

ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ
ΤΕΙ - ΗΡΑΚΛΕΙΟΥ



ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

ΤΜΗΜΑ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΠΟΛΥΜΕΣΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΣΧΕΔΙΑΣΜΟΣ και ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ
ΟΡΓΑΝΩΣΗΣ ΕΤΑΙΡΙΚΩΝ ΠΛΗΡΟΦΟΡΙΩΝ

- Κουτσόπουλος Διονύσιος, Α.Μ. 1550
- Παντελή Παντελάκης, Α.Μ. 1810

Επιβλέπων Καθηγητής: Μαλάμος Αθανάσιος

Ευχαριστίες

Ευχαριστούμε θερμά από καρδιάς, όλους τους ανθρώπους που μας σταθήκανε σε όλες τις δύσκολες στιγμές που περάσαμε για να ολοκληρωθεί η συγκεκριμένη εργασία. Στιγμές χρονικές, οικονομικές αλλά και βεβαρημένες ψυχολογικά.

Κυρίως, χρωστάμε ένα μεγάλο ευχαριστώ στις οικογένειες μας, που μας στήριξαν σε όλο αυτό το χρονικό διάστημα που πέρασε για την εκπόνηση της πτυχιακής εργασίας, που μας υπενθύμιζαν πάντα την προτεραιότητα μας και τον στόχο μας, ακόμα και όταν αποκλίναμε αυτού, ήταν εκεί, καθοδηγητικοί και όχι πιεστικοί.

Ευχαριστούμε τον καθηγητή μας, κ. Γιώργο Μαμάκη, για όλη την βοήθεια που μας έδωσε. Τον ευχαριστούμε για τον μοναδικό τρόπο που μας έδειξε, πώς να μάθουμε κάτι πραγματικά αξιόλογο στον κλάδο. Έναν τρόπο καθοδηγητικό, που μετά από την ολοκλήρωση της εργασίας αυτής, μπορούμε και οι δυο να πούμε πως είμαστε γνώστες του αντικειμένου. Τον ευχαριστούμε επίσης για την επιείκεια και την υπομονή που έκανε για την καθυστέρηση παραχώρησης της εφαρμογής.

Τέλος, ευχαριστούμε τον καθηγητή Α. Μαλάμο, για την προσφορά του να αναλάβει ως επόπτης καθηγητής. Ο επόπτης καθηγητής είναι απαραίτητος για την πραγματοποίηση στην παρουσίαση και παράδοση πτυχιακών εργασιών. Την ημέρα παράδοσης της εργασίας μας, ο κ. Μαμάκης είχε πάψει να εργάζεται στο Τ.Ε.Ι. της Κρήτης ως καθηγητής και σταμάτησε πλέον να έχει το δικαίωμα να παραμείνει ο επιβλέπων καθηγητής της πτυχιακής εργασίας μας.

Abstract

The digital application that we are going to see below, is a complex in construction, but easy to use, work that helps any business, company, store, or any person individually to organize inside his computer, all financial transactions made daily, monthly or whenever he wants to set. Transactions that are related to income or expense, transactions that are classified by type of account and name partner.

All of the above mention the profit that the user has in use of the final result of the application. But, winner isn't only the user of the digital application, but everyone that deals with the sector of computer science and want to learn something about programming, either it refers to knowledge of code, or a program.

The next following pages, contain useful information, because they show a good way to create a similar application, and is a valuable way of syntax for such documents with detailed presentation of the job, from the programmer's eye, but, and from the user's eye.

Σύνοψη

Η ψηφιακή εφαρμογή που θα μελετήσουμε παρακάτω, είναι μια πολύπλοκη στην κατασκευή, μα εύκολη στην χρήση, εργασία η οποία βοηθάει την κάθε επιχείρηση, εταιρία, κατάσταση αλλά και οποιονδήποτε άνθρωπο μεμονωμένα να οργανώσει μέσα στον υπολογιστή του, όλες τις οικονομικές συναλλαγές που κάνει ημερησίως, μηνιαίως ή όποτε αυτός θέλει να ορίσει. Συναλλαγές που αφορούν έσοδα ή έξοδα, συναλλαγές που κατηγοριοποιούνται ανά είδος λογαριασμού και επωνυμία συνεργάτη.

Όλα τα παραπάνω, αναφέρονται στο κέρδος που έχει ο χρήστης στην χρήση του τελικού αποτελέσματος της εφαρμογής. Όμως, κερδισμένος δεν βγαίνει μόνο ο χρήστης της ψηφιακής εφαρμογής, αλλά και όποιος ασχολείται με τον χώρο της πληροφορικής και θέλει να μάθει κάποια πράγματα για τον προγραμματισμό, είτε αυτό αφορά γνώση κώδικα, είτε κάποιο πρόγραμμα.

Οι επόμενες σελίδες που ακολουθούν, περιέχουν χρήσιμες πληροφορίες, γιατί δείχνουν έναν καλό τρόπο δημιουργίας κάποιας παρόμοιας εφαρμογής, καθώς και αποτελεί έναν αξιόλογο τρόπο σύνταξης εγγράφου για την λεπτομερή παρουσίαση της δουλειάς, από το μάτι του προγραμματιστή, αλλά και του χρήστη.

Περιεχόμενα

Ευχαριστίες.....	I
Abstract.....	II
Σύνοψη.....	III
Περιεχόμενα.....	IV
Πίνακας Εικόνων.....	VI
1. Κεφάλαιο 1 - Εισαγωγή.....	1
1.1 – Περίληψη.....	1
1.2 – Κίνητρο για την διεξαγωγή της Εργασίας.....	1
1.3 – Σκοπός και Στόχοι Εργασίας.....	1
1.4 – Δομή Εργασίας.....	2
2. Κεφάλαιο 2 - Προγράμματα.....	3
2.1 – Εισαγωγή.....	3
2.2 – MySQL Server 5.1.....	3
2.2.1 – Εισαγωγή στον MySQL Server.....	3
2.2.2 – Εγκατάσταση MySQL Server 5.1.....	3
2.3 – MySQL GUI Tools.....	14
2.3.1 – Εισαγωγή στα MySQL GUI Tools.....	14
2.3.2 – Εγκατάσταση MySQL GUI Tools.....	14
2.4 – JDK 6 Update 25.....	16
2.4.1 – Εισαγωγή στο JDK.....	16
2.4.2 – Εγκατάσταση JDK 6.....	16
2.5 – Netbeans IDE 6.9.1.....	21
2.5.1 – Εισαγωγή στον Netbeans.....	21
2.5.2 – Εγκατάσταση Netbeans IDE 6.9.1.....	21
2.6 – Apache Tomcat 6.0.....	26
2.6.1 – Εισαγωγή στον Apache Tomcat.....	26
2.6.2 – Εγκατάσταση Apache Tomcat 6.0.....	27
2.7 – MySQL Connector/J 5.1.....	28
2.7.1 – Εισαγωγή στον MySQL Connector/J.....	28
2.7.2 – Εγκατάσταση MySQL Connector/J.....	28
3. Κεφάλαιο 3 - Κατασκευή και Ανάπτυξη Web Application με την χρήση Netbeans.....	30
3.1 – Εισαγωγή.....	30
3.2 – Εξοικείωση με τον Netbeans IDE 6.9.1.....	30
3.3 – Εισαγωγή στο Project.....	31
3.3.1 – Δημιουργία ενός νέου Java Web Application στο IDE.....	31
3.3.2 – Δημιουργία ενός νέου JSP αρχείου.....	33
3.2.3 – Δημιουργία ενός νέου πακέτου JAVA.....	34
3.3.4 – Δημιουργία ενός νέου JAVA Servlet.....	36
3.3.5 – Δημιουργία ενός νέου JAVA Class.....	37
3.3.6 – Δημιουργία ενός νέου JavaScript αρχείου.....	39
3.3.7 – Δημιουργία ενός νέου CSS αρχείου.....	41
3.3.8 – Σύνδεση του Web Application με τον MySQL Connector/J.....	43
3.3.9 – Σύνδεση του Web Application με τον Tomcat Server 6.0.....	45
4. Κεφάλαιο 4 - JSP, Servlets και Γλώσσες Προγραμματισμού.....	49
4.1 – Εισαγωγή.....	49
4.2 – JAVA.....	49
4.2.1 – Εισαγωγή στην JAVA.....	49
4.2.2 – Βασικά Χαρακτηριστικά JSP.....	52

4.2.3 – Αρχεία JSP	52
4.2.4 – UML	53
4.2.5 – Βασικά Χαρακτηριστικά των JAVA Servlets	53
4.2.6 – Αρχεία JAVA Servlets	54
4.2.7 – Βασικά Χαρακτηριστικά των JAVA Class	59
4.2.8 – Αρχεία JAVA Class	60
4.3 – MySQL	85
4.3.1 – Βασικά Χαρακτηριστικά MySQL	85
4.3.2 – Πίνακες της MySQL Βάσης Δεδομένων(financial)	86
4.4 – CSS.....	91
4.4.1 – Βασικά Χαρακτηριστικά CSS	91
4.4.2 – Αρχεία CSS	91
4.5 – JavaScript.....	92
4.5.1 – Βασικά Χαρακτηριστικά JavaScript	92
4.5.2 – Αρχεία JavaScript.....	92
5. Κεφάλαιο 5 - Εγχειρίδιο Χρήστη	93
5.1 – Εισαγωγή.....	93
5.2 – Αρχική	94
5.2.1 – Στοιχεία Λογαριασμού	94
5.2.2 – Πληρωμές	95
5.2.3 – Ενέργειες	97
5.2.4 – Γενέθλια	97
5.3 – Εταιρίες	97
5.3.1 – Εταιρίες	97
5.3.2 – Πελάτες/Προμηθευτές	100
5.3.3 – Έσοδα/Έξοδα	103
5.4 – Ημερολόγιο.....	105
5.4.1 – Γενέθλια	105
5.4.2 – Ενέργειες	105
5.5 – Τηλεφωνικός Κατάλογος	107
5.5.1 – Εισαγωγή Νέας Επαφής	108
5.5.2 – Ευρετήριο	109
5.5.3 – Αναζήτηση	110
5.6 – Οικονομική Επισκόπηση	111
5.6.1 – Επιλογή Εταιρίας.....	111
5.6.2 – Φόρμα Αναζήτησης.....	112
5.6.3 – Έσοδα	113
5.6.4 – Έξοδα	119
5.6.5 – Συνολικά.....	124
5.7 – Έξοδος	125
Βιβλιογραφία	126
Παράρτημα Α: Υπόδειξη τιμών	127
Παράρτημα Β: Διαφάνειες Παρουσίασης Πτυχιακής Εργασίας	129
Παράρτημα Γ: Περιεχόμενα CD.....	139

Πίνακας Εικόνων

2. Κεφάλαιο 2 - Προγράμματα	3
2.2 MySQL Server 5.1	3
Εικόνα 2.2.1. Λήψη MySQL Server.....	4
Εικόνα 2.2.2. Εγκατάσταση MySQL Server	4
Εικόνα 2.2.3. Εγκατάσταση MySQL Server	5
Εικόνα 2.2.4. Εγκατάσταση MySQL Server	5
Εικόνα 2.2.5. Εγκατάσταση MySQL Server	6
Εικόνα 2.2.6. Εγκατάσταση MySQL Server	6
Εικόνα 2.2.7. Εγκατάσταση MySQL Server	7
Εικόνα 2.2.8. Εγκατάσταση MySQL Server	7
Εικόνα 2.2.9. Εγκατάσταση - Ρυθμίσεις MySQL Server	8
Εικόνα 2.2.10. Εγκατάσταση - Ρυθμίσεις MySQL Server	8
Εικόνα 2.2.11. Εγκατάσταση - Ρυθμίσεις MySQL Server	9
Εικόνα 2.2.12. Εγκατάσταση - Ρυθμίσεις MySQL Server	9
Εικόνα 2.2.13. Εγκατάσταση - Ρυθμίσεις MySQL Server	10
Εικόνα 2.2.14. Εγκατάσταση - Ρυθμίσεις MySQL Server	10
Εικόνα 2.2.15. Εγκατάσταση - Ρυθμίσεις MySQL Server	11
Εικόνα 2.2.16. Εγκατάσταση - Ρυθμίσεις MySQL Server	11
Εικόνα 2.2.17. Εγκατάσταση - Ρυθμίσεις MySQL Server	12
Εικόνα 2.2.18. Εγκατάσταση - Ρυθμίσεις MySQL Server	12
Εικόνα 2.2.19. Εγκατάσταση - Ρυθμίσεις MySQL Server	13
Εικόνα 2.2.20. Εγκατάσταση - Ρυθμίσεις MySQL Server	13
2.3 MySQL GUI Tools	14
Εικόνα 2.3.1. Λήψη MySQL GUI Tools.....	14
Εικόνα 2.3.2. Αποθήκευση MySQL GUI Tools.....	14
Εικόνα 2.3.3. Προβολή MySQL GUI Tools	15
Εικόνα 2.3.4. Είσοδος στο MySQL Administrator	15
Εικόνα 2.3.5. MySQL Administrator	16
2.4 JDK 6 Update 25.....	16
Εικόνα 2.4.1. Λήψη JDK.....	17
Εικόνα 2.4.2. Εγκατάσταση JDK	17
Εικόνα 2.4.3. Εγκατάσταση JDK	18
Εικόνα 2.4.4. Εγκατάσταση JDK	18
Εικόνα 2.4.5. Εγκατάσταση JDK	19
Εικόνα 2.4.6. Εγκατάσταση JDK	19
Εικόνα 2.4.7. Εγκατάσταση JDK	20
Εικόνα 2.4.8. Εγκατάσταση JDK	20
2.5 Netbeans IDE 6.9.1	21
Εικόνα 2.5.1. Λήψη Netbeans	21
Εικόνα 2.5.2. Φόρτωση Εγκατάστασης Netbeans.....	22
Εικόνα 2.5.3. Εγκατάσταση Netbeans.....	22
Εικόνα 2.5.4. Εγκατάσταση Netbeans.....	23
Εικόνα 2.5.5. Εγκατάσταση Netbeans.....	23
Εικόνα 2.5.6. Εγκατάσταση Netbeans.....	24
Εικόνα 2.5.7. Εγκατάσταση Netbeans.....	24
Εικόνα 2.5.8. Εγκατάσταση Netbeans.....	25
Εικόνα 2.5.9. Εγκατάσταση Netbeans.....	25
Εικόνα 2.5.10. Εγκατάσταση Netbeans.....	26

2.6 Apache Tomcat 6.0	26
Εικόνα 2.6.1. Λήψη Apache Tomcat.....	27
Εικόνα 2.6.2. Αποθήκευση Apache Tomcat	28
2.7 MySQL Connector/J	28
Εικόνα 2.7.1. Λήψη MySQL Connector/J.....	29
Εικόνα 2.7.2. Αποθήκευση MySQL Connector/J	29
3. Κεφάλαιο 3 - Κατασκευή και Ανάπτυξη Web Application με την χρήση Netbeans	30
3.2 Εξοικείωση με το Netbeans IDE 9	30
Εικόνα 3.2.1. Εκκίνηση Netbeans IDE 6.9.1	30
Εικόνα 3.2.2. Netbeans IDE 6.9.1	31
3.3 Εισαγωγή στο Project	31
3.3.1 Δημιουργία ενός νέου Java Web Application στο IDE	31
Εικόνα 3.3.1.1. Δημιουργία Web Application.....	31
Εικόνα 3.3.1.2. Δημιουργία Web Application.....	32
Εικόνα 3.3.1.3. Δημιουργία Web Application.....	32
3.3.2 Δημιουργία ενός νέου JSP αρχείου	33
Εικόνα 3.3.2.1. Δημιουργία JSP	33
Εικόνα 3.3.2.2. Δημιουργία JSP	33
Εικόνα 3.3.2.3. Δημιουργία JSP	34
3.3.3 Δημιουργία ενός νέου πακέτου JAVA	34
Εικόνα 3.3.3.1. Δημιουργία Java Package.....	34
Εικόνα 3.3.3.2. Δημιουργία Java Package.....	35
Εικόνα 3.3.3.3. Δημιουργία Java Package.....	35
3.3.4 Δημιουργία ενός νέου JAVA Servlet	36
Εικόνα 3.3.4.1. Δημιουργία Servlet.....	36
Εικόνα 3.3.4.2. Δημιουργία Servlet.....	36
Εικόνα 3.3.4.3. Δημιουργία Servlet.....	37
3.3.5 Δημιουργία ενός νέου Java Class	37
Εικόνα 3.3.5.1. Δημιουργία Java Class	37
Εικόνα 3.3.5.2. Δημιουργία Java Class	38
Εικόνα 3.3.5.3. Δημιουργία Java Class	38
3.3.6 Δημιουργία ενός νέου JavaScript αρχείου	39
Εικόνα 3.3.6.1. Δημιουργία JavaScript	39
Εικόνα 3.3.6.2. Δημιουργία JavaScript	39
Εικόνα 3.3.6.3. Δημιουργία JavaScript	40
Εικόνα 3.3.6.4. Δημιουργία JavaScript	40
3.3.7 Δημιουργία ενός νέου CSS αρχείου	41
Εικόνα 3.3.7.1. Δημιουργία CSS	41
Εικόνα 3.3.7.2. Δημιουργία CSS	41
Εικόνα 3.3.7.3. Δημιουργία CSS	42
Εικόνα 3.3.7.4. Δημιουργία CSS	42
3.3.8 Σύνδεση του Web Application με τον MySQL Connector/J	43
Εικόνα 3.3.8.1. Σύνδεση MySQL Connector/J.....	43
Εικόνα 3.3.8.2. Σύνδεση MySQL Connector/J.....	44
Εικόνα 3.3.8.3. Σύνδεση MySQL Connector/J.....	44
Εικόνα 3.3.8.4. Σύνδεση MySQL Connector/J.....	45
3.3.9 Σύνδεση του Web Application με τον Tomcat Server 6.0	45
Εικόνα 3.3.9.1. Σύνδεση Apache Tomcat.....	45
Εικόνα 3.3.9.2. Σύνδεση Apache Tomcat.....	46
Εικόνα 3.3.9.3. Σύνδεση Apache Tomcat.....	46
Εικόνα 3.3.9.4. Σύνδεση Apache Tomcat.....	47

Εικόνα 3.3.9.5. Σύνδεση Apache Tomcat.....	47
Εικόνα 3.3.9.6. Σύνδεση Apache Tomcat.....	48
Εικόνα 3.3.9.7. Σύνδεση Apache Tomcat.....	48
4. Κεφάλαιο 4 - JSP, Servlets και Γλώσσες Προγραμματισμού.....	49
4.2 – JAVA	49
Εικόνα 4.2.1. UML του Servlet activities	54
Εικόνα 4.2.2. UML του Servlet addviewcust	55
Εικόνα 4.2.3. UML του Servlet calendar	55
Εικόνα 4.2.4. UML του Servlet companies.....	56
Εικόνα 4.2.5. UML του Servlet finview.....	56
Εικόνα 4.2.6. UML του Servlet home	57
Εικόνα 4.2.7. UML του Servlet incomesExpenses.....	58
Εικόνα 4.2.8. UML του Servlet loginServlet	58
Εικόνα 4.2.9. UML του Servlet phonebook	59
Εικόνα 4.2.10. UML του Java Class previewCompanyAgainst.....	60
Εικόνα 4.2.11. UML του Java Class previewCompanyFor.....	62
Εικόνα 4.2.12. UML του Java Class previewTotal	65
Εικόνα 4.2.13. UML του Java Class checkClass(Γνωρίσματα).....	68
Εικόνα 4.2.14. UML του Java Class checkClass(Λειτουργίες).....	69
Εικόνα 4.2.15. UML του Java Class ServletBean	85
5. Κεφάλαιο 5 - Εγχειρίδιο Χρήστη.....	93
5.1 Εισαγωγή.....	93
Εικόνα 5.1.1. Σύνδεση Χρήστη.....	93
5.2 Αρχική	94
Εικόνα 5.2.1. Στοιχεία Λογαριασμού.....	94
Εικόνα 5.2.2. Αλλαγή Στοιχείων Λογαριασμού.....	94
Εικόνα 5.2.3. Άδεια Φόρμα Πληρωμών που Εκκρεμούν.....	95
Εικόνα 5.2.4. Φόρμα Πληρωμών που Εκκρεμούν	96
Εικόνα 5.2.5. Φόρμα Ενεργειών που Εκκρεμούν.....	97
Εικόνα 5.2.6. Φόρμα Γενεθλίων Τρέχουσας Ημέρας.....	97
5.3 Εταιρίες	97
Εικόνα 5.3.1. Εταιρίες.....	97
Εικόνα 5.3.2. Προσθήκη Νέας Εταιρίας	98
Εικόνα 5.3.3. Προβολή Εταιριών	99
Εικόνα 5.3.4. Επισκόπηση Εταιρίας.....	99
Εικόνα 5.3.5. Διαγραφή Εταιρίας.....	100
Εικόνα 5.3.6. Πελάτες/Προμηθευτές.....	100
Εικόνα 5.3.7. Εισαγωγή Νέου Πελάτη/Προμηθευτή	101
Εικόνα 5.3.8. Προβολή Πελατών/Προμηθευτών	101
Εικόνα 5.3.9. Διαγραφή Πελάτη/Προμηθευτή.....	102
Εικόνα 5.3.10. Επεξεργασία Πελάτη/Προμηθευτή	102
Εικόνα 5.3.11. Έσοδα/Εξόδα	103
Εικόνα 5.3.12. Εισαγωγή Εσόδου/Εξόδου	103
Εικόνα 5.3.13. Προβολή Εσόδων/Εξόδων	104
Εικόνα 5.3.14. Διαγραφή Εσόδου/Εξόδου	104
5.4 Ημερολόγιο.....	105
Εικόνα 5.4.1. Γενέθλια	105
Εικόνα 5.4.2. Ενέργειες.....	105
Εικόνα 5.4.3. Προσθήκη Νέας Ενέργειας	106
Εικόνα 5.4.4. Προβολή Ενεργειών.....	106
Εικόνα 5.4.5. Διαγραφή Ενέργειας.....	107

Εικόνα 5.4.6. Επεξεργασία Ενέργειας.....	107
5.5 Τηλεφωνικός Κατάλογος	107
Εικόνα 5.5.1. Φόρμα Προσθήκης Νέας Επαφής	108
Εικόνα 5.5.2. Εισαγωγή Νέας Επαφής.....	108
Εικόνα 5.5.3. Ευρετήριο.....	109
Εικόνα 5.5.4. Μήνυμα Ευρετηρίου Χωρίς Αποτελέσματα	109
Εικόνα 5.5.5. Μήνυμα Ευρετηρίου Με Αποτελέσματα	109
Εικόνα 5.5.6. Διαγραφή Επαφής	109
Εικόνα 5.5.7. Επεξεργασία Επαφής	110
Εικόνα 5.5.8. Αναζήτηση Επαφής.....	110
Εικόνα 5.5.9. Αποτελέσματα Αναζήτησης με Χρήση OR	111
Εικόνα 5.5.10. Αποτελέσματα Αναζήτησης με Χρήση AND	111
5.6 Οικονομική Επισκόπηση	111
Εικόνα 5.6.1. Επιλογή Εταιρίας	112
Εικόνα 5.6.2. Φόρμα Αναζήτησης	112
Εικόνα 5.6.3. Άδεια Φόρμα Εσόδου ή Εξόδου	113
Εικόνα 5.6.4. Έσοδα.....	114
Εικόνα 5.6.5. Εισαγωγή Εσόδου	114
Εικόνα 5.6.6. Περισσότερα Εσόδου.....	115
Εικόνα 5.6.7. Δόσεις Εσόδου	116
Εικόνα 5.6.8. Επεξεργασία/Εισαγωγή Δόσης Εσόδου	117
Εικόνα 5.6.9. Διαγραφή Εσόδου	118
Εικόνα 5.6.10. Έξοδα	119
Εικόνα 5.6.11. Εισαγωγή Εξόδου.....	120
Εικόνα 5.6.12. Περισσότερα Εξόδου	121
Εικόνα 5.6.13. Δόσεις Εξόδου	121
Εικόνα 5.6.14. Εισαγωγή/Επεξεργασία Δόσης Εξόδου	122
Εικόνα 5.6.15. Διαγραφή Εξόδου.....	123
Εικόνα 5.6.16. Συνολικά	124
Παράρτημα Β. Διαφάνειες Παρουσίασης Πτυχιακής Εργασίας.....	129
Εικόνα Β1. Εξώφυλλο Παρουσίασης.....	129
Εικόνα Β2. Περιεχόμενα Παρουσίασης	129
Εικόνα Β3. Εισαγωγή.....	130
Εικόνα Β4. Εργαλεία	130
Εικόνα Β5. Financial – Τι είναι.....	131
Εικόνα Β6. Financial – Από τι αποτελείται.....	131
Εικόνα Β7. Financial – Σύνδεση	132
Εικόνα Β8. Financial – Αρχική	132
Εικόνα Β9. Financial – Εταιρίες(Μέρος 1 ^ο).....	133
Εικόνα Β10. Financial – Εταιρίες(Μέρος 2 ^ο).....	133
Εικόνα Β11. Financial – Ημερολόγιο(Μέρος 1 ^ο).....	134
Εικόνα Β12. Financial – Ημερολόγιο(Μέρος 2 ^ο).....	134
Εικόνα Β13. Financial – Τηλεφωνικός Κατάλογος(Μέρος 1 ^ο).....	135
Εικόνα Β14. Financial – Τηλεφωνικός Κατάλογος(Μέρος 2 ^ο)	135
Εικόνα Β15. Financial – Οικονομική Επισκόπηση(Μέρος 1 ^ο)	136
Εικόνα Β16. Financial – Οικονομική Επισκόπηση (Μέρος 2 ^ο)	136
Εικόνα Β17. Financial – Οικονομική Επισκόπηση (Μέρος 3 ^ο)	137
Εικόνα Β18. Financial – Έξοδος.....	137
Εικόνα Β19. Επίδειξη Εφαρμογής.....	138
Εικόνα Β20. Τέλος Παρουσίασης Πτυχιακής Εργασίας	138

Λίστα Πινάκων

Παράρτημα Α.....	127
Πίνακας Α1. Τύποι Συμπλήρωσης Πεδίων	127
Πίνακας Α2. Συμπλήρωση πεδίων στην σελίδα Αρχική	127
Πίνακας Α3. Συμπλήρωση πεδίων στην σελίδα Εταιρίες	127
Πίνακας Α4. Συμπλήρωση πεδίων στην σελίδα Ημερολόγιο	128
Πίνακας Α5. Συμπλήρωση πεδίων στην σελίδα Τηλεφωνικός Κατάλογος	128
Πίνακας Α6. Συμπλήρωση πεδίων στην σελίδα Οικονομική Επισκόπηση.....	128

ΚΕΦΑΛΑΙΟ 1

Εισαγωγή

Η πτυχιακή εργασία συνιστά ένα εννοιολογικά και μεθοδολογικά συμπαγές έργο που καταδεικνύει τις μεθοδολογικές, κριτικές, αναλυτικές, συγγραφικές -και ενίοτε- ερευνητικές ικανότητες και δεξιότητες των φοιτητών. Ειδικότερα στο πλαίσιο της πτυχιακής εργασίας θα πρέπει να καταδεικνύεται η ικανότητα του τελειόφοιτου σπουδαστή να :

- Αναζητά, διερευνά, επιλέγει, χρησιμοποιεί και να καταγράφει ορθά στοιχεία από την ελληνική και διεθνή βιβλιογραφία.
- Προσδιορίζει με ακρίβεια το θέμα του, να οργανώνει, να δομεί και να επιμερίζει την εργασία του ώστε να αναδεικνύεται η ουσία του θέματος.
- Ακολουθεί σαφή και λογική πορεία ανάλυσης του θέματος με εμπειριστατωμένη και πλήρη επιχειρηματολογία.
- Υποστηρίζει την εργασία ενώπιον επιτροπής και κοινού, διατυπώνοντας με ορθότητα και σαφήνεια τα βασικά σημεία και απαντώντας εμπειριστατωμένα σε σχετικές ερωτήσεις ειδικού ή γενικότερου ενδιαφέροντος.

1.1– Περίληψη

Στην πτυχιακή εργασία δημιουργείται μια δικτυακή υπηρεσία που δίνει τη δυνατότητα στο χρήστη να οργανώσει τα δεδομένα και τις πληροφορίες μιας εταιρίας.

Κάνοντας χρήση της υπηρεσίας, ο χρήστης μπορεί να δει τις υποχρεώσεις, τα έσοδα, τα έξοδα και πολλά άλλα που έχουν να κάνουν με τους συνεργάτες και τους πελάτες μιας επιχείρησης. Δημιουργούνται υπηρεσίες Τηλεφωνικού Καταλόγου όπου μπορεί να γίνει καταχώρηση επαφών, Ημερολογίου όπου υπάρχει η δυνατότητα προβολής των γενεθλίων των επαφών του Τηλεφωνικού Καταλόγου και των Ενεργειών που δραστηριοποιούνται από την εταιρία και όχι μόνο. Ακόμη υπάρχει η υπηρεσία της Οικονομικής Επισκόπησης όπου εκεί παρουσιάζονται αναλυτικά τα έσοδα και τα έξοδα της επιχείρησης, που έχουν να κάνουν με τις συνεργαζόμενες εταιρίες, όπως επίσης τα κέρδη και οι απώλειες που έχει κάνει η επιχείρηση.

1.2– Κίνητρο για την Διεξαγωγή της Εργασίας

Βασικό κίνητρο για την διεξαγωγή της εργασίας αυτής, η μεταστροφή όλο και περισσότερων χρηστών στην χρήση της τεχνολογίας, είτε από θέληση είτε από ανάγκη. Σημαντικό ρόλο έπαιξε επίσης η γνώση σύγχρονων γλωσσών προγραμματισμού, όπως **JAVA**, **MySQL**, **JavaScript**, καθώς και προγραμμάτων, από την σύλληψη της ιδέας έως και την υλοποίηση τους, για την ανάγκη της κάλυψης της εργασίας.

1.3– Σκοπός και Στόχοι Εργασίας

Σκοπός της παρούσας πτυχιακής εργασίας είναι η πραγματοποίηση μιας ψηφιακής εφαρμογής η οποία **οργανώνει** και **ταξινομεί** με εύχρηστο τρόπο, **πληροφορίες** που αφορούν μια επιχείρηση. Η εφαρμογή απευθύνεται στον ιδιοκτήτη της επιχείρησης ή στον υπεύθυνο που έχει τεθεί στα οικονομικά ή στην διοίκηση της επιχείρησης.

Επιμέρους στόχοι της συγκεκριμένης Πτυχιακής Εργασίας είναι:

- a) Η έρευνα των πληροφοριών και δεδομένων που χρειάζονται για την κάλυψη όλων των αναγκών της συγκεκριμένης εφαρμογής.
- b) Ο έλεγχος των σωστών αποδόσεων των εννοιών, όρων και πράξεων.
- c) Η ευπαρουσίαστη απεικόνιση της εφαρμογής ώστε να μην γίνεται βαρετή προς τον χρήστη.
- d) Η ασφάλεια της εφαρμογής από τυχόν επιθέσεις.
- e) Η ανάπτυξη όλων των παραπάνω στην πράξη.

1.4– Δομή Εργασίας

Στα παρακάτω κεφάλαια θα γνωρίσουμε την σειρά η οποία ακολουθήθηκε για να ολοκληρωθεί η πτυχιακή εργασία, η εφαρμογή αλλά και το έντυπο. Επίσης, θα γνωρίσουμε αναλυτικά τις γλώσσες προγραμματισμού που χρησιμοποιήθηκαν, τον τρόπο που συνδέονται μεταξύ τους αλλά και τα αρχεία που δομήθηκαν για να φτάσουμε στο επιθυμητό αποτέλεσμα. Θα έχουμε την δυνατότητα να γνωρίσουμε τα προγράμματα που χρησιμοποιήθηκαν για την πραγματοποίηση της πτυχιακής εργασίας στον πρακτικό τομέα, πού τα βρίσκουμε, πως εγκαθίστανται αλλά θα βρείτε και παραδείγματα δημιουργίας αρχείων μέσα από τα προγράμματα, όπου χρειάζεται. Τέλος θα δείξουμε έναν πρότυπο τρόπο δημιουργίας **Εγχειρίδιου χρήστη(User Manual)**, στην περίπτωση που ,κάποιος θέλει να προσφέρει το application που έχει κάνει μαζί με ένα manual.

ΚΕΦΑΛΑΙΟ 2

Προγράμματα

2.1 - Εισαγωγή

Το κεφάλαιο αυτό ασχολείται με την αναφορά, επίδειξη, εγκατάσταση και σύνδεση μεταξύ τους, προγραμμάτων. Να σημειωθεί ότι η χρήση των προγραμμάτων πραγματοποιήθηκε σε λειτουργικό **Windows XP**. Τα προγράμματα που χρησιμοποιήθηκαν για την ολοκλήρωση της εφαρμογής ήταν:

- **MySQL Server 5.1**
- **MySQL GUI tools**
- **JDK 6 update 25**
- **Netbeans IDE 6.9.1**
- **Apache Tomcat Server 6.0**
- **MySQL Connector/J 5.1**
- **Web Browser**

2.2- MySQL Server 5.1

2.2.1 – Εισαγωγή στον MySQL Server

Ο MySQL Server είναι λογισμικό δημιουργίας και διαχείρισης MySQL βάσεων δεδομένων. Η MySQL διατίθεται σε δύο βασικές εκδόσεις:

- **MySQL Enterprise:** η διαχείριση και η βελτιστοποίηση προσφέρεται από επαγγελματίες συμβούλους της MySQL. Η έκδοση αυτή έχει συνδρομή.
- **MySQL Community Server:** Ανά τακτά χρονικά διαστήματα, η τελευταία έκδοση της MySQL Enterprise περνά στην έκδοση MySQL Community Server, στην οποία ο κώδικας είναι ελεύθερος και η χρήση της γίνεται υπό την άδεια GPL (GNU General Public License <http://www.gnu.org/licenses/license-list.html>)

Θα χρησιμοποιήσετε τον MySQL Community Server, έκδοση 5.1, την οποία μπορείτε να κατεβάσετε από τον δικτυακό τόπο της MySQL: <http://dev.mysql.com/downloads/mysql/5.0.html> . Παρέχεται υποστήριξη για πλήθος λειτουργικών συστημάτων και αρχιτεκτονικών υπολογιστών. Περισσότερες πληροφορίες για την εγκατάσταση δείτε στο εγχειρίδιο χρήστη: <http://dev.mysql.com/doc/refman/5.0/en/>

Το **εγχειρίδιο** της MySQL (**MySQL Reference Manual**), στο οποίο μπορείτε να ανατρέχετε, βρίσκεται στη διεύθυνση: <http://dev.mysql.com/doc/>, απ' όπου μπορείτε να το κατεβάσετε είτε να το δείτε online. Στην συνέχεια θα παρουσιαστεί η εγκατάσταση της έκδοσης για Windows.

2.2.2 – Εγκατάσταση MySQL Server 5.1

1. Κατεβάστε δωρεάν το MySQL Server Community Server Edition. Μπορείτε να κατεβάσετε το λογισμικό από την ιστοσελίδα της MySQL(<http://dev.mysql.com>). Βεβαιωθείτε ότι κατεβάζετε το **zip** αρχείο που αντιστοιχεί στο δικό σας σύστημα. Στην δικιά μας περίπτωση, χρειάστηκε το Windows(x86, 32-bit), ZIP Archive.

MySQL Community Server 5.5.13

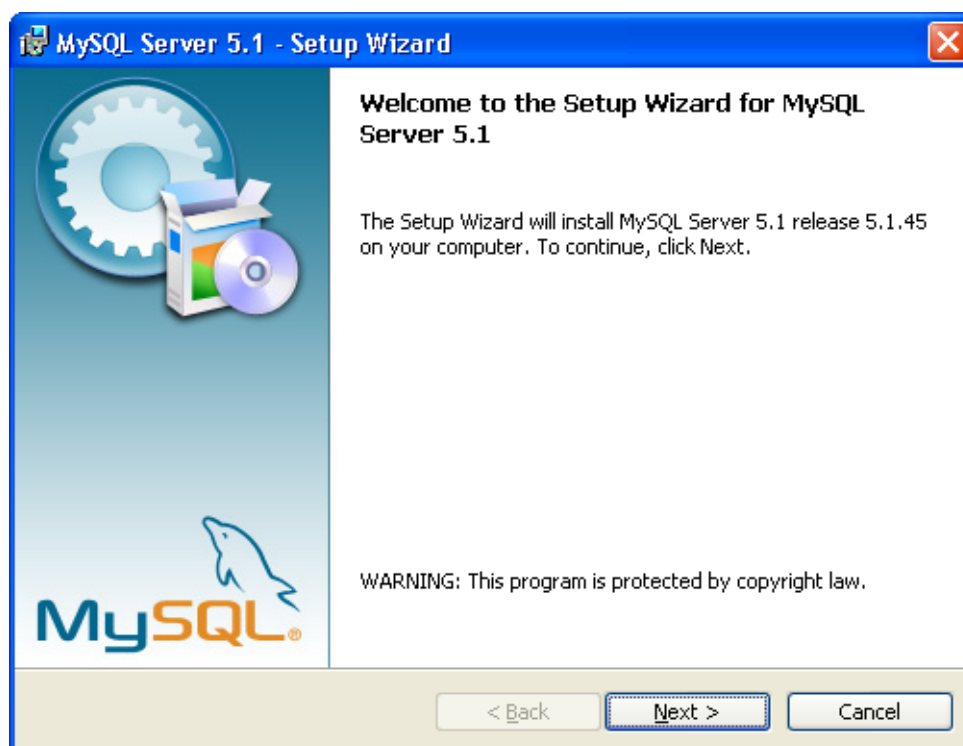
Select Platform: Looking for previous GA versions?

Microsoft Windows

Windows (x86, 32-bit), MSI Installer (mysql-5.5.13-win32.msi)	5.5.13	27.7M	Download
Windows (x86, 64-bit), MSI Installer (mysql-5.5.13-winx64.msi)	5.5.13	28.8M	Download
Windows (x86, 32-bit), ZIP Archive (mysql-5.5.13.zip)	5.5.13	27.0M	Download
Windows (x86, 32-bit), ZIP Archive (mysql-5.5.13-win32.zip)	5.5.13	133.3M	Download
Windows (x86, 64-bit), ZIP Archive (mysql-5.5.13-winx64.zip)	5.5.13	135.5M	Download

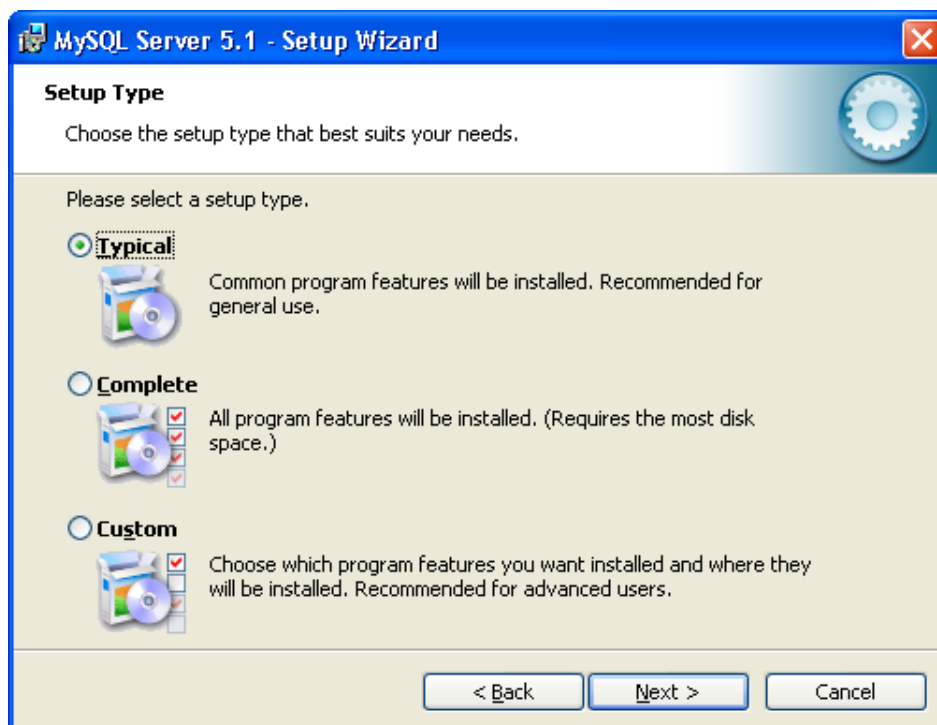
Εικόνα 2.2.1. Λήψη MySQL Server

- Μετά την λήψη του αρχείου, ανοίξτε το zip αρχείο με κάποιο πρόγραμμα από-συμπίεσης που έχετε στο σύστημα σας. Μόλις ανοίξει ο συμπιεσμένος φάκελος, μια λίστα από διάφορα αρχεία εμφανίζεται. Αναζητήστε το εκτελέσιμο αρχείο .exe με την ονομασία "**Setup.exe**". Πατήστε διπλό κλικ πάνω του χωρίς να το αποσυμπιέσετε. Στο παράθυρο που θα εμφανιστεί, επιλέξτε Next.



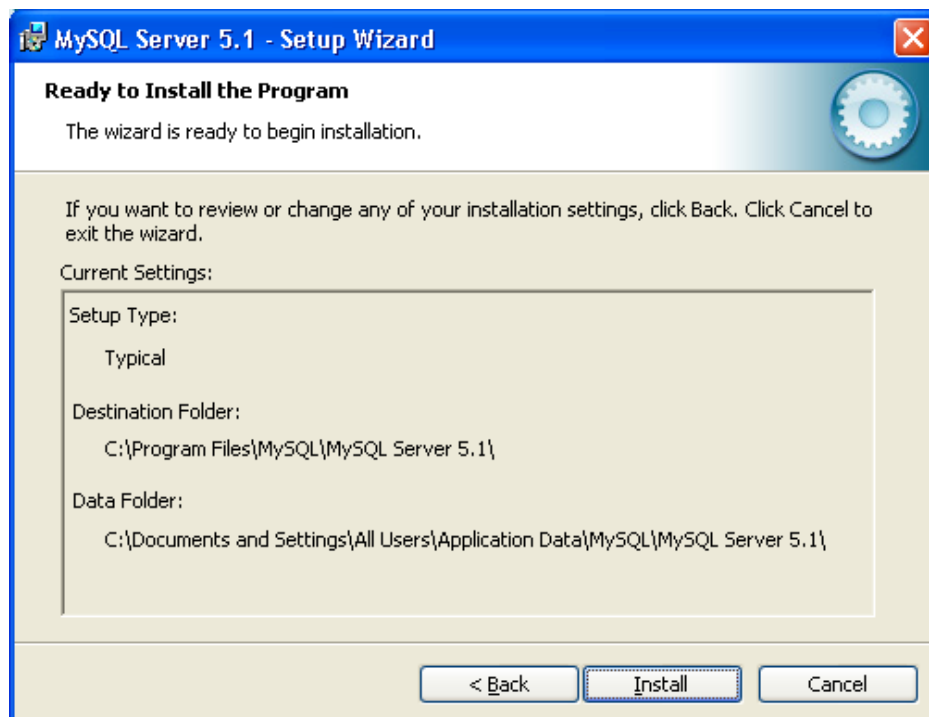
Εικόνα 2.2.2. Εγκατάσταση MySQL Server

3. Στο επόμενο παράθυρο, επιλέξτε την "**Typical**" εγκατάσταση και στη συνέχεια κάντε κλικ στο κουμπί Next.



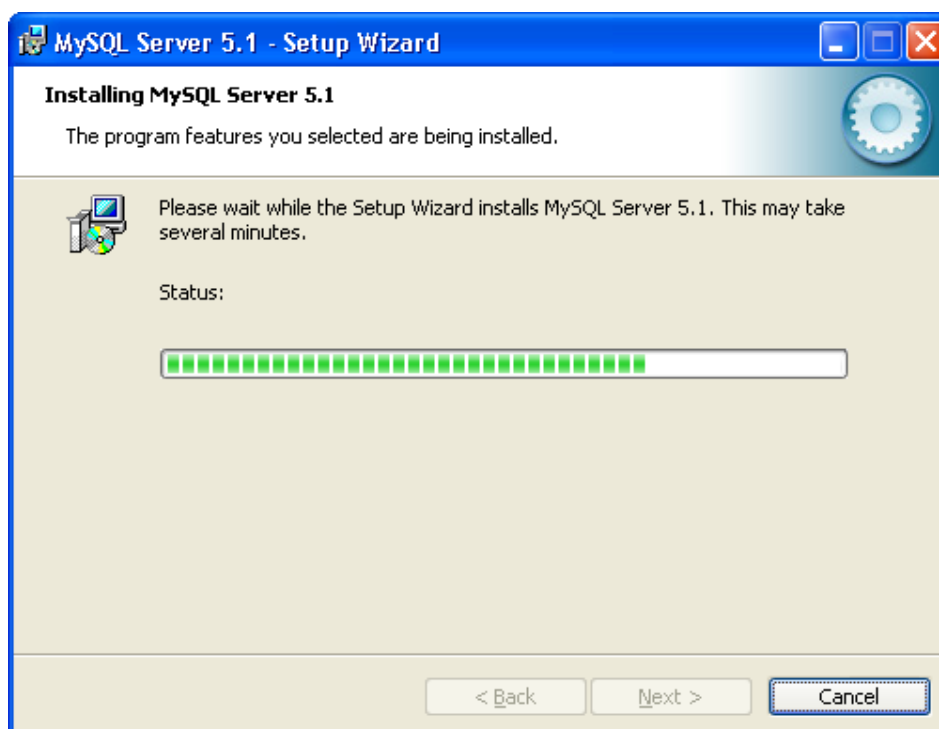
Εικόνα 2.2.3. Εγκατάσταση MySQL Server

4. Τώρα, η MySQL είναι έτοιμη να εγκατασταθεί. Στο επόμενο παράθυρο, επιλέξτε Install για την εκκίνηση της εγκατάστασης.



Εικόνα 2.2.4. Εγκατάσταση MySQL Server

5. Αναμείνατε μέχρι την ολοκλήρωση της εγκατάστασης της MySQL.



Εικόνα 2.2.5. Εγκατάσταση MySQL Server

6. Όταν τελειώσει η εγκατάσταση θα εμφανιστεί κάποια ενημέρωση για την έκδοση MySQL Enterprise.



Εικόνα 2.2.6. Εγκατάσταση MySQL Server

7. Με διαδοχικά Next θα προσπεράσετε την ενημέρωση και θα φτάσετε στο επόμενο.



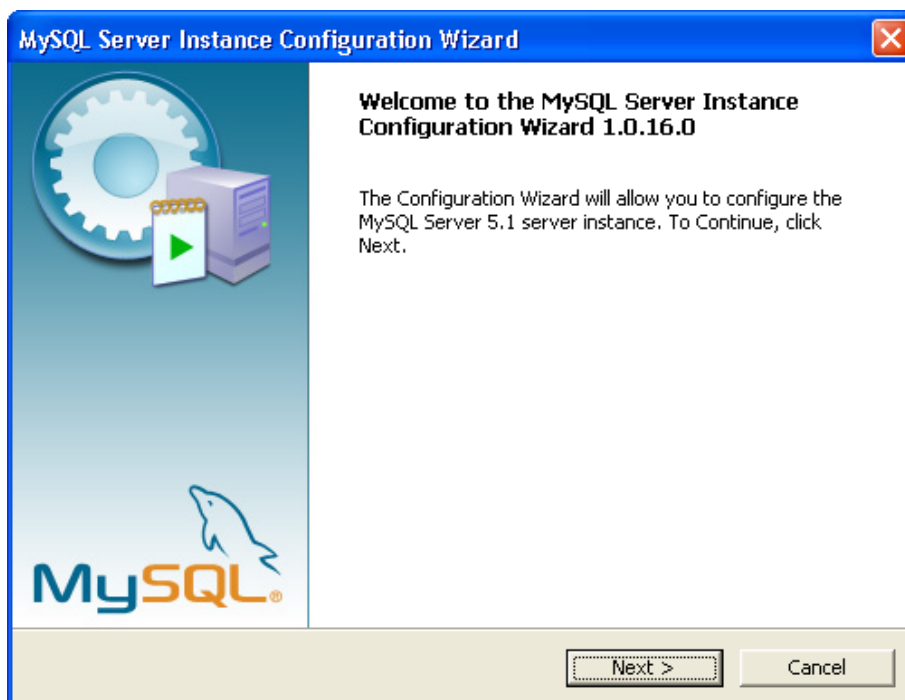
Εικόνα 2.2.7. Εγκατάσταση MySQL Server

8. Με διαδοχικά Next θα προσπεράσετε την ενημέρωση και θα φτάσετε στο παρακάτω παράθυρο. Τσεκάρτε την πρώτη επιλογή **Configure the MySQL Server Now**, και αφήστε την άλλη, **Register the MySQL Server Now**, κενή. Αφορά την εγγραφή στην MySQL η οποία δεν είναι απαραίτητη. Επιλέξτε Finish.



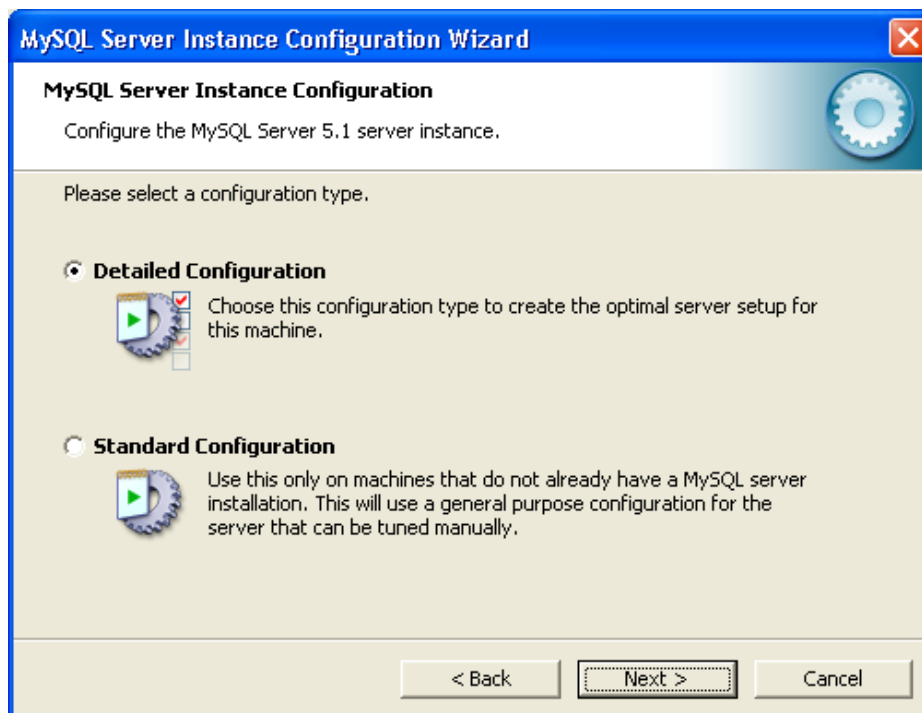
Εικόνα 2.2.8. Εγκατάσταση MySQL Server

9. Σε αυτό το στάδιο η διαδικασία της εγκατάστασης της MySQL έχει τελειώσει. Απομένει η ρύθμιση των παραμέτρων του προγράμματος. Αυτό γίνεται στο παρακάτω παράθυρο που καλέσαμε στο προηγούμενο βήμα, στον **MySQL Server Instance Configuration Wizard**. Επιλέξτε Next.



Εικόνα 2.2.9. Εγκατάσταση - Ρυθμίσεις MySQL Server

10. Επιλέξτε την πρώτη περίπτωση **Detailed Configuration** και πατήστε Next.



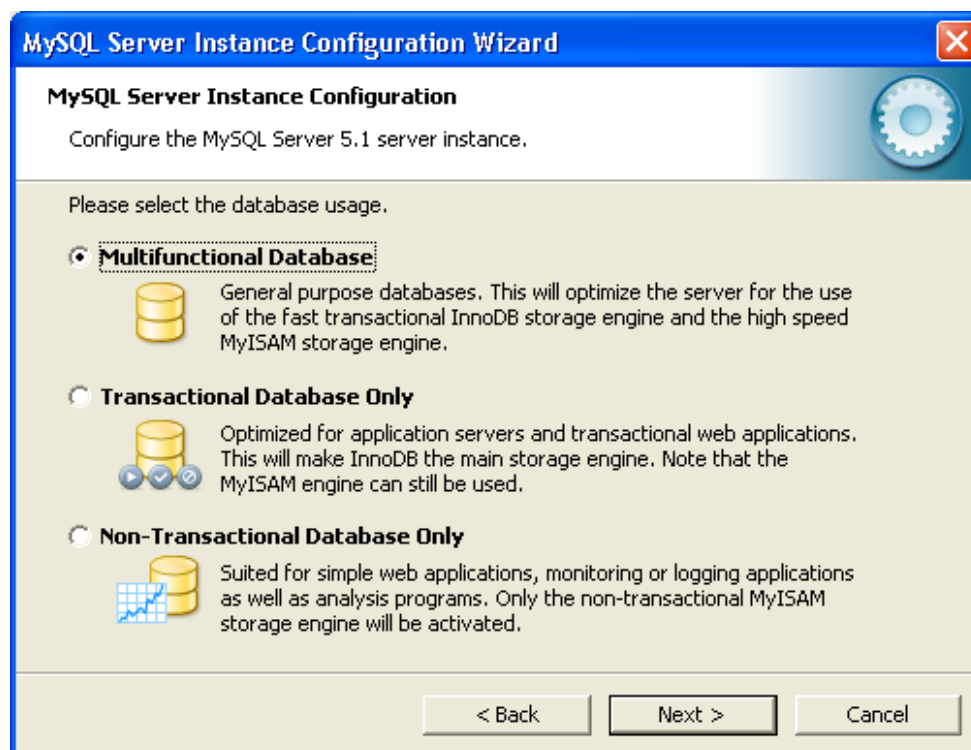
Εικόνα 2.2.10. Εγκατάσταση - Ρυθμίσεις MySQL Server

11. Στο επόμενο παράθυρο, επιλέξτε την πρώτη περίπτωση **Developer Machine** και πατήστε Next.



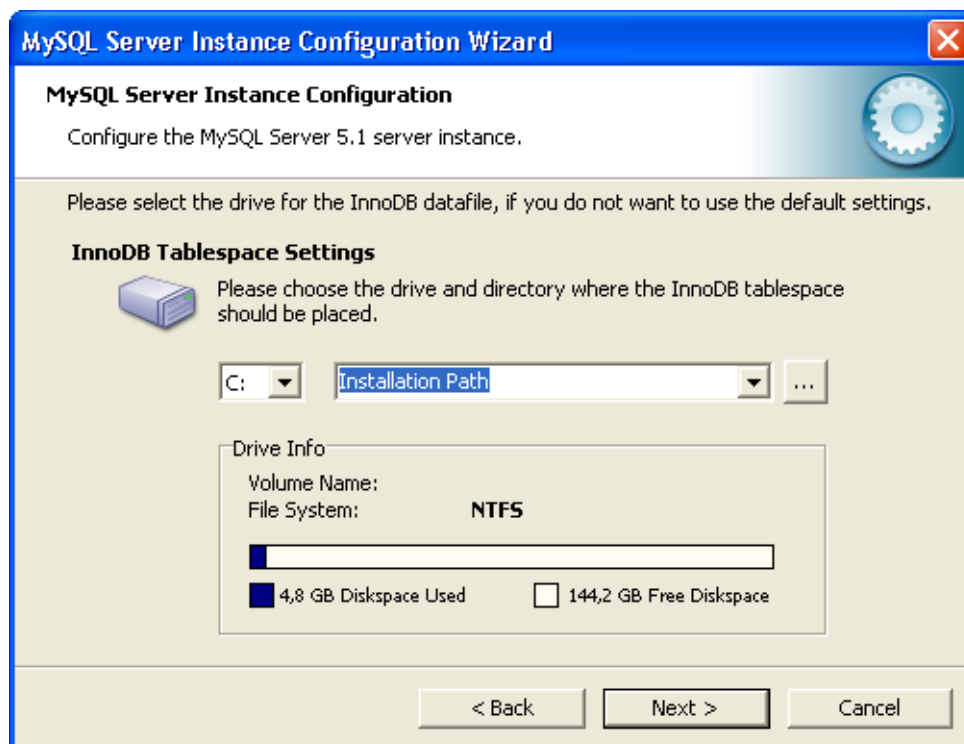
Εικόνα 2.2.11. Εγκατάσταση - Ρυθμίσεις MySQL Server

12. Τώρα, επιλέξτε **Multifunctional Database** και πατήστε Next.



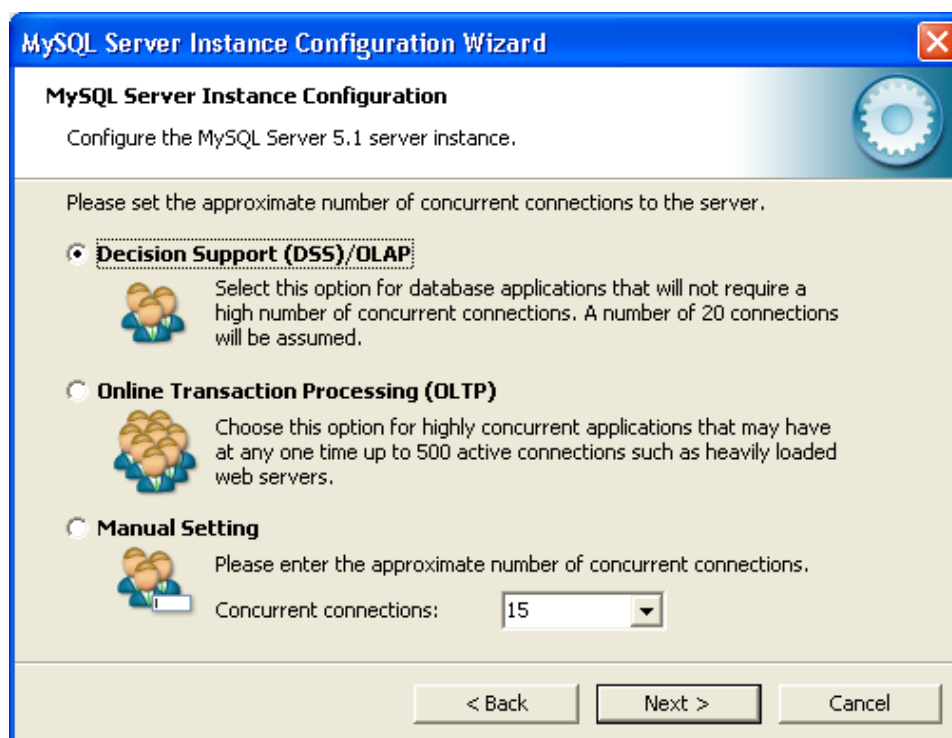
Εικόνα 2.2.12. Εγκατάσταση - Ρυθμίσεις MySQL Server

13. Σαν προορισμό αποθήκευσης επιλέξτε τον "C:", ή τον δίσκο που έχετε εσείς εγκατεστημένο το λειτουργικό σας και "Installation Path". Πατήστε Next.



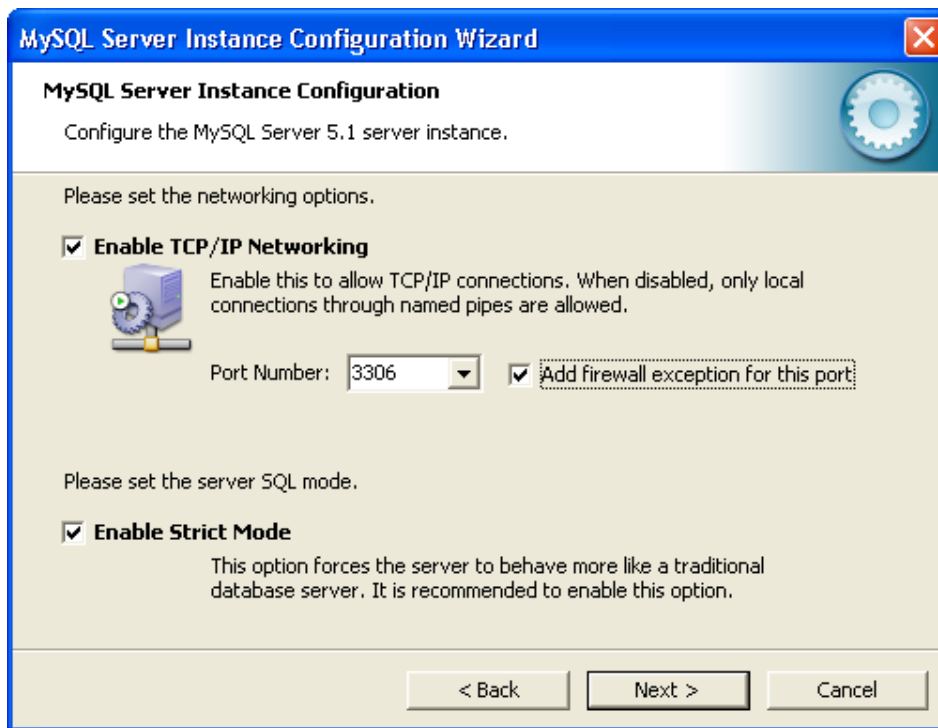
Εικόνα 2.2.13. Εγκατάσταση - Ρυθμίσεις MySQL Server

14. Στο επόμενο παράθυρο, επιλέξτε **Decision Support (DSS)/OLAP** και πατήστε Next.



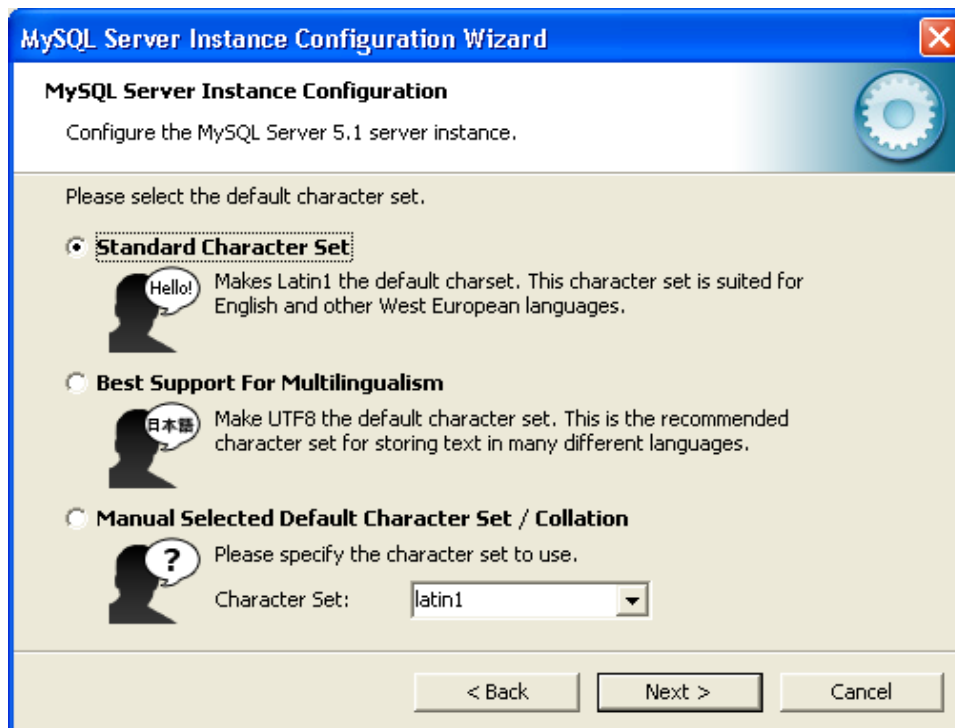
Εικόνα 2.2.14. Εγκατάσταση - Ρυθμίσεις MySQL Server

15. Στο επόμενο βήμα, τσεκάρετε και τις δυο επιλογές **Enable TCP/IP Networking** και **Enable Strict Model**. Μην ξεχάσετε να τσεκάρετε την εξαίρεση του firewall για την συγκεκριμένη θύρα. Πατήστε Next.



Εικόνα 2.2.15. Εγκατάσταση - Ρυθμίσεις MySQL Server

16. Τώρα, επιλέξτε **Standard Character Set** και πατήστε Next.



Εικόνα 2.2.16. Εγκατάσταση - Ρυθμίσεις MySQL Server

17. Στο παράθυρο που ακολουθεί, τσεκάρτε εκτός από την πρώτη περίπτωση που είναι ήδη επιλεγμένη, και την δεύτερη που τοποθετεί την μεταβλητή της MySQL μέσα στο PATH των Windows. Πατήστε Next.



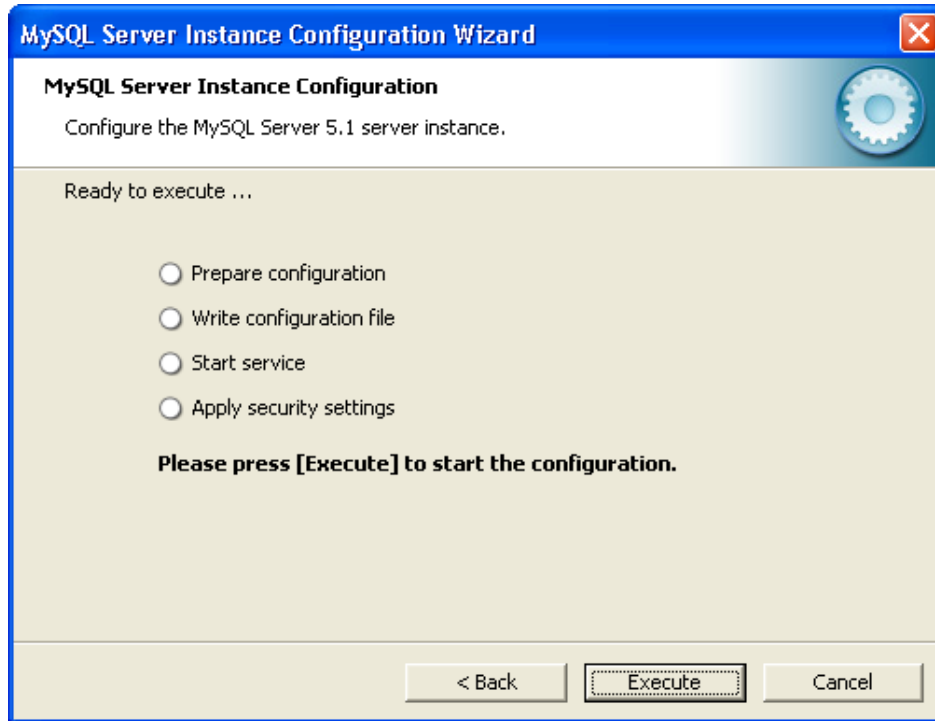
Εικόνα 2.2.17. Εγκατάσταση - Ρυθμίσεις MySQL Server

18. Τώρα, επιλέξτε **Modify Security Settings**, και για κωδικό του Server γράψτε “1234” στα δυο πεδία. Πατήστε Next.



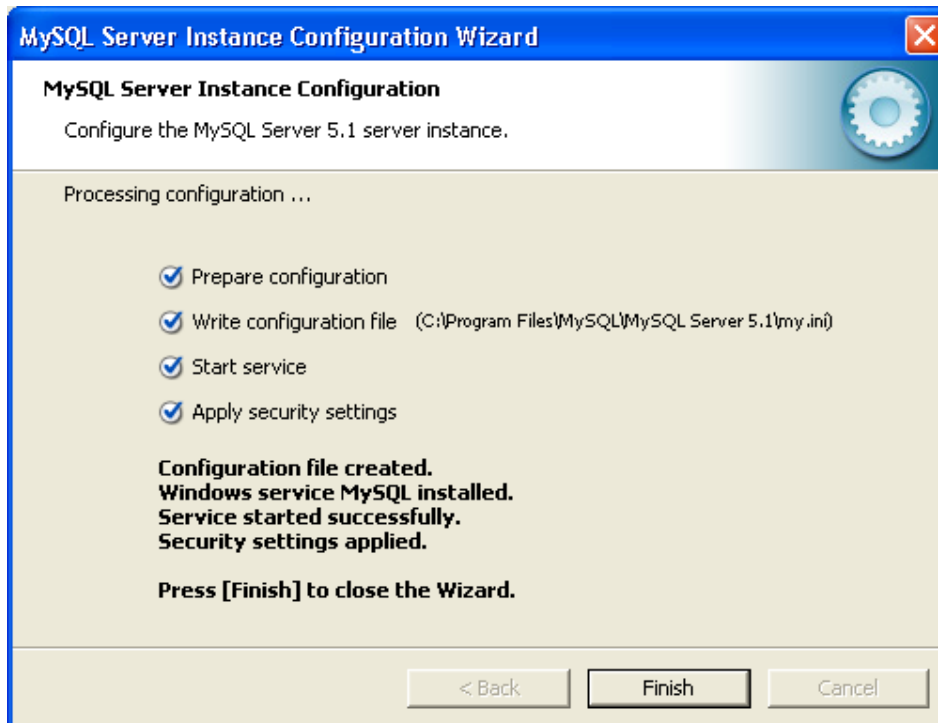
Εικόνα 2.2.18. Εγκατάσταση - Ρυθμίσεις MySQL Server

19. Οι επιλογές όλων των παραμέτρων έχουν ολοκληρωθεί, και είναι έτοιμες για εφαρμογή. Πατήστε Execute.



Εικόνα 2.2.19. Εγκατάσταση - Ρυθμίσεις MySQL Server

20. Όταν η εφαρμογή και αποθήκευση των ολοκληρωθούν, τότε θα εμφανιστεί το παρακάτω μήνυμα. Η MySQL είναι έτοιμη για χρήση. Πατήστε Finish για το κλείσιμο του παραθύρου.



Εικόνα 2.2.20. Εγκατάσταση - Ρυθμίσεις MySQL Server

2.3 – MySQL GUI Tools

2.3.1 – Εισαγωγή στα MySQL GUI Tools

Πρόκειται για μια συλλογή από εργαλεία(**MySQL Administrator**, **MySQL Query Browser** και **Migration Toolkit**) που το καθένα χωριστά μπορεί να χρησιμοποιηθεί για την σύνδεση, διαχείριση και χρήση ενός MySQL Server. Είναι δωρεάν διαθέσιμα για διάφορα λειτουργικά συστήματα

2.3.2 – Εγκατάσταση MySQL GUI Tools

1. Μπορείτε να κατεβάσετε το MySQL GUI Tools από την ιστοσελίδα της MySQL (<http://dev.mysql.com/downloads/gui-tools/5.0.html>). Βεβαιωθείτε ότι κατεβάζετε το σωστό τύπο αρχείου που αντιστοιχεί στο δικό σας σύστημα. Σε αυτή την περίπτωση, η χρήση γίνεται πάνω σε Windows. Βεβαιωθείτε ότι κατεβάζετε το αντίστοιχο αρχείο χωρίς Installer.

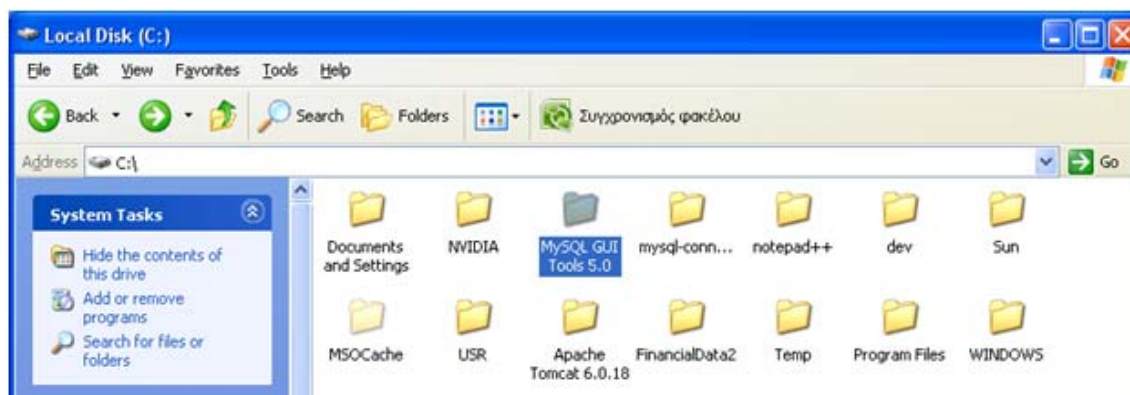
MySQL GUI Tools: Archived Downloads

Windows downloads
The install package uses the Windows Installer, which is built in to Windows XP and more recent Microsoft Windows versions.
Download an updated [Windows Installer for Windows 2000](#)

Windows (x86)	5.0-r17	16.9M	Download
MD5: a389ba3821bfb8497ebafab81ae8a7db			
Without installer (unzip in C:\)	5.0-r17	16.1M	Download
MD5: 54593f476f2fb19a2f6c5ac4a200299e			
Mac OSX downloads			
MAC OSX 10.4 (Universal binaries)	5.0-r12	12.4M	Download
MD5: 87989a094c5e4b6efb28cdad50dd1a44			

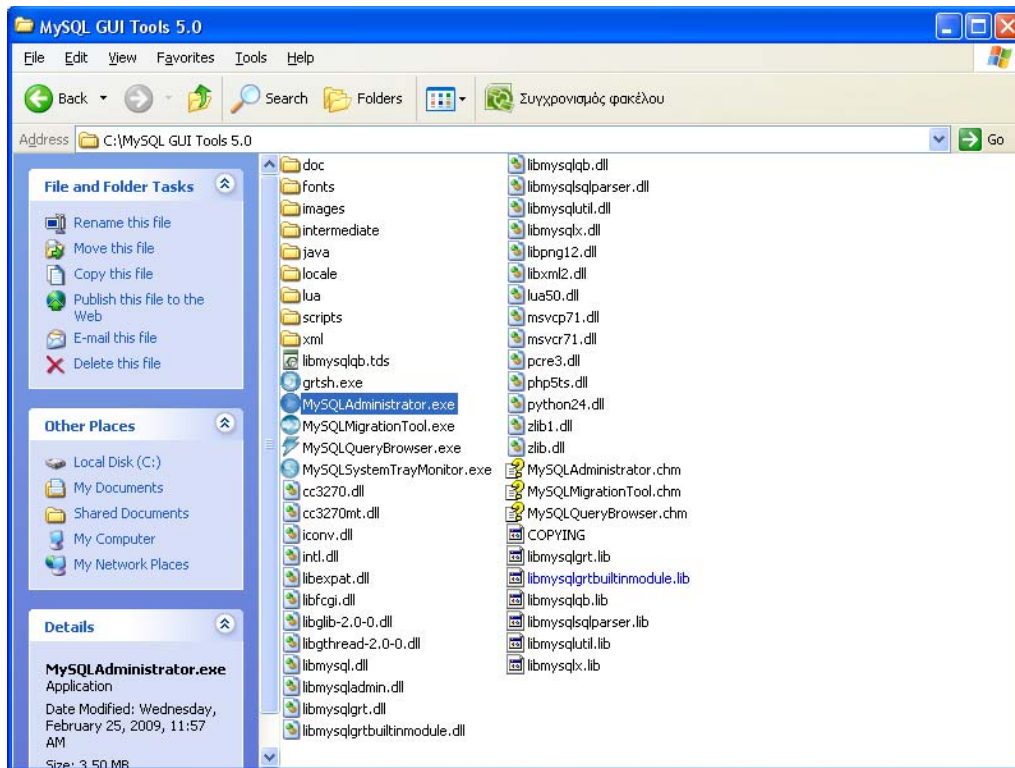
Εικόνα 2.3.1. Λήψη MySQL GUI Tools

2. Όταν η λήψη ολοκληρωθεί, ανοίξτε το **zip** αρχείο με κάποιο πρόγραμμα από-συμπίεσης. Μέσα βρίσκεται ένας φάκελος με την ονομασία **MySQL GUI Tools**. Κάντε τον από-συμπίεση και τοποθετήστε τον μέσα στον “C:\” ή σε όποιον δίσκο έχετε εσείς εγκατεστημένο το λειτουργικό σας σύστημα.



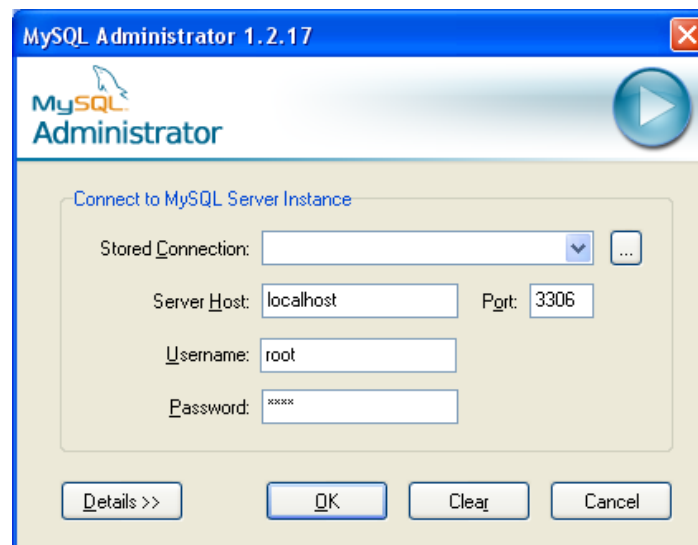
Εικόνα 2.3.2. Αποθήκευση MySQL GUI Tools

3. Ανοίξτε τον φάκελο. Μέσα υπάρχουν διάφορα αρχεία, όμως το πιο σημαντικό, αυτό δηλαδή που θα χρησιμοποιηθεί για την ευκολία δημιουργίας και τροποποίησης βάσεων δεδομένων και παραμέτρων είναι ο **MySQL Administrator**. Κάντε διπλό κλικ πάνω στο .exe αρχείο.



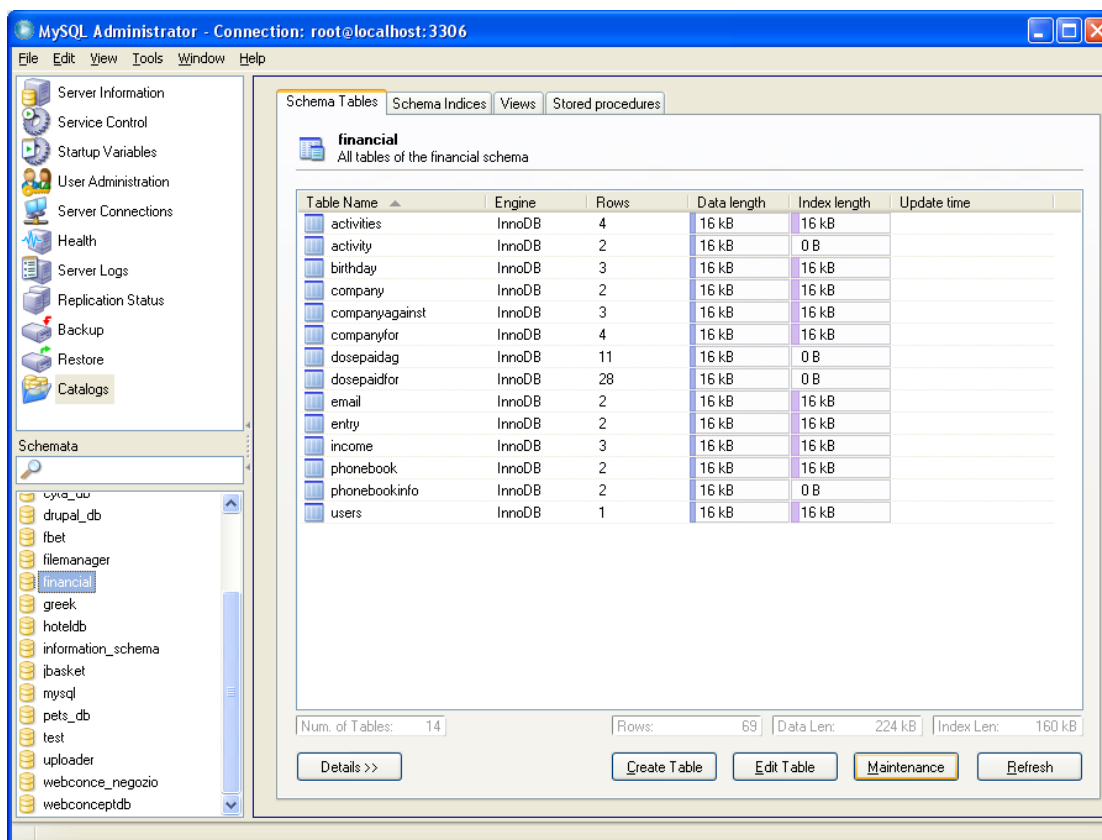
Εικόνα 2.3.3. Προβολή MySQL GUI Tools

4. Ένα νέο παράθυρο εμφανίζεται. Απαιτεί σωστά στοιχεία για την είσοδο στον MySQL Administrator. Το πεδίο Server Host συμπληρώστε με **localhost**. Αν κατά την εγκατάσταση του MySQL Server ακολουθήσατε τα βήματα και τους κωδικούς που δόθηκαν, τότε στο πεδίο Username συμπληρώστε **root** και στο Password **1234**, αλλιώς γράψτε του κωδικούς που εσείς δώσατε κατά την εγκατάσταση του MySQL Server.



Εικόνα 2.3.4. Είσοδος στο MySQL Administrator

5. Τώρα ο MySQL Administrator είναι έτοιμος για χρήση. Εδώ μπορείτε να δείτε όλες τις βάσεις δεδομένων που υπάρχουν στο σύστημα σας, μπορείτε να εισάγετε μία νέα βάση δεδομένων, να δημιουργήσετε αντίγραφα ασφαλείας(back-up) και βεβαίως, να τροποποιήσετε τις βάσεις, τους πίνακες τους αλλά και τις παραμέτρους τους καθώς και άλλες χρήσιμες υπηρεσίες.



Εικόνα 2.3.5. MySQL Administrator

2.4 – JDK 6 update 25

2.4.1 – Εισαγωγή στο JDK

Το **JDK** είναι ακρωνύμιο από τις λέξεις «Java Development Kit». Πρόκειται για όλη την πλατφόρμα της java όπως αυτή σχεδιάστηκε και υλοποιήθηκε από τη SUN. Περιέχει:


















- API classes(βιβλιοθήκη κώδικα ήδη υλοποιημένου Application Program Interface/Διεπαφή Προγραμματικών Εφαρμογών)
- Java compiler (μεταγλωττιστής)
- Java Virtual Machine interpreter (διερμηνευτής εικονικής μηχανής Java)

Χρησιμοποιείται για compile και το τρέξιμο των Java applications και των applets

- applets : εφαρμογές για υπολογιστές που τρέχουν μέσα από browsers, όπως ο Firefox.

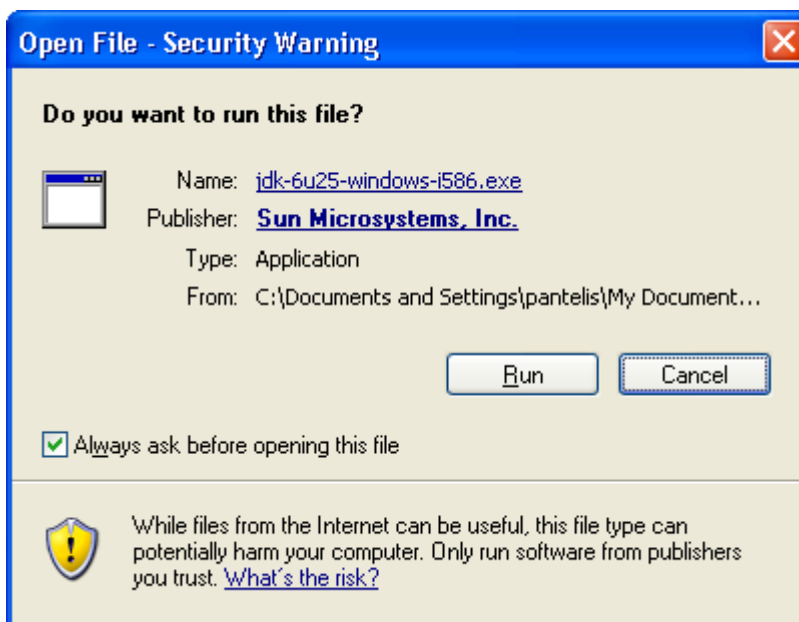
2.4.2 – Εγκατάσταση JDK 6

1. Πηγαίνετε στο Oracle.com για να κατεβάσετε την τελευταία έκδοση του JDK. Κατεβάστε από τις διαθέσιμες εκδόσεις αυτή που υποστηρίζει το σύστημα σας, Εμείς χρησιμοποιούμε Windows 32 bit .

Java SE Development Kit 6 Update 26		
Product / File Description	File Size	Download
Linux x86 - RPM Installer	76.93 MB	 jdk-6u26-linux-i586-rpm.bin
Linux x86 - Self Extracting Installer	81.20 MB	 jdk-6u26-linux-i586.bin
Linux Intel Itanium - RPM Installer	60.25 MB	 jdk-6u26-linux-ia64-rpm.bin
Linux Intel Itanium - Self Extracting Installer	67.92 MB	 jdk-6u26-linux-ia64.bin
Linux x64 - RPM Installer	77.15 MB	 jdk-6u26-linux-x64-rpm.bin
Linux x64 - Self Extracting Installer	81.45 MB	 jdk-6u26-linux-x64.bin
Solaris x86 - Self Extracting Binary	81.08 MB	 jdk-6u26-solaris-i586.sh
Solaris x86 - Packages - tar.Z	136.89 MB	 jdk-6u26-solaris-i586.tar.Z
Solaris SPARC - Self Extracting Binary	86.05 MB	 jdk-6u26-solaris-sparc.sh
Solaris SPARC - Packages - tar.Z	141.37 MB	 jdk-6u26-solaris-sparc.tar.Z
Solaris SPARC 64-bit - Self Extracting Binary	12.24 MB	 jdk-6u26-solaris-sparcv9.sh
Solaris SPARC 64-bit - Packages - tar.Z	15.58 MB	 jdk-6u26-solaris-sparcv9.tar.Z
Solaris x64 - Self Extracting Binary	8.50 MB	 jdk-6u26-solaris-x64.sh
Solaris x64 - Packages - tar.Z	12.24 MB	 jdk-6u26-solaris-x64.tar.Z
Windows x86	76.81 MB	 jdk-6u26-windows-i586.exe
Windows Intel Itanium	63.32 MB	 jdk-6u26-windows-ia64.exe
Windows x64	67.42 MB	 jdk-6u26-windows-x64.exe

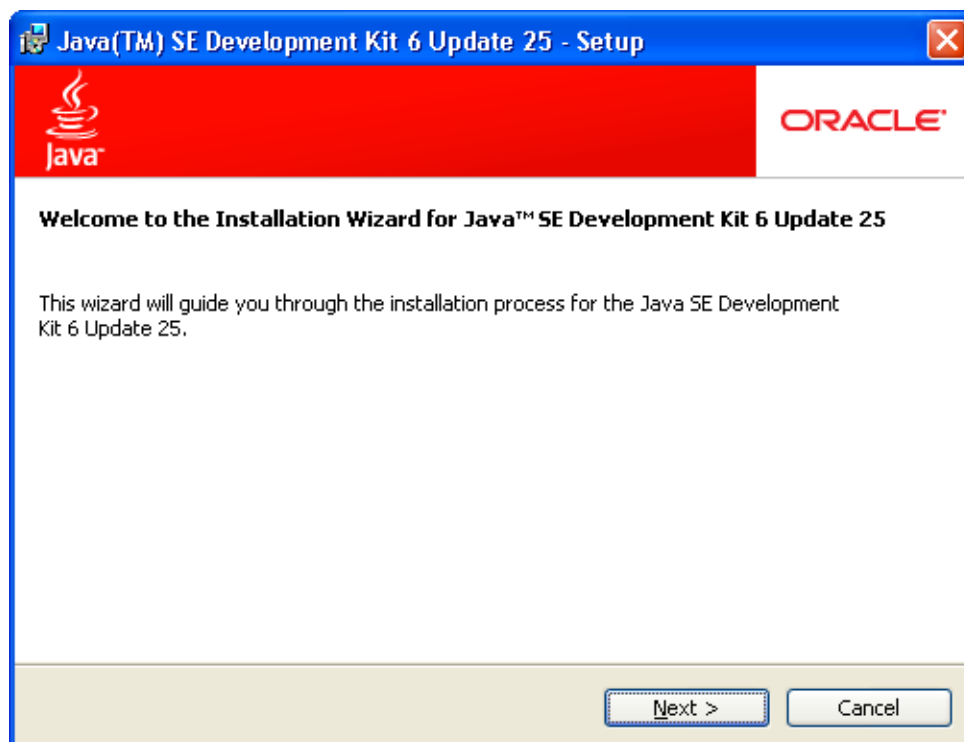
Εικόνα 2.4.1. Λήψη JDK

- Εφόσον πατήσετε διπλό κλικ στο αρχείο .exe που κατεβάσατε, ένα νέο παράθυρο εμφανίζεται για την επιβεβαίωση της εγκατάστασης του JDK. Επιλέξτε Run.



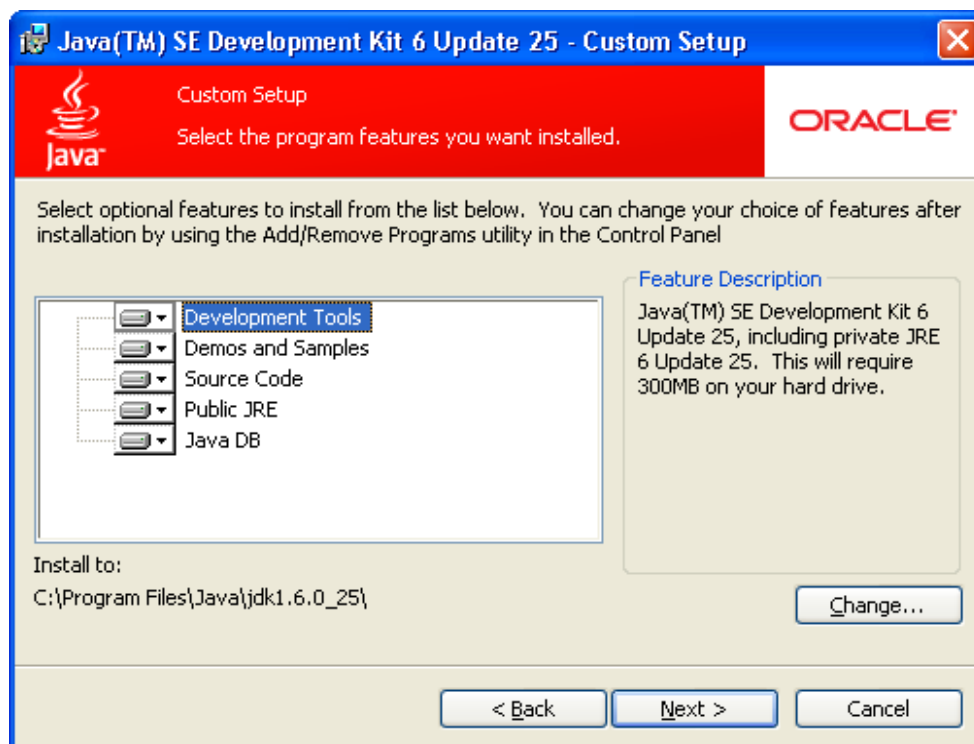
Εικόνα 2.4.2. Εγκατάσταση JDK

3. Ένα παράθυρο καλωσορίσματος εμφανίζεται. Πατήστε Next.



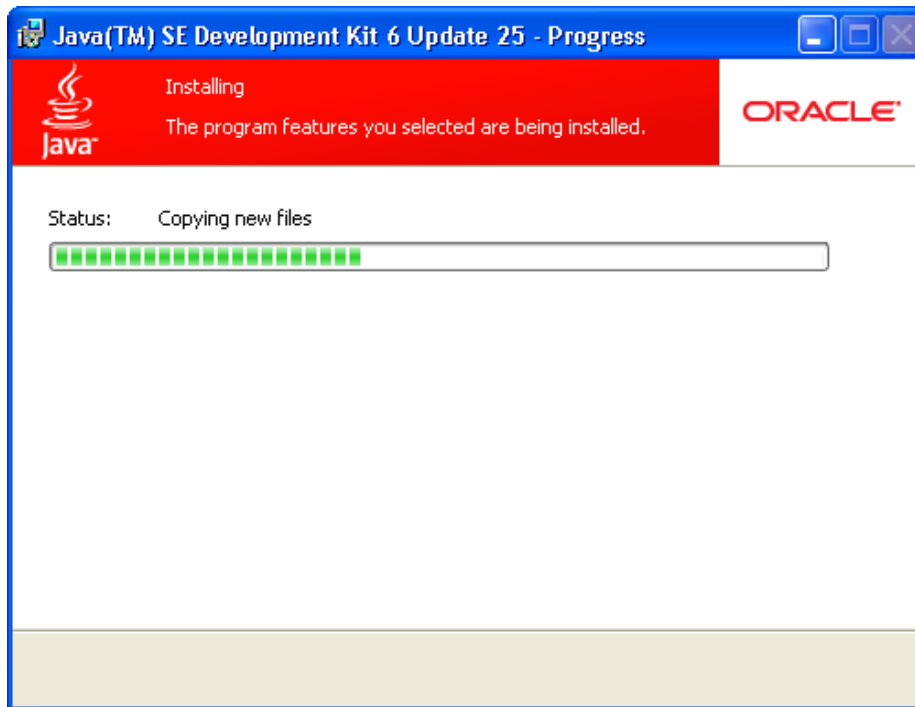
Εικόνα 2.4.3. Εγκατάσταση JDK

4. Στο επόμενο παράθυρο εμφανίζονται τα εργαλεία και οι υπηρεσίες που επιλέγετε να εγκαταστήσετε. Μην κάνετε καμία τροποποίηση. Πατήστε Next.



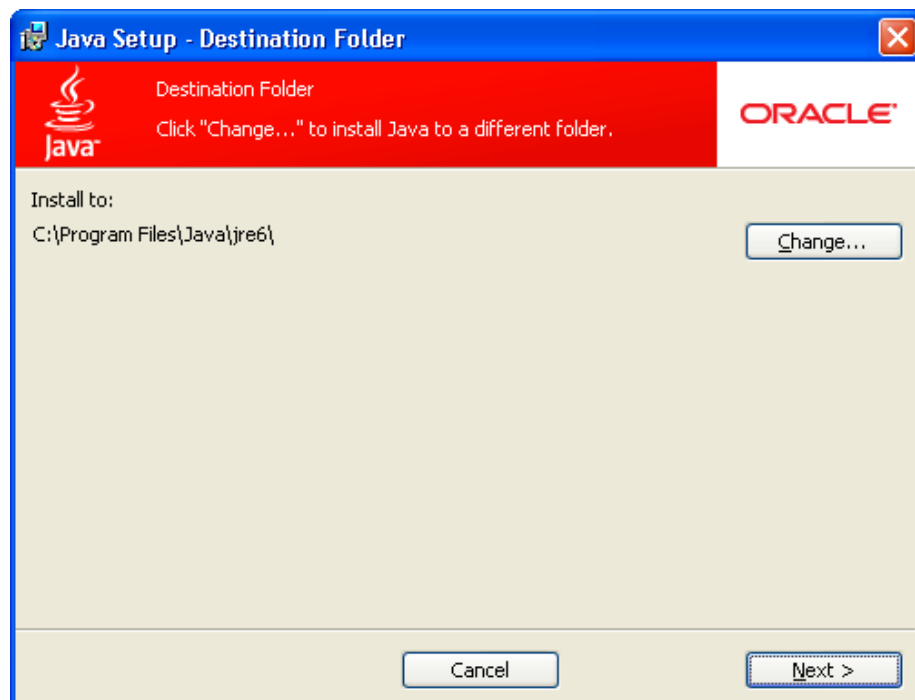
Εικόνα 2.4.4. Εγκατάσταση JDK

5. Περιμένετε λίγη ώρα μέχρι την ολοκλήρωση της εγκατάστασης.



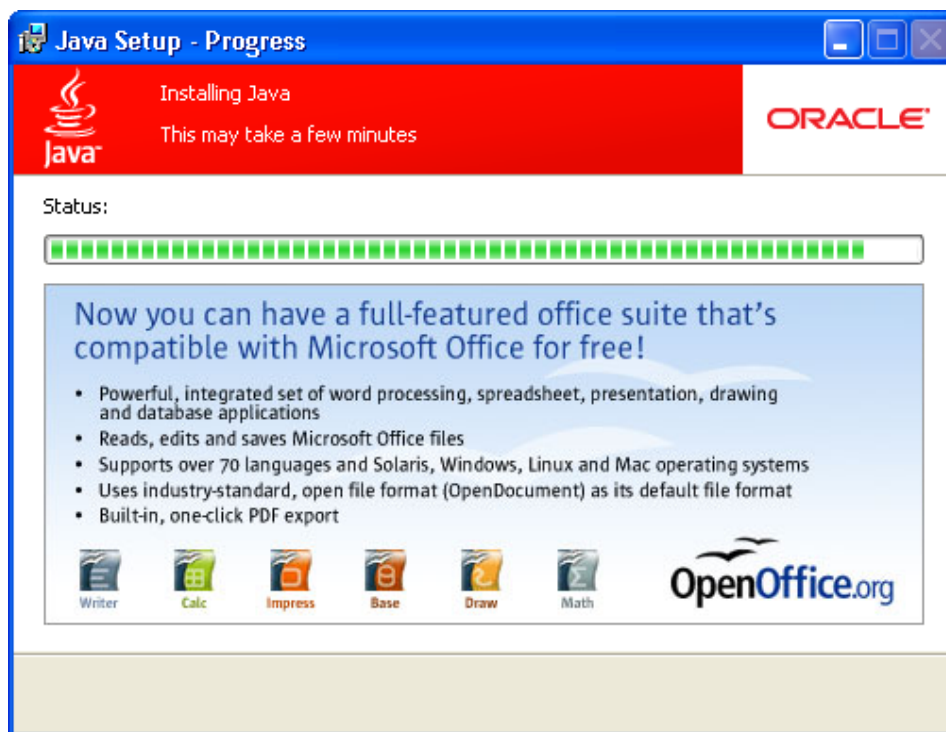
Εικόνα 2.4.5. Εγκατάσταση JDK

6. Αμέσως μετά την εγκατάσταση του JDK, ένα νέο παράθυρο εμφανίζεται για την εγκατάσταση του JRE, που είναι η έκδοση της JAVA. Αφήστε τον προεπιλεγμένο προορισμό αποθήκευσης. Πατήστε Next.



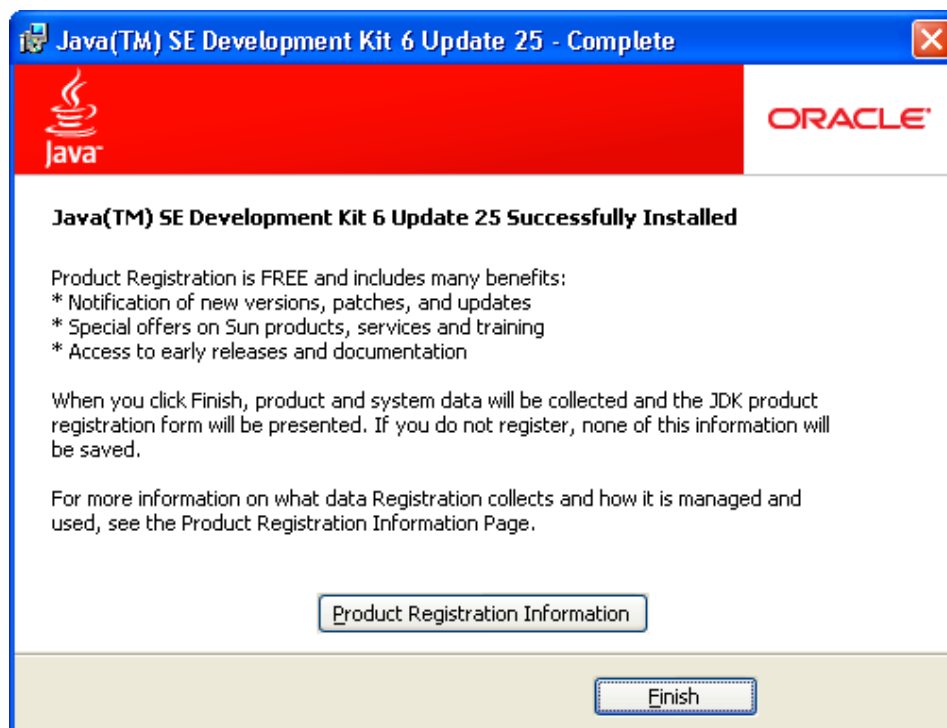
Εικόνα 2.4.6. Εγκατάσταση JDK

7. Η εγκατάσταση ξεκινάει. Περιμένετε μέχρι την ολοκλήρωση της εγκατάστασης της JAVA.



Εικόνα 2.4.7. Εγκατάσταση JDK

8. Η εγκατάσταση του JDK και της JAVA έχουν ολοκληρωθεί. Επιλέξτε Finish.



Εικόνα 2.4.8. Εγκατάσταση JDK

2.5 – Netbeans IDE 6.9.1

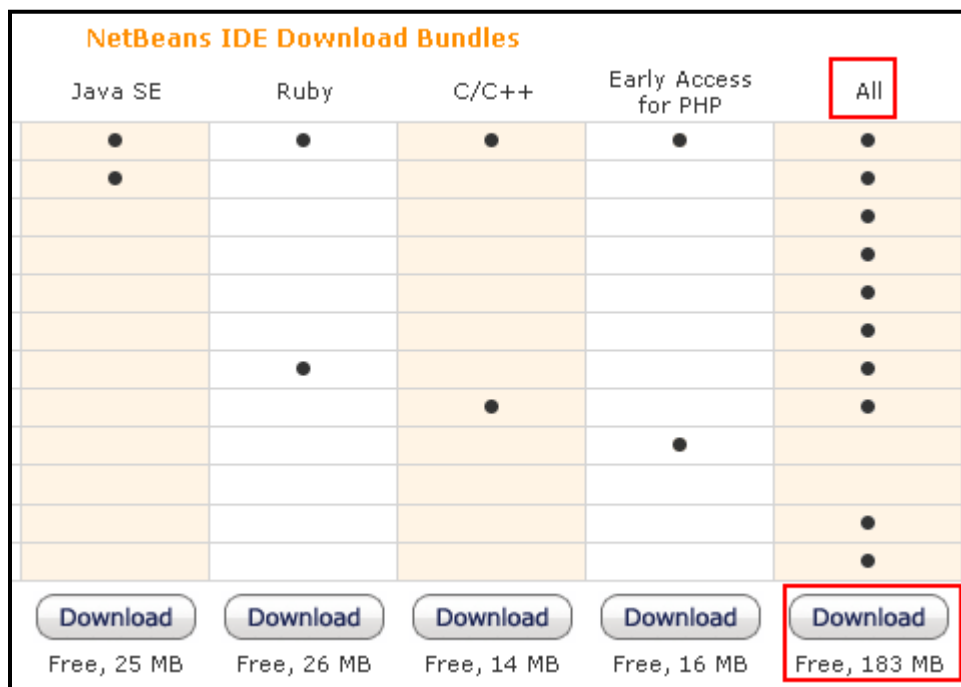
2.5.1 – Εισαγωγή στον Netbeans

Το εργαλείο NetBeans παρέχει μια πλατφόρμα για την ανάπτυξη διάφορων εφαρμογών που είτε θα χρησιμοποιηθούν στα πλαίσια κάποιου δικτύου είτε θα χρησιμοποιηθούν ως ανεξάρτητες (Stand-alone) εφαρμογές.

Το εργαλείο NetBeans υποστηρίζει διάφορες γλώσσες προγραμματισμού. (Java, C, C++, PHP, Python, JavaScript). Παρέχει ένα πλήρες περιβάλλον ανάπτυξης εφαρμογών (IDE: Integrated Development Environment). Το περιβάλλον ανάπτυξης γενικά παρέχει βοηθήματα στον προγραμματιστή τα οποία κάνουν ευκολότερη και αποδοτικότερη την ανάπτυξη των εφαρμογών. Επίσης τα βασικότερα από τα βοηθήματα αυτά αφορούν κυρίως τον Source Code Editor (συντακτικού στον κώδικα, live επισήμανση και live parsing, pop up παράθυρα βοήθειας). Ακόμη παρέχονται βοηθήματα για κατασκευή GUI άλλα και βοηθήματα διαχείρισης βάσεων δεδομένων

2.5.2 – Εγκατάσταση Netbeans IDE 6.9.1

1. Πηγαίνετε στο Netbeans.org για να κατεβάσετε την τελευταία έκδοση του NetBeans. Κατεβάστε την πλήρη έκδοση που είναι το πλήρες "πακέτο" πακέτο.

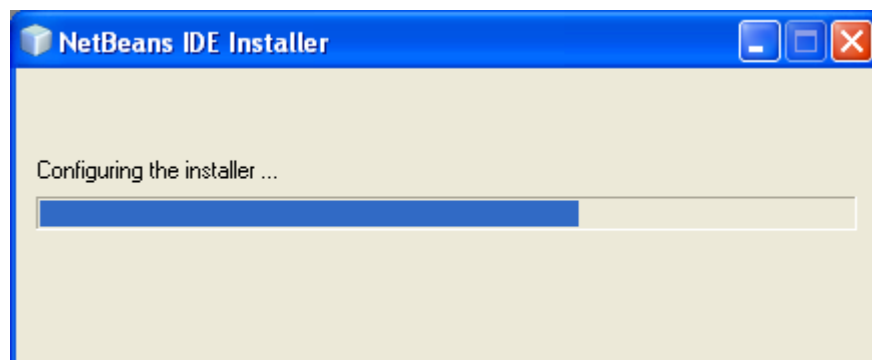


Εικόνα 2.5.1. Λήψη Netbeans

Μπορείτε να προσθέσετε ή να αφαιρέσετε πακέτα αργότερα χρησιμοποιώντας το IDE's Plugin Manager (Tools> Plugins).

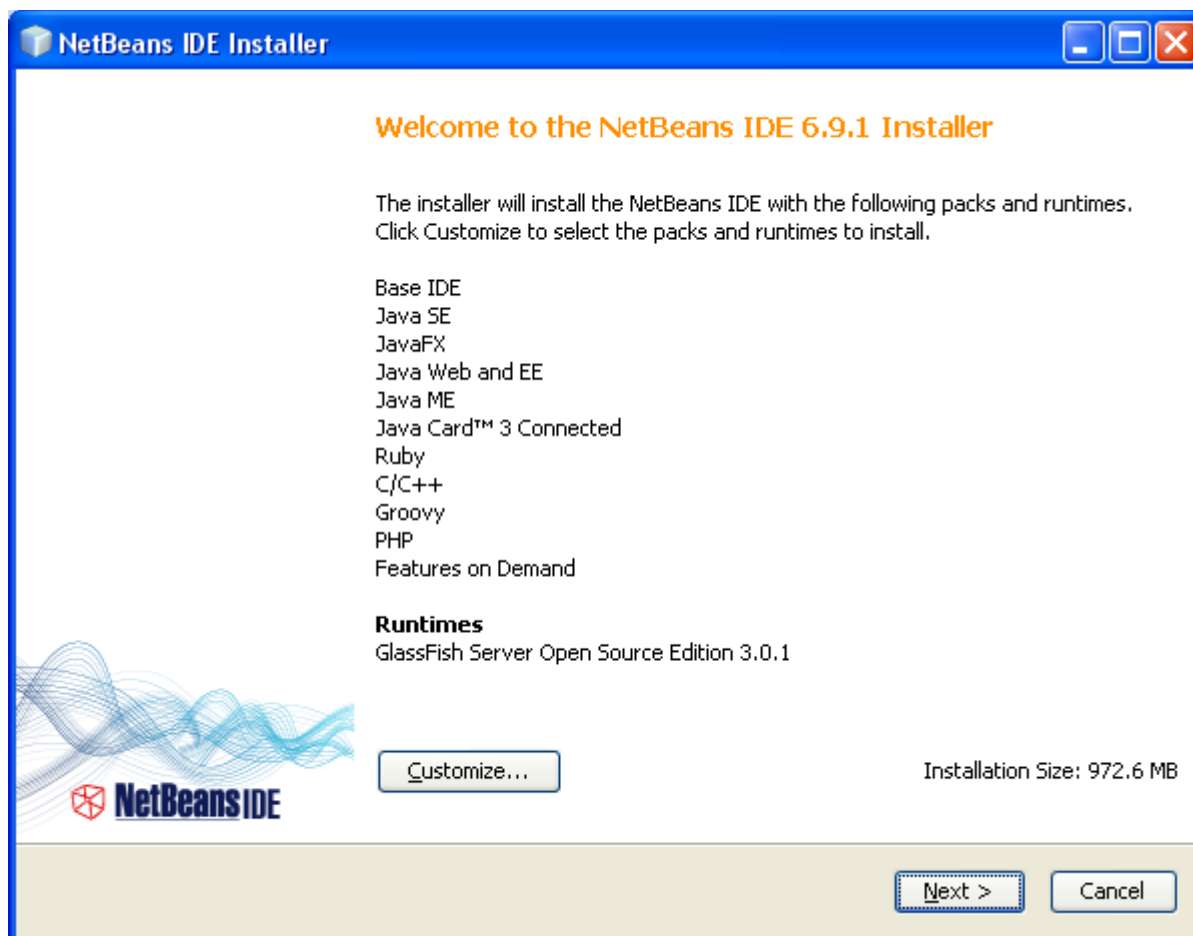
2. Κάντε διπλό κλικ στο .exe αρχείο αυτόματης εξαγωγής.

3. Το NetBeans IDE πρόγραμμα εγκατάστασης θα ξεκινήσει.



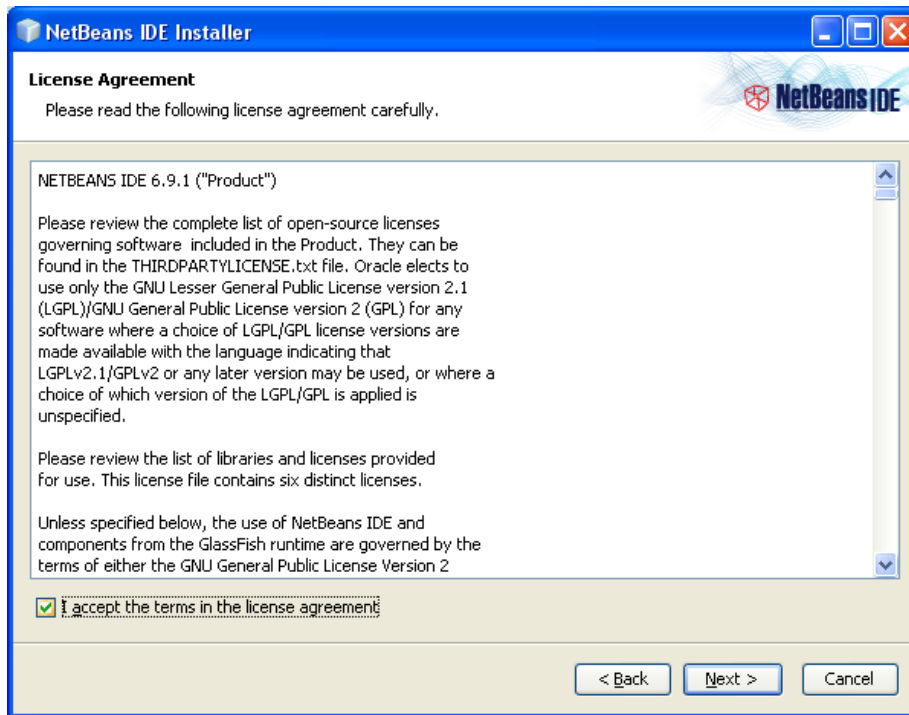
Εικόνα 2.5.2. Φόρτωση Εγκατάστασης Netbeans

4. Στη συνέχεια, εμφανίζεται η σελίδα υποδοχής του NetBeans IDE installer wizard. Μια λίστα των default πακέτων εμφανίζονται στην σελίδα υποδοχής. Κάντε κλικ στο κουμπί Next.



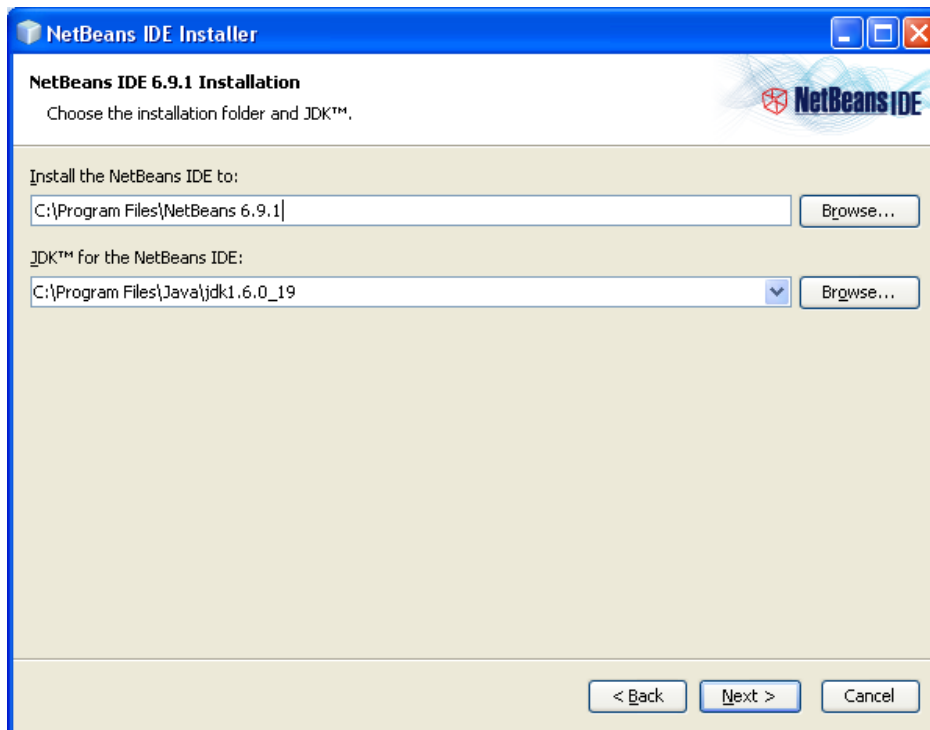
Εικόνα 2.5.3. Εγκατάσταση Netbeans

5. Αποδεχτείτε την Άδεια Χρήσης και πατήστε Next.



Εικόνα 2.5.4. Εγκατάσταση Netbeans

6. Επιλέξτε τη διαδρομή εγκατάστασης. Κάντε κλικ στο κουμπί Browse, εάν θέλετε να αλλάξετε. Η Java JDK έχει ρυθμιστεί στην επιθυμητή διαδρομή για αυτό το μηχάνημα. Στην περίπτωση μας δεχόμαστε τις προτεινόμενες διαδρομές. Κάντε κλικ στο κουμπί Next.



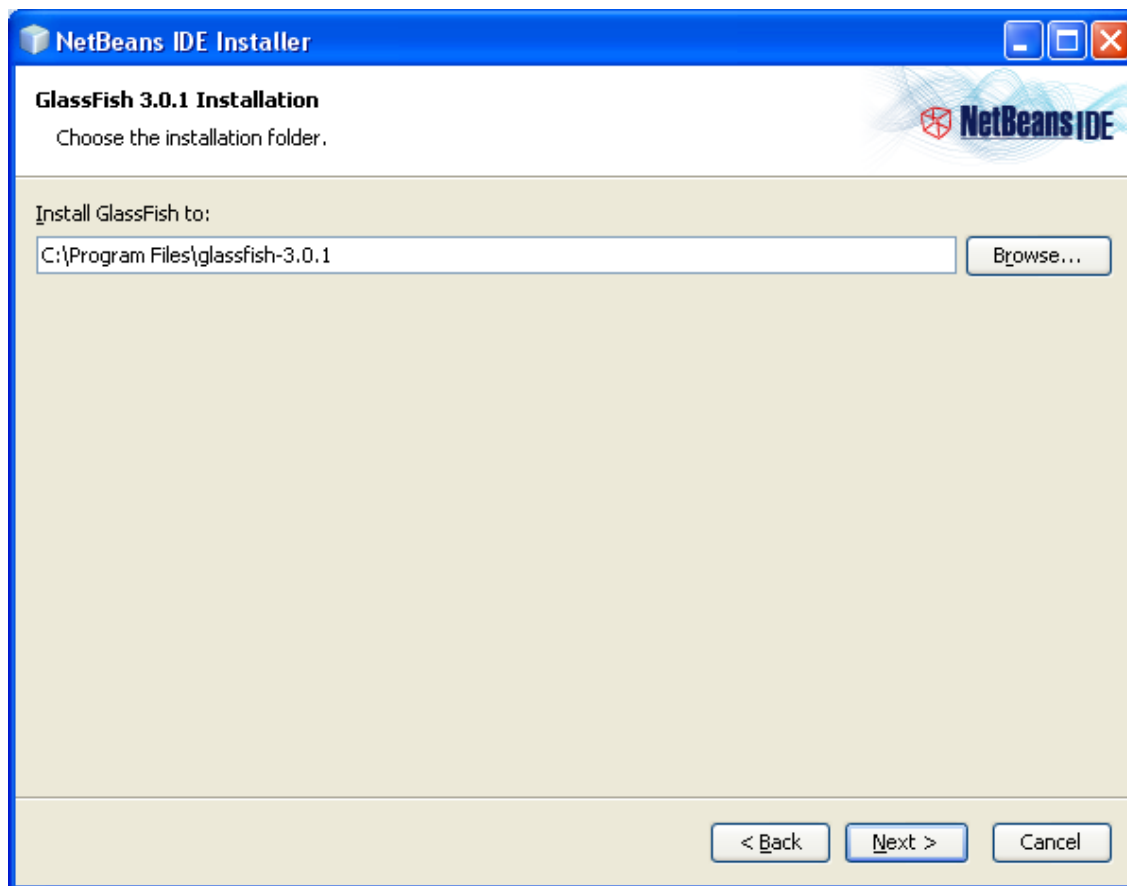
Εικόνα 2.5.5. Εγκατάσταση Netbeans

7. Αν το ακόλουθο Windows Security Alert μήνυμα εμφανιστεί, κάντε κλικ στο κουμπί Απεμπλοκή.



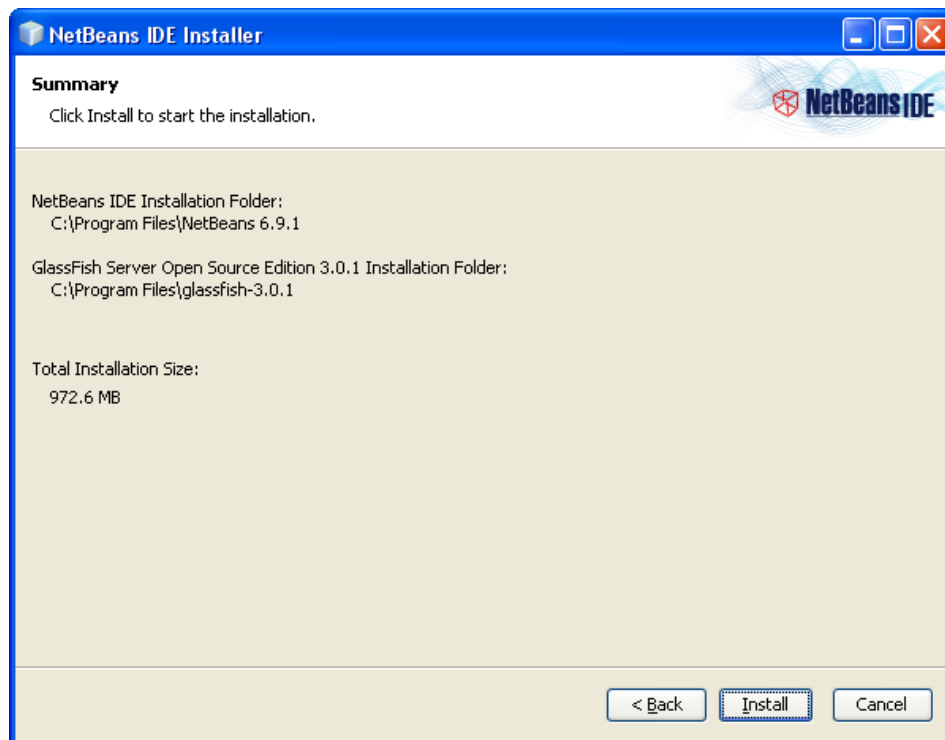
Εικόνα 2.5.6. Εγκατάσταση Netbeans

8. Επόμενη, είναι η επιλογή διαδρομής εγκατάστασης του GlassFish web server. Εμείς, απλά αποδεχόμαστε την δοσμένη προεπιλεγμένη διαδρομή. Κάντε κλικ στο κουμπί Next.



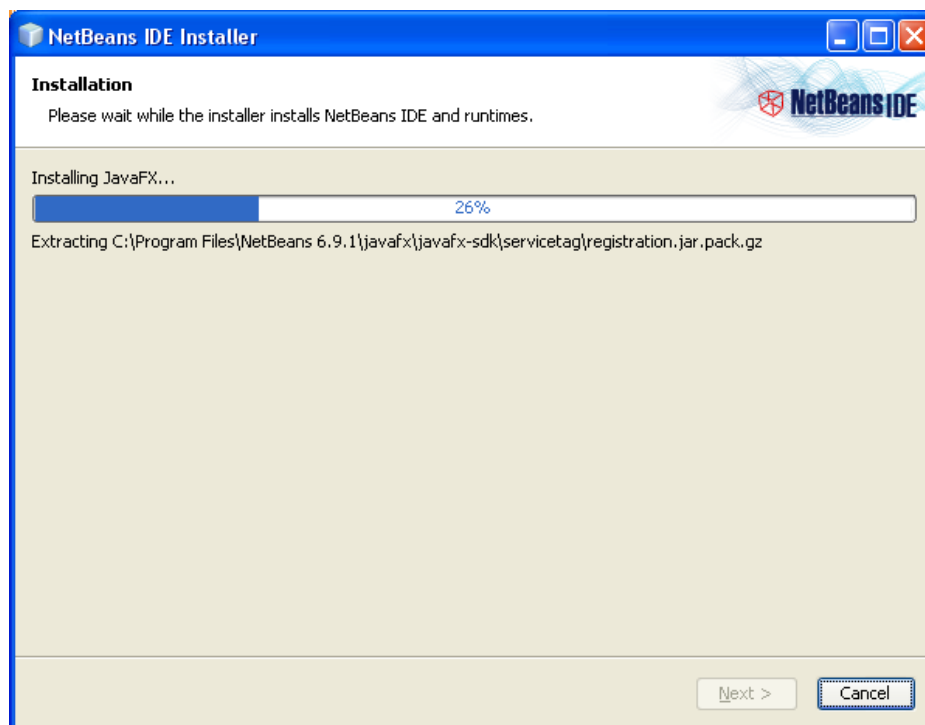
Εικόνα 2.5.7. Εγκατάσταση Netbeans

9. Επόμενη, είναι ολική εγκατάσταση του NetBeans IDE. Αν θέλετε να αλλάξετε τις ήδη επιλεγμένες ρυθμίσεις, κάντε κλικ στο κουμπί Back, αλλιώς κάντε κλικ στο κουμπί Install



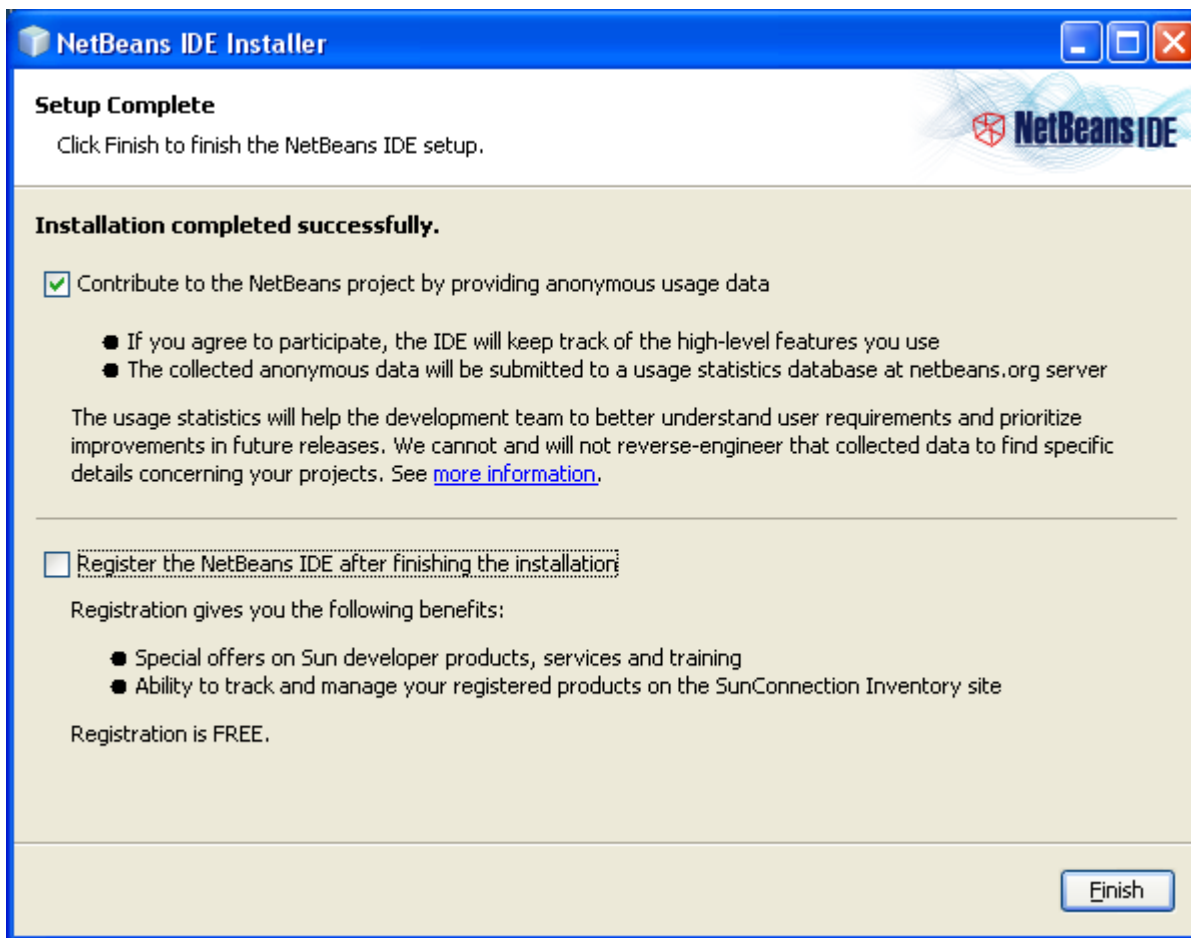
Εικόνα 2.5.8. Εγκατάσταση Netbeans

10. Η εγκατάσταση ξεκινάει. Περιμένετε μέχρι την ολοκλήρωση της εγκατάστασης.



Εικόνα 2.5.9. Εγκατάσταση Netbeans

11. Αν το μήνυμα του βήματος 8 εμφανιστεί ξανά, κάντε κλικ στο κουμπί Απεμπλοκή.
12. Η εγκατάσταση ολοκληρώθηκε με επιτυχία, όπως φαίνεται στο παρακάτω σχήμα. Απενεργοποιήστε την δεύτερη επιλογή, δεν είναι απαραίτητο να εγγραφείτε στην ιστοσελίδα του **Netbeans** για την χρήση του. Κάντε κλικ στο κουμπί Finish.



Εικόνα 2.5.10. Εγκατάσταση Netbeans

13. Η ανάλογη συντόμευση έχει προστεθεί στο μενού Έναρξη των Windows και την επιφάνεια εργασίας. Κάντε διπλό κλικ στο εικονίδιο στην επιφάνεια εργασίας σας ή κάντε κλικ στο μενού Έναρξη.

2.6 – Apache Tomcat 6.0

2.6.1 – Εισαγωγή στον Apache Tomcat

Οι ιστοσελίδες για να είναι διαθέσιμες στο διαδίκτυο πρέπει να φορτωθούν σε έναν διακομιστή. Στην δική μας εργασία χρησιμοποιήθηκε ο Apache Tomcat.

Ο Apache Tomcat (ή **Jakarta Tomcat** ή απλά **Tomcat**) είναι λογισμικό ανοιχτού κώδικα (ελεύθερο λογισμικό) που αναπτύχθηκε από την Apache Software Foundation (ASF). Ο Tomcat αναπτύσσει προδιαγραφές για Java Servlet και το JavaServer Pages (JSP) από την Sun Microsystems, και παρέχει ένα "καθαρά Java" HTTP web server περιβάλλον για τρέξιμο κώδικα Java.

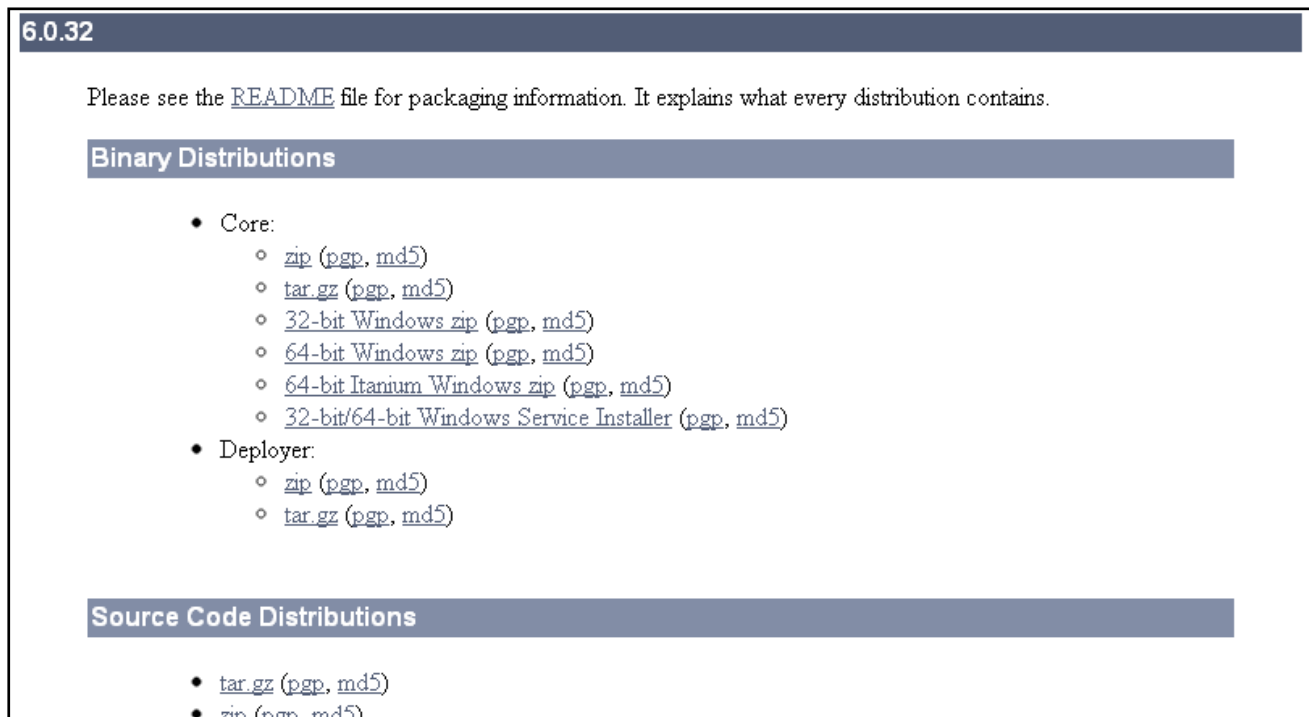
Ο Tomcat δεν πρέπει να συγχέεται με τον web server Apache, ο οποίος είναι μια εκτέλεση C ενός HTTP διακομιστή. Αυτοί οι δυο web servers δεν διατίθενται μαζί. Ο Apache Tomcat περιλαμβάνει εργαλεία για τη διαμόρφωση και τη διαχείριση, αλλά μπορεί επίσης να ρυθμιστεί με επεξεργασία XML αρχείων.

Με το εργαλείο παρέχεται η δυνατότητα:

- ρύθμισης του διαχειριστή ασφάλειας της Java
- χρησιμοποίησης πηγών δεδομένων μέσω JDBC
- υποστήριξης του πρωτοκόλλου SSL για κρυπτογραφημένες συνδέσεις
- χρήσης διαμεσολαβητών (proxy)
- εικονικής λειτουργίας (virtual hosting)
- εκτέλεσης σεναρίων CGI

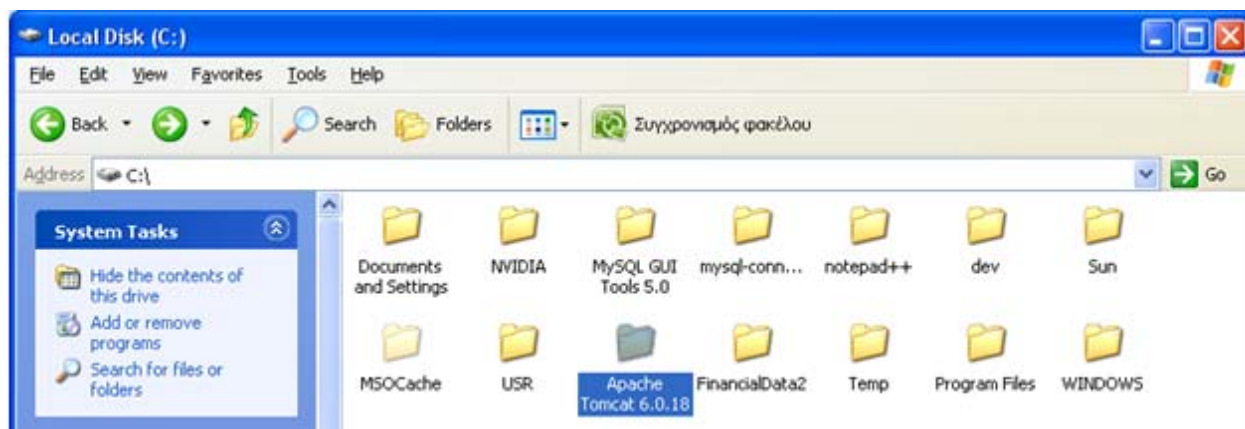
2.6.2 – Εγκατάσταση Apache Tomcat 6.0

1. Πηγαίνετε στο <http://tomcat.apache.org/> για να κατεβάσετε τον Apache Tomcat 6.0 (<http://tomcat.apache.org/download-60.cgi>) που χρησιμοποιείται, ή όποια έκδοση επιθυμείτε εσείς. Διαλέξτε το αρχείο **zip** που υποστηρίζει το σύστημα σας από την κατηγορία **Binary Distributions/Core**.



Εικόνα 2.6.1. Λήψη Apache Tomcat

2. Ο συγκεκριμένος φάκελος δεν περιέχει κάποιο εκτελέσιμο αρχείο εγκατάστασης και η εκκίνηση της υπηρεσίας του server είναι πολύπλοκη. Γι' αυτό, προσωρινά, αποθηκεύστε ολόκληρο τον φάκελο που περιέχει το **zip**(apache-tomcat-6.0.32) μέσα στον τοπικό δίσκο (C:\).



Εικόνα 2.6.2. Αποθήκευση Apache Tomcat

2.7 – MySQL Connector/J

2.7.1 – Εισαγωγή στον MySQL Connector/J

Το MySQL Connector/J αποτελεί εφαρμογή της Sun JDBC 3.0 API για την MySQL σχεσιακή βάση δεδομένων του server. Προσπαθεί να συμμορφωθεί όσο το δυνατόν περισσότερο στην JDBC API όπως καθορίζεται από την JavaSoft. Με πιο απλά λόγια το MySQL Connector/J χρησιμοποιείται σαν κάποιος πρόσθετος οδηγός που συνδέει την γλώσσα JAVA αλλά και προγράμματα που την υποστηρίζουν, με την MySQL. Είναι γνωστό ότι λειτουργεί με πολλά άλλα προϊόντα άλλων κατασκευαστών, μεταξύ των οποίων:

Διακομιστές Εφαρμογών(Application Servers)

- Apache Tomcat [<http://jakarta.apache.org/>]
- JBoss [<http://www.jboss.org/>]
- Weblogic [<http://www.beasys.com/>]
- IBM WebSphere [<http://www-3.ibm.com/software/info1/websphere/>]

Εργαλεία Σχεσιακής Χαρτογράφησης Αντικειμένων(Object Relational Mapping Tools)

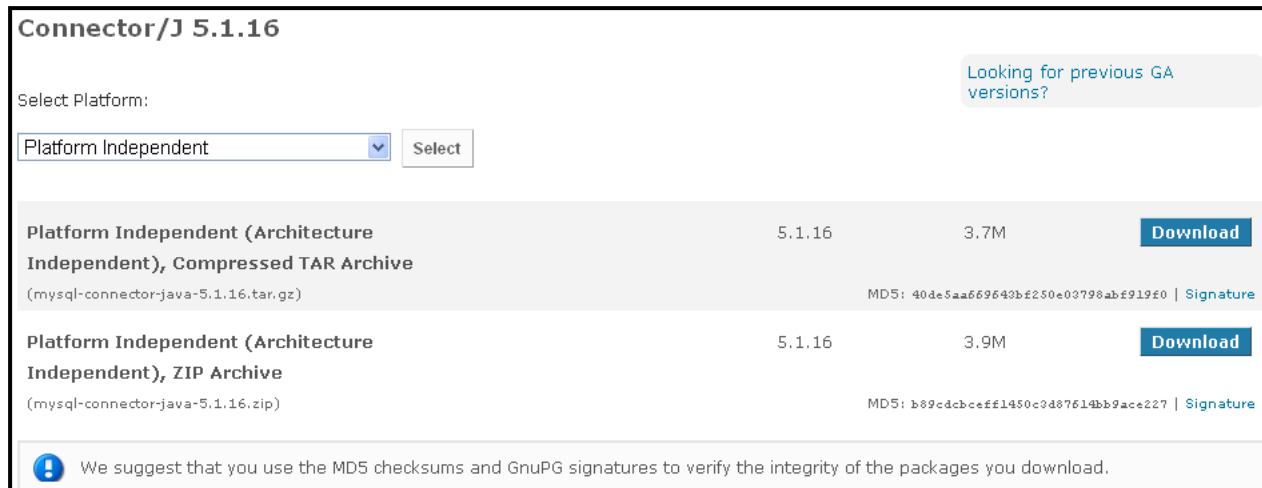
- Hibernate [<http://hibernate.sourceforge.net/>]
- Apache ObjectRelationalBridge [<http://db.apache.org/ojb/>]
- CocoBase [<http://www.thoughtinc.com/>]
- Kodo [<http://www.solarmetric.com/>]

Περιβάλλοντα Ανάπτυξης(Development Environments)

- Eclipse [<http://www.eclipse.org/>]
- Borland JBuilder [<http://www.borland.com/jbuilder/>]
- IBM WebSphere Studio [<http://www-3.ibm.com/software/awdtools/studioappdev/>]

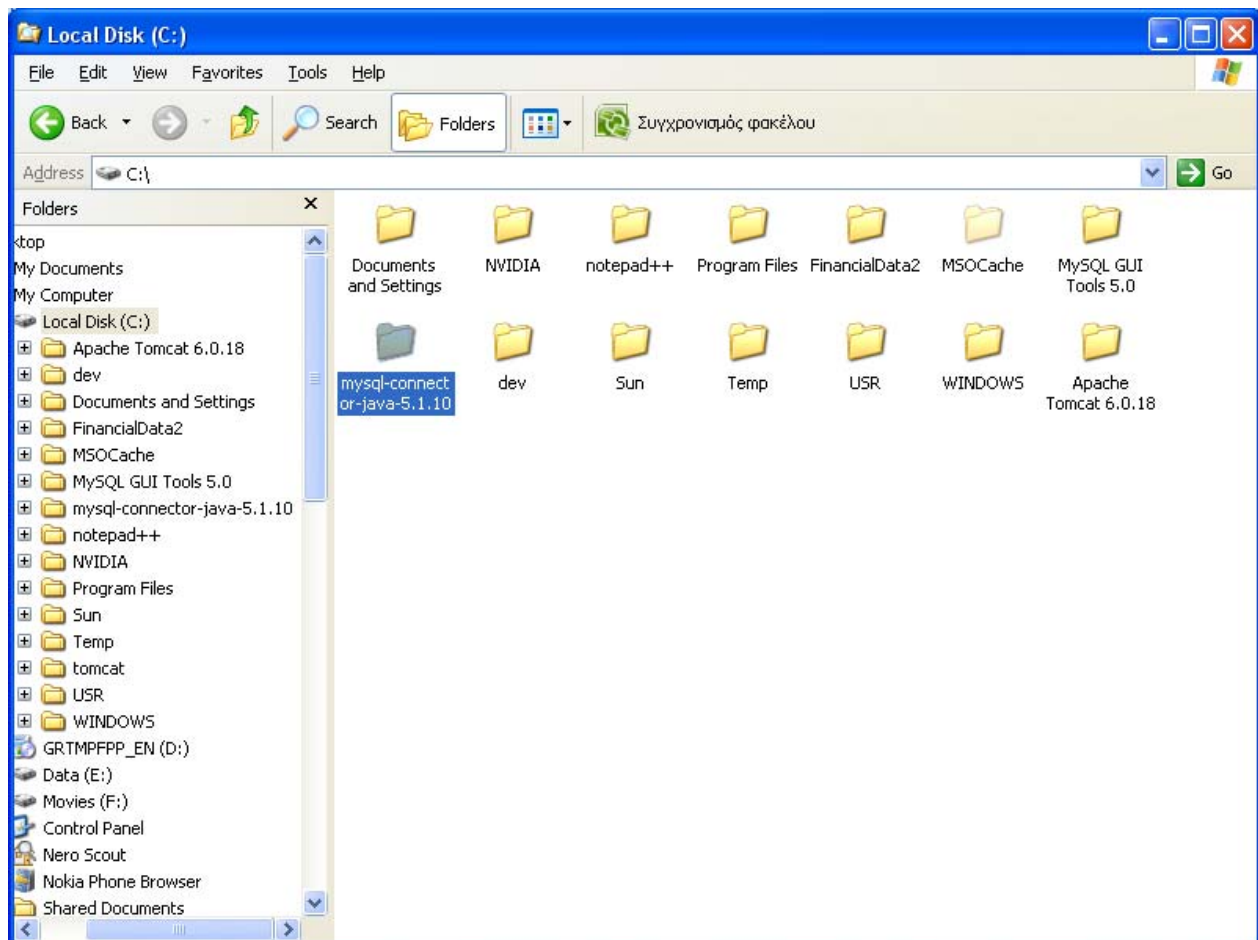
2.7.2 – Εγκατάσταση MySQL Connector/J

1. Πηγαίνετε στο <http://www.mysql.com/> για να κατεβάσετε το MySQL Connector/J 5.1.10 (<http://www.mysql.com/downloads/connector/j/>) που χρησιμοποιείται, ή όποια έκδοση επιθυμείτε εσείς. Διαλέξτε το αρχείο zip για να κατεβάσετε.



Εικόνα 2.7.1. Λήψη MySQL Connector/J

2. Ο συγκεκριμένος φάκελος δεν περιέχει κάποιο εκτελέσιμο αρχείο εγκατάστασης. Το MySQL Connector/J δεν είναι εφαρμογή εγκατάστασης, αλλά χρησιμοποιείται σαν κάποιο **plugin** που χρειάζονται διάφορα προγράμματα, όπως το Netbeans στην προκειμένη περίπτωση. Γι' αυτό, προσωρινά, αποθηκεύστε ολόκληρο τον φάκελο που περιέχει το **zip(mysql-connector-java-5.1.10)** μέσα στον τοπικό δίσκο (C:\).



Εικόνα 2.7.2. Αποθήκευση MySQL Connector/J

ΚΕΦΑΛΑΙΟ 3

Κατασκευή και Ανάπτυξη Web Application με την χρήση του Netbeans

3.1 – Εισαγωγή

Το κεφάλαιο αυτό προσφέρει μια πρώτη γνωριμία με το βασικό πρόγραμμα που χρησιμοποιήθηκε για την υλοποίηση της ψηφιακής εφαρμογής, το **Netbeans 9.0**. Στα παρακάτω, δίνονται τα βήματα για την δημιουργία όλων των αρχείων και όλων των προσθέτων που χρησιμοποιούνται, για την καθοδήγηση του προγραμματιστή και την αποφυγή λαθών μέσα στις πολλαπλές επιλογές που προσφέρει το πρόγραμμα αυτό. Τα αρχεία που χρησιμοποιήθηκαν στο συγκεκριμένο web application είναι , αρχεία jsp, αρχεία servlet και java, αρχεία css, javascript. Επίσης, υποδεικνύεται ο τρόπος σύνδεσης του Netbeans με τα **Mysql Connector/J** και **Apache Tomcat Server**.

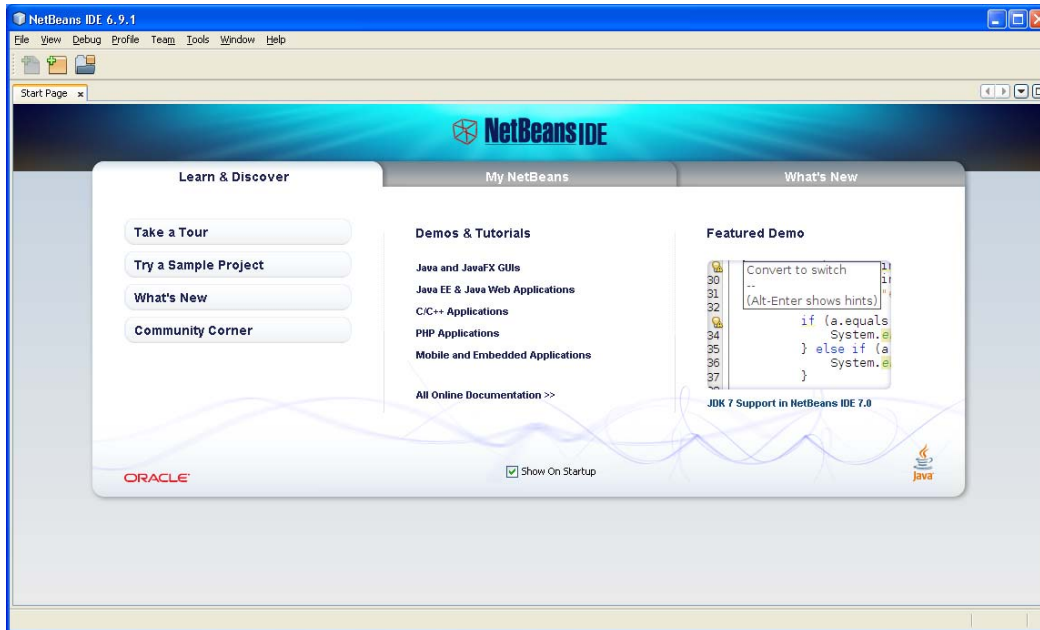
3.2 – Εξοικείωση με τον Netbeans IDE 6.9.1

1. Η παρακάτω εικόνα δείχνει την οθόνη έναρξης του NetBeans IDE . Ο χρόνος φόρτωσης είναι ανάλογος της επίδοσης του υπολογιστή σας.



Εικόνα 3.2.1. Εκκίνηση Netbeans IDE 6.9.1

2. Το παρακάτω είναι το NetBeans IDE 6.9.1.

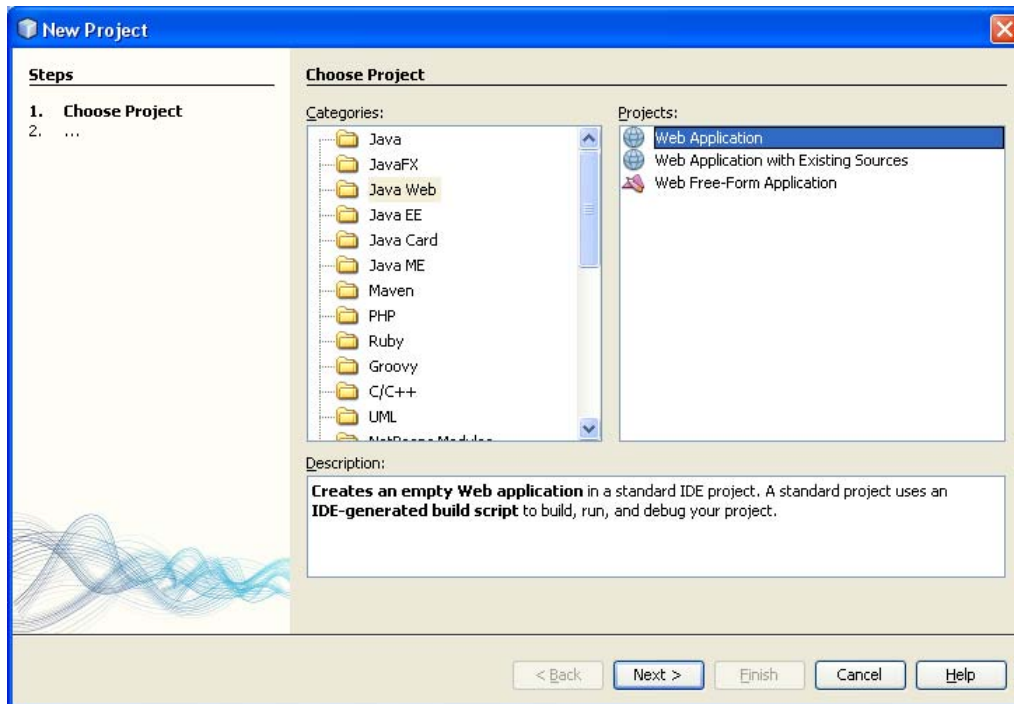


Εικόνα 3.2.2. Netbeans IDE 6.9.1

3.3 – Εισαγωγή στο Project

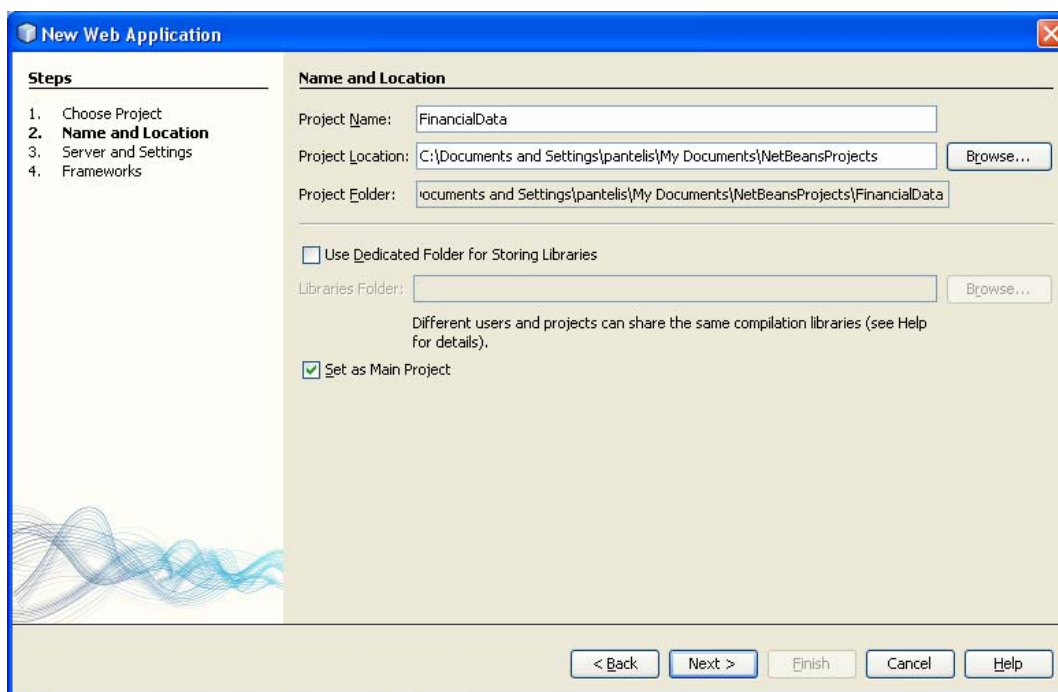
3.3.1 – Δημιουργία ενός νέου Java Web Application στο IDE

1. Επιλέξτε File > New Project (Ctrl+Shift+N) από το κύριο μενού. Επιλέξτε την κατηγορία Java Web , στη συνέχεια, επιλέξτε Web Application. Κάντε κλικ στο Next.



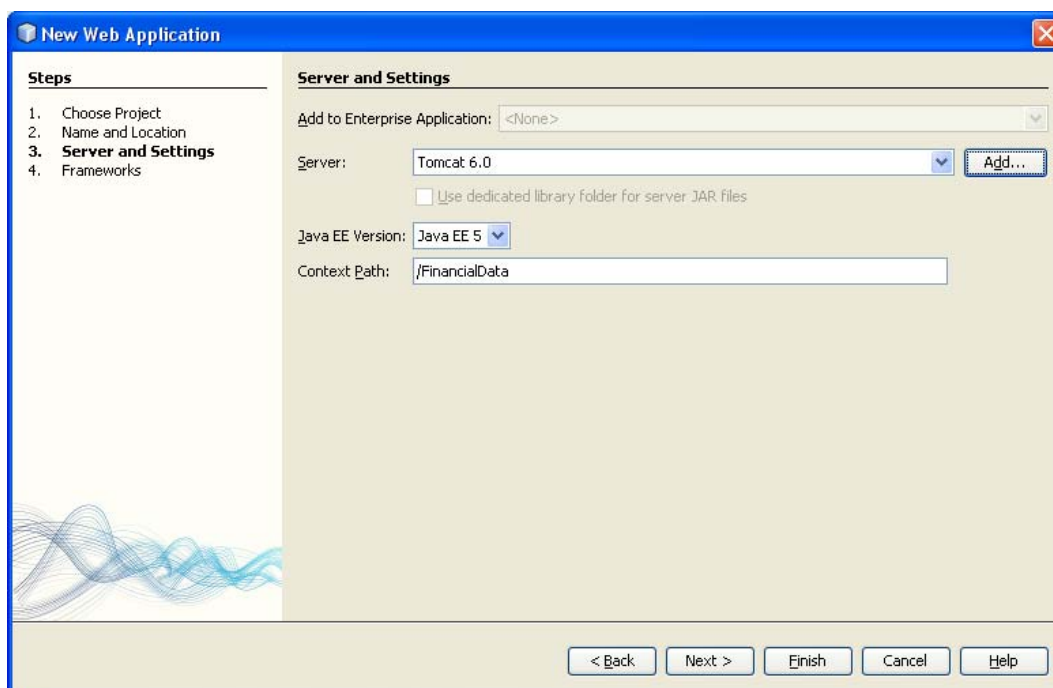
Εικόνα 3.3.1.1. Δημιουργία Web Application

2. Σαν όνομα του **project**, πληκτρολογήστε **FinancialData**. Επίσης, καθορίστε τη θέση για το project στον υπολογιστή σας. (Από προεπιλογή, το IDE τοποθετεί τα projects στον φάκελο NetBeansProjects που βρίσκεται στον home φάκελο του χρήστη.) Κάντε κλικ στο Next.



Εικόνα 3.3.1.2. Δημιουργία Web Application

3. Στο πάνελ του Server και των Ρυθμίσεων, καθορίστε τον Tomcat 6.0 ως Server που θα χρησιμοποιηθεί για να τρέξει η εφαρμογή. Στο πεδίο της έκδοσης Java EE Version, επιλέξτε Java EE 5.

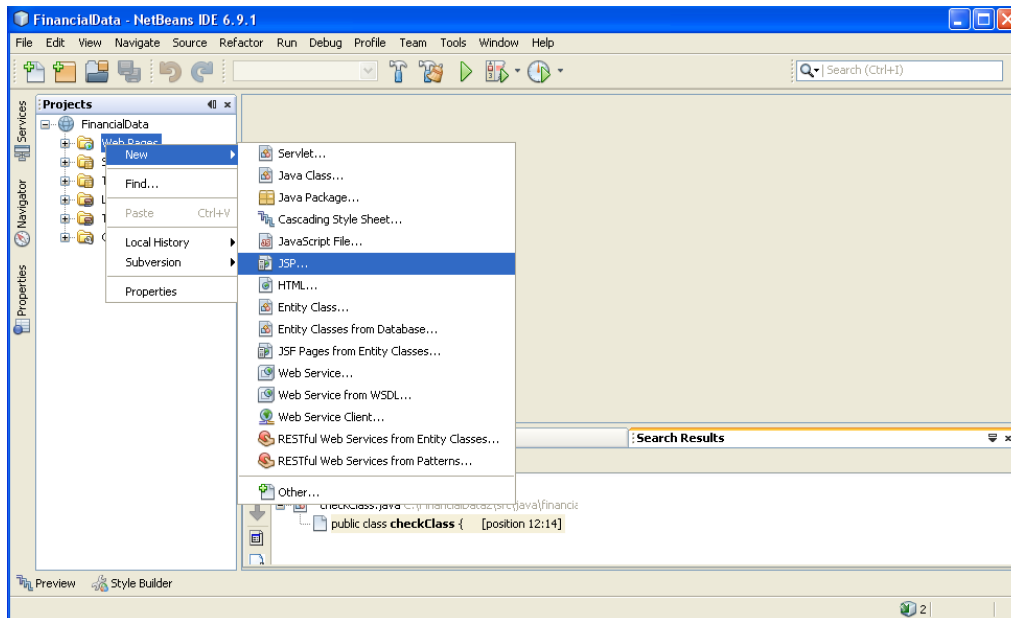


Εικόνα 3.3.1.3. Δημιουργία Web Application

4. Κάντε κλικ στο κουμπί Finish. Το IDE δημιουργεί ένα πρότυπο σχέδιο για το σύνολο της αίτησης, και ανοίγει μια κενή σελίδα (**index.jsp**) JSP στον επεξεργαστή μέσα στον φάκελο *Web Pages*. Το αρχείο **index.jsp** χρησιμεύει ως η σελίδα υποδοχής για την εφαρμογή.

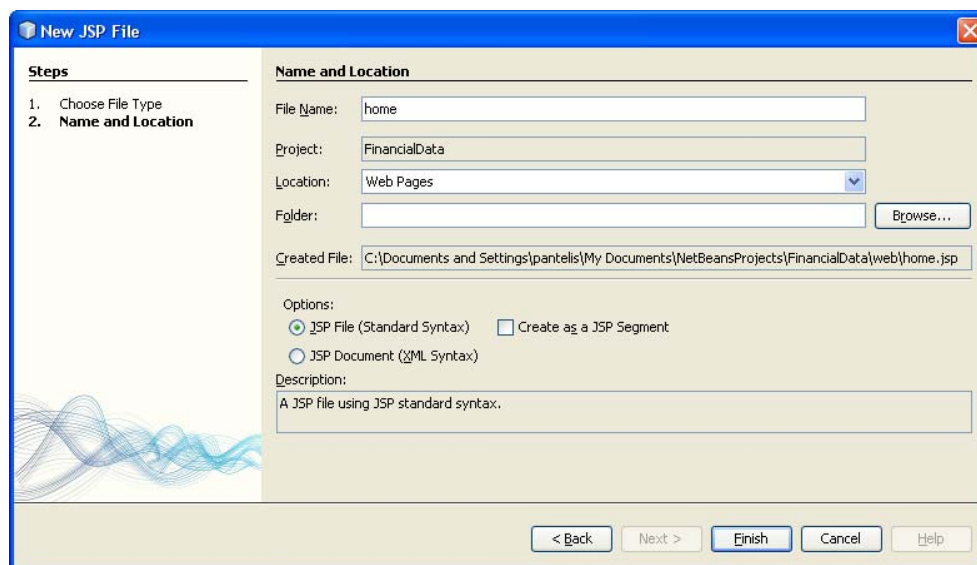
3.3.2 – Δημιουργία ενός νέου JSP αρχείου

1. Αφότου έχει δημιουργηθεί το **Web Application**(FinancialData), κάντε δεξί κλικ στον φάκελο **Web Pages**. Επιλέξτε **New**, και πατήστε κλικ στην επιλογή **JSP...**.



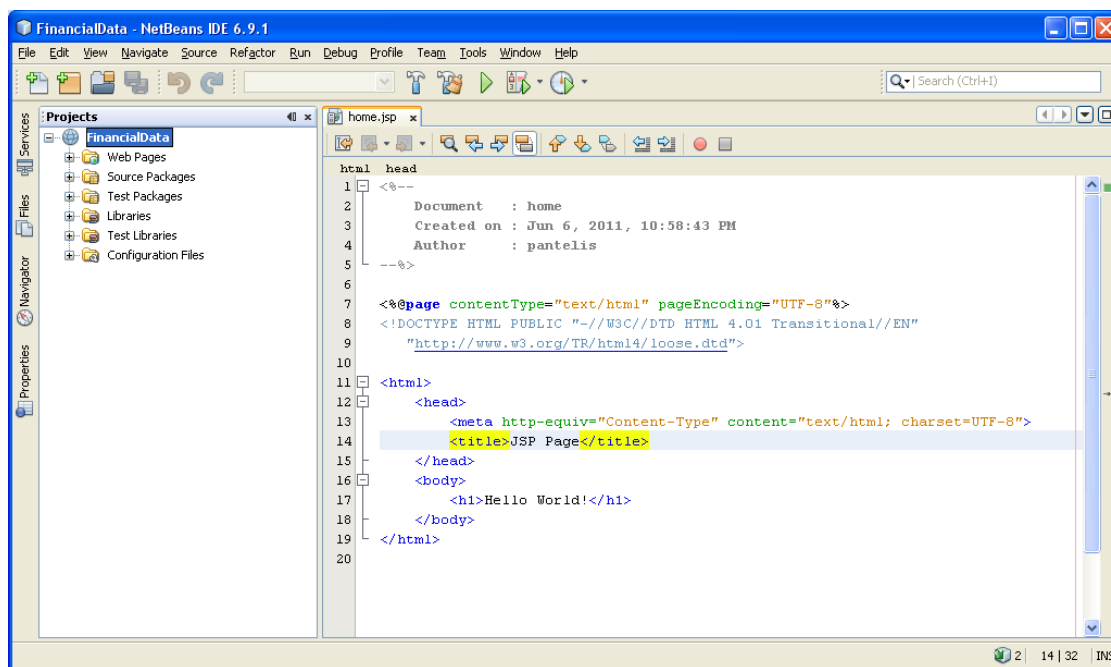
Εικόνα 3.3.2.1. Δημιουργία JSP

2. Στο παράθυρο που εμφανίζεται συμπληρώστε το πεδίο **File Name** με την ονομασία που θέλετε να δώσετε στο αρχείο .jsp(στο παράδειγμα δημιουργούμε το αρχείο **home.jsp**). Μην κάνετε καμία άλλη τροποποίηση, επιλέξτε το κουμπί **Finish**.



Εικόνα 3.3.2.2. Δημιουργία JSP

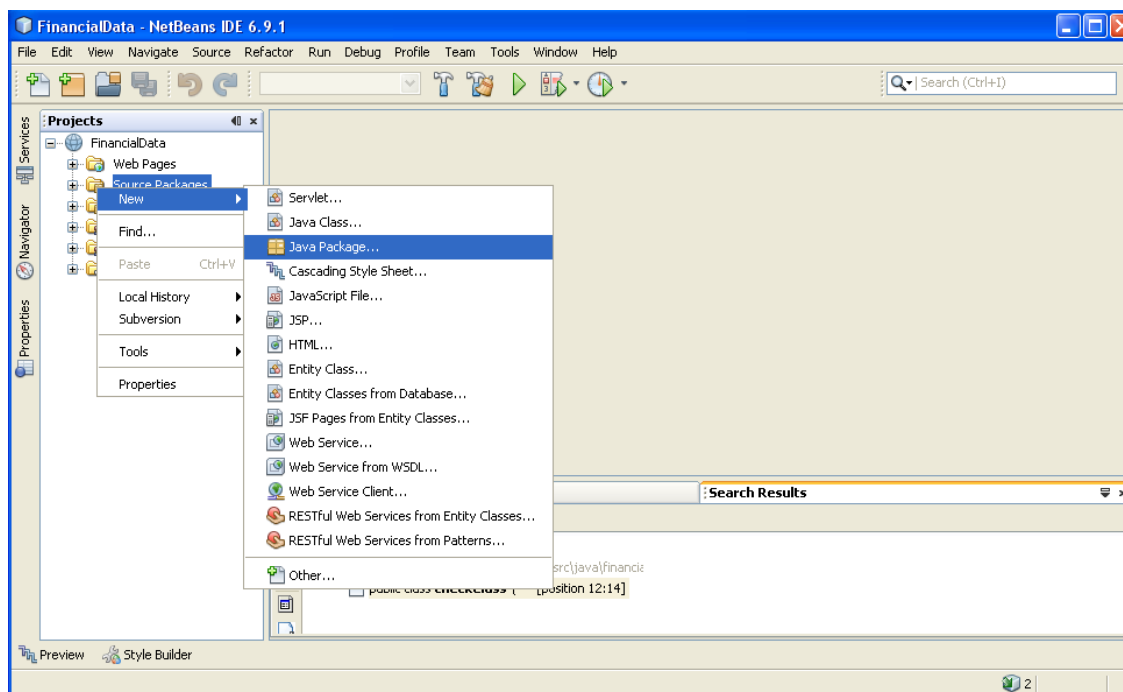
3. Το αρχείο έχει δημιουργηθεί και είναι έτοιμο για επεξεργασία.



Εικόνα 3.3.2.3. Δημιουργία JSP

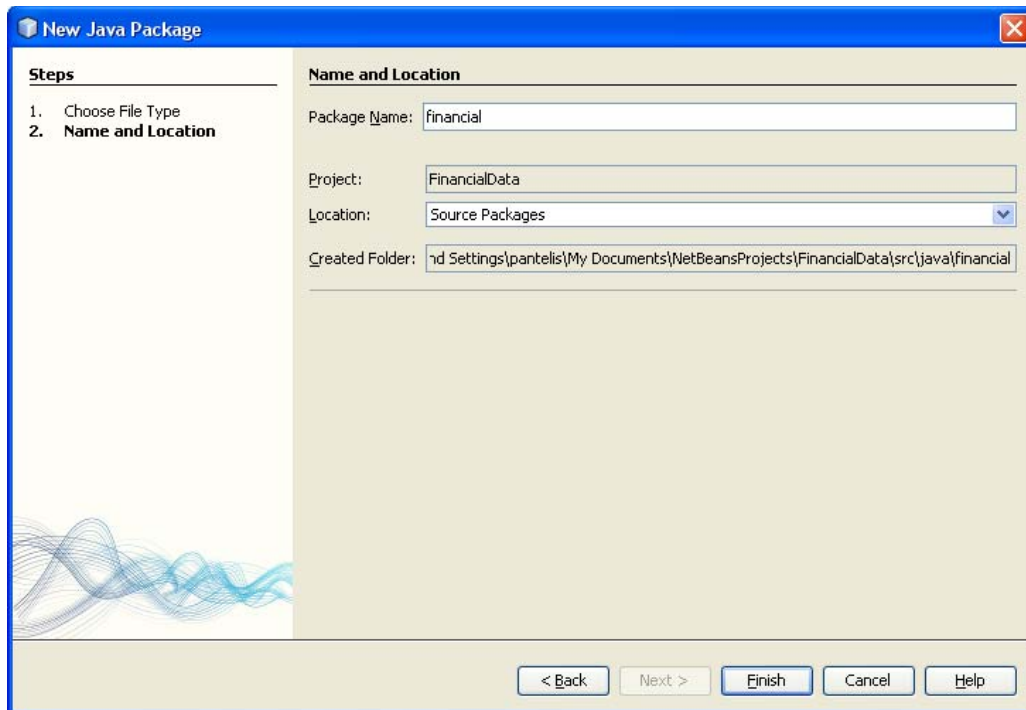
3.3.3 – Δημιουργία ενός νέου πακέτου JAVA

1. Η χρήση του πακέτου είναι να ομαδοποιήσει τα αρχεία JAVA. Αφότου έχει δημιουργηθεί το **Web Application**(FinancialData), κάντε δεξί κλικ στον φάκελο **Source Packages**. Επιλέξτε **New**, και πατήστε κλικ στην επιλογή **Java Package**....



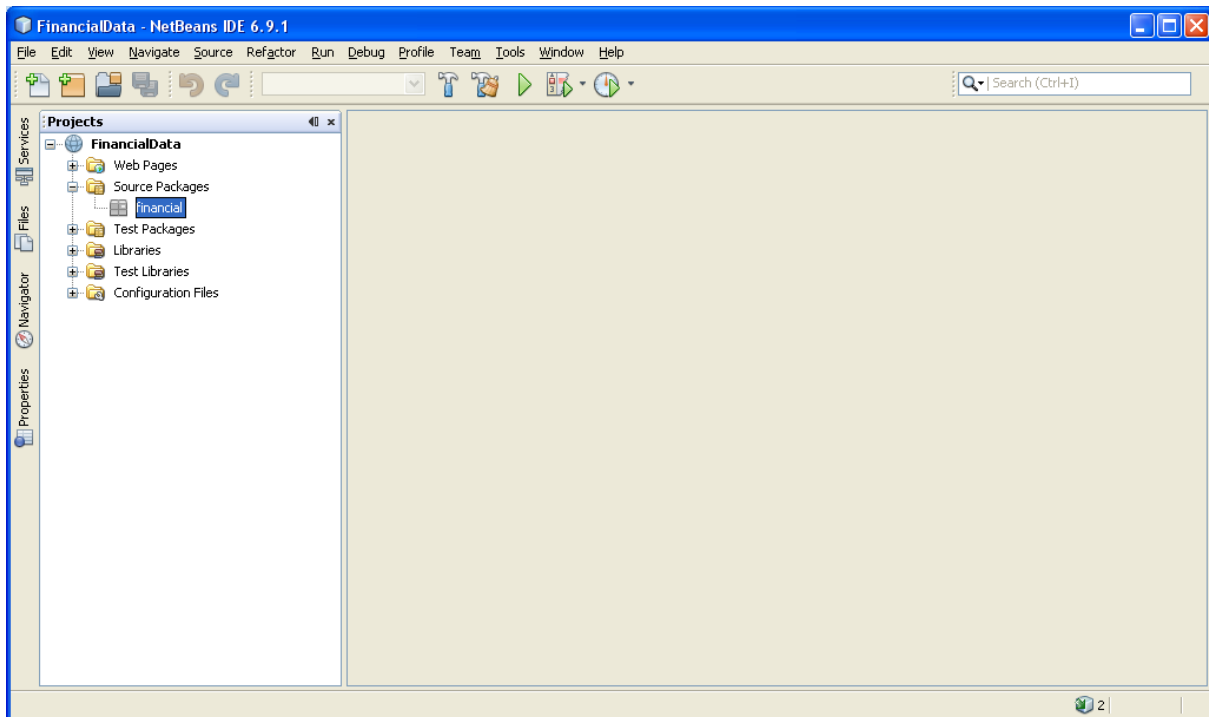
Εικόνα 3.3.3.1. Δημιουργία Java Package

2. Στο παράθυρο που εμφανίζεται συμπληρώστε το πεδίο **Package Name** με την ονομασία που θέλετε να δώσετε στο πακέτο (στο δικό μας project δίδεται η ονομασία **financial**). Μην κάνετε καμία άλλη τροποποίηση, επιλέξτε το κουμπί Finish.



Εικόνα 3.3.3.2. Δημιουργία Java Package

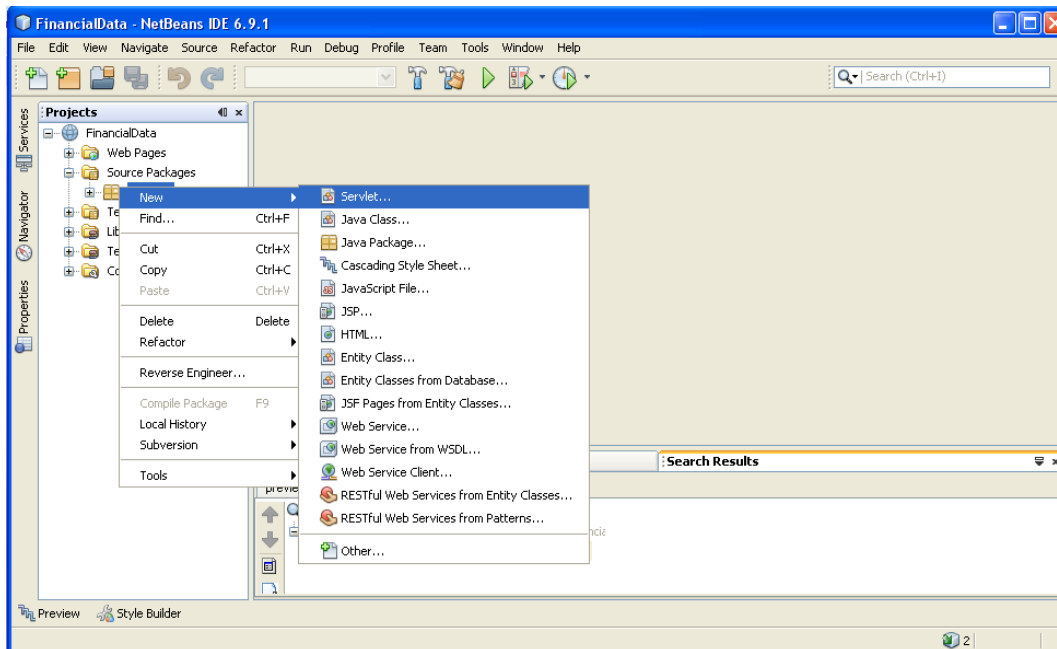
3. Το αρχείο έχει δημιουργηθεί και είναι έτοιμο για επεξεργασία.



Εικόνα 3.3.3.3. Δημιουργία Java Package

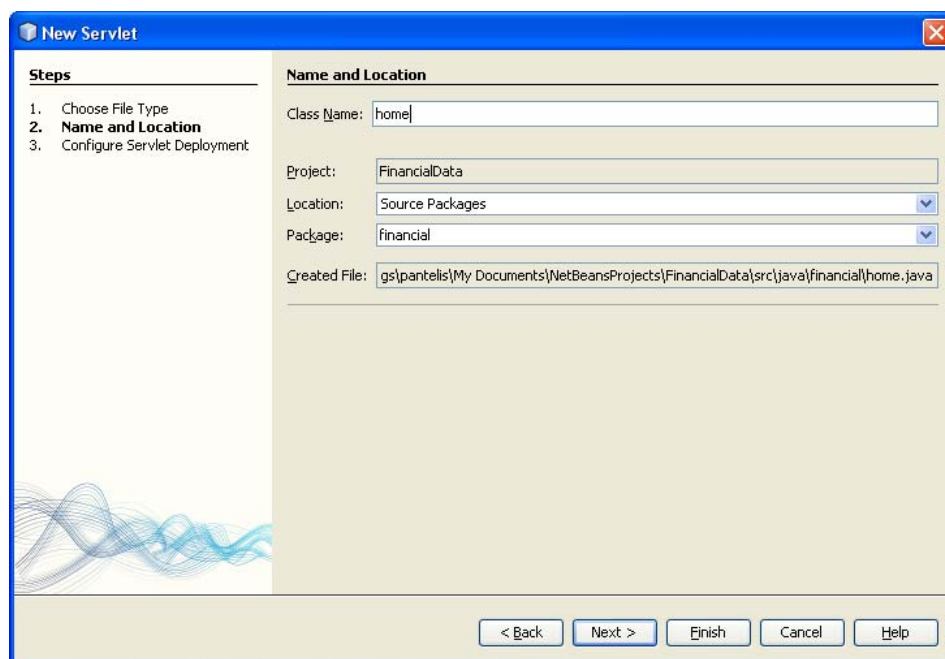
3.3.4 – Δημιουργία ενός νέου JAVA Servlet

1. Αφότου έχουν δημιουργηθεί τα **Web Application**(FinancialData), και το **πακέτο JAVA**(financial) κάντε δεξί κλικ στο πακέτο **financial** μέσα στον φάκελο **Source Packages**. Επιλέξτε **New**, και πατήστε κλικ στην επιλογή **Servlet...**



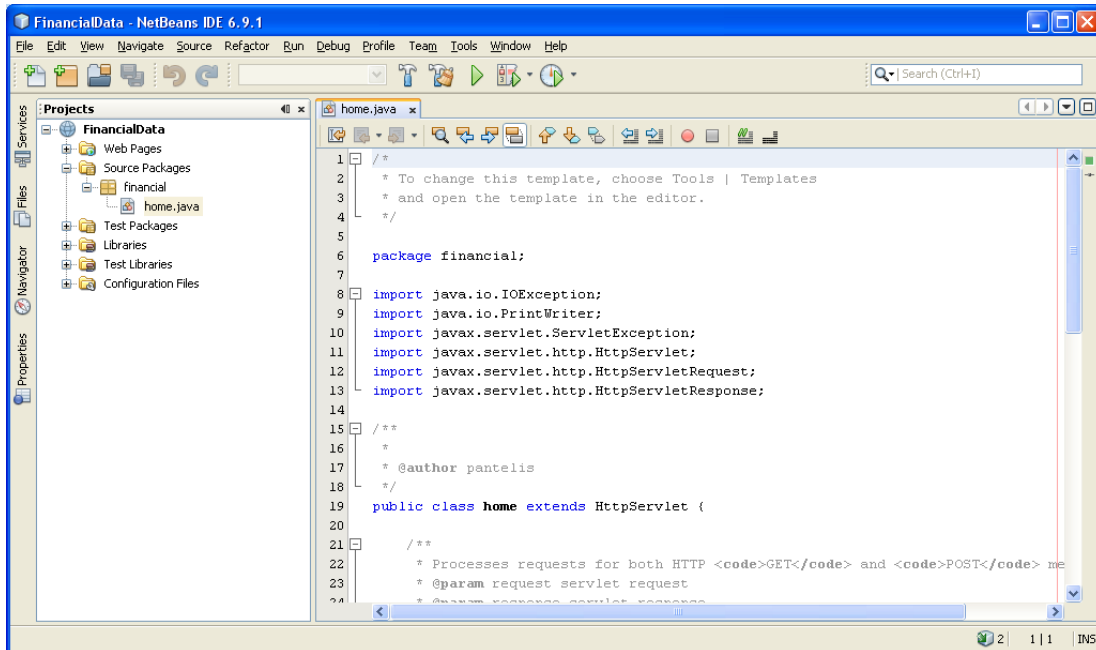
Εικόνα 3.3.4.1. Δημιουργία Servlet

2. Στο παράθυρο που εμφανίζεται συμπληρώστε το πεδίο **Class Name** με την ονομασία που θέλετε να δώσετε στο αρχείο .java(στο παράδειγμα δημιουργούμε το αρχείο home.java). Μην κάνετε καμία άλλη τροποποίηση, επιλέξτε το κουμπί **Finish**.



Εικόνα 3.3.4.2. Δημιουργία Servlet

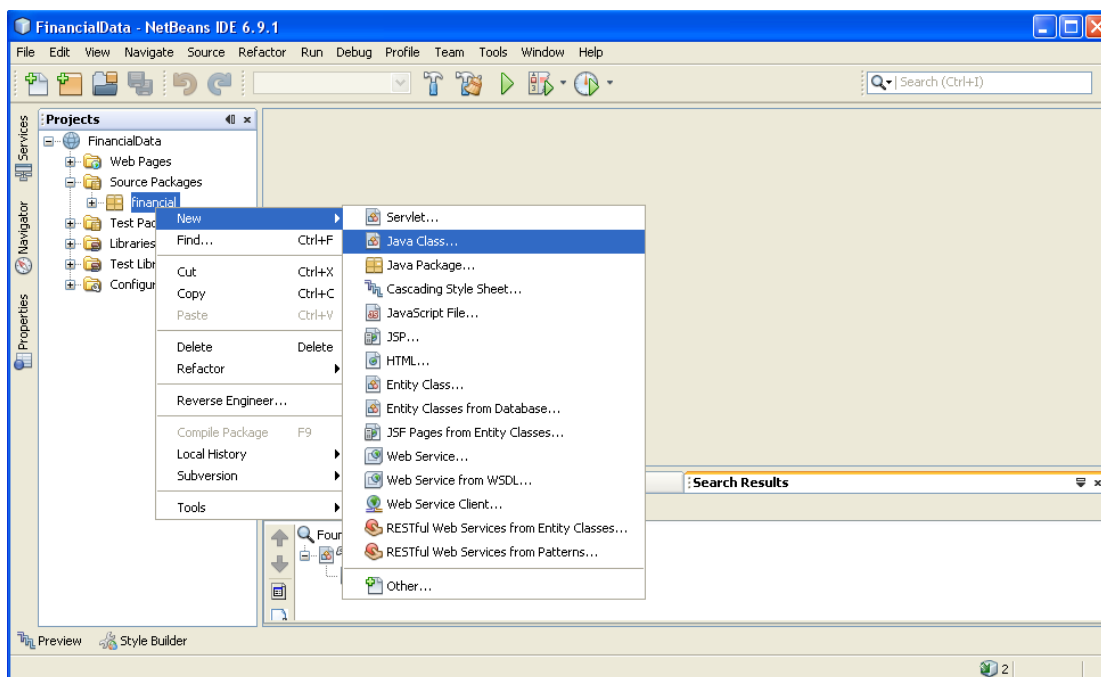
3. Το αρχείο έχει δημιουργηθεί και είναι έτοιμο για επεξεργασία.



Εικόνα 3.3.4.3. Δημιουργία Servlet

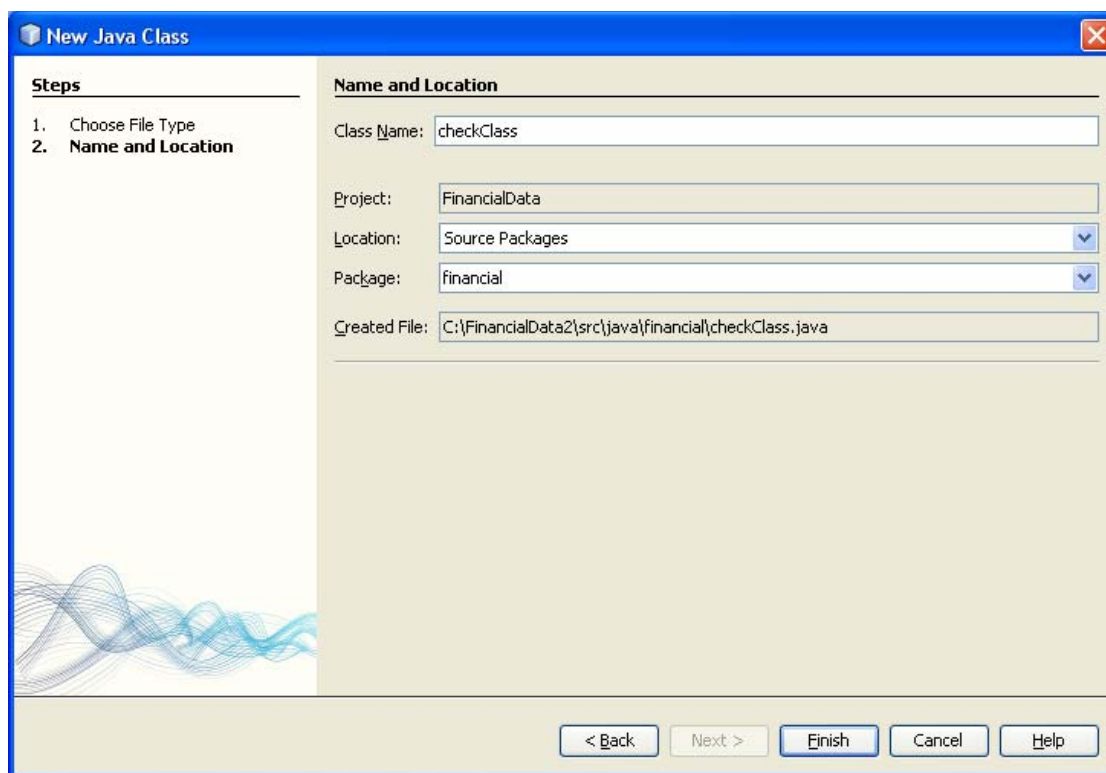
3.3.5 – Δημιουργία ενός νέου JAVA Class

1. Αφότου έχουν δημιουργηθεί τα **Web Application**(FinancialData), και το **πακέτο JAVA**(financial) κάντε δεξί κλικ στο πακέτο **financial** μέσα στον φάκελο **Source Packages**. Επιλέξτε **New**, και πατήστε κλικ στην επιλογή **Java Class**...



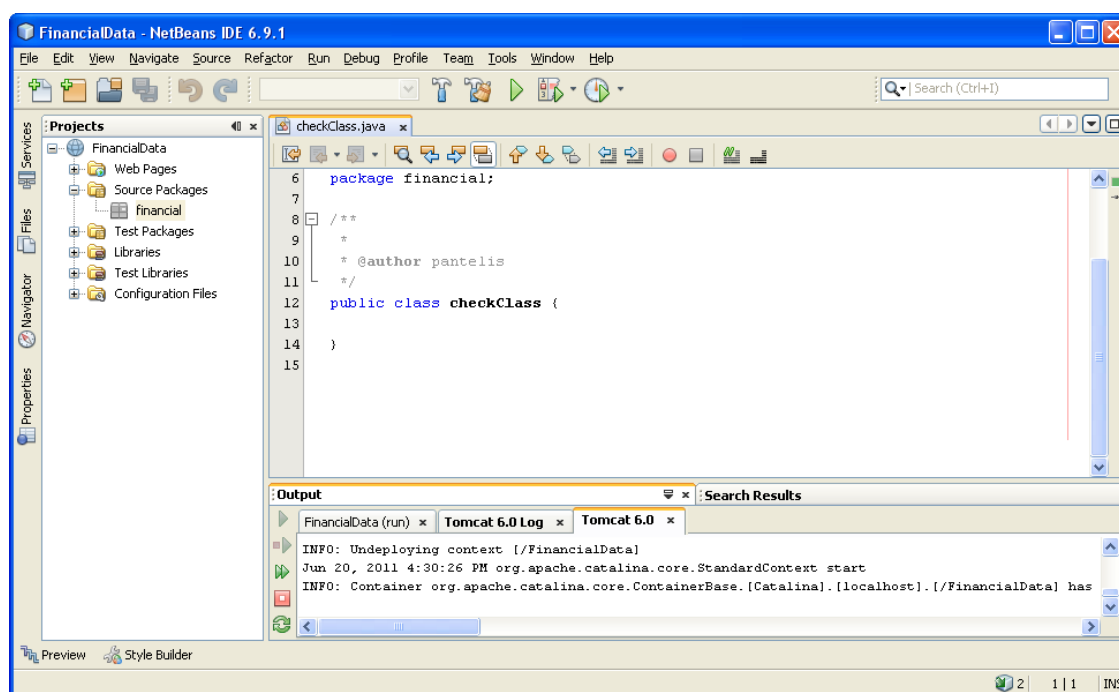
Εικόνα 3.3.5.1. Δημιουργία Java Class

2. Στο παράθυρο που εμφανίζεται συμπληρώστε το πεδίο **Class Name** με την ονομασία που θέλετε να δώσετε στο αρχείο .java(στο παράδειγμα δημιουργούμε το αρχείο checkClass.java). Μην κάνετε καμία άλλη τροποποίηση, επιλέξτε το κουμπί Finish.



Εικόνα 3.3.5.2. Δημιουργία Java Class

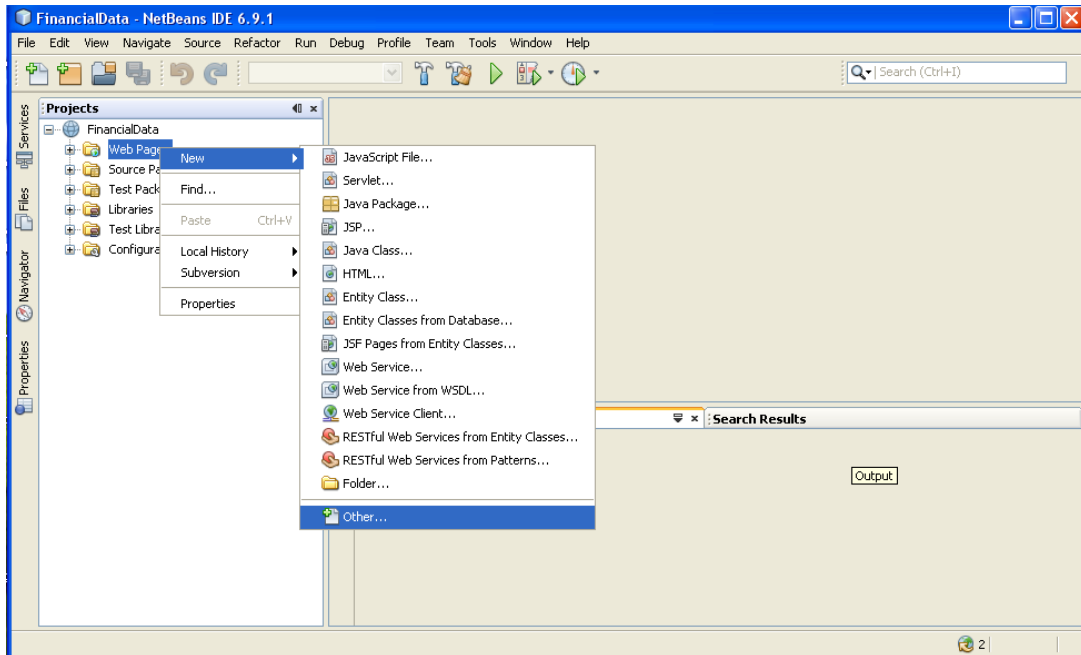
3. Το αρχείο έχει δημιουργηθεί και είναι έτοιμο για επεξεργασία.



Εικόνα 3.3.5.3. Δημιουργία Java Class

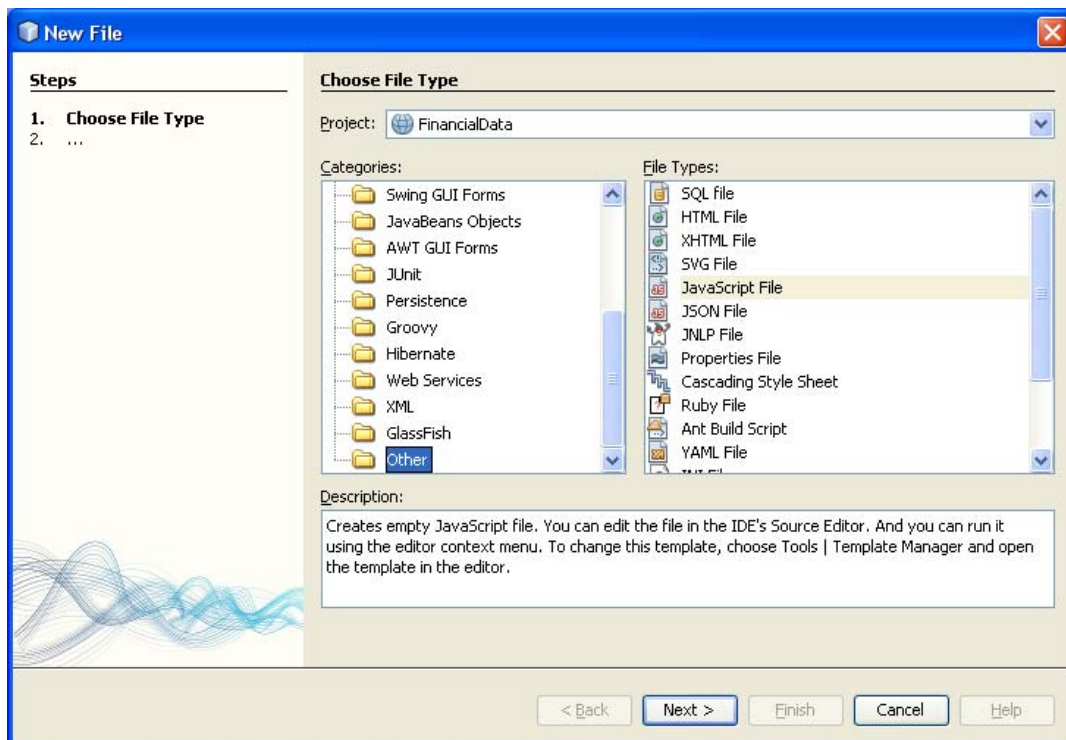
3.3.6 – Δημιουργία ενός νέου JavaScript αρχείου

1. Αφότου έχει δημιουργηθεί το **Web Application(FinancialData)**, κάντε δεξί κλικ στον φάκελο **Web Pages**. Επιλέξτε **New**, και πατήστε κλικ στην επιλογή **Other...**



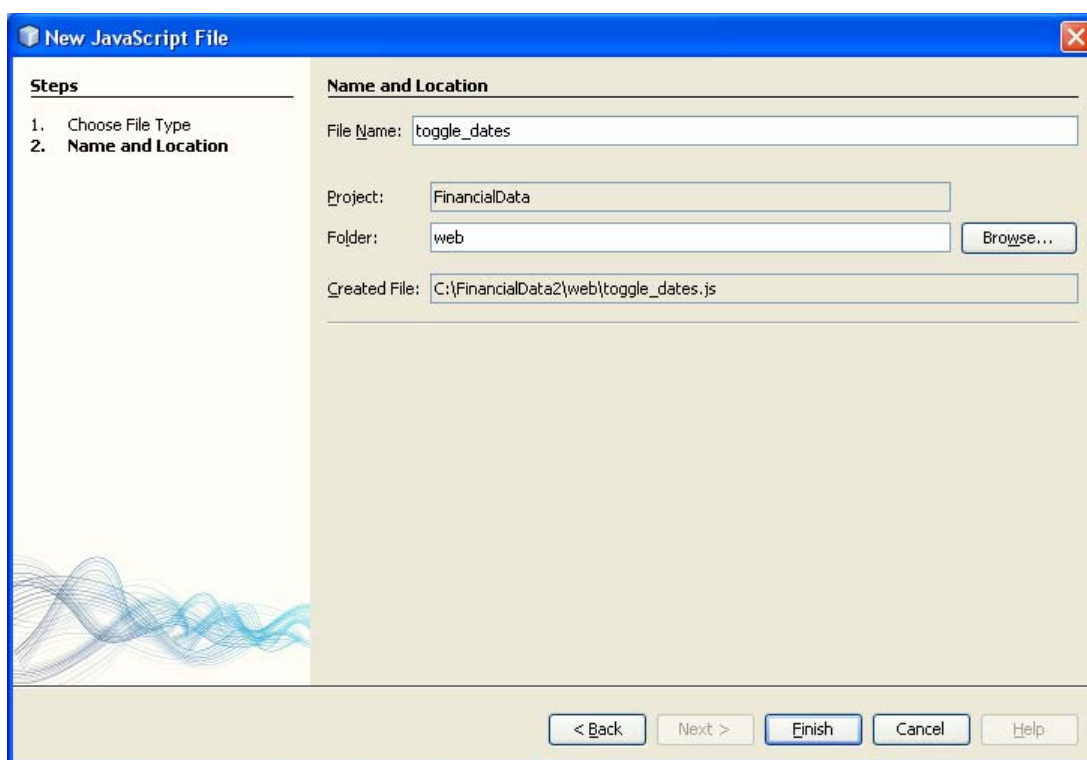
Εικόνα 3.3.6.1. Δημιουργία JavaScript

2. Στο παράθυρο που εμφανίζεται επιλέξτε από την αριστερή στήλη τον φάκελο **Other** και από την δεξιά στήλη **JavaScript File**. Επιλέξτε το κουμπί **Next**.



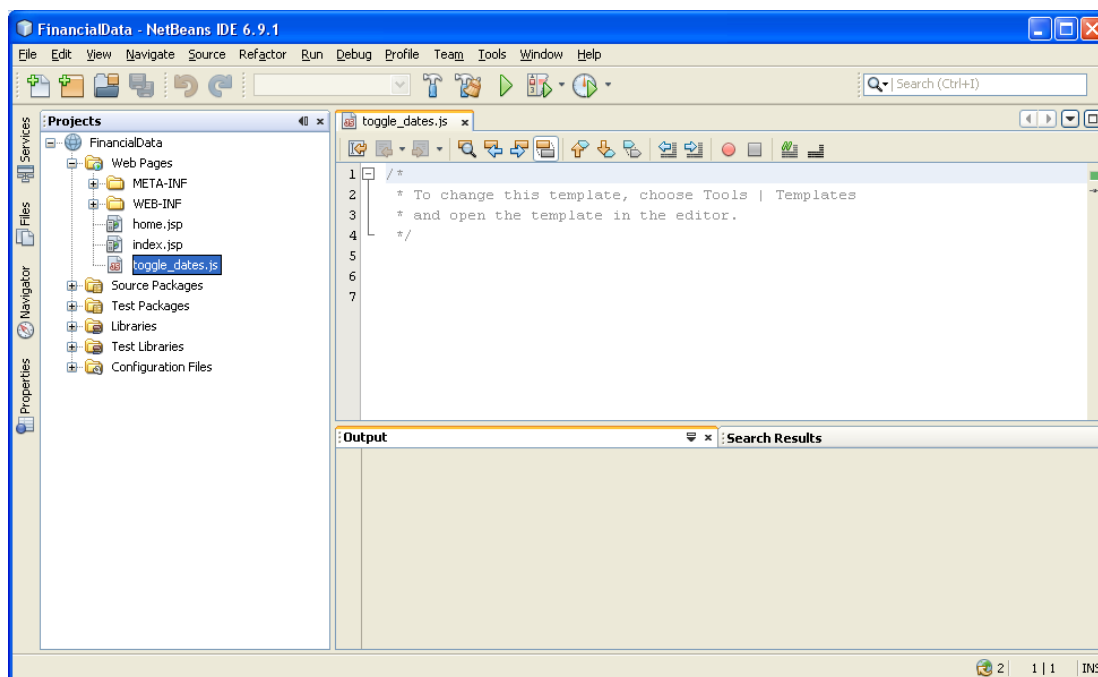
Εικόνα 3.3.6.2. Δημιουργία JavaScript

3. Στο παράθυρο που εμφανίζεται συμπληρώστε το πεδίο **File Name** με την ονομασία που θέλετε να δώσετε στο αρχείο .js(στο παράδειγμα δημιουργούμε το αρχείο toggle_dates.js). Μην κάνετε καμία άλλη τροποποίηση, επιλέξτε το κουμπί **Finish**.



Εικόνα 3.3.6.3. Δημιουργία JavaScript

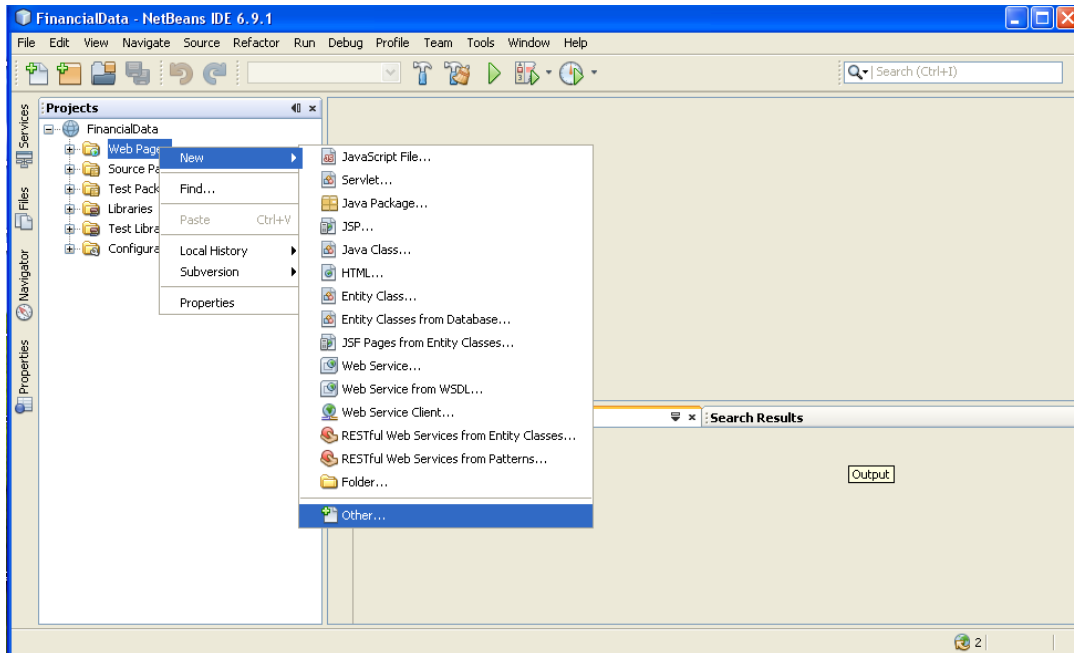
4. Το αρχείο έχει δημιουργηθεί και είναι έτοιμο για επεξεργασία.



Εικόνα 3.3.6.4. Δημιουργία JavaScript

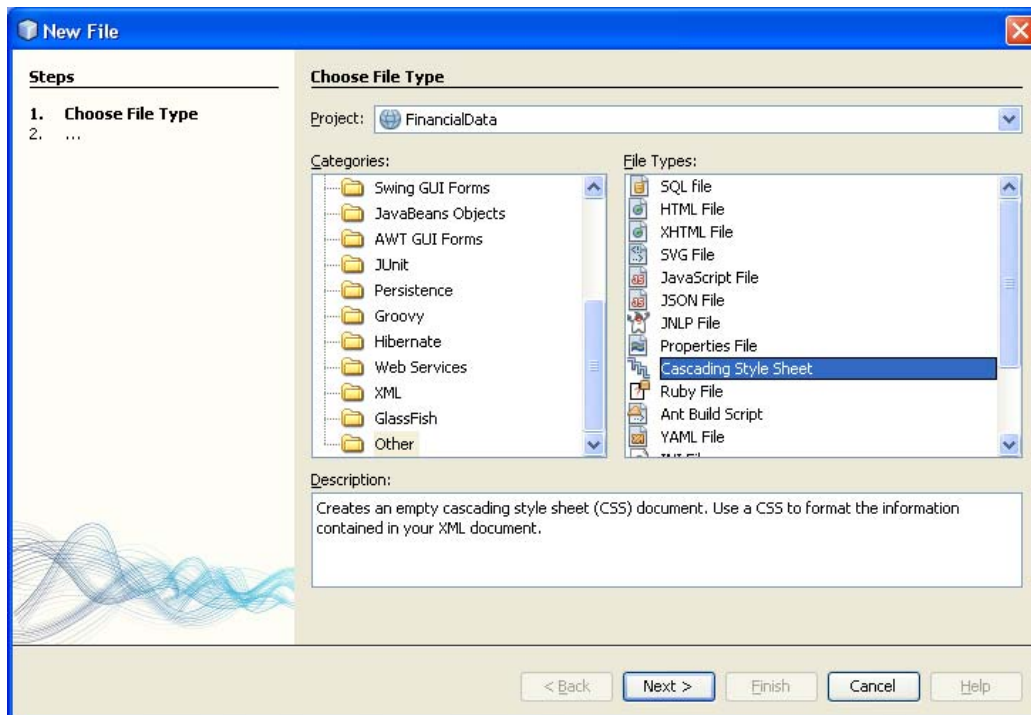
3.3.7 – Δημιουργία ενός νέου CSS αρχείου

1. Αφότου έχει δημιουργηθεί το **Web Application**(FinancialData), κάντε δεξί κλικ στον φάκελο **Web Pages**. Επιλέξτε **New**, και πατήστε κλικ στην επιλογή **Other...**



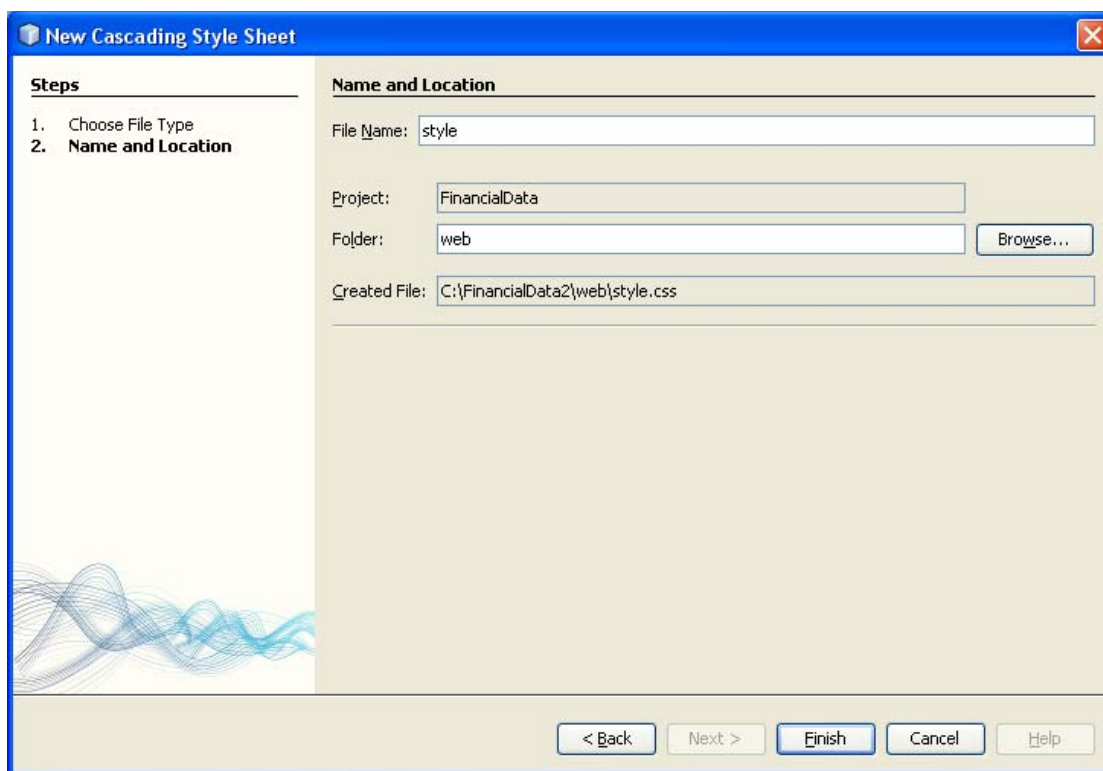
Εικόνα 3.3.7.1. Δημιουργία CSS

2. Στο παράθυρο που εμφανίζεται επιλέξτε από την αριστερή στήλη τον φάκελο **Other** και από την δεξιά στήλη **Cascading Style Sheet**. Επιλέξτε το κουμπί **Next**.



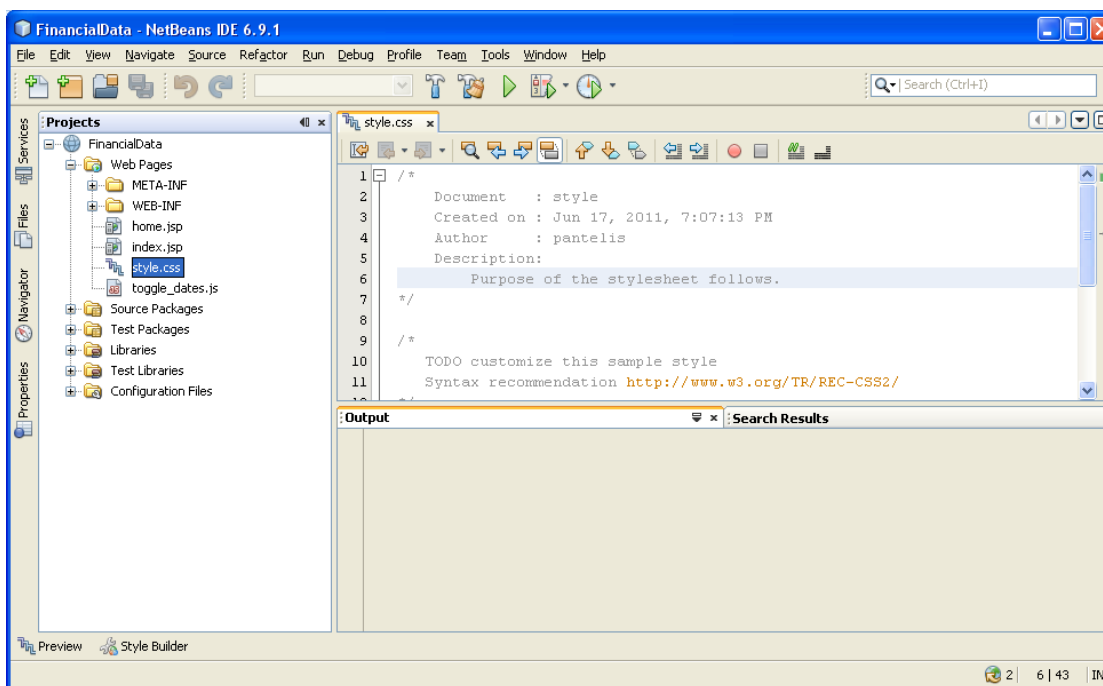
Εικόνα 3.3.7.2. Δημιουργία CSS

3. Στο παράθυρο που εμφανίζεται συμπληρώστε το πεδίο **File Name** με την ονομασία που θέλετε να δώσετε στο αρχείο .css(στο παράδειγμα δημιουργούμε το αρχείο style.css). Μην κάνετε καμία άλλη τροποποίηση, επιλέξτε το κουμπί Finish.



Εικόνα 3.3.7.3. Δημιουργία CSS

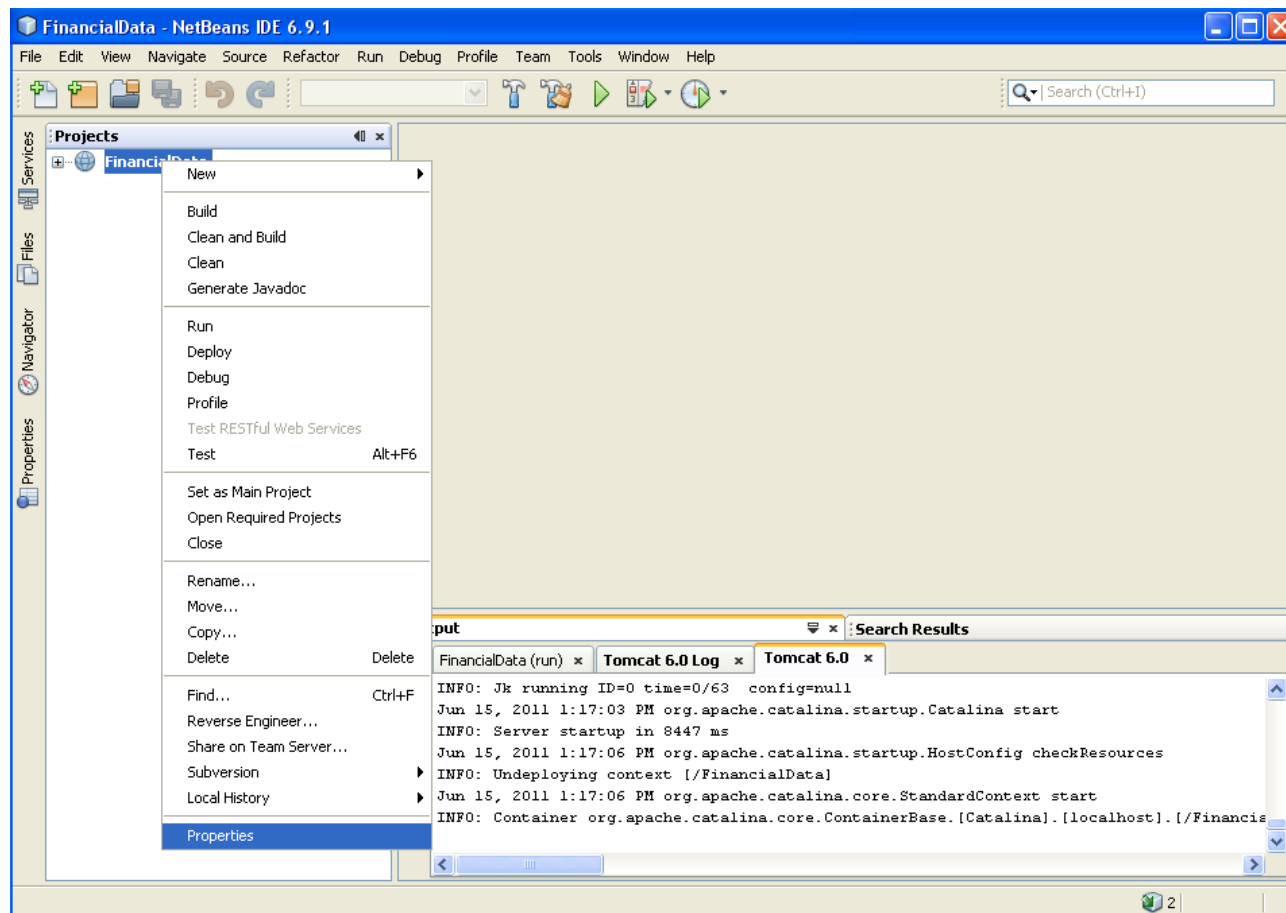
4. Το αρχείο έχει δημιουργηθεί και είναι έτοιμο για επεξεργασία.



Εικόνα 3.3.7.4. Δημιουργία CSS

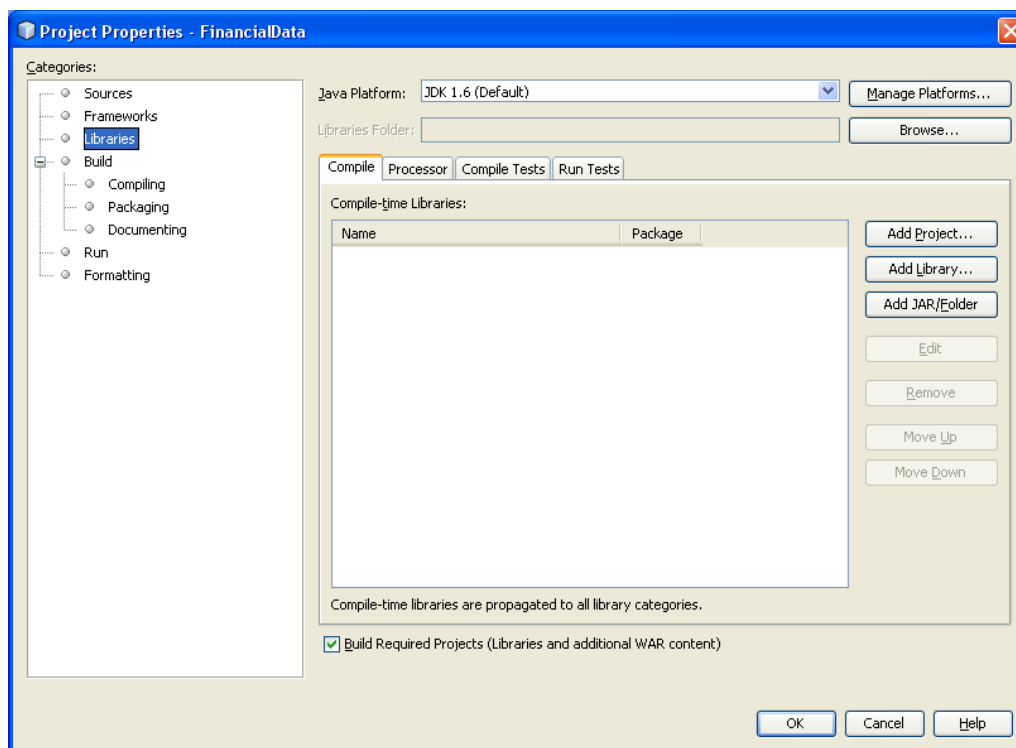
3.3.8 – Σύνδεση του Web Application με τον MySQL Connector /J

1. Για να μπορέσει το **Web Application**(FinancialData) να φορτώσει σωστά, πρέπει να συνδεθεί με τα δεδομένα της βάσης(MySQL). Για να γίνει αυτό πρέπει να συνδέσουμε την εφαρμογή με τον **mysql Connector Java**. Πατήστε δεξί κλικ στον φάκελο του **Web Application**(FinancialData) και επιλέξτε **Properties**.



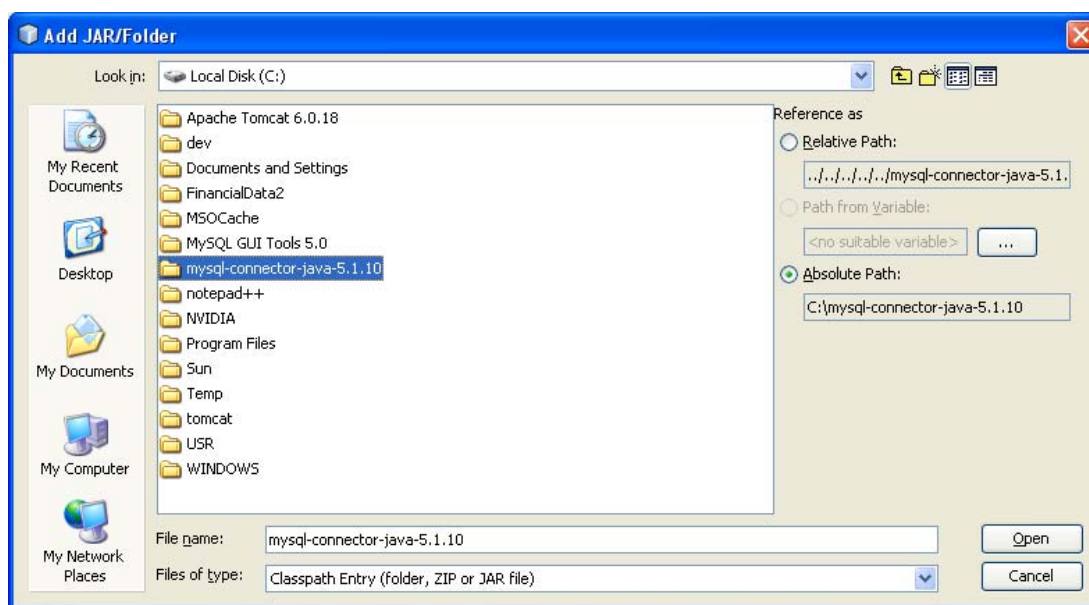
Εικόνα 3.3.8.1. Σύνδεση MySQL Connector/J

2. Θα εμφανιστεί ένα νέο παράθυρο. Πατήστε κλικ στην καρτέλα **Libraries**. Στα δεξιά υπάρχει η επιλογή **Add JAR/Folder....** Επιλέξτε την κάνοντας κλικ επάνω.



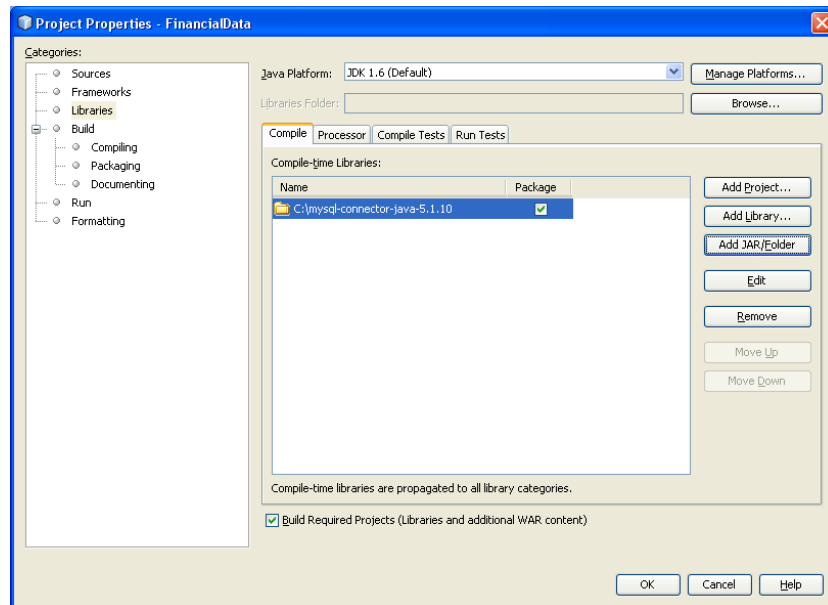
Εικόνα 3.3.8.2. Σύνδεση MySQL Connector/J

- Ένα νέο παράθυρο εμφανίζεται. Από εδώ πρέπει να πλοηγηθείτε μέχρι να βρείτε την θέση που είναι αποθηκευμένος ο φάκελος του **mysql connector java**. Προσοχή, πρέπει να φορτώσετε όλον τον φάκελο. Από την εγκατάσταση του **mysql Connector Java** η προεπιλεγμένη τοποθεσία αποθήκευσης είναι: *C:\mysql-connector-java-5.1.10*. Όταν φτάσετε στο φάκελο που δείχνει η Εικόνα 3.3.8.3(Σύνδεση MySQL Connector/J) πατήστε Open.



Εικόνα 3.3.8.3. Σύνδεση MySQL Connector/J

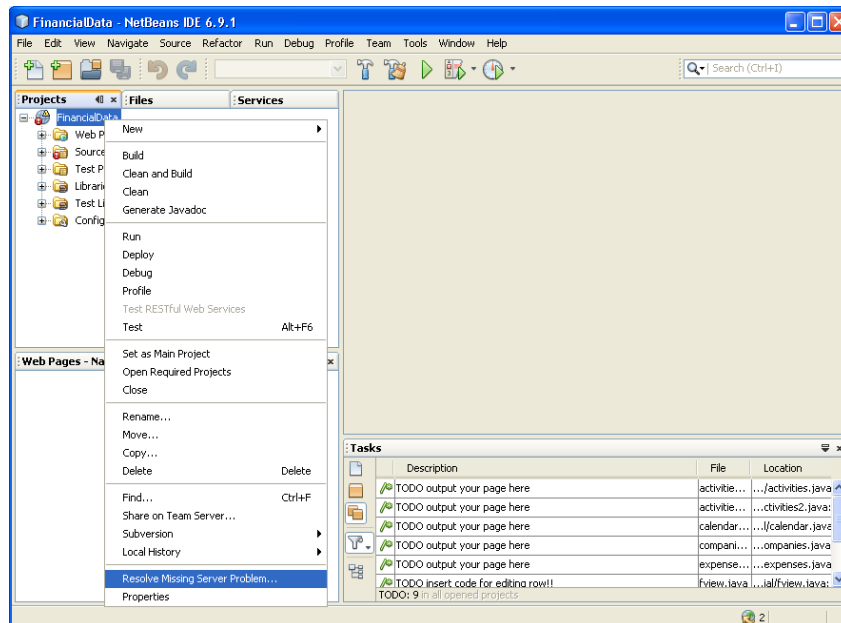
4. Αν όλα γίνανε σωστά, τότε πρέπει να φαίνεται στον υπολογιστή σας ότι δείχνει και η παρακάτω εικόνα. Αν όντως είναι έτσι, τότε η σύνδεση με τον **mysql Connector Java** είναι επιτυχής. Πατήστε **OK** για να κλείσετε το παράθυρο.



Εικόνα 3.3.8.4. Σύνδεση MySQL Connector/J

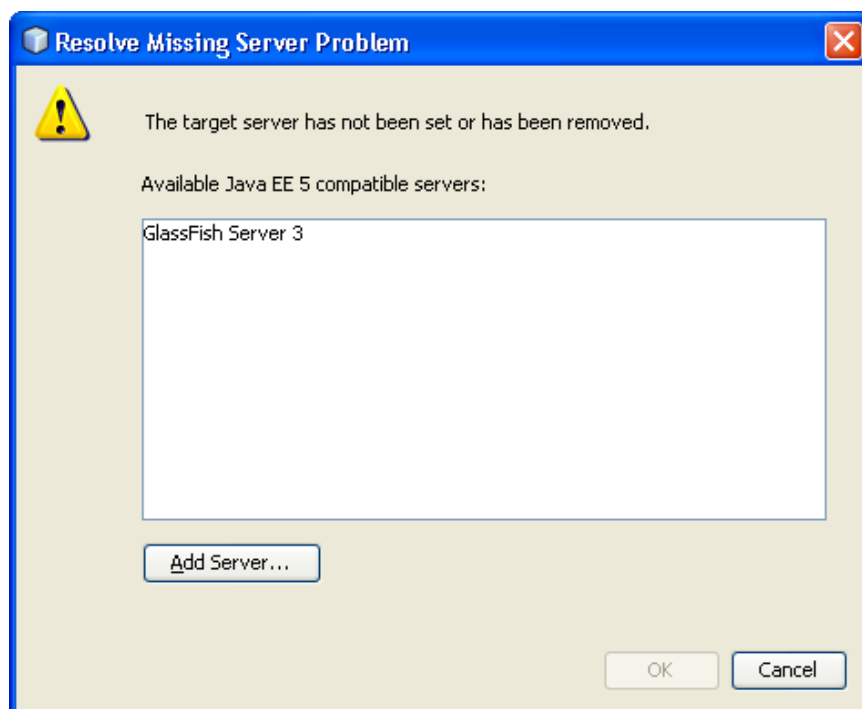
3.3.9 – Σύνδεση του Web Application με τον Tomcat Server 6.0

1. Εκτός από τον **mysql Connector Java**, που χρησιμοποιείται για την σύνδεση μεταξύ της **MySQL** και της εφαρμογής, για να μπορέσει το **Web Application(FinancialData)** να φορτώσει σωστά, πρέπει να συνδεθεί με κάποιον **server**. Στο συγκεκριμένο project χρησιμοποιήθηκε ο **Tomcat Server 6.0**. Για να συνδέσετε τον **Tomcat** πατήστε δεξί κλικ στον φάκελο του **Web Application(FinancialData)** και επιλέξτε **Resolve Missing Server Problem...**



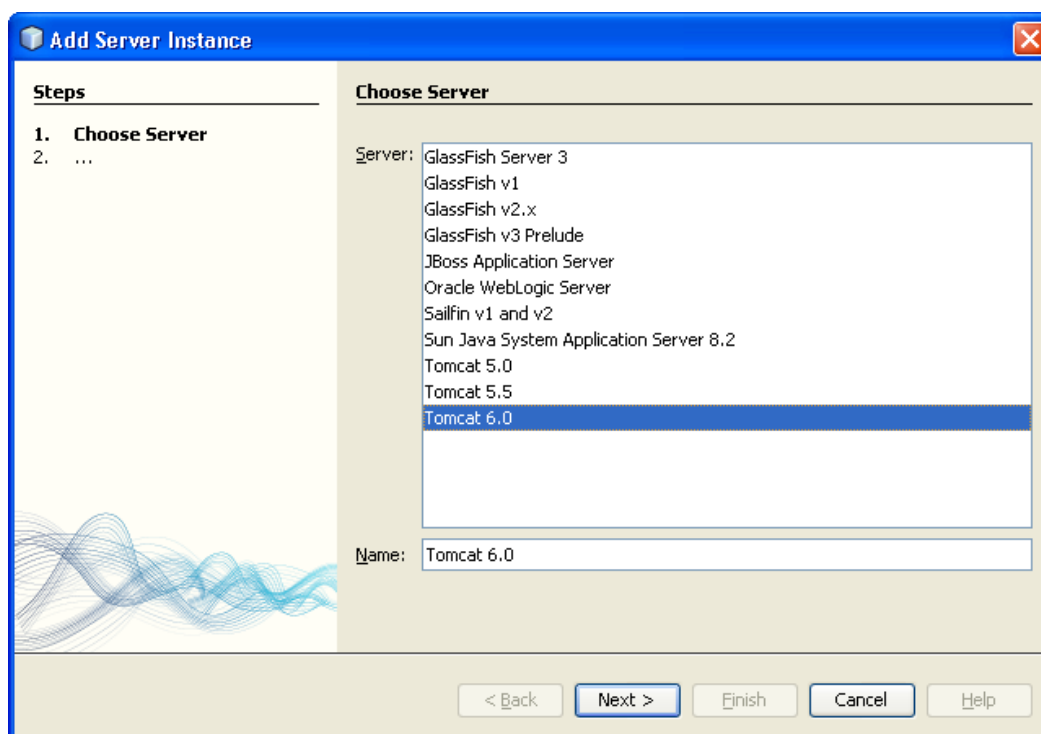
Εικόνα 3.3.9.1. Σύνδεση Apache Tomcat

2. Θα εμφανιστεί ένα νέο παράθυρο. Πατήστε κλικ στο κουμπί **Add Server...**



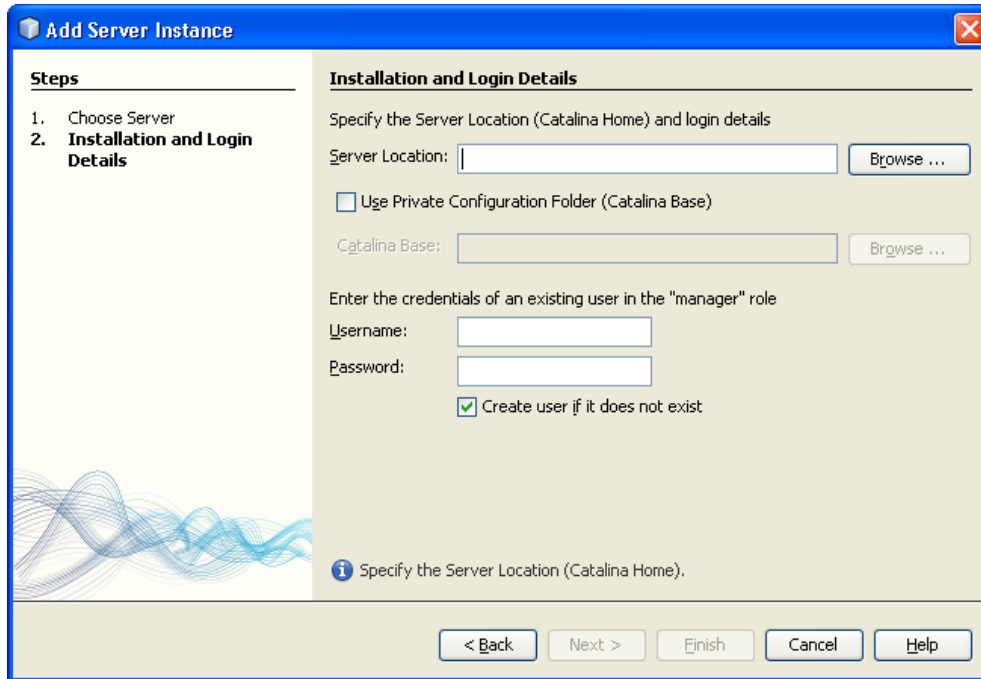
Εικόνα 3.3.9.2. Σύνδεση Apache Tomcat

3. Ένα νέο παράθυρο εμφανίζεται. Δίνεται μια λίστα από διάφορους εξυπηρετητές. Επιλέξτε τον **Tomcat 6.0** και πατήστε Next.



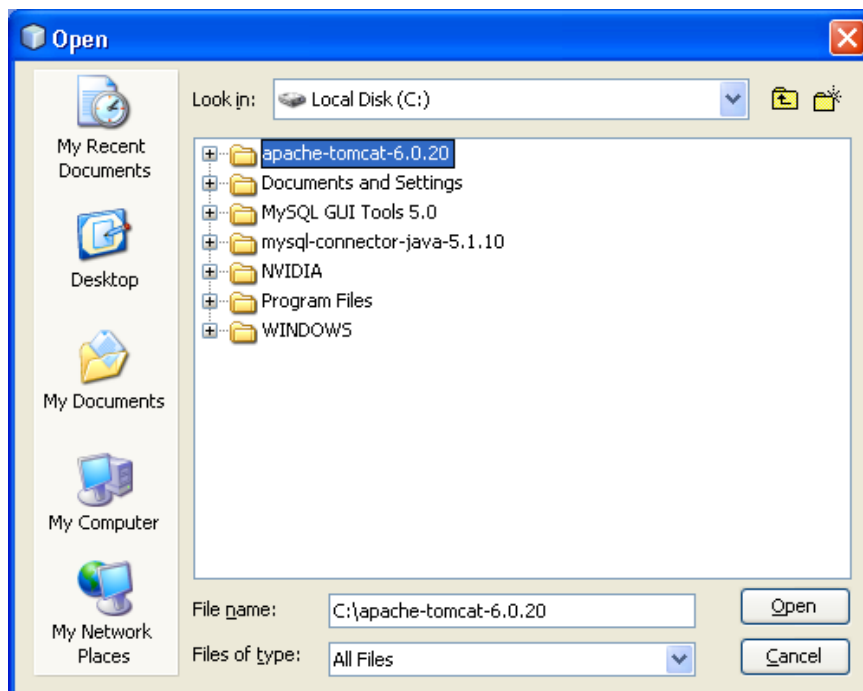
Εικόνα 3.3.9.3. Σύνδεση Apache Tomcat

4. Στην συνέχεια πρέπει να δώσετε την διαδρομή που βρίσκεται αποθηκευμένος ο **Tomcat 6.0**. Δίπλα από το πεδίο **Server Location** πατήστε Browse.



Εικόνα 3.3.9.4. Σύνδεση Apache Tomcat

5. Ένα νέο παράθυρο εμφανίζεται. Από εδώ πρέπει να πλοηγηθείτε μέχρι να βρείτε την θέση που είναι αποθηκευμένος ο φάκελος του **Tomcat 6.0**. Προσοχή, πρέπει να φορτωθεί όλος ο φάκελος. Από την εγκατάσταση του **Tomcat 6.0** η προεπιλεγμένη τοποθεσία αποθήκευσης είναι: *C:\apache-tomcat-6.0.20*. Όταν βρείτε τον φάκελο, επιλέξτε τον και πατήστε Open.

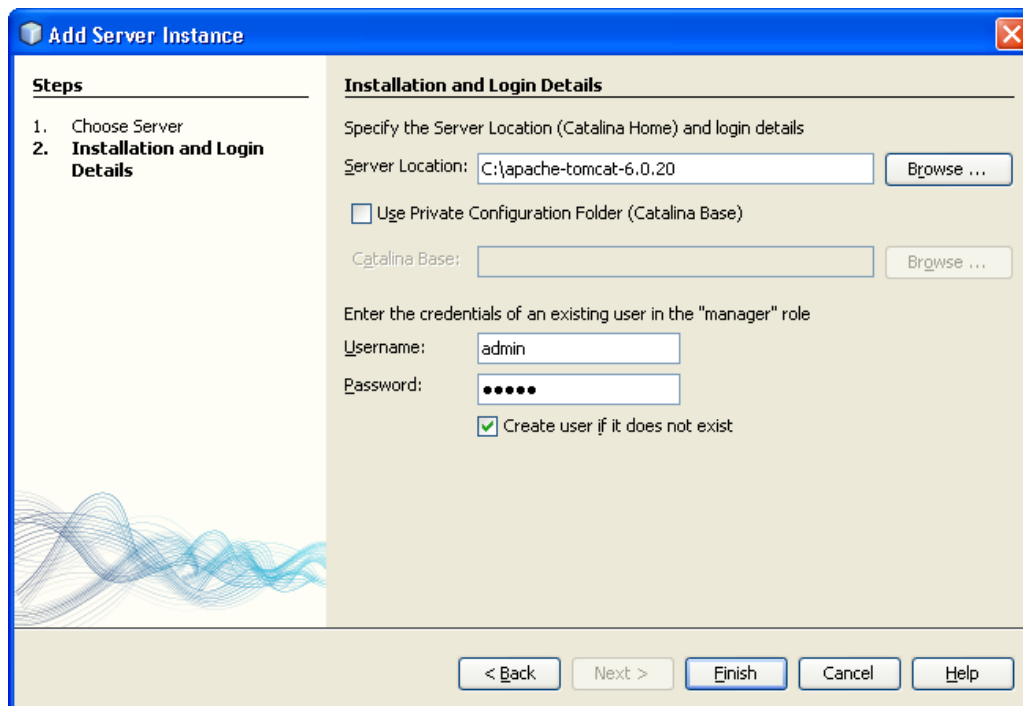


Εικόνα 3.3.9.5. Σύνδεση Apache Tomcat

6. Το μόνο που μένει, είναι να συμπληρώσετε τα πεδία του **Username** και **Password**. Δώστε τους κωδικούς του **Tomcat 6.0** στα πεδία και πατήστε **Finish**. Μην κάνετε καμία άλλη τροποποίηση.

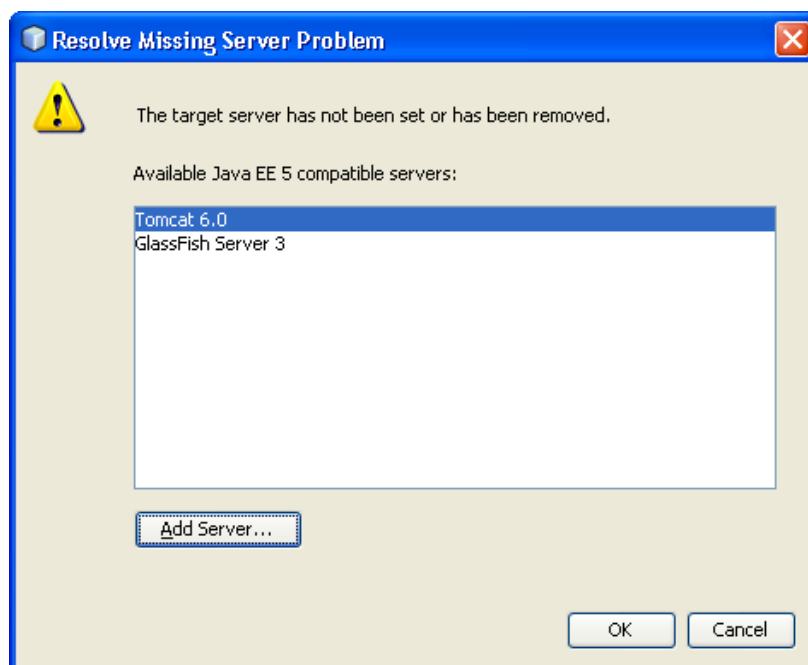
Οι προεπιλεγμένοι κωδικοί του **Tomcat Server 6.0 είναι:

- + Username: **admin**
- + Password: **admin**



Εικόνα 3.3.9.6. Σύνδεση Apache Tomcat

7. Στο παράθυρο που απέμεινε πατήστε **OK**. Η σύνδεση με τον **Tomcat 6.0** έχει ολοκληρωθεί.



Εικόνα 3.3.9.7. Σύνδεση Apache Tomcat

ΚΕΦΑΛΑΙΟ 4

JSP, Servlets και Γλώσσες Προγραμματισμού

4.1 – Εισαγωγή

Σε αυτό το κεφάλαιο θα αναφερθούν όλες οι γλώσσες προγραμματισμού που χρησιμοποιήθηκαν για την επίτευξη του αποτελέσματος. Με πιο σημαντική την γλώσσα JAVA, που χρησιμοποιήθηκε για την δημιουργία αρχείων JSP και Servlets, θα γνωριστούν τα χαρακτηριστικά αλλά και τα αρχεία που χρησιμοποιήθηκαν σε άλλες γλώσσες όπως η MySQL, η JavaScript και η CSS.

4.2- JAVA

4.2.1 – Εισαγωγή στην JAVA

Η γλώσσα προγραμματισμού *Java* είναι ένα πρόσφατο προϊόν της *Sun Microsystems Inc.*. Ξεκίνησε σαν μέρος ενός μεγαλύτερου σχεδίου που αφορούσε την ανάπτυξη λογισμικού για καταναλωτικά ηλεκτρονικά. Πρόκειται για μικρές, αξιόπιστες, φορητές, πραγματικού χρόνου συσκευές που στην αρχή βασιζόντουσαν στην C++. Αρκετά προβλήματα όμως παρουσιάστηκαν και η γλώσσα C++ δεν μπόρεσε να εφαρμοστεί τελικά. Χρειάστηκε να αναπτυχθεί μία νέα γλώσσα: η Java. Η Java, στην τελική της μορφή, βρήκε περαιτέρω εφαρμογή στην επίλυση μερικών προβλημάτων του σημερινού προγραμματισμού, όπως animation, την αλληλεπίδραση πραγματικού χρόνου (*real-time interaction*) και την εξερεύνηση του Web (*Web browsing*).

Ας πάρουμε το World Wide Web για παράδειγμα. Οι Web browsers περιορίζονται σε ότι μπορεί να προσφέρει η HTML γλώσσα: σελίδες δυο διαστάσεων με κείμενο, φωτογραφίες ή γραφικά και συνδέσμους (*links*). Κάποια μορφή επεκτασιμότητας προσέφερε η JavaScript από την *Netscape Corporation*, αλλά οι σελίδες παραμένουν βαρετές, στατικές και χαζές σε περιεχόμενο. Οι Web browsers δεν διαθέτουν ακόμα την ιδιότητα της αλληλεπίδρασης πραγματικού χρόνου. Χρησιμοποιώντας την Java γλώσσα, η Sun κατασκεύασε τον νέο World Wide Web browser, τον *HotJava*. Πλέον, η σχέση μεταξύ server και client αλλάζει. Ο νέος browser εκμεταλλεύεται την υπολογιστική δύναμη του client και αντί να κατεβάζει απλές στατικές σελίδες κειμένου και εικόνων, ο *HotJava* κατεβάζει μικρά προγράμματα σε Java που τρέχουν στον client. Ο browser είναι γρήγορος, ευέλικτος, πολύπλευρος και εύκολα προσαρμόσιμος. Από την ημέρα της δημοσιοποίησης της τον Μάιο του 1995, η Java έχει εξαπλωθεί σε όλο το Internet. Η Java έχει κιόλας λύσει τα περισσότερα προβλήματα στο μοντέλο client/server και έχει προάγει την χρήση του World Wide Web.

Παρ' όλα αυτά, επειδή η Java έχει σχεδιαστεί για να χρησιμοποιείται σε ανοικτά, δικτυωμένα περιβάλλοντα, έχει βαθιές συνέπειες στην ασφάλεια ενός συστήματος υπολογιστών. Επίσης, η δυνατότητα της να διανέμει αυτόματα εκτελέσιμα αρχεία μέσω του Internet επηρεάζει, με την σειρά της, την ασφάλεια των δικτύων. Τα εκτελέσιμα αρχεία από το Internet συχνά κρύβουν παγίδες και δύσκολα μπορεί να είναι κανείς σίγουρος ότι ο κώδικας είναι ελεύθερος από ιούς.

Πλεονεκτήματα της JAVA

Η Java χαρακτηρίζεται από τα εξής: απλή, αντικειμενοστραφής, συμβατή με δικτυακά πρωτόκολλα, ουδέτερη της υποκείμενης αρχιτεκτονικής, φορητή ασφαλής, υψηλής απόδοσης, δυναμική, σταθερή, interpreted και multithreaded. Στις ακόλουθες παραγράφους θα αναλύσουμε καθεμία από αυτές τις έννοιες.

Απλή

Στόχος της ομάδας της Sun που ανέπτυξε την Java, ήταν μια γλώσσα εύκολη στην χρήση, που δεν απαιτεί πολλή εξάσκηση και εκπαίδευση. Οι περισσότεροι προγραμματιστές στις μέρες μας δουλεύουν είτε με την C είτε με την C++. Έτσι, μολονότι η C++ δεν ήταν η κατάλληλη για το αρχικό σχέδιο, η Java σχεδιάστηκε βάσει της C++, με σκοπό να γίνει όσο το δυνατόν περισσότερο κατανοητή.

Η Java παραλείπει πολλά από τα σπανίως χρησιμοποιούμενα και δυσκολονόητα χαρακτηριστικά της C++, που δεν ωφελούν και πολύ την ευελιξία της γλώσσας. Προστέθηκαν διεργασίες, όπως η αυτόματη συλλογή των "σκουπιδιών" (*automatic garbage collection*), διευκολύνοντας τον προγραμματισμό σε Java. Μια κοινή πηγή πολυπλοκότητας της C++ και της C είναι η διαχείριση της μνήμης. Με την καινούργια διεργασία της αυτόματης συλλογής "σκουπιδιών", που συνιστάται από την περιοδική αποδέσμευση της μνήμης που δεν χρησιμοποιείται, μεγάλο μέρος από την δουλεία των προγραμματιστών αυτοματοποιείται και μειώνονται τα bugs.

Ένα πλεονέκτημα της Java που οφείλεται στην απλότητα της είναι ότι το μέγεθος των απαραίτητων εργαλείων. Ο *Java interpreter* και οι βασικές βιβλιοθήκες είναι μικρές και ο κώδικάς της Java είναι τόσο περιορισμένος σε μέγεθος που μπορεί άνετα να τρέξει σε οποιαδήποτε μικρή μηχανή και να κατέβει από το δίκτυο.

Αντικειμενοστραφής

Λέγοντας ότι μία γλώσσα προγραμματισμού είναι αντικειμενοστραφής, εννοούμε η τεχνική σχεδιασμού ενός προγράμματος συγκεντρώνεται σε αντικείμενα. Ένα αντικείμενο είναι ο συνδυασμός δεδομένων, διαδικασιών και λειτουργιών με βασική ιδιότητα την απόκρυψη του συνδυασμού αυτού. Το κάθε αντικείμενο, δηλαδή, αντιμετωπίζεται σαν ένα "μαύρο κουτί". Τα αντικείμενα δεν είναι ανεξάρτητα μεταξύ τους, αλλά βρίσκονται σε σχέση αλληλεξάρτησης με τα υπόλοιπα. Υπάρχει η έννοια της κληρονομικότητας μεταξύ των αντικειμένων, δηλαδή ένα αντικείμενο μπορεί να κληρονομήσει δεδομένα από άλλα αντικείμενα.

Οι γλώσσες αντικειμενοστραφή προγραμματισμού είναι γλώσσες υψηλού επιπέδου, αφαιρετικές, αποτελεσματικές, γρήγορες και χρησιμοποιούνται για την δημιουργία μεγάλων και σημαντικών εφαρμογών. Οι αντικειμενοστραφής ευκολίες της Java είναι ίδιες με αυτές της C++, με επεκτάσεις από την Objectine C.

Συμβατή με Δίκτυα

Η Java έχει μια μεγάλη βιβλιοθήκη από ρουτίνες για την επιτυχημένη συνεργασία με τα πρωτόκολλα HTTP και FTP. Κατ' αυτόν τον τρόπο, οι δικτυακές συνδέσεις δημιουργούνται ευκολότερα από ότι με την C ή την C++. Τα προγράμματα σε Java μπορούν να έχουν πρόσβαση μέσω δικτύου σε αντικείμενα, με την ίδια άνεση που ένας χρήστης προσπελάζει ένα τοπικό σύστημα αρχείων.

Σταθερή

Η Java προορίζεται για την σύνταξη προγραμμάτων που θα είναι αξιόπιστα από όλες τις πλευρές. Δίνεται έμφαση στον από νωρίς έλεγχο για πιθανά προβλήματα και στον έλεγχο σε πραγματικό χρόνο και στην εξάλειψη καταστάσεων που προκαλούν λάθη.

Η μεγαλύτερη διαφορά μεταξύ Java και C/C++ είναι το γεγονός ότι η Java έχει ένα μοντέλο δεικτών που εξαφανίζει την πιθανότητα της επαναχρησιμοποίησης της μνήμης και την καταστροφή των δεδομένων. Αντί για αριθμητικούς δείκτες (*pointer arithmetic*), η Java έχει πραγματικούς πίνακες (*true arrays*). Οι προγραμματιστές της Java δεν έχουν να φοβηθούν την ακούσια (ή μη) τροποποίηση της μνήμης, γιατί δεν υπάρχουν δείκτες (*pointers*). Εξάλλου, τα προγράμματα σε Java δεν μπορούν να αποκοτήσουν μη εγκεκριμένη πρόσβαση στην μνήμη.

Ασφαλής

Η Java προορίζεται για χρήση σε ανοικτά, δικτυωμένα περιβάλλοντα. Γι' αυτό το λόγο, ιδιαίτερη προσοχή έχει δοθεί στην ασφάλεια που παρέχει η γλώσσα. Η Java επιτρέπει την κατασκευή προγραμμάτων ελεύθερων από ιούς και η τροποποίηση τους είναι αδύνατη. Οι τεχνικές πιστοποίησης ταυτότητας βασίζονται στην ασύμμετρη κρυπτογραφία.

Υπάρχει μεγάλη σχέση μεταξύ του τρόπου διαχείρισης της μνήμης και της παρεχόμενης ασφάλειας. Αλλαγές στην σημασιολογία των δεικτών της μνήμης κάνουν αδύνατη την μη έγκυρη πρόσβαση στα δεδομένα της μνήμης ή της πρόσβασης των δεδομένων των αντικειμένων. Με αυτόν τον τρόπο καταπολεμούνται οι περισσότεροι ιοί.

Ουδέτερη της Υποκείμενης Αρχιτεκτονικής

Η Java έχει σχεδιαστεί για να υποστηρίζει δικτυακές εφαρμογές. Ένα δίκτυο, όμως, αποτελείται από ποικιλία διαφορετικών συστημάτων, με διαφορετικές CPU και λειτουργικά συστήματα. Για να μπορούν οι Java εφαρμογές να εκτελούνται παντού στο δίκτυο, το πρόγραμμα Java πρέπει να περάσει από δύο διαδικασίες ώστε να καταλήξει σε εκτελέσιμη μορφή. Πρώτα ο μεταγλωττιστής, μετατρέπει τον πηγαίο κώδικα του προγράμματος σε μία ενδιάμεση γλώσσα που καλείται *Java bytecodes*. Τα Java bytecodes είναι ανεξάρτητα της πλατφόρμας και με χρήση του ερμηνευτή (*interpreter*) κάθε bytecode εντολή μετατρέπεται σε κατάλληλη δυαδική μορφή για να τρέξει στον εκάστοτε υπολογιστή. Η μεταγλώττιση (*compilation*) συμβαίνει μόνο μια φορά για κάθε Java πρόγραμμα, η ερμηνεία (*interpretation*) γίνεται κάθε φορά που το πρόγραμμα εκτελείται. Το παρακάτω σχήμα επιδεικνύει πως λειτουργεί αυτή η φιλοσοφία.

Τα Java bytecodes μπορούμε να τα φανταστούμε σαν την γλώσσα μηχανής για την *Java Virtual Machine (JVM)*. Κάθε Java ερμηνευτής (π.χ. ένας Web browser που μπορεί να τρέχει applets) είναι μια λογισμική εφαρμογή του της Java Virtual Machine. Η JVM αναλαμβάνει να μετατρέψει τα bytecodes σε κατάλληλη εκτελέσιμη μορφή, ανάλογα με το υποκείμενο software και hardware.

Η τεχνική που περιγράφηκε παραπάνω καλείται "*write once, run anywhere*". Το Java πρόγραμμα μεταγλωττίζεται μία φορά σε Java bytecodes με τον μεταγλωττιστή της Java. Έπειτα, τα bytecodes μπορούν να τρέξουν σε οποιαδήποτε μηχανή που έχει μία εφηρμοσμένη JVM (ο ερμηνευτής).

Φορητή

Το γεγονός ότι είναι ανεξάρτητη της υποκείμενης πλατφόρμας αποτελεί μεγάλο μέρος του ότι είναι φορητή, άλλα υπάρχουν και άλλα σημεία που χαρακτηρίζουν την φορητότητα της.

Σε αντίθεση με την C/C++ δεν υπάρχουν καθόλου χαρακτηριστικά που εξαρτούνται από την CPU του υπολογιστή. Έτσι, τα μεγέθη των πρωταρχικών τύπων δεδομένων είναι καθορισμένα και η συμπεριφορά τους είναι παντού η ίδια. Για παράδειγμα, "*int*" σημαίνει πάντα έναν 32 bit ακέραιο και "*float*" πάντα αντιπροσωπεύει έναν 32 bit floating αριθμό.

Interpreted

Τα Java bytecodes μεταφράζονται σε πραγματικό χρόνο σε εντολές μηχανής που εξαρτώνται από την εκάστοτε πλατφόρμα, και δεν αποθηκεύονται πουθενά. Η διαδικασία είναι γρήγορη και πιο αποτελεσματική. Μαζί με τα bytecodes μεταφέρονται πληροφορίες που μπορούν να χρησιμοποιηθούν κατά την εκτέλεση και παρέχουν την βάση για τους ελέγχους που πραγματοποιεί ο συνδετής (*linker*). Επίσης τα προγράμματα γίνονται πιο επιδεκτικά σε debugging διαδικασίες.

Υψηλής Απόδοσης

Η διαδικασία παραγωγής των εντολών μηχανής είναι απλή και γρήγορη. Ο κώδικας που προκύπτει είναι αποτελεσματικός. Ο μεταγλωττιστής από την μεριά του εφαρμόζει αυτόματη κατανομή των καταχωρητών (*automatic register allocation*) όταν παράγει τα bytecodes. Η τελική μορφή του κώδικα (εκτελέσιμη δυαδική μορφή) είναι μικρή σε μέγεθος και ταχύτατη στην εκτέλεση.

Multithreaded

Τα προγράμματα σε Java έχουν την δυνατότητα να αντιμετωπίζουν πολλές καταστάσεις και διαδικασίες ταυτόχρονα. Σε αντίθεση, η C και C++ είναι *single-threaded* γλώσσες. Τα πλεονεκτήματα του *multithreading* είναι η καλύτερη πραγματικού χρόνου συμπεριφορά και η καλύτερη αλληλεπιδραστική ανταπόκριση.

Δυναμική

Η Java είναι πιο δυναμική γλώσσα από την C ή C++. Έχει αναπτυχθεί για να προσαρμοστεί σε ένα εξελισσόμενο περιβάλλον. Οι βιβλιοθήκες εργαλείων αναπτύσσονται ελεύθερα με την πρόσθεση νέων μεθόδων και μεταβλητών, χωρίς να επηρεάζονται οι ήδη υπάρχουσες εφαρμογές.

4.2.2 – Βασικά Χαρακτηριστικά JSP

Οι *JSPs* (*JavaServer Pages*) είναι μια τεχνολογία που έχει δημιουργηθεί από την εταιρεία Sun Microsystems για να μπορεί να δημιουργεί δυναμικό περιεχόμενο (dynamic content) στο Web. Πρόκειται για HTML έγγραφα (ιστοσελίδες) τα οποία αναμειγνύονται με τη γλώσσα προγραμματισμού Java, η οποία και έχει τη δυνατότητα να παρέχει (δημιουργεί) αυτό το δυναμικό περιεχόμενο.

Οι JSPs είναι μια εφαρμογή στην πλευρά του server (server-side application), που σημαίνει ότι δέχονται μια αίτηση (request) και παράγουν μια απόκριση ή απάντηση (response). Σε γενικές γραμμές, οι αιτήσεις γίνονται από έναν Web client και η απόκριση είναι ένα παραγόμενο HTML έγγραφο (ιστοσελίδα) το οποίο στέλνεται πίσω στον Web client. Επειδή οι JSPs είναι μια εφαρμογή στην πλευρά του server, έχουν πρόσβαση σε πηγές (resources) στον server, όπως είναι τα Servlets, JavaBeans, EJBs, αλλά και σε βάσεις δεδομένων.

4.2.3 – Αρχεία JSP

- **index.jsp:**

Η index.jsp είναι η κεντρική σελίδα του project, όπου σ' αυτήν προβάλλονται δυναμικά όλες οι υπόλοιπες σελίδες jsp εμφανίζοντας την κάθε μια ανάλογα με το attribute που του δίνουμε στο link. Η κάθε jsp σελίδα φορτώνεται στην **index** χρησιμοποιώντας την εντολή ***jsp:include***. Στην συγκεκριμένη σελίδα υλοποιείται το κεντρικό μενού και όλο το εικαστικό περιβάλλον της εφαρμογής. Τα αρχεία jsp που συμπεριλαμβάνονται σε αυτή την σελίδα είναι:

- calendar.jsp
- companies.jsp
- financial_view.jsp
- home.jsp
- login.jsp
- phonebook.jsp

- **calendar.jsp:**

Είναι το αρχείο, το οποίο προβάλλει την σελίδα Ημερολόγιο, όπου δυναμικά με την χρήση iframe φορτώνονται τα αρχεία :

- calendar.java
- activities.java

- **companies.jsp:**

Είναι το αρχείο, το οποίο προβάλλει την σελίδα Εταιρίες, όπου δυναμικά με την χρήση iframe φορτώνονται τα αρχεία :

- companies.java
- addviewcust.java
- incomesExpenses.java

- **financial_view.jsp:**

Είναι το αρχείο, το οποίο προβάλλει την σελίδα Οικονομική Επισκόπηση, όπου δυναμικά με την χρήση iframe φορτώνονται τα αρχεία :

- finview.java

- **home.jsp:**

Είναι το αρχείο, το οποίο προβάλλει την σελίδα Αρχική, όπου δυναμικά με την χρήση iframe φορτώνονται τα αρχεία :

- home.java

- **login.jsp:**

Είναι το αρχείο, το οποίο προβάλλει την σελίδα σύνδεσης του χρήστη με τον λογαριασμό του για την είσοδο του στην εφαρμογή.

- **phonebook.jsp:**

Είναι το αρχείο, το οποίο προβάλλει την σελίδα Τηλεφωνικός Κατάλογος, όπου δυναμικά με την χρήση iframe φορτώνονται τα αρχεία :

- phonebook.java

4.2.4 – UML

Τα παρακάτω αρχεία, **Java Servlets** και **Java Class**, μελετούνται σε διαγράμματα **UML**. Υπάρχουν τρεις τύποι διαγραμμάτων στην UML. Αυτά είναι,α)τα Δομικά Διαγράμματα (**Structural Diagrams**),β) τα Διαγράμματα Συμπεριφοράς (**Behavior Diagrams**) και γ) τα Διαγράμματα Διαχείρισης Μοντέλου (**Model Management Diagrams**). Εμείς θα χρησιμοποιήσουμε τον πρώτο τύπο UML, τα **Δομικά Διαγράμματα**.

Τα Δομικά Διαγράμματα (**Structural Diagrams**) περιγράφουν την εσωτερική λογική δομή ενός συστήματος, δηλαδή τα συστατικά του και τις σχέσεις μεταξύ τους.

- Διάγραμμα Κλάσεων (**Class Diagram**)
- Διάγραμμα Αντικειμένων (**Object Diagram**)
- Διάγραμμα Συνιστωσών (**Component Diagram**)
- Παραταξιακό Διάγραμμα (**Deployment Diagram**)

4.2.5 – Βασικά Χαρακτηριστικά των Java Servlets

Είναι ένα είδος προγράμματος που μπορεί να εκτελεστεί όταν ζητηθεί από έναν browser. Τα servlets είναι γραμμένα σε Java, φορτώνονται στον διακομιστή Web και εκτελούνται όταν μία κατάλληλη εντολή HTTP που ζητά την εκτέλεση τους λαμβάνεται από τον διακομιστή. Τα servlets έχουν ορισμένα σημαντικά χαρακτηριστικά:

- Μπορούν να εκτελεστούν χωρίς αλλαγές σε διάφορους τύπους διακομιστών.
- Από την στιγμή που τα servlets είναι γραμμένα σε Java έχουν πρόσβαση σε διάφορα εργαλεία της Java όπως η CORBA, RMI, εργαλεία ασφαλείας της Java και εργαλεία σύνδεσης βάσεων δεδομένων.
- Τα servlets είναι εγκατεστημένα στην μνήμη. Με τον προγραμματισμό CGI σε γλώσσες όπως η Perl, κάθε φορά που ένα καινούργιο αίτημα προωθείται από τον Web browser μια καινούργια επεξεργασία πρέπει να ξεκινήσει και να τερματιστεί με την λήξη της σύνδεσης και το κατάλληλο πρόγραμμα πρέπει να φορτωθεί και να ξεφορτωθεί από τη μνήμη. Αυτό σημαίνει ότι το φορτίο μπορεί να είναι μεγάλο για τον διακομιστή Web.
- Από την στιγμή που τα servlets βασίζονται στην Java περιλαμβάνουν όλα τα χαρακτηριστικά ασφαλείας της γλώσσας.
- Τα servlets μπορούν να διατηρήσουν την κατάσταση τους, να διατηρήσουν κάποιο είδος μνήμης, ανάμεσα στα αιτήματα, κάτι που, στα πρώτα βήματα του Παγκόσμιου Ιστού, ήταν δύσκολο. Αυτό σημαίνει ότι τα servlets μπορούν να θυμηθούν δεδομένα και λεπτομέρειες ενός προηγούμενου αιτήματος.
- Το αντικειμενοστραφές μοντέλο της Java βοηθά τον κώδικα της να είναι πιο κομψός από κώδικα γραμμένο σε άλλες γλώσσες όπως η C και η Perl. Αν εξετάσετε τα προγράμματα Perl CGI θα δείτε, συχνά, ότι είναι τεράστια, μονοκόμματα και δυσνόητα. Η προσανατολισμένη στ' αντικείμενα φύση της Java παρέχει επίσης και την δυνατότητα μεγαλύτερης επαναχρησιμοποίησης κώδικα.
- Τα servlets, επειδή ενσωματώνονται καλά με τις τεχνολογίες ασφαλείας που σχετίζονται με την Java, μπορούν να διαμορφωθούν ώστε να παρέχουν υψηλά επίπεδα ασφαλείας.

4.2.6 – Αρχεία Java Servlet

Η βασική δομή ενός Servlet αποτελείται από κάποιες συγκεκριμένες μεθόδους και μια μεταβλητή. Καθένα από τα αρχεία Servlet που χρησιμοποιήθηκαν περιέχει τα παραπάνω. Για να μην αναφέρονται συνεχώς σε καθένα από τα αρχεία, θα εξηγηθούν αμέσως τώρα.

- `protected void processRequest(HttpServletRequest request, HttpServletResponse response)`
Μέθοδος όπου επιστρέφει το αποτέλεσμα που δημιουργείται στο HttpServlet. Εδώ γράφουμε τον HTML κώδικα που θέλουμε να εμφανίσουμε.
 - `protected void doGet(HttpServletRequest request, HttpServletResponse response)`
Χειρίζεται HTTP αιτήματα τύπου GET. Η μέθοδος GET προσαρτά τα δεδομένα στο URL και τα στέλνει στον Server με αυτόν τον τρόπο. Αυτό γίνεται μέσω HTML φόρμας χρησιμοποιώντας την ιδιότητα(METHOD).
 - `protected void doPost(HttpServletRequest request, HttpServletResponse response)`
Χειρίζεται HTTP αιτήματα τύπου POST. Η Μέθοδος POST συσκευάζει τα δεδομένα σε ένα πακέτο και στέλνει το πακέτο στον Server. Αυτό γίνεται μέσω HTML φόρμας χρησιμοποιώντας την ιδιότητα(METHOD).
 - `public String getServletInfo()`
Επιστρέφει μια σύντομη περιγραφή του Servlet
 - `public void connectSql()`
Μέθοδος που δημιουργεί την σύνδεση με την βάση δεδομένων.
 - `public Connection con`
Κλάση που παρέχει υπηρεσίες σύνδεσης με μια βάση δεδομένων. Η εκτέλεση δηλώσεων SQL σχετίζεται με μία βάση δεδομένων συνδεδεμένη με ένα αντικείμενο Connection.
- **activities.java:**
Σε αυτό το αρχείο, γίνεται η σχεδίαση για την προβολή, εισαγωγή, επεξεργασία και διαγραφή των ενεργειών του διαχειριστή, όπως ραντεβού, υποχρεώσεις κ.α.
Φορτώνεται μέσα στο αρχείο `calendar.jsp` κάνοντας χρήση του `iframe`.



Εικόνα 4.2.1. UML του Servlet activities

Μέθοδοι:

- **Public boolean datevalidate(String date)**

Η μέθοδος αυτή ελέγχει αν η ημερομηνία που έχει δοθεί έχει τον σωστό τύπο σύνταξης για ημερομηνίες, ή αν η ημερομηνία είναι υπαρκτή.

Αν η τιμή που επιστρέφει είναι αποδεκτή, τότε η μέθοδος παίρνει τιμή **true**, αλλιώς παίρνει τιμή **false**.

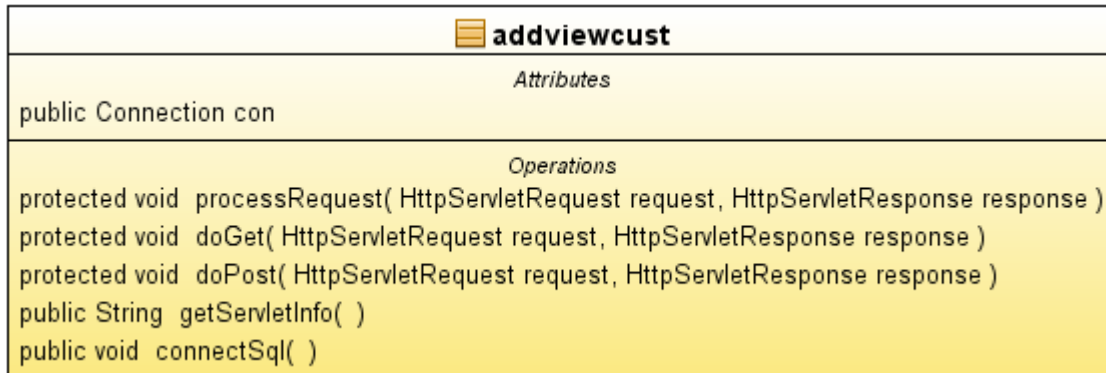
- **String date:** Μεταβλητή τύπου String η οποία για να επιστρέψει η μέθοδος **true** πρέπει να

είναι ημερομηνία της μορφής “yyyy-mm-dd”. Την μεταβλητή αυτή την παίρνουμε από τα πεδία της **Ημ. Έναρξης** και **Ημ. Λήξης** κατά την καταχώρηση και επεξεργασία των **Ενεργειών** στη σελίδα **Ημερολόγιο**.

- **addviewcust.java:**

Σε αυτό το αρχείο, γίνεται η σχεδίαση για την προβολή, εισαγωγή, επεξεργασία και διαγραφή των πελατών ή των προμηθευτών του διαχειριστή.

Φορτώνεται μέσα στο αρχείο companies.jsp κάνοντας χρήση του `iframe`.

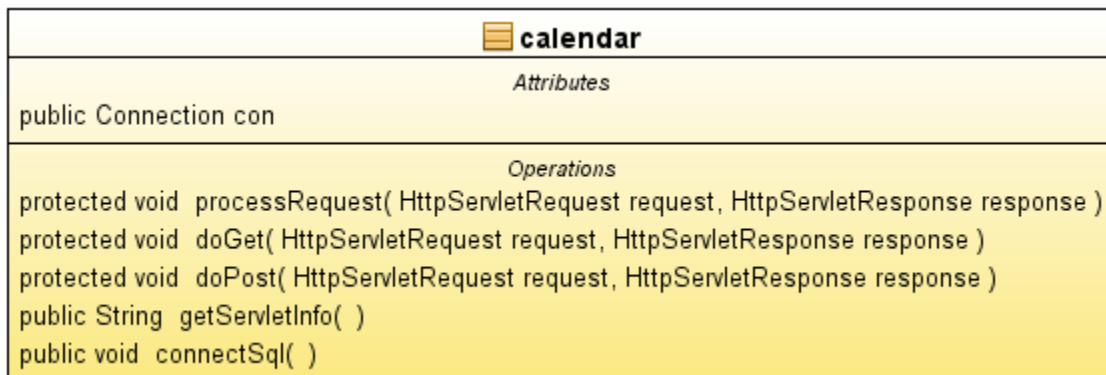


Εικόνα 4.2.2. UML του Servlet addviewcust

- **calendar.java:**

Σε αυτό το αρχείο, γίνεται μια προβολή γενεθλίων των επαφών που είναι αποθηκευμένες στον τηλεφωνικό κατάλογο, ξεκινώντας από την τρέχουσα ημέρα έως και τις επόμενες 5.

Φορτώνεται μέσα στο αρχείο calendar.jsp κάνοντας χρήση του `iframe`.

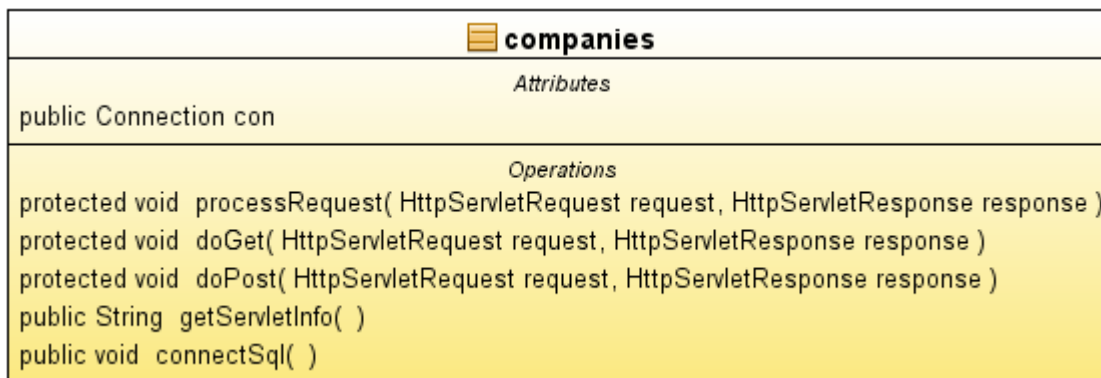


Εικόνα 4.2.3. UML του Servlet calendar

- **companies.java:**

Σε αυτό το αρχείο, γίνεται η σχεδίαση για την προβολή, εισαγωγή, επεξεργασία και διαγραφή των συνεργαζόμενων εταιριών του διαχειριστή.

Φορτώνεται μέσα στο αρχείο companies.jsp κάνοντας χρήση του `iframe`.

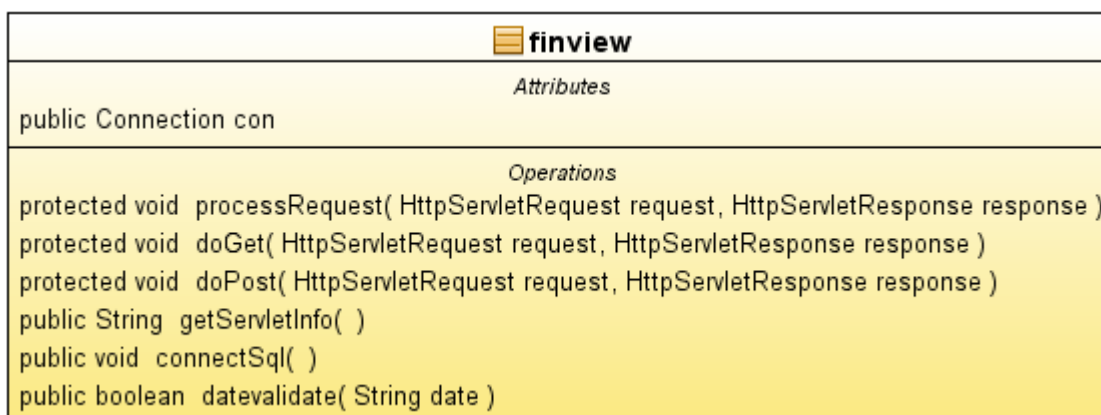


Εικόνα 4.2.4. UML του Servlet companies

▪ **finview.java:**

Σε αυτό το αρχείο, γίνεται η σχεδίαση για την προβολή, εισαγωγή, επεξεργασία και διαγραφή των εσόδων και των εξόδων του διαχειριστή. Επίσης, αναπτύσσεται ένα μενού αναζήτησης για την εύρεση εσόδων και εξόδων βάσει διαφόρων παραμέτρων, όπως ημερομηνιών, είδος πληρωμής, προμηθευτών/πελατών κ.α.

Φορτώνεται μέσα στο αρχείο financial_view.jsp κάνοντας χρήση του iframe.



Εικόνα 4.2.5. UML του Servlet finview

Μέθοδοι:

- **Public boolean datevalidate(String date)**

Η μέθοδος αυτή ελέγχει αν η ημερομηνία που έχει δοθεί έχει τον σωστό τύπο σύνταξης για ημερομηνίες, ή αν η ημερομηνία είναι υπαρκτή.

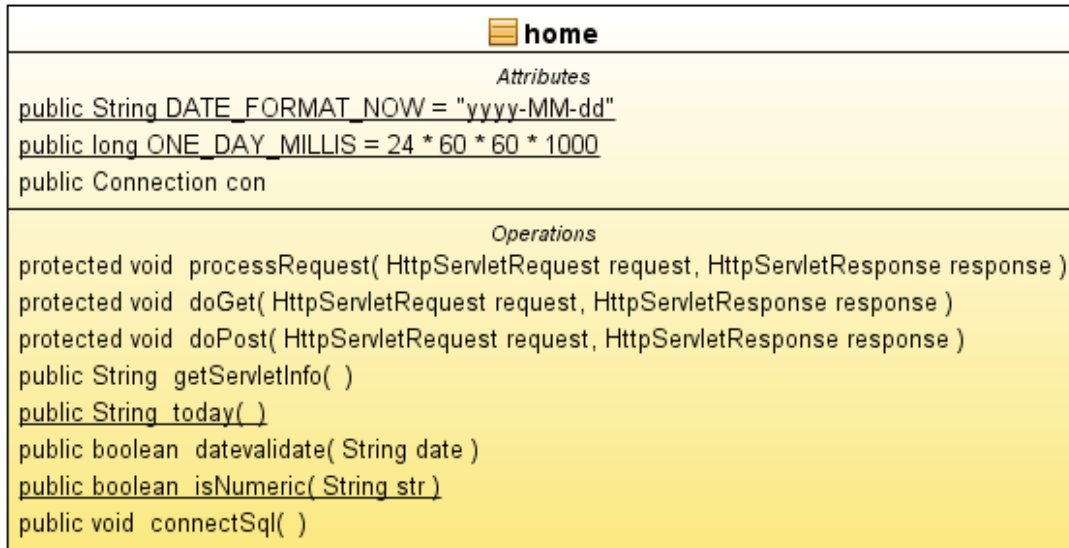
Αν η τιμή που επιστρέφει είναι αποδεκτή, τότε η μέθοδος παίρνει τιμή **true**, αλλιώς παίρνει τιμή **false**.

- **String date:** Μεταβλητή τύπου String η οποία για να επιστρέψει η μέθοδος **true** πρέπει να είναι ημερομηνία της μορφής “yyyy-mm-dd”. Την μεταβλητή αυτή την παίρνουμε από τα πεδία των **ημερομηνιών** κατά την καταχώρηση και επεξεργασία όλων των δεδομένων της σελίδας **Οικονομική Επισκόπηση**.

▪ **home.java:**

Σε αυτό το αρχείο, υλοποιείται η αρχική σελίδα στην οποία προβάλλονται οι υποχρεώσεις που εκκρεμούν. Τέτοιες υπηρεσίες είναι πληρωμές, εσόδων και εξόδων, ενέργειες και γενέθλια επαφών.

Φορτώνεται μέσα στο αρχείο home.jsp κάνοντας χρήση του iframe.



Εικόνα 4.2.6. UML του Servlet home

Μεταβλητές:

- **Public String DATE_FORMAT_NOW= "yyyy-MM-dd"**
Μεταβλητή που επιστρέφει το συγκεκριμένο τύπο ημερομηνίας.
- **Public long ONE_DAY_MILLIS= 24*60*60*1000**
Μεταβλητή που μετατρέπει την μέρα σε milliseconds.

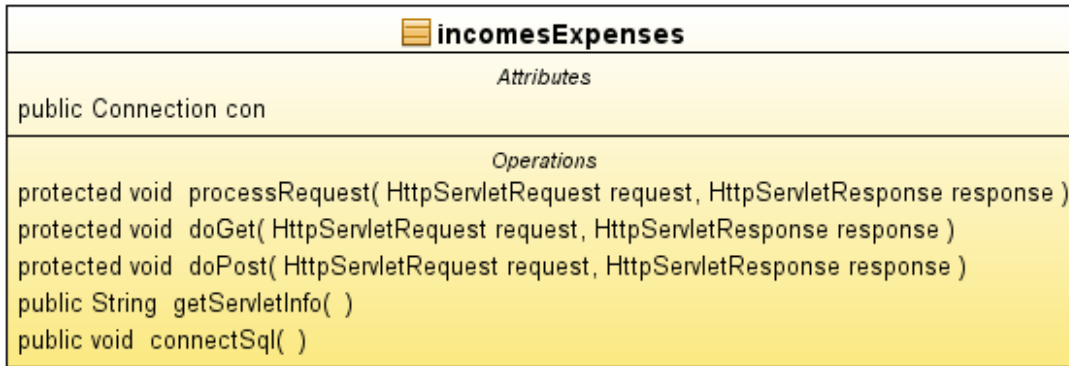
Μέθοδοι:

- **Public String today()**
Η μέθοδος αυτή επιστρέφει την τρέχουσα ημερομηνία.
- **Public boolean datevalidate(String date)**
Η μέθοδος αυτή ελέγχει αν η ημερομηνία που έχει δοθεί έχει τον σωστό τύπο σύνταξης για ημερομηνίες, ή αν η ημερομηνία είναι υπαρκτή.
Αν η τιμή που επιστρέφει είναι αποδεκτή, τότε η μέθοδος παίρνει τιμή **true**, αλλιώς παίρνει τιμή **false**.
 - **String date:** Μεταβλητή τύπου String η οποία για να επιστρέψει η μέθοδος **true** πρέπει να είναι ημερομηνία της μορφής "yyyy-mm-dd". Την μεταβλητή αυτή την παίρνουμε από τα πεδία των **ημερομηνιών** κατά την καταχώρηση των **Εσόδων** και **Εξόδων** στη σελίδα Αρχική.
- **Public boolean isNumeric(String str)**
Μέθοδος όπου παίρνει σαν όρισμα μια μεταβλητή String και ελέγχει αν είναι αριθμός.
 - **String str:** Μεταβλητή τύπου String που ορίζεται στην μέθοδο isNumeric. Την τιμή της την παίρνουμε από τα πεδία των **Ποσών** στην Αρχική σελίδα. Για να ισχύσει η μέθοδος πρέπει να είναι αριθμός **ακέραιος ή δεκαδικός**. Μόνο τότε μας επιστρέφει **true**.

▪ **incomesExpenses.java:**

Σε αυτό το αρχείο, γίνεται η σχεδίαση για την προβολή, εισαγωγή, επεξεργασία και διαγραφή των ειδών πληρωμής του διαχειριστή.

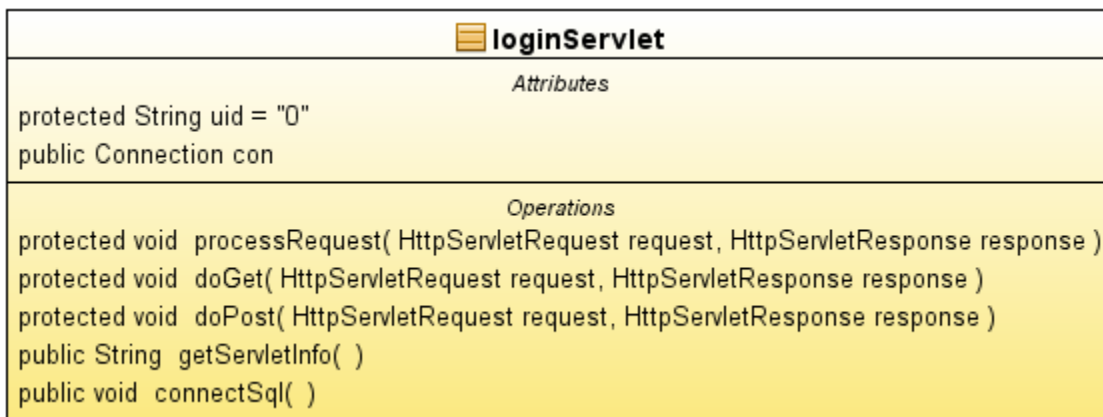
Φορτώνεται μέσα στο αρχείο companies.jsp κάνοντας χρήση του iframe.



Εικόνα 4.2.7. UML του Servlet incomesExpenses

▪ **loginServlet.java:**

Είναι το αρχείο στο οποίο γίνεται ο έλεγχος του **username** και του **password** του χρήστη που συνδέεται στο σύστημα και αν όντως γίνει η πιστοποίηση τότε δημιουργείται το **Session** βάσει του **id** του χρήστη.



Εικόνα 4.2.8. UML του Servlet loginServlet

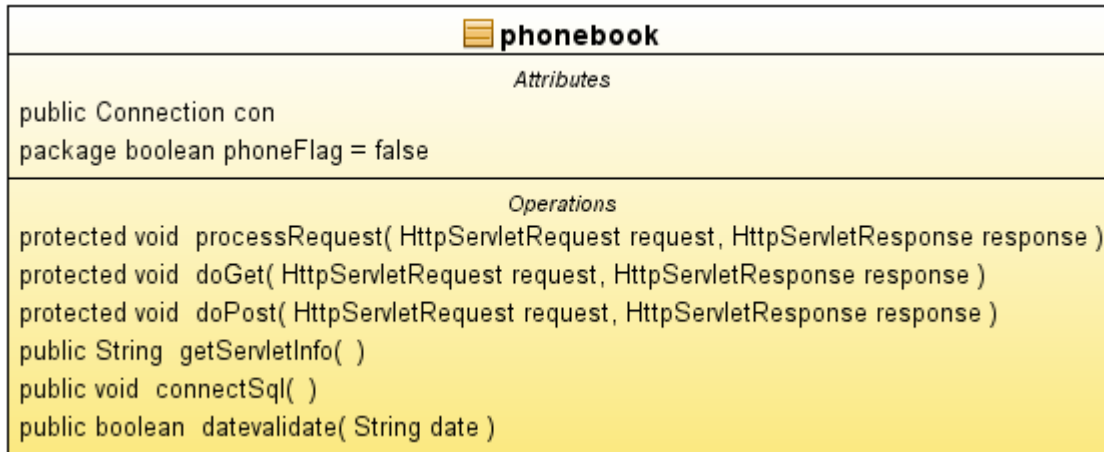
Μεταβλητές:

- **Protected String uid= "0"**

Μεταβλητή που γίνεται ίση με το id του χρήστη μετά την είσοδο του στο σύστημα.

▪ **phonebook.java:**

Στο συγκεκριμένο αρχείο, υπάρχει όλος ο κώδικας που περιέχει η σελίδα, Τηλεφωνικός Κατάλογος. Εδώ γίνεται η εισαγωγή κάποιου συνεργάτη, συγγενή, φίλου με χρήσιμες πληροφορίες, όπως όνομα, επίθετο, γενέθλια, τηλέφωνο. Επίσης, μπορεί να γίνει επεξεργασία, ή διαγραφή κάποιου. Επίσης έχει την δυνατότητα αναζήτησης ονόματος, είτε βάσει ευρετηρίου είτε γράφοντας αρχικά ονόματος ή επίθετου.



Εικόνα 4.2.9. UML του Servlet phonebook

Μεταβλητές:

- **Package Boolean phoneFlag= false**

Μεταβλητή που επιστρέφει true ή false αν το πεδίο του τηλεφώνου είναι κενό κατά την καταχώρηση νέας επαφής.

Μέθοδοι:

- **Public boolean datevalidate(String date)**

Η μέθοδος αυτή ελέγχει αν η ημερομηνία που έχει δοθεί έχει τον σωστό τύπο σύνταξης για ημερομηνίες, ή αν η ημερομηνία είναι υπαρκτή.

Αν η τιμή που επιστρέφει είναι αποδεκτή, τότε η μέθοδος παίρνει τιμή **true**, αλλιώς παίρνει τιμή **false**.

- **String date:** Μεταβλητή τύπου String η οποία για να επιστρέψει η μέθοδος **true** πρέπει να είναι ημερομηνία της μορφής “yyyy-mm-dd”. Την μεταβλητή αυτή την παίρνουμε από τα πεδία των **ημερομηνιών** κατά την καταχώρηση και επεξεργασία των **Επαφών** στη σελίδα **Τηλεφωνικός Κατάλογος**.

4.2.7 – Βασικά Χαρακτηριστικά Java Class

Η **κλάση** (class) είναι ένα πρότυπο που χρησιμοποιείται για δημιουργία πολλαπλών αντικειμένων με παρόμοια χαρακτηριστικά .

Οι κλάσεις περιέχουν όλα τα χαρακτηριστικά ενός συγκεκριμένου συνόλου αντικειμένων. Όταν γράφετε ένα πρόγραμμα σε μια αντικειμενοστρεφή γλώσσα προγραμματισμού, δεν ορίζετε μεμονωμένα αντικείμενα αλλά κλάσεις αντικειμένων.

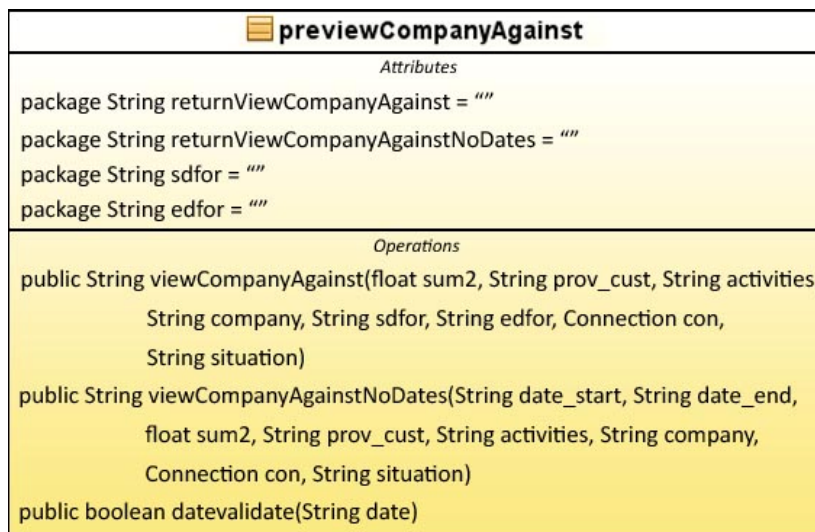
Στον πραγματικό κόσμο υπάρχουν πολλά αντικείμενα από το ίδιο είδος. Για παράδειγμα το ποδήλατο κάποιου είναι ένα από τα πολλά ποδήλατα που κυκλοφορούν. Στον αντικειμενοστραφή προγραμματισμό το προαναφερθέν ποδήλατο είναι ένα αντικείμενο (instance - ομότυπο) της κλάσης των ποδηλάτων. Γενικά τα ποδήλατα έχουν μερικά στοιχεία που χαρακτηρίζουν το είδος τους (π.χ. έχουν φρένα, έχουν ταχύτητες) και μερικά χαρακτηριστικά που είναι διαφορετικά σε κάθε ποδήλατο (π.χ. τρέχουσα ταχύτητα, τρέχων ρυθμός, μέγεθος ρόδας). Όταν ένα εργοστάσιο φτιάχνει ποδήλατα τα φτιάχνει με τα ίδια χαρακτηριστικά . Θα ήταν πολύ αναποτελεσματικό αν έφτιαχνε για κάθε ποδήλατο ένα καινούργιο πατρών. Στο συγκεκριμένο παράδειγμα μια κλάση είναι το πατρών από ένα είδος ποδηλάτου και τα αντικείμενα της είναι όλα τα παραγόμενα ποδήλατα από αυτό το πατρών.

Τα παραπάνω ισχύουν και στον αντικειμενοστραφή προγραμματισμό. Ο προγραμματιστής έχει πλεονεκτήματα δημιουργώντας ένα προσχέδιο για αντικείμενα με ίδιες ιδιότητες. Δηλαδή μπορεί να λεχθεί σε πιο ελεύθερη μορφή ότι κλάση είναι το προσχέδιο ή το πρωτότυπο, το οποίο ορίζει κοινές μεταβλητές και κοινές μεθόδους (συναρτήσεις) για τα αντικείμενα του ίδιου είδους.

4.2.8 – Αρχεία Java Class

▪ `previewCompanyAgainst.java`:

Στην κλάση `previewCompanyAgainst` δημιουργούνται οι μέθοδοι που έχουν σχέση με τον πίνακα των **Εξόδων** μετά την επιλογή του κουμπιού εμφάνιση με ή και χωρίς ημερομηνίες(προβολή αποτελεσμάτων ανεξάρτητα από τις ημερομηνίες) . Η κλάση αυτή φορτώνεται στην σελίδα της **Οικονομικής Επισκόπησης**.



Εικόνα 4.2.10. UML του Java Class `previewCompanyAgainst`

Μεταβλητές:

- **Package String returnUrlCompanyAgainst= ""**
Μεταβλητή που επιστρέφει τα αποτελέσματα μετά την επιλογή του κουμπιού **Εμφάνιση** στον πίνακα των εξόδων.
- **Package String returnUrlCompanyAgainstNoDates= ""**
Μεταβλητή που επιστρέφει τα αποτελέσματα μετά την επιλογή του κουμπιού **Εμφάνιση χωρίς Ημερομηνίες** στον πίνακα των εξόδων. Σε αυτή την περίπτωση οι ημερομηνίες είναι απενεργοποιημένες και εμφανίζονται όλα τα αποτελέσματα που υπάρχουν στην βάση δεδομένων.
- **Public String sdfor= ""**
Μεταβλητή που ισούται με την **Ημερομηνία Έναρξης** μετά την **Εμφάνιση**.
- **Public String edfor= ""**
Μεταβλητή που ισούται με την **Ημερομηνία Λήξης** μετά την **Εμφάνιση**.

Μέθοδοι:

- **public String viewCompanyAgainst(float sum2, String prov_cust, String activities, String company, String sdfor, String edfor, Connection con, String situation)**
Μέθοδος στην οποία μετά την επιλογή του κουμπιού **Εμφάνιση** δημιουργείται ο πίνακας των εξόδων και εμφανίζεται στην σελίδα **Οικονομική Επισκόπησης**.
 - **Float sum2**: Μεταβλητή τύπου **Float**. Παίρνει σαν τιμή το **συνολικό ποσό των εξόδων** μετά την επιλογή του κουμπιού Εμφάνιση στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String prov_cust**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Προμηθευτή/Πελάτη** που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String activities**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Είδους** των εσόδων ή εξόδων που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String company**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της

- **Οικονομικής Επισκόπησης.**
 - **String sdfor:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String edfor:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Connection con:** Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση Βεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String situation:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή αυτή της επιλογής της **Κατάστασης Πληρωμής** μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
- **public String viewCompanyAgainstNoDates(String date_start, String date_end, float sum2, String prov_cust, String activities, String company, Connection con, String situation)**

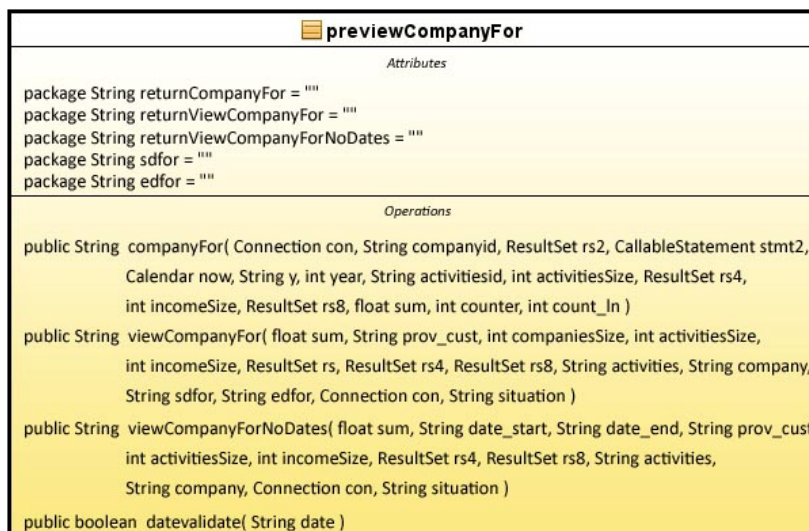
Μέθοδος στην οποία μετά την επιλογή του κουμπιού **Εμφάνιση**, χωρίς τον έλεγχο ημερομηνιών, δημιουργείται ο πίνακας των εξόδων και εμφανίζεται στην σελίδα **Οικονομική Επισκόπηση**.

 - **String date_start:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String date_end:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Float sum2:** Μεταβλητή τύπου **Float**. Παίρνει σαν τιμή το **συνολικό ποσό των εξόδων** μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String activities:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Είδους** των εσόδων ή εξόδων που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String company:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Connection con:** Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String situation:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή αυτή της επιλογής της **Κατάστασης Πληρωμής** μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
- **public boolean datevalidate(String date)**

Η μέθοδος αυτή ελέγχει αν η ημερομηνία που έχει δοθεί έχει τον σωστό τύπο σύνταξης για ημερομηνίες, ή αν η ημερομηνία είναι υπαρκτή.
Αν η τιμή που επιστρέφει είναι αποδεκτή, τότε η μέθοδος παίρνει τιμή **true**, αλλιώς παίρνει τιμή **false**.

 - **String date:** Μεταβλητή τύπου **String** η οποία για να επιστρέψει η μέθοδος **true** πρέπει να είναι ημερομηνία της μορφής **"yyyy-mm-dd"**. Την μεταβλητή αυτή την παίρνουμε από τα πεδία των **ημερομηνιών** κατά την καταχώρηση και επεξεργασία όλων των δεδομένων της σελίδας **Οικονομική Επισκόπηση**.
- **previewCompanyFor.java:**

Στην κλάση **previewCompanyAgainst** δημιουργούνται οι μέθοδοι που έχουν σχέση με τον πίνακα των **Εσόδων**. Η κλάση αυτή φορτώνεται στην σελίδα της **Οικονομικής Επισκόπησης**.



Εικόνα 4.2.11. UML του Java Class previewCompanyFor

Μεταβλητές:

- **Package String returnCompanyFor = ""**
Μεταβλητή που επιστρέφει τα αποτελέσματα κατά την επιλογή εταιρίας στην σελίδα **Οικονομική Επισκόπηση** στον πίνακα των εσόδων. Δημιουργείται η φόρμα της αναζήτησης.
- **Package String returnViewCompanyFor= ""**
Μεταβλητή που επιστρέφει τα αποτελέσματα μετά την επιλογή του κουμπιού **Εμφάνιση** στον πίνακα των εσόδων. Δημιουργείται η φόρμα της αναζήτησης.
- **Package String returnViewCompanyForNoDates= ""**
Μεταβλητή που επιστρέφει τα αποτελέσματα μετά την επιλογή του κουμπιού **Εμφάνιση χωρίς Ημερομηνίες** στον πίνακα των εσόδων. Σε αυτή την περίπτωση οι ημερομηνίες είναι απενεργοποιημένες και εμφανίζονται όλα τα αποτελέσματα που υπάρχουν στην βάση δεδομένων. Δημιουργείται η φόρμα της αναζήτησης.
- **Public String sdfor= ""**
Μεταβλητή που ισούται με την **Ημερομηνία Λήξης** μετά την **Εμφάνιση**.
- **Public String edfor= ""**
Μεταβλητή που ισούται με την **Ημερομηνία Λήξης** μετά την **Εμφάνιση**.

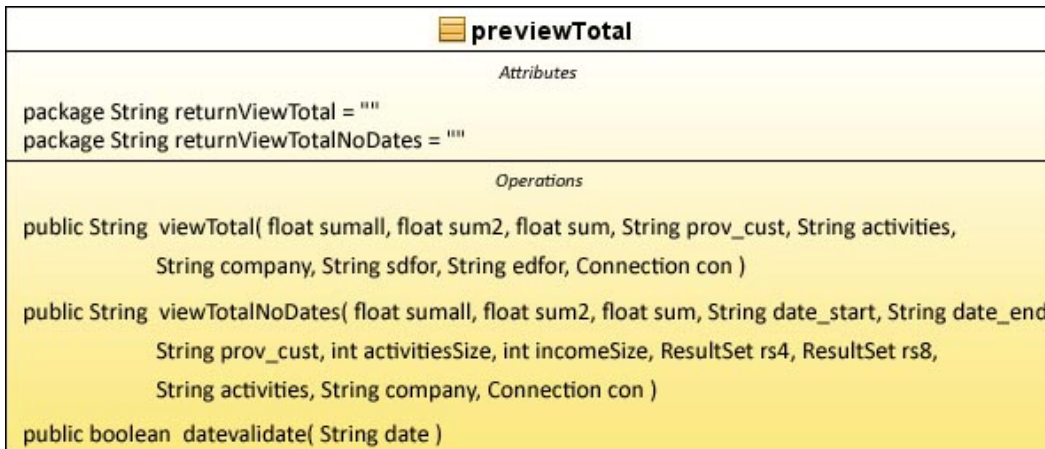
Μέθοδοι:

- **public String companyFor(Connection con, String companyid, ResultSet rs2, CallableStatement stmt2, Calendar now, String y, int year, String activitiesid, int activitiesSize, ResultSet rs4, int incomeSize, ResultSet rs8, float sum, int counter, int count_In)**
Μέθοδος στην οποία δημιουργείται ο πίνακας των εσόδων για το τρέχον έτος και εμφανίζεται κατά την είσοδο στην σελίδα **Οικονομική Επισκόπηση**.
 - **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String companyid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **ResultSet rs2**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **όλων των εσόδων της συγκεκριμένης εταιρίας**.
 - **CallableStatement stmt2**: Μεταβλητή τύπου **CallableStatement**. Παίρνει σαν τιμή μία **εντολή σε γλώσσα SQL**. Αυτό δημιουργεί τα αποτελέσματα της εντολής που παίρνει.
 - **Calendar now**: Μεταβλητή τύπου **Calendar**. Παίρνει σαν τιμή **ένα στιγμιότυπο** με πληροφορίες που έχουν να κάνουν με τη **σημερινή ημερομηνία**.

- **String y**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **σημερινή χρονολογία**. Π.χ. 2011.
- **Int year**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή την **σημερινή χρονολογία**. Π.χ. 2011
- **String activitiesid**: Μεταβλητή τύπου **String** όπου παίρνει σαν τιμή το 'id' του **Είδους** των εσόδων ή εξόδων που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- **Int activitiesSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή **το πλήθος των Ειδών** των εσόδων ή εξόδων που υπάρχουν στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- **ResultSet rs4**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **όλων των Ειδών** που υπάρχουν στη Βάση Δεδομένων και είναι ενεργά.
- **Int incomeSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή **το πλήθος των Προμηθευτών/Πελατών** που υπάρχουν στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- **ResultSet rs8**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **όλων των Προμηθευτών/Πελατών** που υπάρχουν στη Βάση Δεδομένων.
- **Float sum**: Μεταβλητή τύπου **Float**. Παίρνει σαν τιμή το **συνολικό ποσό των εσόδων** στη σελίδα της **Οικονομικής Επισκόπησης**. Σε αυτή την περίπτωση η τιμή του είναι ίση με 0.
- **Int counter**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή την **τρέχουσα γραμμή** που κρατάει ο πίνακας αποτελεσμάτων ενός ResultSet. Αυξάνει κάθε φορά που υπάρχει κάποιο αποτέλεσμα. Σε αυτή την περίπτωση η τιμή του είναι 0.
- **Int count_In**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή την **τρέχουσα γραμμή** που κρατάει ο πίνακας αποτελεσμάτων ενός ResultSet. Αυξάνει κάθε φορά που κάποιο αποτέλεσμα τυπώνετε στην οθόνη. Σε αυτή την περίπτωση η τιμή του είναι 1.
- **public String viewCompanyFor(float sum, String prov_cust, int companiesSize, int activitiesSize, int incomeSize, ResultSet rs, ResultSet rs4, ResultSet rs8, String activities, String company, String sdfor, String edfor, Connection con, String situation)**
Μέθοδος στην οποία μετά την επιλογή του κουμπιού **Εμφάνιση** δημιουργείται ο πίνακας των εσόδων και εμφανίζεται στην σελίδα **Οικονομική Επισκόπησης**.
 - **Float sum**: Μεταβλητή τύπου **Float**. Παίρνει σαν τιμή το **συνολικό ποσό των εσόδων** μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String prov_cust**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Προμηθευτή/Πελάτη** που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Int companiesSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή **το πλήθος των Εταιριών** που υπάρχουν στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Int activitiesSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή **το πλήθος των Ειδών** των εσόδων ή εξόδων που υπάρχουν στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Int incomeSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή **το πλήθος των Προμηθευτών/Πελατών** που υπάρχουν στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **ResultSet rs**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **των πληροφοριών όλων των Εταιριών** που υπάρχουν στη Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** και είναι ενεργές.
 - **ResultSet rs4**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **όλων των Ειδών** που υπάρχουν στη Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** και είναι ενεργά.

- **ResultSet rs8**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα όλων των **Προμηθευτών/Πελατών** που υπάρχουν στη Βάση Δεδομένων.
 - **String activities**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Είδους** των εσόδων ή εξόδων που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String company**: Μεταβλητή τύπου **String** όπου παίρνει σαν τιμή το 'id' της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String sdfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String edfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String situation**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή αυτή της επιλογής της **Κατάστασης Πληρωμής** μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
- **public String viewCompanyForNoDates(float sum, String date_start, String date_end, String prov_cust, int activitiesSize, int incomeSize, ResultSet rs4, ResultSet rs8, String activities, String company, Connection con, String situation)**
Μέθοδος στην οποία μετά την επιλογή του κουμπιού **Εμφάνιση**, χωρίς τον έλεγχο ημερομηνιών, δημιουργείται ο πίνακας των εσόδων και εμφανίζεται στην σελίδα **Οικονομική Επισκόπηση**.
- **Float sum**: Μεταβλητή τύπου **Float**. Παίρνει σαν τιμή το **συνολικό ποσό των εσόδων** μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String date_start**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String date_end**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String prov_cust**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Προμηθευτή/Πελάτη** που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Int activitiesSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των Ειδών** των εσόδων ή εξόδων που υπάρχουν στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Int incomeSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των Προμηθευτών/Πελατών** που υπάρχουν στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **ResultSet rs4**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα όλων των **Ειδών** που υπάρχουν στη Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες και είναι ενεργά.
 - **ResultSet rs8**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα όλων των

- Προμηθευτών/Πελατών που υπάρχουν στη Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες.
- **String activities**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Είδους** των εσόδων ή εξόδων που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String company**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String situation**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή αυτή της επιλογής της **Κατάστασης Πληρωμής** μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
- **public boolean datevalidate(String date)**
 Η μέθοδος αυτή ελέγχει αν η ημερομηνία που έχει δοθεί έχει τον σωστό τύπο σύνταξης για ημερομηνίες, ή αν η ημερομηνία είναι υπαρκτή.
 Αν η τιμή που επιστρέφει είναι αποδεκτή, τότε η μέθοδος παίρνει τιμή **true**, αλλιώς παίρνει τιμή **false**.
- **String date**: Μεταβλητή τύπου **String** η οποία για να επιστρέψει η μέθοδος **true** πρέπει να είναι ημερομηνία της μορφής "yyyy-mm-dd". Την μεταβλητή αυτή την παίρνουμε από τα πεδία των **ημερομηνιών** κατά την καταχώρηση και επεξεργασία όλων των δεδομένων της σελίδας **Οικονομική Επισκόπηση**.
- **previewTotal.java**:
 Στην κλάση **previewTotal** δημιουργούνται οι μέθοδοι που έχουν σχέση με τον πίνακα **Συνολικά** μετά την επιλογή του κουμπιού εμφάνιση με ή και χωρίς ημερομηνίες(προβολή αποτελεσμάτων ανεξάρτητα από τις ημερομηνίες) . Η κλάση αυτή φορτώνεται στην σελίδα της **Οικονομικής Επισκόπησης**.



Εικόνα 4.2.12. UML του Java Class previewTotal

Μεταβλητές:

- **Package String returnViewTotal = ""**
 Μεταβλητή που επιστρέφει τα αποτελέσματα μετά την επιλογή του κουμπιού **Εμφάνιση** στον πίνακα **Συνολικά**.
- **Package String returnViewTotalNoDates=""**
 Μεταβλητή που επιστρέφει τα αποτελέσματα μετά την επιλογή του κουμπιού **Εμφάνιση** ,**χωρίς Ημερομηνίες**, στον πίνακα **Συνολικά**. Σε αυτή την περίπτωση οι ημερομηνίες είναι απενεργοποιημένες και εμφανίζονται όλα τα αποτελέσματα που υπάρχουν στην βάση δεδομένων.

Μέθοδοι:

- **public String viewTotal(float sumall, float sum2, float sum, String prov_cust, String activities, String company, String sdfor, String edfor, Connection con)**

Μέθοδος όπου υπολογίζει το σύνολο των **Εσόδων** και **Εξόδων** και μας εμφανίζει το συνολικό αποτέλεσμα(κέρδη ή απώλειες) για συγκεκριμένο χρονικό διάστημα που του ορίζουμε από της ημερομηνίες στην αναζήτηση. Η μέθοδος φορτώνεται στη σελίδα **Οικονομική Επισκόπηση** μετά την επιλογή του κουμπιού **Εμφάνιση**.

- **Float sumall:** Μεταβλητή τύπου **Float**. Παίρνει σαν τιμή το αποτέλεσμα της **αφαίρεσης των εσόδων και των εξόδων** μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**. Σαν αρχική τιμή παίρνει το 0 και έπειτα μέσα στην μέθοδο γίνεται ίσο με την αφαίρεση.
 - **Float sum2:** Μεταβλητή τύπου **Float**. Παίρνει σαν τιμή το **συνολικό ποσό των εξόδων** μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**. Σαν αρχική τιμή παίρνει το 0 και έπειτα μέσα στην μέθοδο γίνεται ίσο με το άθροισμα όλων των εξόδων που επιλέγηκαν.
 - **Float sum:** Μεταβλητή τύπου **Float**. Παίρνει σαν τιμή το **συνολικό ποσό των εσόδων** μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**. Σαν αρχική τιμή παίρνει το 0 και έπειτα μέσα στην μέθοδο γίνεται ίσο με το άθροισμα όλων των εσόδων που επιλέγηκαν.
 - **String prov_cust:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Προμηθευτή/Πελάτη** που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String activities:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Είδους** των εσόδων ή εξόδων που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String company:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String sdfor:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String edfor:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Connection con:** Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου στη σελίδα της **Οικονομικής Επισκόπησης**.
- **public String viewTotalNoDates(float sumall, float sum2, float sum, String date_start, String date_end, String prov_cust, int activitiesSize, int incomeSize, ResultSet rs4, ResultSet rs8, String activities, String company, Connection con)**

Μέθοδος όπου υπολογίζει το σύνολο των **Εσόδων** και **Εξόδων** και μας εμφανίζει το συνολικό αποτέλεσμα(κέρδη ή απώλειες) για συγκεκριμένο χρονικό διάστημα που του ορίζουμε από της ημερομηνίες στην αναζήτηση. Η μέθοδος φορτώνεται στη σελίδα **Οικονομική Επισκόπηση** μετά την επιλογή του κουμπιού **Εμφάνιση** όταν έχουμε επιλέξει το **Χωρίς Ημερομηνίες(προβολή αποτελεσμάτων ανεξάρτητα από τις ημερομηνίες)**.

- **Float sumall:** Μεταβλητή τύπου **Float**. Παίρνει σαν τιμή το αποτέλεσμα της **αφαίρεσης των εσόδων και των εξόδων** μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**. Σαν αρχική τιμή παίρνει το 0 και έπειτα μέσα στην μέθοδο γίνεται ίσο με την αφαίρεση.
- **Float sum2 :** Μεταβλητή τύπου **Float**. Παίρνει σαν τιμή το **συνολικό ποσό των εξόδων** μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**. Σαν αρχική τιμή παίρνει το 0 και έπειτα μέσα στην μέθοδο γίνεται ίσο με το άθροισμα όλων των εξόδων που επιλέγηκαν.

- **Float sum**: Μεταβλητή τύπου **Float**. Παίρνει σαν τιμή το **συνολικό ποσό των εσόδων** μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**. Σαν αρχική τιμή παίρνει το 0 και έπειτα μέσα στην μέθοδο γίνεται ίσο με το άθροισμα όλων των εσόδων που επιλέγηκαν.
- **String date_start**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String date_end**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String prov_cust**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Προμηθευτή/Πελάτη** που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** στη σελίδα της **Οικονομικής Επισκόπησης**.
- **Int activitiesSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των Ειδών** των εσόδων ή εξόδων που υπάρχουν στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
- **Int incomeSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των Προμηθευτών/Πελατών** που υπάρχουν στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
- **ResultSet rs4**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **όλων των Ειδών** που υπάρχουν στη Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες και είναι ενεργά.
- **ResultSet rs8**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **όλων των Προμηθευτών/Πελατών** που υπάρχουν στη Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες.
- **String activities**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Είδους** των εσόδων ή εξόδων που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String company**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
- **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση Δεδομένων μετά την επιλογή του κουμπιού **Εμφάνιση** όπου οι ημερομηνίες είναι απενεργοποιημένες στη σελίδα της **Οικονομικής Επισκόπησης**.
- **public boolean datevalidate(String date)**

Η μέθοδος αυτή ελέγχει αν η ημερομηνία που έχει δοθεί έχει τον σωστό τύπο σύνταξης για ημερομηνίες, ή αν η ημερομηνία είναι υπαρκτή.
Αν η τιμή που επιστρέφει είναι αποδεκτή, τότε η μέθοδος παίρνει τιμή **true**, αλλιώς παίρνει τιμή **false**.

 - **String date**: Μεταβλητή τύπου **String** η οποία για να επιστρέψει η μέθοδος **true** πρέπει να είναι ημερομηνία της μορφής **"yyyy-mm-dd"**. Την μεταβλητή αυτή την παίρνουμε από τα πεδία των **ημερομηνιών** κατά την καταχώρηση και επεξεργασία όλων των δεδομένων της σελίδας **Οικονομική Επισκόπησης**.
- **checkClass**:

Στην κλάση **checkClass** δημιουργούνται οι μέθοδοι που έχουν σχέση με τις περισσότερες λειτουργίες που πραγματοποιούνται στη σελίδα της **Οικονομικής Επισκόπησης**. **Προσθήκη, Επεξεργασία,**

Διαγραφή, Προβολή Εσόδων και Εξόδων όπως επίσης και Δόσεων για Έσοδα και Έξοδα δημιουργούνται σε αυτή την κλάση.



Εικόνα 4.2.13. UML του Java Class checkClass(Γνωρίσματα)

```

Operations

public String today( )
public String createDropDown( Connection con, ResultSet rs, int companiesSize )
public String addNewCompanyFor( String company, Connection con, ResultSet rs, ResultSet rs4, ResultSet rs8,
int companiesSize, int activitiesSize, int incomeSize )
public String addNewCompanyAgainst( String company, Connection con, ResultSet rs, ResultSet rs4, ResultSet rs8,
int companiesSize, int activitiesSize, int incomeSize )
public String deleteCompanyFor( String counter, String sdfor, String edfor, String company, String approval,
String deletefor, String del_for, String recursive, String title_distinctive, String activity, String price,
String ds_for, String de_for, String prov_cust, String notes, String whenrecursive, Connection con )
public String deleteCompanyAgainst( String counter, String sdfor, String edfor, String company, String approval,
String deleteag, String del_ag, String recursive, String title_distinctive, String activity, String price,
String ds_for, String de_for, String prov_cust, String notes, String whenrecursive, Connection con )
public String editCompanyFor( String counter, String activid_formenu, String provid_formenu, String sdfor,
String edfor, String companyfor, Connection con, String company, String activities, String prov_cust,
ResultSet rs, ResultSet rs4, ResultSet rs8, int companiesSize, int activitiesSize, int incomeSize )
public String editCompanyAgainst( String counter, String activid_formenu, String provid_formenu, String sdfor,
String edfor, String companyag, Connection con, String company, String activities, String prov_cust,
ResultSet rs, ResultSet rs4, ResultSet rs8, int companiesSize, int activitiesSize, int incomeSize )
public String saveCompanyFor( String companyid, String activitiesid, String price, String ds_for, String de_for,
String prov_cust, String notes, String recurs, String whenrecurs, String compl, String when, String differ,
Connection con, boolean incomeFlag, String days_or_months )
public String saveCompanyAgainst( String companyid, String activitiesid, String price, String ds_for, String de_for,
String prov_cust, String notes, String recurs, String whenrecurs, String compl, String when, String differ,
Connection con, boolean incomeFlag, String days_or_months )
public String saveEditCompanyFor( String companyforid, String companyid, String activitiesid, String price,
String ds_for, String de_for, String prov_cust, String notes, String recurs, String whenrecurs, String compl,
String when, String differ, Connection con, boolean incomeFlag, String days_or_months )
public String saveEditCompanyAgainst( String companyagid, String companyid, String activitiesid, String price,
String ds_for, String de_for, String prov_cust, String notes, String recurs, String whenrecurs, String compl,
String when, String differ, Connection con, boolean incomeFlag, String days_or_months )
public String viewMoreCompanyFor( String In_sum, int check, String counter, String activid_formenu,
String provid_formenu, String sdfor, String edfor, Connection con, String companyforid, String company,
String activitiesid, String prov_custid, String title_distinctive, String kind_of_activity, String price,
String date_start, String date_end, String income_type, String notes, String recursive,
String whenrecursive )
public String viewMoreCompanyAgainst( String In_sum2, int check, String counter, String activid_formenu,
String provid_formenu, String sdfor, String edfor, Connection con, String companyagid, String company,
String activitiesid, String prov_custid )
public String addNewDoseCompanyFor( String In_sum, String check, String counter, String activid_formenu,
String provid_formenu, String sdfor, String edfor, Connection con, String companyforid, String company,
String activitiesid, String prov_custid )
public String addNewDoseCompanyAgainst( String In_sum2, String check, String counter, String activid_formenu,
String provid_formenu, String sdfor, String edfor, Connection con, String companyagid, String company,
String activitiesid, String prov_custid )
public String saveAddNewDoseCompanyFor( String companyforid, String dose_compl, String price, String when,
String potential, Connection con, boolean incomeFlag )
public String saveAddNewDoseCompanyAgainst( String companyagid, String dose_compl, String price, String when,
String potential, Connection con, boolean incomeFlag )
public String editDoseCompanyFor( String In_sum, String check, String counter, String dosefor_id,
String activid_formenu, String provid_formenu, String sdfor, String edfor, Connection con,
String companyforid, String company, String activitiesid, String prov_custid )
public String editDoseCompanyAgainst( String In_sum2, String check, String counter, String doseag_id,
String activid_formenu, String provid_formenu, String sdfor, String edfor, Connection con,
String companyagid, String company, String activitiesid, String prov_custid )
public String saveEditDoseCompanyFor( String companyforid, String dose_compl, String price, String when,
String potential, Connection con, boolean incomeFlag, String dosepaidid )
public String saveEditDoseCompanyAgainst( String companyagid, String dose_compl, String price, String when,
String potential, Connection con, boolean incomeFlag, String dosepaidid )
public boolean datevalidate( String date )
public boolean isNumeric( String str )

```

Εικόνα 4.2.14. UML του Java Class checkClass(Λειτουργίες)

Μεταβλητές:

- **Package String returnCreateDropDown = ""**
Μεταβλητή που επιστρέφει το **αρχικό μενού** επιλογής Εταιριών.
- **Package String returnAddNewCompanyFor = ""**
Μεταβλητή που επιστρέφει τον πίνακα **εισαγωγής Νέου Εσόδου** για συγκεκριμένη Εταιρία.
- **Package String returnAddNewCompanyAgainst = ""**
Μεταβλητή που επιστρέφει τον πίνακα **εισαγωγής Νέου Εξόδου** για συγκεκριμένη Εταιρία.
- **Package String returnDeleteCompanyFor = ""**
Μεταβλητή που επιστρέφει τον πίνακα **διαγραφής Εσόδου** συγκεκριμένης Εταιρίας.
- **Package String returnDeleteCompanyAgainst = ""**
Μεταβλητή που επιστρέφει τον πίνακα **διαγραφής Εσόδου** συγκεκριμένης Εταιρίας.
- **Package String returnEditCompanyFor = ""**
Μεταβλητή που επιστρέφει τον πίνακα **επεξεργασίας Εσόδου** συγκεκριμένης Εταιρίας.
- **Package String returnEditCompanyAgainst = ""**
Μεταβλητή που επιστρέφει τον πίνακα **επεξεργασίας Εξόδου** συγκεκριμένης Εταιρίας.
- **Package String returnSaveCompanyFor = ""**
Μεταβλητή που αποθηκεύει ένα νέο **Έσοδο και τα δεδομένα του** για μια συγκεκριμένη Εταιρία στη Βάση Δεδομένων.
- **Package String returnSaveCompanyAgainst = ""**
Μεταβλητή που αποθηκεύει ένα νέο **Έξοδο και τα δεδομένα του** για μια συγκεκριμένη Εταιρία στη Βάση Δεδομένων.
- **Package String returnEditSaveCompanyFor = ""**
Μεταβλητή που αποθηκεύει το **Έσοδο και τα δεδομένα του** για μια συγκεκριμένη Εταιρία στη Βάση Δεδομένων μετά από επεξεργασία του.
- **Package String returnEditSaveCompanyAgainst = ""**
Μεταβλητή που αποθηκεύει το **Έξοδο και τα δεδομένα του** για μια συγκεκριμένη Εταιρία στη Βάση Δεδομένων μετά από επεξεργασία του.
- **Package String returnViewMoreCompanyFor = ""**
Μεταβλητή που επιστρέφει τον πίνακα **προβολής περισσότερων λεπτομερειών Εσόδου** για μια συγκεκριμένη Εταιρία.
- **Package String returnViewMoreCompanyAgainst = ""**
Μεταβλητή που επιστρέφει τον πίνακα **προβολής περισσότερων λεπτομερειών Εξόδου** για μια συγκεκριμένη Εταιρία.
- **Package String returnEditDoseCompanyFor = ""**
Μεταβλητή που επιστρέφει τον πίνακα **επεξεργασίας Δόσης ενός Εσόδου** για μια συγκεκριμένη Εταιρία.
- **Package String returnEditDoseCompanyAgainst = ""**
Μεταβλητή που επιστρέφει τον πίνακα **επεξεργασίας Δόσης ενός Εξόδου** για μια συγκεκριμένη Εταιρία.
- **Package String returnSaveEditDoseCompanyFor = ""**
Μεταβλητή που αποθηκεύει τη **Δόση ενός Εσόδου και τις λεπτομέρειες της** για μια συγκεκριμένη Εταιρία στη Βάση Δεδομένων μετά από επεξεργασία της.
- **Package String returnSaveEditDoseCompanyAgainst = ""**
Μεταβλητή που αποθηκεύει τη **Δόση ενός Εξόδου και τις λεπτομέρειες της** για μια συγκεκριμένη Εταιρία στη Βάση Δεδομένων μετά από επεξεργασία της.
- **Package String returnAddNewDoseCompanyFor = ""**
Μεταβλητή που επιστρέφει τον πίνακα **εισαγωγής Νέας Δόσης ενός Εσόδου** για μια συγκεκριμένη Εταιρία.
- **Package String returnAddNewDoseCompanyAgainst = ""**
Μεταβλητή που επιστρέφει τον πίνακα **εισαγωγής Νέας Δόσης ενός Εξόδου** για μια συγκεκριμένη Εταιρία.

- **Package String returnSaveDoseCompanyFor = ""**
Μεταβλητή που αποθηκεύει μια Νέα Δόση ενός Εσόδου και τις λεπτομέρειες της για μια συγκεκριμένη Εταιρία στη Βάση.
- **Package String returnSaveDoseCompanyAgainst = ""**
Μεταβλητή που αποθηκεύει μια Νέα Δόση ενός Εξόδου και τις λεπτομέρειες της για μια συγκεκριμένη Εταιρία στη Βάση.
- **Package int isrecursive = ""**
Μεταβλητή που ορίζει αν κάποιο έσοδο ή έξοδο είναι αναδρομικό ή όχι. Παίρνει τιμή 1 όταν είναι αναδρομικό και τιμή 0 όταν είναι μη αναδρομικό.
- **Package int whenrecursive = ""**
Μεταβλητή που ορίζει το πλήθος των ημερών ή μηνών όταν το έσοδο ή έξοδο είναι αναδρομικό.
- **Package int completed = ""**
Μεταβλητή που ορίζει αν κάποιο έσοδο ή έξοδο έχει ολοκληρωθεί. Παίρνει τιμή 1 όταν είναι ολοκληρωμένο και τιμή 0 όταν είναι δεν είναι ολοκληρωμένο.
- **Package int dose_completed = ""**
Μεταβλητή που ορίζει αν κάποια δόση εσόδου ή εξόδου έχει ολοκληρωθεί. Παίρνει τιμή 1 όταν είναι ολοκληρωμένο και τιμή 0 όταν είναι δεν είναι ολοκληρωμένο.
- **Public String DATE_FORMAT_NOW = "yyyy-MM-dd"**
Μεταβλητή που επιστρέφει το συγκεκριμένο τύπο ημερομηνίας.
- **Public long ONE_DAY_MILLIS = 24 * 60 * 60 * 1000**
Μεταβλητή που μετατρέπει την μέρα σε milliseconds.

Μέθοδοι:

- **public String today()**
Η μέθοδος αυτή επιστρέφει την τρέχουσα ημερομηνία.
- **public String createDropDown(Connection con, ResultSet rs, int companiesSize)**
Μέθοδος που δημιουργεί το μενού επιλογής στην Οικονομική Επισκόπηση.
 - **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της Οικονομικής Επισκόπησης.
 - **ResultSet rs**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα των πληροφοριών όλων των Εταιριών που υπάρχουν στη Βάση Δεδομένων και είναι ενεργές.
 - **Int CompaniesSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το πλήθος των Εταιριών που υπάρχουν στην Βάση στη σελίδα της Οικονομικής Επισκόπησης.
- **public String addNewCompanyFor(String company, Connection con, ResultSet rs, ResultSet rs4, ResultSet rs8, int companiesSize, int activitiesSize, int incomeSize)**
Δημιουργεί το πίνακα εισαγωγής νέου εσόδου στην Οικονομική Επισκόπηση.
 - **String company**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' της Εταιρίας που υπάρχει στην Βάση Δεδομένων στη σελίδα της Οικονομικής Επισκόπησης.
 - **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της Οικονομικής Επισκόπησης.
 - **ResultSet rs**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα των πληροφοριών όλων των Εταιριών που υπάρχουν στη Βάση Δεδομένων και είναι ενεργές.
 - **ResultSet rs4**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα όλων των Ειδών που υπάρχουν στη Βάση Δεδομένων και είναι ενεργά.
 - **ResultSet rs8**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα όλων των Προμηθευτών/Πελατών που υπάρχουν στη Βάση Δεδομένων.

- **Int CompaniesSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των Εταιριών** που υπάρχουν στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- **Int ActivitiesSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των Ειδών** των εσόδων ή εξόδων που υπάρχουν στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- **Int incomeSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των Προμηθευτών/Πελατών** που υπάρχουν στην Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- **public String addNewCompanyAgainst(String company, Connection con, ResultSet rs, ResultSet rs4, ResultSet rs8, int companiesSize, int activitiesSize, int incomeSize)**
Δημιουργεί το πίνακα εισαγωγής νέου εξόδου στην **Οικονομική Επισκόπηση**.
 - **String company**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **ResultSet rs**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **των πληροφοριών όλων των Εταιριών** που υπάρχουν στη Βάση Δεδομένων και είναι ενεργές.
 - **ResultSet rs4**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **όλων των Ειδών** που υπάρχουν στη Βάση Δεδομένων και είναι ενεργά.
 - **ResultSet rs8**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **όλων των Προμηθευτών/Πελατών** που υπάρχουν στη Βάση Δεδομένων.
 - **Int CompaniesSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των Εταιριών** που υπάρχουν στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Int ActivitiesSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των Ειδών** των εσόδων ή εξόδων που υπάρχουν στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Int incomeSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των Προμηθευτών/Πελατών** που υπάρχουν στην Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- **public String deleteCompanyFor(String counter, String sdfor, String edfor, String company, String approval, String deletefor, String del_for, String recursive, String title_distinctive, String activity, String price, String ds_for, String de_for, String prov_cust, String notes, String whenrecursive, Connection con)**

Μέθοδος που δημιουργεί τον πίνακα **διαγραφής εσόδου** στην **Οικονομική Επισκόπηση**.

- **String counter**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των εσόδων** που υπάρχουν στο ResultSet που χρησιμοποιούμε για να εμφανίσουμε στην οθόνη τα έσοδα. Για κάθε έσοδο που βρίσκει αυξάνει κατά 1 έτσι ώστε να του δίνει μοναδικό όνομα.
- **String sdfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String edfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String company**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String approval**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **Ναι ή Όχι** ανάλογα με το αν θέλουμε να γίνει διαγραφή του εσόδου.
- **String deletefor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του εσόδου που

- πρόκειται να διαγραφεί.
- ***String del_for***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του εσόδου που πρόκειται να διαγραφεί.
- ***String recursive***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή 1 ή 0. Αντιστοιχεί στο πεδίο του εσόδου που το ορίζει ως **αναδρομικό ή μη αναδρομικό**.
- ***String title_distinctive***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή τον **ονομαστικό τίτλο** της εταιρίας του συγκεκριμένου εσόδου που πρόκειται να διαγραφεί.
- ***String activity***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **όνομα του είδους** του εσόδου που πρόκειται να διαγραφεί.
- ***String price***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **ποσό** του εσόδου που πρόκειται να διαγραφεί.
- ***String ds_for***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Έναρξης** του εσόδου που πρόκειται να διαγραφεί.
- ***String de_for***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Λήξης** του εσόδου που πρόκειται να διαγραφεί.
- ***String prov_cust***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **όνομα του Προμηθευτή/Πελάτη** του εσόδου που πρόκειται να διαγραφεί.
- ***String notes***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή τα **σχόλια** του εσόδου που πρόκειται να διαγραφεί.
- ***String whenrecursive***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή ένα αριθμό που ορίζει το **μήκος της αναδρομικότητας** του εσόδου που πρόκειται να διαγραφεί.
- ***Connection con***: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- ***public String deleteCompanyAgainst(String counter, String sdfor, String edfor, String company, String approval, String deleteag, String del_ag, String recursive, String title_distinctive, String activity, String price, String ds_for, String de_for, String prov_cust, String notes, String whenrecursive, Connection con)***

Μέθοδος που δημιουργεί τον πίνακα **διαγραφής εξόδου** στην **Οικονομική Επισκόπηση**.

- ***String counter***: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των εξόδων** που υπάρχουν στο ResultSet που χρησιμοποιούμε για να εμφανίσουμε στην οθόνη τα έξοδα. Για κάθε έξοδο που βρίσκει αυξάνει κατά 1 έτσι ώστε να του δίνει μοναδικό όνομα.
- ***String sdfor***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** στη σελίδα της **Οικονομικής Επισκόπησης**.
- ***String edfor***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** στη σελίδα της **Οικονομικής Επισκόπησης**.
- ***String company***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- ***String approval***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **Ναι ή Όχι** ανάλογα με το αν θέλουμε να γίνει διαγραφή του εξόδου.
- ***String deleteag***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του εξόδου που πρόκειται να διαγραφεί.
- ***String del_ag***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του εξόδου που πρόκειται να διαγραφεί.
- ***String recursive***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή 1 ή 0. Αντιστοιχεί στο πεδίο του εξόδου που το ορίζει ως **αναδρομικό ή μη αναδρομικό**.
- ***String title_distinctive***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή τον **ονομαστικό τίτλο** της εταιρίας του συγκεκριμένου εξόδου που πρόκειται να διαγραφεί.
- ***String activity***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **όνομα του είδους** του εξόδου που πρόκειται να διαγραφεί.
- ***String price***: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **ποσό** του εξόδου που πρόκειται

- να διαγραφεί.
- **String ds_for**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Έναρξης** του εξόδου που πρόκειται να διαγραφεί.
- **String de_for**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Λήξης** του εξόδου που πρόκειται να διαγραφεί.
- **String prov_cust**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **όνομα του Προμηθευτή/Πελάτη** του εξόδου που πρόκειται να διαγραφεί.
- **String notes**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή τα **σχόλια** του εξόδου που πρόκειται να διαγραφεί.
- **String whenrecursive**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή ένα αριθμό που ορίζει το **μήκος της αναδρομικότητας** του εξόδου που πρόκειται να διαγραφεί.
- **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- **public String editCompanyFor(String counter, String activid_formenu, String provid_formenu, String sdfor, String edfor, String companyfor, Connection con, String company, String activities, String prov_cust, ResultSet rs, ResultSet rs4, ResultSet rs8, int companiesSize, int activitiesSize, int incomeSize)**

Μέθοδος που δημιουργεί τον πίνακα **επεξεργασίας εσόδου** στην **Οικονομική Επισκόπηση**.

- **String counter**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των εσόδων** που υπάρχουν στο **ResultSet** που χρησιμοποιούμε για να εμφανίσουμε στην οθόνη τα έσοδα. Για κάθε έσοδο που βρίσκει αυξάνει κατά 1 έτσι ώστε να του δίνει μοναδικό όνομα.
- **String activid_formenu**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Είδους** από το μενού της **Εμφάνισης**.
- **String provid_formenu**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Προμηθευτή/Πελάτη** από το μενού της **Εμφάνισης**.
- **String sdfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String edfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String companyfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του εσόδου.
- **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String company**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String activities**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Είδους** του εσόδου που υπάρχει στη Βάση Δεδομένων.
- **String prov_cust**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Προμηθευτή/Πελάτη** του εσόδου που υπάρχει στη Βάση Δεδομένων.
- **ResultSet rs**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **των πληροφοριών όλων των Εταιριών** που υπάρχουν στη Βάση Δεδομένων και είναι ενεργές.
- **ResultSet rs4**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **όλων των Ειδών** που υπάρχουν στη Βάση και είναι ενεργά.
- **ResultSet rs8**: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **όλων των Προμηθευτών/Πελατών** που υπάρχουν στη Βάση.
- **Int CompaniesSize**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή το **πλήθος των Εταιριών** που υπάρχουν στην Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.

- *Int ActivitiesSize*: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή **το πλήθος των Ειδών** των εσόδων ή εξόδων που υπάρχουν στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- *Int incomeSize*: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή **το πλήθος των Προμηθευτών/Πελατών** που υπάρχουν στην Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- *public String editCompanyAgainst(String counter, String activid_formenu, String provid_formenu, String sdfor, String edfor, String companyag, Connection con, String company, String activities, String prov_cust, ResultSet rs, ResultSet rs4, ResultSet rs8, int companiesSize, int activitiesSize, int incomeSize)*

Μέθοδος που δημιουργεί τον πίνακα **επεξεργασίας εξόδου** στην **Οικονομική Επισκόπηση**.

- *String counter*: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή **το πλήθος των εξόδων** που υπάρχουν στο ResultSet που χρησιμοποιούμε για να εμφανίσουμε στην οθόνη τα έξοδα. Για κάθε έξοδο που βρίσκει αυξάνει κατά 1 έτσι ώστε να του δίνει μοναδικό όνομα.
- *String activid_formenu*: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **το 'id' του Είδους** από το μενού της **Εμφάνισης**.
- *String provid_formenu*: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **το 'id' του Προμηθευτή/Πελάτη** από το μενού της **Εμφάνισης**.
- *String sdfor*: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** στη σελίδα της **Οικονομικής Επισκόπησης**.
- *String edfor*: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** στη σελίδα της **Οικονομικής Επισκόπησης**.
- *String companyag*: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **το 'id' του εξόδου**.
- *Connection con*: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- *String company*: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **το 'id' της Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- *String activities*: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **το 'id' του Είδους** του εξόδου που υπάρχει στη Βάση Δεδομένων.
- *String prov_cust*: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **το 'id' του Προμηθευτή/Πελάτη** του εξόδου που υπάρχει στη Βάση Δεδομένων.
- *ResultSet rs*: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **των πληροφοριών όλων των Εταιριών** που υπάρχουν στη Βάση Δεδομένων και είναι ενεργές.
- *ResultSet rs4*: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **όλων των Ειδών** που υπάρχουν στη Βάση και είναι ενεργά.
- *ResultSet rs8*: Μεταβλητή τύπου **ResultSet**. Παίρνει σαν τιμή τα αποτελέσματα που δημιουργεί κάποιο **CallableStatement**. Κρατάει αποτελέσματα **όλων των Προμηθευτών/Πελατών** που υπάρχουν στη Βάση.
- *Int CompaniesSize*: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή **το πλήθος των Εταιριών** που υπάρχουν στην Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- *Int ActivitiesSize*: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή **το πλήθος των Ειδών** των εσόδων ή εξόδων που υπάρχουν στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- *Int incomeSize*: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή **το πλήθος των Προμηθευτών/Πελατών** που υπάρχουν στην Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- *public String saveCompanyFor(String companyid, String activitiesid, String price, String ds_for, String de_for, String prov_cust, String notes, String recurs, String whenrecurs, String compl, String*

when, String differ, Connection con, boolean incomeFlag, String days_or_months)

Μέθοδος που αποθηκεύει ένα νέο έσοδο στην **Οικονομική Επισκόπηση**.

- **String companyid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String activitiesid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Είδους** του εσόδου που πρόκειται να αποθηκευτεί.
 - **String price**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **ποσό** του εσόδου που πρόκειται να αποθηκευτεί.
 - **String ds_for**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Έναρξης** του εσόδου που πρόκειται να αποθηκευτεί.
 - **String de_for**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Λήξης** του εσόδου που πρόκειται να αποθηκευτεί.
 - **String prov_cust**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Προμηθευτή/Πελάτη** του εσόδου που πρόκειται να αποθηκευτεί.
 - **String notes**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή τα **σχόλια** του εσόδου που πρόκειται να αποθηκευτεί.
 - **String recurs**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **1** ή **0**. Καθορίζει την **αναδρομικότητα** ενός εσόδου που πρόκειται να αποθηκευτεί.
 - **String whenrecurs**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή ένα αριθμό που ορίζει το **μήκος της αναδρομικότητας** του εσόδου που πρόκειται να αποθηκευτεί.
 - **String compl**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **1** ή **0**. Καθορίζει την **ολοκλήρωση** του εσόδου που πρόκειται να αποθηκευτεί.
 - **String when**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή μια ημερομηνία που ορίζει πότε ολοκληρώθηκε το έσοδο.
 - **String differ**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή κάποιο **αριθμό** που μας βοηθάει να δούμε την διαφορά του αρχικού ποσού του εσόδου με του τελικού που πληρώθηκε.
 - **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Boolean incomeFlag**: Μεταβλητή τύπου **Boolean**. Σαν αρχική τιμή παίρνει **true**. Χρησιμοποιείται για τον **έλεγχο** στις περιπτώσεις που υπάρχουν για να επιτρέψει την αποθήκευση ενός εσόδου. Εάν μείνει true τότε αποθηκεύεται το έσοδο αλλιώς δεν αποθηκεύεται.
 - **String days_or_months**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το πεδίο που θα καθορίσει αν ένα αναδρομικό έσοδο είναι σε **μέρες** ή **μήνες**.
- *public String saveCompanyAgainst(String companyid, String activitiesid, String price, String ds_for, String de_for, String prov_cust, String notes, String recurs, String whenrecurs, String compl, String when, String differ, Connection con, boolean incomeFlag, String days_or_months)*

Μέθοδος που αποθηκεύει ένα νέο έξοδο στην **Οικονομική Επισκόπηση**.

- **String companyid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String activitiesid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Είδους** του εξόδου που πρόκειται να αποθηκευτεί.
- **String price**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **ποσό** του εξόδου που πρόκειται να αποθηκευτεί.
- **String ds_for**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Έναρξης** του εξόδου που πρόκειται να αποθηκευτεί.
- **String de_for**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Λήξης** του εξόδου που πρόκειται να αποθηκευτεί.
- **String prov_cust**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Προμηθευτή/Πελάτη** του εξόδου που πρόκειται να αποθηκευτεί.

- **String notes**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή τα **σχόλια** του εξόδου που πρόκειται να αποθηκευτεί.
- **String recurs**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **1** ή **0**. Καθορίζει την **αναδρομικότητα** ενός εξόδου που πρόκειται να αποθηκευτεί.
- **String whenrecurs**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή ένα αριθμό που ορίζει το **μήκος της αναδρομικότητας** του εξόδου που πρόκειται να αποθηκευτεί.
- **String compl**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **1** ή **0**. Καθορίζει την **ολοκλήρωση** του εξόδου που πρόκειται να αποθηκευτεί.
- **String when**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή μια ημερομηνία που ορίζει πότε ολοκληρώθηκε το έξοδο.
- **String differ**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή κάποιο **αριθμό** που μας βοηθάει να δούμε την διαφορά του αρχικού ποσού του εξόδου με του τελικού που πληρώθηκε.
- **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- **Boolean incomeFlag**: Μεταβλητή τύπου **Boolean**. Σαν αρχική τιμή παίρνει **true**. Χρησιμοποιείται για τον **έλεγχο** στις περιπτώσεις που υπάρχουν για να επιτρέψει την αποθήκευση ενός εξόδου. Εάν μείνει true τότε αποθηκεύεται το έξοδο αλλιώς δεν αποθηκεύεται.
- **String days_or_months**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το πεδίο που θα καθορίσει αν ένα αναδρομικό έξοδο είναι σε **μέρες** ή **μήνες**.
- **public String saveEditCompanyFor(String companyforid, String companyid, String activitiesid, String price, String ds_for, String de_for, String prov_cust, String notes, String recurs, String whenrecurs, String compl, String when, String differ, Connection con, boolean incomeFlag, String days_or_months)**

Μέθοδος που **αποθηκεύει** ένα **έσοδο** μετά από **επεξεργασία** στην **Οικονομική Επισκόπηση**.

- **String companyforid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **εσόδου** που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
- **String companyid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** της **Εταιρίας** του εσόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
- **String activitiesid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Είδους** του εσόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
- **String price**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **ποσό** του εσόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
- **String ds_for**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Έναρξης** του εσόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
- **String de_for**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Λήξης** του εσόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
- **String prov_cust**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Προμηθευτή/Πελάτη** του εσόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
- **String notes**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή τα **σχόλια** του εσόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
- **String recurs**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **1** ή **0**. Καθορίζει την **αναδρομικότητα** ενός εσόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
- **String whenrecurs**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή ένα αριθμό που ορίζει το **μήκος της αναδρομικότητας** του εσόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
- **String compl**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **1** ή **0**. Καθορίζει την **ολοκλήρωση** του εσόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
- **String when**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή μια ημερομηνία που ορίζει πότε ολοκληρώθηκε το έσοδο.
- **String differ**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή κάποιο **αριθμό** που μας βοηθάει

- να δούμε την διαφορά του αρχικού ποσού του εσόδου με του τελικού που πληρώθηκε.
- **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Boolean incomeFlag**: Μεταβλητή τύπου **Boolean**. Σαν αρχική τιμή παίρνει **true**. Χρησιμοποιείται για τον **έλεγχο** στις περιπτώσεις που υπάρχουν για να επιτρέψει την αποθήκευση ενός εσόδου. Εάν μείνει true τότε αποθηκεύεται το έσοδο αλλιώς δεν αποθηκεύεται.
 - **String days_or_months**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το πεδίο που θα καθορίσει αν ένα αναδρομικό έσοδο είναι σε **μέρες** ή **μήνες**.
- **public String saveEditCompanyAgainst(String companyagid, String companyid, String activitiesid, String price, String ds_for, String de_for, String prov_cust, String notes, String recurs, String whenrecurs, String compl, String when, String differ, Connection con, boolean incomeFlag, String days_or_months)**
- Μέθοδος που **αποθηκεύει** ένα **έξοδο** μετά από **επεξεργασία** στην **Οικονομική Επισκόπηση**.
- **String companyagid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **εξόδου** που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
 - **String companyid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String activitiesid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Είδους** του εξόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
 - **String price**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **ποσό** του εξόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
 - **String ds_for**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Έναρξης** του εξόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
 - **String de_for**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Λήξης** του εξόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
 - **String prov_cust**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Προμηθευτή/Πελάτη** του εξόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
 - **String notes**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή τα **σχόλια** του εξόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
 - **String recurs**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **1** ή **0**. Καθορίζει την **αναδρομικότητα** ενός εξόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
 - **String whenrecurs**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή ένα αριθμό που ορίζει το **μήκος της αναδρομικότητας** του εξόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
 - **String compl**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **1** ή **0**. Καθορίζει την **ολοκλήρωση** του εξόδου που επεξεργάστηκε και πρόκειται να αποθηκευτεί.
 - **String when**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή μια ημερομηνία που ορίζει πότε ολοκληρώθηκε το έξοδο.
 - **String differ**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή κάποιο **αριθμό** που μας βοηθάει να δούμε την διαφορά του αρχικού ποσού του εξόδου με του τελικού που πληρώθηκε.
 - **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Boolean incomeFlag**: Μεταβλητή τύπου **Boolean**. Σαν αρχική τιμή παίρνει **true**. Χρησιμοποιείται για τον **έλεγχο** στις περιπτώσεις που υπάρχουν για να επιτρέψει την αποθήκευση ενός εξόδου. Εάν μείνει true τότε αποθηκεύεται το έξοδο αλλιώς δεν αποθηκεύεται.
 - **String days_or_months**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το πεδίο που θα καθορίσει αν ένα αναδρομικό έξοδο είναι σε **μέρες** ή **μήνες**.
- **public String viewMoreCompanyFor(String ln_sum, int check, String counter, String**

activid_formenu, String provid_formenu, String sdfor, String edfor, Connection con, String companyforid, String company, String activitiesid, String prov_custid, String title_distinctive, String kind_of_activity, String price, String date_start, String date_end, String income_type, String notes, String recursive, String whenrecursive)

Μέθοδος που δημιουργεί τον πίνακα **περισσότερων** λεπτομερειών για ένα έσοδο στην **Οικονομική Επισκόπηση**.

- **String In_sum**: Μεταβλητή τύπου **String**. Παίρνει τιμή το **συνολικό ποσό του εσόδου** που επιλέγηκε σχετικά με τις ημερομηνίες.
- **Int check**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή ένα **αριθμό** που ορίζει **ποιο έσοδο επιλέγηκε** για να προβληθεί. Αυτός ο αριθμός μπαίνει στο τέλος της μεταβλητής **In_sum** και είναι αυτός που την κάνει μοναδική.
- **String counter**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **πλήθος των εσόδων** που ανακτήθηκαν από την SQL εντολή.
- **String activid_formenu**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id' του Είδους** από το μενού της **Εμφάνισης**.
- **String provid_formenu**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id' του Προμηθευτή/Πελάτη** από το μενού της **Εμφάνισης**.
- **String sdfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String edfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** στη σελίδα της **Οικονομικής Επισκόπησης**.
- **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String companyforid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id' του εσόδου** που πρόκειται να προβληθεί.
- **String company**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id' της Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String activitiesid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id' του Είδους** του εσόδου που πρόκειται να προβληθεί.
- **String prov_custid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id' του Προμηθευτή/Πελάτη** του εσόδου που πρόκειται να προβληθεί.
- **String title_distinctive**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή τον **ονομαστικό τίτλο** της εταιρίας του συγκεκριμένου εσόδου που πρόκειται να προβληθεί.
- **String kind_of_activity**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **ονομασία του Είδους** του εσόδου που επιλέγηκε και πρόκειται να προβληθεί.
- **String price**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **ποσό** του εσόδου που πρόκειται να προβληθεί.
- **String date_start**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Έναρξης** του εσόδου που πρόκειται να προβληθεί.
- **String date_end**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **Ημ. Λήξης** του εσόδου που πρόκειται να προβληθεί.
- **String income_type**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την **ονομασία του Προμηθευτή/Πελάτη** του εσόδου που επιλέγηκε και πρόκειται να προβληθεί.
- **String notes**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή τα **σχόλια** του εσόδου που επιλέγηκε και πρόκειται να προβληθεί.
- **String recursive**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **1 ή 0**. Καθορίζει την **αναδρομικότητα** του εσόδου που πρόκειται να προβληθεί.
- **String whenrecursive**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή ένα αριθμό που ορίζει το **μήκος της αναδρομικότητας** του εσόδου που πρόκειται να προβληθεί.
- **public String viewMoreCompanyAgainst(String In_sum2, int check, String counter, String**

activid_formenu, String provid_formenu, String sdfor, String edfor, Connection con, String companyagid, String company, String activitiesid, String prov_custid)

Μέθοδος που δημιουργεί τον πίνακα **περισσότερων** λεπτομερειών για ένα **έξοδο** στην **Οικονομική Επισκόπηση**.

- **String ln_sum2**: Μεταβλητή τύπου **String**. Παίρνει τιμή το **συνολικό ποσό του εξόδου** που επιλέγηκε σχετικά με τις ημερομηνίες.
 - **Int check**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή ένα **αριθμό** που ορίζει **ποιο έξοδο επιλέγηκε** για να προβληθεί. Αυτός ο αριθμός μπαίνει στο τέλος της μεταβλητής **ln_sum** και είναι αυτός που την κάνει μοναδική.
 - **String counter**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **πλήθος των εξόδων** που ανακτήθηκαν από την SQL εντολή.
 - **String activid_formenu**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Είδους** από το μενού της **Εμφάνισης**.
 - **String provid_formenu**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Προμηθευτή/Πελάτη** από το μενού της **Εμφάνισης**.
 - **String sdfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String edfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String companyagid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **εξόδου** που πρόκειται να προβληθεί.
 - **String company**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String activitiesid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Είδους** του εξόδου που πρόκειται να προβληθεί.
 - **String prov_custid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Προμηθευτή/Πελάτη** του εξόδου που πρόκειται να προβληθεί.
- **public String addNewDoseCompanyFor(String ln_sum, String check, String counter, String activid_formenu, String provid_formenu, String sdfor, String edfor, Connection con, String companyforid, String company, String activitiesid, String prov_custid)**

Μέθοδος που δημιουργεί τον πίνακα **εισαγωγής νέας δόσης** για κάποιο αναδρομικό **έσοδο** στην **Οικονομική Επισκόπηση**.

- **String ln_sum**: Μεταβλητή τύπου **String**. Παίρνει τιμή το **συνολικό ποσό του εσόδου** που επιλέγηκε σχετικά με τις ημερομηνίες.
- **String check**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή ένα **αριθμό** που ορίζει σε **ποιο έσοδο επιλέγηκε** για να προστεθεί μία δόση.
- **String counter**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **πλήθος των εσόδων** που ανακτήθηκαν από την SQL εντολή.
- **String activid_formenu**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Είδους** από το μενού της **Εμφάνισης**.
- **String provid_formenu**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Προμηθευτή/Πελάτη** από το μενού της **Εμφάνισης**.
- **String sdfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String edfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** στη σελίδα της **Οικονομικής Επισκόπησης**.
- **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής**

- **Επισκόπησης.**
 - **String companyforid:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **εσόδου** που επιλέγει να προστεθεί μία δόση.
 - **String company:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String activitiesid:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Είδους** του εσόδου που επιλέγει να προστεθεί μία δόση.
 - **String prov_custid:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Προμηθευτή/Πελάτη** του εσόδου που επιλέγει να προστεθεί μία δόση.
- **public String addNewDoseCompanyAgainst(String ln_sum2, String check, String counter, String activid_formenu, String provid_formenu, String sdfor, String edfor, Connection con, String companyagid, String company, String activitiesid, String prov_custid)**

Μέθοδος που δημιουργεί τον πίνακα **εισαγωγής νέας δόσης** για κάποιο αναδρομικό **έξοδο** στην **Οικονομική Επισκόπηση**.

 - **String ln_sum2:** Μεταβλητή τύπου **String**. Παίρνει τιμή το **συνολικό ποσό του εξόδου** που επιλέγηκε σχετικά με τις ημερομηνίες.
 - **String check:** Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή ένα **αριθμό** που ορίζει σε **ποιο έξοδο επιλέγηκε** για να προστεθεί μία δόση.
 - **String counter:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **πλήθος των εξόδων** που ανακτήθηκαν από την SQL εντολή.
 - **String activid_formenu:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Είδους** από το μενού της **Εμφάνισης**.
 - **String provid_formenu:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Προμηθευτή/Πελάτη** από το μενού της **Εμφάνισης**.
 - **String sdfor:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String edfor:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Connection con:** Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String companyagid:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **εξόδου** που επιλέγει να προστεθεί μία δόση.
 - **String company:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String activitiesid:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Είδους** του εσόδου που επιλέγει να προστεθεί μία δόση.
 - **String prov_custid:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Προμηθευτή/Πελάτη** του εσόδου που επιλέγει να προστεθεί μία δόση.
- **public String saveAddNewDoseCompanyFor(String companyforid, String dose_compl, String price, String when, String potential, Connection con, boolean incomeFlag)**

Μέθοδος που αποθηκεύει μια νέα **δόση** για κάποιο αναδρομικό **έσοδο** στην **Οικονομική Επισκόπηση**.

 - **String companyforid:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **εσόδου** που επιλέγει να προστεθεί μία δόση.
 - **String dose_compl:** Μεταβλητή τύπου **String**. Παίρνει τιμή **1** ή **0**. Το 1 ορίζει το αν η δόση έχει πληρωθεί και το 0 το αν δεν έχει πληρωθεί.
 - **String price:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **ποσό της δόσης** που μόλις καταχωρήθηκε και που πρέπει να πληρώνεται.
 - **String when:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **μια ημερομηνία** που είναι το **πότε πληρώθηκε** η συγκεκριμένη δόση.

- **String potential**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **μια ημερομηνία** που είναι το **μέχρι πότε πρέπει να πληρωθεί** η συγκεκριμένη δόση.
- **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- **Boolean IncomeFlag**: Μεταβλητή τύπου **Boolean**. Σαν αρχική τιμή παίρνει **true**. Χρησιμοποιείται για τον **έλεγχο** στις περιπτώσεις που υπάρχουν για να επιτρέψει την αποθήκευση μιας δόσης ενός εσόδου. Εάν μείνει true τότε αποθηκεύεται η δόση αλλιώς δεν αποθηκεύεται.
- **public String saveAddNewDoseCompanyAgainst(String companyagid, String dose_compl, String price, String when, String potential, Connection con, boolean incomeFlag)**
Μέθοδος που αποθηκεύει μια νέα **δόση** για κάποιο αναδρομικό **έξοδο** στην **Οικονομική Επισκόπηση**.
 - **String companyagid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **εξόδου** που επιλέγει να προστεθεί μία δόση.
 - **String dose_compl**: Μεταβλητή τύπου **String**. Παίρνει τιμή **1 ή 0**. Το 1 ορίζει το αν η δόση έχει πληρωθεί και το 0 το αν δεν έχει πληρωθεί.
 - **String price**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το ποσό **της δόσης** που μόλις καταχωρήθηκε και που πρέπει να πληρώνεται.
 - **String when**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **μια ημερομηνία** που είναι το **πότε πληρώθηκε** η συγκεκριμένη δόση.
 - **String potential**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **μια ημερομηνία** που είναι το **μέχρι πότε πρέπει να πληρωθεί** η συγκεκριμένη δόση.
 - **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Boolean IncomeFlag**: Μεταβλητή τύπου **Boolean**. Σαν αρχική τιμή παίρνει **true**. Χρησιμοποιείται για τον **έλεγχο** στις περιπτώσεις που υπάρχουν για να επιτρέψει την αποθήκευση μιας δόσης ενός εξόδου. Εάν μείνει true τότε αποθηκεύεται η δόση αλλιώς δεν αποθηκεύεται.
- **public String editDoseCompanyFor(String ln_sum, String check, String counter, String dosefor_id, String activid_formenu, String provid_formenu, String sdfor, String edfor, Connection con, String companyforid, String company, String activitiesid, String prov_custid)**
Μέθοδος που δημιουργεί τον πίνακα **επεξεργασίας** μιας **δόσης** για κάποιο αναδρομικό **έσοδο** στην **Οικονομική Επισκόπηση**.
 - **String ln_sum**: Μεταβλητή τύπου **String**. Παίρνει τιμή το **συνολικό ποσό του εσόδου** που επιλέγηκε σχετικά με τις ημερομηνίες.
 - **String check**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή ένα **αριθμό** που ορίζει **ποιο έσοδο** επιλέγηκε.
 - **String counter**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **πλήθος των εσόδων** που ανακτήθηκαν από την SQL εντολή.
 - **String dosefor_id**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** της **δόσης του εσόδου** που πρόκειται να επεξεργαστεί.
 - **String activid_formenu**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Είδους** από το μενού της **Εμφάνισης**.
 - **String provid_formenu**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **Προμηθευτή/Πελάτη** από το μενού της **Εμφάνισης**.
 - **String sdfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String edfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** στη σελίδα της **Οικονομικής Επισκόπησης**.

- **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String companyforid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **εσόδου**.
- **String company**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
- **String activitiesid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Είδους** του εσόδου που επιλέγει να επεξεργαστεί μία δόση.
- **String prov_custid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Προμηθευτή/Πελάτη** του εσόδου που επιλέγει να επεξεργαστεί μία δόση.
- **public String editDoseCompanyAgainst(String ln_sum2, String check, String counter, String doseag_id, String activid_formenu, String provid_formenu, String sdfor, String edfor, Connection con, String companyagid, String company, String activitiesid, String prov_custid)**
Μέθοδος που δημιουργεί τον πίνακα **επεξεργασίας** μιας **δόσης** για κάποιο αναδρομικό **έξοδο** στην **Οικονομική Επισκόπηση**.
 - **String ln_sum2**: Μεταβλητή τύπου **String**. Παίρνει τιμή το **συνολικό ποσό του εξόδου** που επιλέγηκε σχετικά με τις ημερομηνίες.
 - **String check**: Μεταβλητή τύπου **Int**. Παίρνει σαν τιμή ένα **αριθμό** που ορίζει **ποιο έξοδο επιλέγηκε**.
 - **String counter**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **πλήθος των εξόδων** που ανακτήθηκαν από την SQL εντολή.
 - **String doseag_id**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' της **δόσης του εξόδου** που πρόκειται να επεξεργαστεί.
 - **String activid_formenu**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Είδους** από το μενού της **Εμφάνισης**.
 - **String provid_formenu**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Προμηθευτή/Πελάτη** από το μενού της **Εμφάνισης**.
 - **String sdfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Από** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String edfor**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή την ημερομηνία του πεδίου **Μέχρι** στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Connection con**: Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String companyagid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **εξόδου**.
 - **String company**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' της **Εταιρίας** που υπάρχει στην Βάση Δεδομένων στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **String activitiesid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Είδους** του εξόδου που επιλέγει να επεξεργαστεί μία δόση.
 - **String prov_custid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **Προμηθευτή/Πελάτη** του εξόδου που επιλέγει να επεξεργαστεί μία δόση.
- **public String saveEditDoseCompanyFor(String companyforid, String dose_compl, String price, String when, String potential, Connection con, boolean incomeFlag, String dosepaidid)**
Μέθοδος που αποθηκεύει μια **δόση** κάποιου αναδρομικού **εσόδου** μετά από **επεξεργασία** στην **Οικονομική Επισκόπηση**.
 - **String companyforid**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το 'id' του **εσόδου** που επιλέγει να επεξεργαστεί μία δόση.
 - **String dose_compl**: Μεταβλητή τύπου **String**. Παίρνει τιμή **1** ή **0**. Το 1 ορίζει το αν η δόση έχει πληρωθεί και το 0 το αν δεν έχει πληρωθεί.
 - **String price**: Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **ποσό της δόσης** που μόλις καταχωρήθηκε και που πρέπει να πληρώνεται.

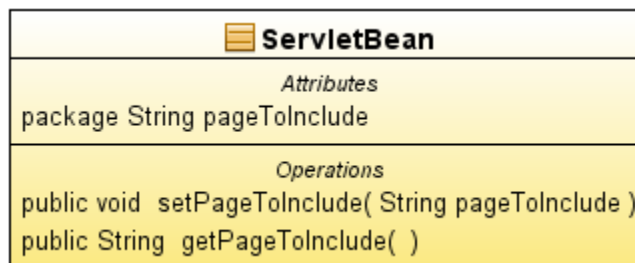
- **String when:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **μια ημερομηνία** που είναι το **πότε** **πληρώθηκε** η συγκεκριμένη δόση.
- **String potential:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **μια ημερομηνία** που είναι το **μέχρι πότε πρέπει να πληρωθεί** η συγκεκριμένη δόση.
- **Connection con:** Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
- **Boolean incomeFlag:** Μεταβλητή τύπου **Boolean**. Σαν αρχική τιμή παίρνει **true**. Χρησιμοποιείται για τον **έλεγχο** στις περιπτώσεις που υπάρχουν για να επιτρέψει την αποθήκευση μιας δόσης ενός εσόδου. Εάν μείνει true τότε αποθηκεύεται η δόση αλλιώς δεν αποθηκεύεται.
- **String dosepaidid:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** της δόσης του εσόδου που επιλέγει να επεξεργαστεί μία δόση.
- **public String saveEditDoseCompanyAgainst(String companyagid, String dose_compl, String price, String when, String potential, Connection con, boolean incomeFlag, String dosepaidid)**
Μέθοδος που αποθηκεύει μια **δόση** κάποιου αναδρομικού **εξόδου** μετά από **επεξεργασία** στην **Οικονομική Επισκόπηση**.
 - **String companyagid:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** του **εξόδου** που επιλέγει να επεξεργαστεί μία δόση.
 - **String dose_compl:** Μεταβλητή τύπου **String**. Παίρνει τιμή **1 ή 0**. Το 1 ορίζει το αν η δόση έχει πληρωθεί και το 0 το αν δεν έχει πληρωθεί.
 - **String price:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το ποσό **της δόσης** που μόλις καταχωρήθηκε και που πρέπει να πληρώνεται.
 - **String when:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **μια ημερομηνία** που είναι το **πότε** **πληρώθηκε** η συγκεκριμένη δόση.
 - **String potential:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή **μια ημερομηνία** που είναι το **μέχρι πότε πρέπει να πληρωθεί** η συγκεκριμένη δόση.
 - **Connection con:** Μεταβλητή τύπου **Connection**. Παίρνει σαν τιμή τις κατάλληλες πληροφορίες που χρειάζονται για τη σύνδεση με τη Βάση στη σελίδα της **Οικονομικής Επισκόπησης**.
 - **Boolean incomeFlag:** Μεταβλητή τύπου **Boolean**. Σαν αρχική τιμή παίρνει **true**. Χρησιμοποιείται για τον **έλεγχο** στις περιπτώσεις που υπάρχουν για να επιτρέψει την αποθήκευση μιας δόσης ενός εξόδου. Εάν μείνει true τότε αποθηκεύεται η δόση αλλιώς δεν αποθηκεύεται.
 - **String dosepaidid:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το **'id'** της δόσης του **εξόδου** που επιλέγει να επεξεργαστεί μία δόση.
- **public boolean datevalidate(String date)**
Η μέθοδος αυτή ελέγχει αν η ημερομηνία που έχει δοθεί έχει σωστό τύπο σύνταξης για ημερομηνίες, ή αν η ημερομηνία είναι υπαρκτή.
Αν η τιμή που επιστρέφει είναι αποδεκτή, τότε η μέθοδος παίρνει τιμή **true**, αλλιώς παίρνει τιμή **false**.
 - **String date:** Μεταβλητή τύπου **String** η οποία για να επιστρέψει η μέθοδος **true** πρέπει να είναι ημερομηνία της μορφής **"yyyy-mm-dd"**. Την μεταβλητή αυτή την παίρνουμε από τα πεδία των **ημερομηνιών** κατά την καταχώρηση και επεξεργασία όλων των δεδομένων της σελίδας **Οικονομική Επισκόπηση**.
- **public static boolean isNumeric(String str)**
Μέθοδος όπου παίρνει σαν όρισμα μια μεταβλητή **String** και ελέγχει αν είναι αριθμός.
 - **String str:** Μεταβλητή τύπου **String** που ορίζεται στην μέθοδο **isNumeric**. Την τιμή της την παίρνουμε από τα πεδία των **Ποσών**. Για να ισχύσει η μέθοδος πρέπει να είναι αριθμός **ακέραιος ή δεκαδικός**. Μόνο τότε μας επιστρέφει **true**.

▪ ServletBean.java:

Είναι η κλάση που δημιουργείται για να χρησιμοποιηθεί στις JSP σελίδες. Αυτή φορτώνεται κάνοντας τρέχων ένα αντικείμενό της και το δεσμεύουμε σε μια μεταβλητή με το όνομα που διευκρινίζεται από το "id".

π.χ. `<jsp:useBean id="financial" class="financial.ServletBean" />`

Για να ορίσουμε τις ιδιότητες του Bean χρησιμοποιούμε την `jsp:setProperty` όπου παίρνει τιμή για `name` ίδια με το `id` του `jsp:useBean` και την ιδιότητα `property` η οποία ορίζει την ιδιότητα ενδιαφέροντος.



Εικόνα 4.2.15. UML του Java Class ServletBean

Μεταβλητές:

- **Package String pageToInclude**

Μεταβλητή η οποία παίρνει σαν όρισμα ποιο Servlet θα εμφανιστεί.

Μέθοδοι:

- **Public void setPageToInclude(String pageToInclude)**

Είναι μια μέθοδος (**Constructor**) που κατασκευάζει ένα αντικείμενο pageToInclude.

- **String pageToInclude:** Μεταβλητή τύπου **String**. Παίρνει σαν τιμή το ποιο **Servlet** θα προβληθεί στην κάθε σελίδα.

- **Public String getPageToInclude()**

Είναι μια μέθοδος (**Constructor**) που επιστρέφει ένα αντικείμενο pageToInclude.

4.3 – MySQL

4.3.1 – Βασικά Χαρακτηριστικά MySQL

Η MySQL είναι ένα πολύ γρήγορο και δυνατό, σύστημα **διαχείρισης βάσεων δεδομένων**. Μια βάση δεδομένων επιτρέπει την αποθήκευση, την αναζήτηση, την ταξινόμηση και την ανάκληση των δεδομένων αποτελεσματικά. Ο MySQL διακομιστής ελέγχει την πρόσβαση στα δεδομένα σας, για να μπορούν να δουλεύουν πολλοί χρήστες ταυτόχρονα, για να παρέχει γρήγορη πρόσβαση και να διασφαλίζει ότι μόνο πιστοποιημένοι χρήστες μπορούν να έχουν πρόσβαση. Συνεπώς η MySQL είναι ένας πολυνηματικός διακομιστής πολλαπλών χρηστών. Χρησιμοποιεί την SQL (Structured Query Language) την τυπική γλώσσα ερωτημάτων για βάσεις δεδομένων, παγκόσμια. Η MySQL είναι διαθέσιμη από το 1996 αλλά η ιστορία της ξεκινά από το 1979.

Πλεονεκτήματα της MySQL

Μερικοί από τους κύριους ανταγωνιστές της MySQL είναι οι PostgreSQL, Microsoft SQL και Oracle. Η MySQL έχει πολλά πλεονεκτήματα, όπως χαμηλό κόστος, εύκολη διαμόρφωση και μάθηση και ο κώδικας προέλευσης είναι διαθέσιμος.

Απόδοση

Η MySQL είναι χωρίς αμφιβολία γρήγορη. Πολλές δοκιμές δείχνουν ότι η MySQL είναι αρκετά πιο γρήγορη από τον ανταγωνισμό.

Χαμηλό κόστος

Η MySQL είναι διαθέσιμη δωρεάν, με άδεια ανοικτού κώδικα (Open Source) ή με χαμηλό κόστος, αν πάρετε εμπορική άδεια, αν απαιτείται από την εφαρμογή σας.

Ευκολία Χρήσης

Οι περισσότερες μοντέρνες βάσεις δεδομένων χρησιμοποιούν SQL. Αν έχετε χρησιμοποιήσει ένα άλλο σύστημα διαχείρισης βάσεων δεδομένων δεν θα έχετε πρόβλημα να προσαρμοστείτε σε αυτό.

Μεταφερσιμότητα

Η MySQL μπορεί να χρησιμοποιηθεί σε πολλά διαφορετικά συστήματα Unix όπως επίσης και στα Microsoft Windows .

Κώδικας Προέλευσης

Μπορείτε να πάρετε και να τροποποιήσετε τον κώδικα προέλευσης της MySQL.

Οι πίνακες δεδομένων που χρησιμοποιήθηκαν στο project είναι:

- **activities**
- **activity**
- **birthday**
- **company**
- **companyagainst**
- **companyfor**
- **dosepaidag**
- **dosepaidfor**
- **email**
- **entry**
- **income**
- **phonebook**
- **phonebookinfo**
- **users**

4.3.2 – Πίνακες της MySQL Βάσης Δεδομένων(financial)

Activities

Αποθηκεύονται πληροφορίες για το είδος της πληρωμής, εσόδου ή εξόδου, και αποτελείται από τα πεδία:

- **id(bigint(20)):**
Ο αριθμός που κάνει μοναδικό, το κάθε είδος
- **kind_of_activity(varchar(40)):**
Η ονομασία του είδους της πληρωμής(εσόδου ή εξόδου)
- **Status(tinyint(1)):**
Η ενεργοποίηση ή απενεργοποίηση εμφάνισης κάποιου από τα είδη πληρωμής, εσόδων ή εξόδων

Activity

Αποθηκεύονται πληροφορίες για τις ενέργειες που έχει ο χρήστης και χρειάζεται την υπενθύμιση τους, όπως ραντεβού κ.α., και αποτελείται από τα πεδία:

- **id(bigint(20)):**
Ο αριθμός που κάνει μοναδική, την κάθε ενέργεια
- **activity(varchar(40)):**
Η ονομασία της ενέργειας
- **date_start(date)::**
Η ημερομηνία που ξεκινάει η ενέργεια

- **date_end(date):**
Η ημερομηνία που λήγει η ενέργεια
- **notes(varchar(40)):**
Οι σημειώσεις που περιγράφουν την ενέργεια

Birthday

Αποθηκεύονται πληροφορίες για τα γενέθλια των επαφών που βρίσκονται στον τηλεφωνικό κατάλογο, και αποτελείται από τα πεδία:

- **id(bigint(20)):**
Ο αριθμός που κάνει μοναδικά, τα γενέθλια της κάθε επαφής
- **notes(varchar(40)):**
Σημειώσεις που περιγράφουν την ιδιότητα της επαφής
- **date(date):**
Η ημερομηνία γέννησης της επαφής

Company

Αποθηκεύονται πληροφορίες για τα στοιχεία που περιγράφουν μια εταιρία και αποτελείται από τα πεδία:

- **id(bigint(20)):**
Ο αριθμός που κάνει μοναδική, την κάθε εταιρία
- **title_distinctive(varchar(40)):**
Η σύντομη επωνυμία της εταιρίας
- **full_name(varchar(40)):**
Η πλήρης επωνυμία της εταιρίας
- **business:**
Το αντικείμενο απασχόλησης της εταιρίας
- **AFM(varchar(40)):**
Ο Α.Φ.Μ (Αριθμός Φορολογικού Μητρώου) της εταιρίας
- **DOY(varchar(40)):**
Η Δ.Ο.Υ. (Διεύθυνση Οικονομικών Υπηρεσιών) της εταιρίας
- **full_address(varchar(40)):**
Η διεύθυνση στην οποία έχει έδρα η εταιρία
- **fax(varchar(40)):**
Το fax της εταιρίας
- **website(varchar(40)):**
Η ιστοσελίδα που έχει η επιχείρηση
- **percentage_owned(varchar(40)):**
Το ποσοστό ιδιοκτησίας της εταιρίας, εάν υπάρχει
- **company_status(tinyint(1)):**
Η ενεργοποίηση ή απενεργοποίηση εμφάνισης κάποιας από τις εταιρίες

Companyagainst

Αποθηκεύονται πληροφορίες που περιγράφουν τα έξοδα του διαχειριστή προς τις συνεργαζόμενες εταιρίες και αποτελείται από τα πεδία:

- **id(bigint(20)):**
Ο μοναδικός αριθμός που κάνει μοναδικό, το κάθε έξοδο
- **company_id(bigint(20)):**
Ο μοναδικός αριθμός που αντιστοιχεί στην συγκεκριμένη εταιρία, τον οποίο παίρνουμε από τον πίνακα company
- **activities_id(bigint(20)):**

Ο μοναδικός αριθμός που αντιστοιχεί στο συγκεκριμένο είδος πληρωμής, τον οποίο παίρνουμε από τον πίνακα activities

- **price(float(12,1)):**
Το ποσό πληρωμής
- **date_start(date):**
Η ημερομηνία που ξεκινάει η προθεσμία πληρωμής
- **date_end(date):**
Η ημερομηνία που λήγει η προθεσμία πληρωμής
- **income_id(bigint(20)):**
Ο μοναδικός αριθμός που αντιστοιχεί στο συγκεκριμένο είδος προμηθευτή ή πελάτη, τον οποίο παίρνουμε από τον πίνακα income
- **notes(varchar(100)):**
Περαιτέρω σημειώσεις που εξηγούν το συγκεκριμένο έξοδο
- **isrecursive(tinyint(1)):**
Η επιλογή αναδρομικότητας του εξόδου
- **whenrecursive(varchar(40)):**
Η συχνότητα πληρωμής του εξόδου αν είναι αναδρομικό
- **completed(tinyint(1)):**
Η επιλογή εξόφλησης του εξόδου
- **when(date):**
Η ημερομηνία που εξοφλήθη το έξοδο
- **difference(float(12,1)):**
Η τυχόν διαφορά ανάμεσα στο πόσο πληρωμής και το ποσό εξόφλησης
- **days_or_months(varchar(10)):**
Η επιλογή συχνότητας του εξόδου σε μέρες ή μήνες

Companyfor

Αποθηκεύονται πληροφορίες που περιγράφουν τα έσοδα του διαχειριστή από τις συνεργαζόμενες εταιρίες και αποτελείται από τα πεδία:

- **id(bigint(20)):**
Ο μοναδικός αριθμός που κάνει μοναδικό, το κάθε έσοδο
- **company_id(bigint(20)):**
Ο μοναδικός αριθμός που αντιστοιχεί στην συγκεκριμένη εταιρία, τον οποίο παίρνουμε από τον πίνακα company
- **activities_id(bigint(20)):**
Ο μοναδικός αριθμός που αντιστοιχεί στο συγκεκριμένο είδος πληρωμής, τον οποίο παίρνουμε από τον πίνακα activities
- **price(float(12,2)):**
Το ποσό πληρωμής
- **date_start(price):**
Η ημερομηνία που ξεκινάει η προθεσμία πληρωμής
- **date_end(price):**
Η ημερομηνία που λήγει η προθεσμία πληρωμής
- **income_id(bigint(20)):**
Ο μοναδικός αριθμός που αντιστοιχεί στο συγκεκριμένο είδος προμηθευτή ή πελάτη, τον οποίο παίρνουμε από τον πίνακα income
- **notes(varchar(100)):**
Περαιτέρω σημειώσεις που εξηγούν το συγκεκριμένο έσοδο
- **isrecursive(tinyint(1)):**
Η επιλογή αναδρομικότητας του εσόδου
- **whenrecursive(varchar(40)):**

- Η συχνότητα πληρωμής του εσόδου αν είναι αναδρομικό
- **completed**(tinyint(1)):
Η επιλογή εξόφλησης του εσόδου
- **when**(price):
Η ημερομηνία που εξοφλήθηκε το έσοδο
- **difference**(float(12,2)):
Η τυχόν διαφορά ανάμεσα στο πόσο πληρωμής και το ποσό εξόφλησης
- **days_or_months**(varchar(10)):
Η επιλογή συχνότητας του εσόδου σε μέρες ή μήνες

Dosepaidag

Αποθηκεύονται πληροφορίες που περιγράφουν τις δόσεις του εξόδου του διαχειριστή προς μια εταιρία και αποτελείται από τα πεδία:

- **id**(bigint(10)):
Ο μοναδικός αριθμός που κάνει μοναδική, την κάθε δόση από κάποιο έξοδο
- **companyagainst_id**(bigint(10)):
Ο μοναδικός αριθμός που αντιστοιχεί σε ένα συγκεκριμένο έξοδο, τον οποίο παίρνουμε από τον πίνακα companyagainst
- **when**(date):
Η ημερομηνία που πληρώθηκε η δόση του εξόδου
- **completed**(tinyint(1)):
Η επιλογή ολοκλήρωσης της πληρωμής της δόσης
- **potential**(date):
Η ημερομηνία που λήγει η προθεσμία της πληρωμής της δόσης
- **price**(float(12,2)):
Το ποσό που πληρώθηκε

Dosepaidfor

Αποθηκεύονται πληροφορίες που περιγράφουν τις δόσεις του εσόδου του διαχειριστή από μια εταιρία και αποτελείται από τα πεδία:

- **id**(bigint(10)):
Ο μοναδικός αριθμός που κάνει μοναδική, την κάθε δόση από κάποιο έσοδο
- **companyfor_id**(bigint(10)):
Ο μοναδικός αριθμός που αντιστοιχεί σε ένα συγκεκριμένο έσοδο, τον οποίο παίρνουμε από τον πίνακα companyfor
- **when**(date):
Η ημερομηνία που πληρώθηκε η δόση του εσόδου
- **completed**(tinyint(1)):
Η επιλογή ολοκλήρωσης της πληρωμής της δόσης
- **potential**(date):
Η ημερομηνία που λήγει η προθεσμία της πληρωμής της δόσης
- **price**(float(12,2)):
Το ποσό που πληρώθηκε

Email

Αποθηκεύονται τα e-mail των επαφών που βρίσκονται στον τηλεφωνικό κατάλογο και αποτελείται από τα πεδία:

- **id**(bigint(20)):
Ο μοναδικός αριθμός που κάνει μοναδικό, το κάθε email
- **email**(varchar(40)):
Το email της επαφής

Entry

Αποθηκεύεται το ονοματεπώνυμο των επαφών που βρίσκονται στον τηλεφωνικό κατάλογο και αποτελείται από τα πεδία:

- **id(bigint(20))**:
Ο μοναδικός αριθμός που κάνει μοναδικό, το κάθε ονοματεπώνυμο
- **name(varchar(40))**:
Το όνομα της επαφής
- **surname(varchar(40))**:
Το επίθετο της επαφής

Income

Αποθηκεύονται τα στοιχεία των προμηθευτών ή των πελατών και αποτελείται από τα πεδία:

- **id(bigint(20))**:
Ο μοναδικός αριθμός που κάνει μοναδικό, τον κάθε προμηθευτή ή πελάτη
- **income_type(varchar(40))**:
Η περιγραφή του προμηθευτή ή πελάτη
- **status(tinyint(1))**:
Η ενεργοποίηση ή απενεργοποίηση εμφάνισης κάποιου από τους προμηθευτές/πελάτες

Phonebook

Αποθηκεύεται το τηλέφωνο των επαφών που βρίσκονται στον τηλεφωνικό κατάλογο και αποτελείται από τα πεδία:

- **id(bigint(20))**:
Ο μοναδικός αριθμός που κάνει μοναδικό, το κάθε τηλέφωνο
- **phonebook(varchar(40))**:
Το τηλέφωνο της επαφής

Phonebookinfo

Αποθηκεύονται πληροφορίες που περιγράφουν στοιχεία των επαφών και αποτελείται από τα πεδία:

- **id(bigint(20))**:
Ο μοναδικός αριθμός που κάνει μοναδικό, το κάθε τηλέφωνο
- **phonebook_id(bigint(20))**:
Ο μοναδικός αριθμός που αντιστοιχεί σε ένα συγκεκριμένο αριθμό τηλεφώνου, τον οποίο παίρνουμε από τον πίνακα phonebook
- **entry_id(bigint(20))**:
Ο μοναδικός αριθμός που αντιστοιχεί σε ένα συγκεκριμένο ονοματεπώνυμο, τον οποίο παίρνουμε από τον πίνακα entry
- **user_id(bigint(20))**:
Ο μοναδικός αριθμός που αντιστοιχεί στον συγκεκριμένο διαχειριστή, τον οποίο παίρνουμε από τον πίνακα user
- **birthday_id(bigint(20))**:
Ο μοναδικός αριθμός που αντιστοιχεί σε συγκεκριμένα γενέθλια, τον οποίο παίρνουμε από τον πίνακα birthday
- **email_id(bigint(20))**:
Ο μοναδικός αριθμός που αντιστοιχεί σε ένα συγκεκριμένο e-mail, τον οποίο παίρνουμε από τον πίνακα email

Users

Αποθηκεύονται πληροφορίες που περιγράφουν τον χρήστη που διαχειρίζεται την εφαρμογή και αποτελείται από τα πεδία:

- **id(bigint(20)):**
Ο μοναδικός αριθμός που κάνει μοναδικό, τον κάθε χρήστη
- **user(varchar(40)):**
Το όνομα του χρήστη
- **username(varchar(40)):**
Το username εισόδου του
- **password(varchar(40)):**
Το password εισόδου του

4.4 – CSS

4.4.1 – Βασικά Χαρακτηριστικά CSS

Το CSS είναι μια απλή γλώσσα που μας βοηθάει να ορίσουμε με σαφήνεια και ιδιαίτερη ευελιξία τον τρόπο με τον οποίο θα εμφανίζονται τα διάφορα στοιχεία στην ιστοσελίδα μας.

Πλεονεκτήματα της MySQL

Μερικοί από τους κύριους ανταγωνιστές της MySQL είναι οι PostgreSQL, Microsoft SQL και Oracle. Η MySQL έχει πολλά πλεονεκτήματα, όπως χαμηλό κόστος, εύκολη διαμόρφωση και μάθηση και ο κώδικας προέλευσης είναι διαθέσιμος.

Μεγαλύτερη Ευελιξία

Το CSS κατέστησε εφικτές μορφοποιήσεις οι οποίες ήταν αδύνατες ή πολύ δύσκολες με την κλασσική HTML.

Ευκολότερη Συντήρηση των Ιστοσελίδων

Η εμφάνιση ενός ολόκληρου site μπορεί να ελέγχεται από ένα μόνο εξωτερικό αρχείο CSS. Έτσι, κάθε αλλαγή στο στυλ της ιστοσελίδας μπορεί να γίνεται με μια μοναδική αλλαγή σε αυτό το αρχείο, αντί για την επεξεργασία πολλών σημείων σε κάθε σελίδα που υπάρχει στο site.

Μικρότερο Μέγεθος Αρχείου

Δεδομένου ότι ο κάθε κανόνας μορφοποίησης γράφεται μόνο μια φορά και όχι σε κάθε σημείο που εφαρμόζεται.

Καλύτερο SEO(Search Engine Optimization)

Οι μηχανές αναζήτησης δεν «μπερδεύονται» ανάμεσα σε περιεχόμενο και τη μορφοποίηση του, αλλά έχουν πρόσβαση στο περιεχόμενο σκέτο, οπότε είναι πολύ ευκολότερο να το καταγράψουν και να το αρχειοθετήσουν (indexing).

Γρηγορότερες Σελίδες

Όταν χρησιμοποιούμε εξωτερικό αρχείο CSS (βλ. *Πως εισάγουμε CSS στη σελίδα μας:*), ο browser την πρώτη φορά που θα φορτώσει κάποια σελίδα του site μας το αποθηκεύει στην cache, οπότε δεν χρειάζεται να το κατεβάσει ξανά κάθε φορά που κατεβάζει ο χρήστης του κάποια άλλη σελίδα του site μας.

Χρησιμοποιήθηκαν 2 αρχεία CSS για να καλύψουν τις απεικονιστικές ανάγκες της εφαρμογής.

- **Style**
- **Calendar**

4.4.2 – Αρχεία CSS

- **Style.css:**
Είναι το βασικό αρχείο CSS που χρησιμοποιείται για τις εικαστικές και όχι μόνο επιλογές της απεικόνισης της εφαρμογής.
- **Calendar.css:**
Κατευθύνει το στυλ του `calendar_db.js`.

4.5 – JavaScript

4.5.1 – Βασικά Χαρακτηριστικά JavaScript

Η **JavaScript** είναι γλώσσα προγραμματισμού η οποία έχει σαν σκοπό την παραγωγή δυναμικού περιεχομένου και την εκτέλεση κώδικα στην πλευρά του πελάτη (client-side) σε ιστοσελίδες. Το πρότυπο της γλώσσας κατά τον οργανισμό τυποποίησης ECMA ονομάζεται **ECMAScript**.

Η αρχική έκδοση της Javascript βασίστηκε στη σύνταξη στη γλώσσα προγραμματισμού C, αν και έχει εξελιχθεί, ενσωματώνοντας πια χαρακτηριστικά από νεότερες γλώσσες.

Παρά την ευρεία χρήση της Javascript για συγγραφή προγραμμάτων σε περιβάλλον φυλλομετρητή, αξίζει να σημειωθεί ότι από την αρχή χρησιμοποιήθηκε και για τη συγγραφή κώδικα από την πλευρά του διακομιστή, από την ίδια τη Netscape στο προϊόν LiveWire, με μικρή επιτυχία. Η χρήση της Javascript στο διακομιστή εμφανίζεται πάλι σήμερα, με τη διάδοση του Node.js, ενός μοντέλου προγραμματισμού βασισμένο στα γεγονότα (events).

Αρχικά χρησιμοποιήθηκε για προγραμματισμό από την πλευρά του πελάτη (client), που ήταν ο φυλλομετρητής (browser) του χρήστη, και χαρακτηρίστηκε σαν *client-side γλώσσα προγραμματισμού*. Αυτό σημαίνει ότι η επεξεργασία του κώδικα Javascript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στο διακομιστή, αλλά στο πρόγραμμα περιήγησης των επισκεπτών, ενώ μπορεί να ενσωματωθεί σε στατικές σελίδες HTML. Αντίθετα, άλλες γλώσσες όπως η PHP εκτελούνται στο διακομιστή (*server-side γλώσσες προγραμματισμού*).

Η Javascript δεν θα πρέπει να συγχέεται με τη Java, που είναι διαφορετική γλώσσα προγραμματισμού και με διαφορετικές εφαρμογές. Η χρήση της λέξης "Java" στο όνομα της γλώσσας έχει περισσότερη σχέση με το προφίλ του προϊόντος που έπρεπε να έχει και λιγότερο με κάποια πιθανή συμβατότητα ή άλλη στενή σχέση με τη Java. Ρόλο σε αυτήν τη σύγχυση έπαιξε και ότι η Java και η Javascript έχουν δεχτεί σημαντικές επιρροές από τη γλώσσα C, ειδικά στο συντακτικό, ενώ είναι και οι δύο αντικειμενοστρεφείς γλώσσες. Τονίζεται ότι ο σωστός τρόπος γραφής της είναι "**JavaScript**" και όχι 'Java script' σαν δύο λέξεις, όπως λανθασμένα γράφεται ορισμένες φορές.

Χρησιμοποιήθηκαν 2 αρχεία **JavaScript** για να καλύψουν τις απεικονιστικές ανάγκες της εφαρμογής.

- **timer**
- **calendar**
- **Toggle_boxes**
- **Toggle_dates**

4.5.2 – Αρχεία JavaScript

- **timer.js:**

Το αρχείο αυτό χρησιμοποιείται για την εμφάνιση ενός ψηφιακού ρολογιού όπου δείχνει την τρέχουσα ώρα, όπως επίσης και την τρέχουσα ημερομηνία.

- **calendar_db.js:**

Το αρχείο αυτό χρησιμοποιείται για την επιλογή ημερομηνίας εμφανίζοντας ένα γραφικό ημερολόγιο για την σωστή σύνταξη της και για ευκολία στον χρήστη.

Τα παρακάτω δυναμικά αρχεία JAVASCRIPT στην πραγματικότητα εκτελούν την ίδια λειτουργία, όμως το καθένα από αυτά χρησιμεύει σε διαφορετική περίπτωση. Μέσα σ' αυτά, υπάρχει μια συνάρτηση-μέθοδος, όπου, όταν κάποιο **checkbox επιλεγεί, ενεργοποιεί ή απενεργοποιεί κάποια πεδία κειμένου(textboxes).*

- **toggle_boxes.js:**

Καλώντας την συνάρτηση μέσα στο αρχείο αυτό, όταν το checkbox επιλεγεί, ενεργοποιεί **δυο** πεδία κειμένου(textboxes), και αντίστοιχα, όταν από-επιλεγεί, τα απενεργοποιεί.

- **toggle_dates.js:**

Καλώντας την συνάρτηση μέσα στο αρχείο αυτό, όταν το checkbox επιλεγεί, απενεργοποιεί **δυο** πεδία κειμένου(textboxes), και αντίστοιχα, όταν από-επιλεγεί, τα ενεργοποιεί.

ΚΕΦΑΛΑΙΟ 5

Εγχειρίδιο Χρήστη

5.1 – Εισαγωγή

Η εφαρμογή **Financial** μπορεί να χρησιμοποιηθεί σε έναν Internet Browser, όπως ο Internet Explorer, Mozilla Firefox, Google chrome κ.α. Η εφαρμογή για διευκόλυνση χρήσης αποτελείται από κάποια στατικά μέρη.

Αυτά είναι:

- a) **Το μενού**
 - Βρίσκεται στην κορυφή της εφαρμογής.
- b) **Η προβολή του συνδεδεμένου χρήστη**
 - Βρίσκεται δεξιά κάτω από το μενού
- c) **Η τρέχουσα ημερομηνία.**
 - Βρίσκεται αριστερά στο τέλος της σελίδας
- d) **Η τρέχουσα ώρα.**
 - Βρίσκεται δεξιά στο τέλος της σελίδας

Τα παραπάνω είναι τα στοιχεία που θα συναντώνται σε κάθε σελίδα της εφαρμογής. Για να μετακινήστε στις διάφορες σελίδες-λειτουργίες χρησιμοποιείτε το μενού. Οι **σελίδες** είναι οι παρακάτω:

- Αρχική
- Εταιρίες
- Ημερολόγιο
- Τηλεφωνικός Κατάλογος
- Οικονομική Επισκόπηση
- Έξοδος

, για τα οποία θα μιλήσουμε αναλυτικά στις επόμενες σελίδες. Πρώτα όμως, για να χρησιμοποιηθούν όλες οι λειτουργίες της εφαρμογής, θα πρέπει ο χρήστης να συνδεθεί με τον του.



The image shows a login form with the title "ΣΥΝΔΕΣΗ" in blue. It contains two input fields: "Username:" with the text "user" and "Password:" with masked characters "....". Below the fields is a "Login" button.

Εικόνα 5.1.1. Σύνδεση Χρήστη

τα οποία Για να γίνει η σύνδεση πρέπει να εισαχθεί ένα σωστό username με το κωδικό του. Αν τα στοιχεία είναι σωστά, τότε η σύνδεση γίνεται επιτυχώς, αλλιώς πρέπει να ξαναγίνει προσπάθεια σύνδεσης.

5.2 – Αρχική

Εδώ μπορείτε να δείτε όλα τα στοιχεία που αφορούν την σημερινή ημέρα. *Πληρωμές, γενέθλια και ενέργειες* που περιέχονται στην σημερινή ημέρα βρίσκονται εδώ για να σας υπενθυμίσουν τις υποχρεώσεις σας

Επίσης, σε αυτή την σελίδα μπορείτε να τροποποιήσετε τα στοιχεία του λογαριασμού σας. Μπορείτε δηλαδή να αλλάξετε το *Username*, το *Password* το *Όνοματεπώνυμο* σας.

Στην αρχική σελίδα το σύστημα υπενθυμίζει στον χρήστη τις υποχρεώσεις της ημέρας που έχει να κάνει με:

- **Πληρωμές**
 - *Εσοδα*
 - *Εξοδα*
- **Ενέργειες**
- **Γενέθλια**

5.2.1 – Στοιχεία Λογαριασμού

Όνοματεπώνυμο:	<input type="text" value="pantelis"/>
Όνομα Χρήστη:	<input type="text" value="admin"/>
Κωδικός:	<input type="password" value="....."/>
Επιστροφή	Αλλαγή στοιχείων

Εικόνα 5.2.1. Στοιχεία Λογαριασμού

Μετά το πάτημα του κουμπιού **Στοιχεία Λογαριασμού** γίνεται η προβολή των στοιχείων του χρήστη. Αυτά είναι:

- ✚ Όνοματεπώνυμο – Όνομα και επίθετο του χρήστη
- ✚ Όνομα Χρήστη – Username Εισόδου
- ✚ Κωδικός – Κωδικός Εισόδου

a) Αλλαγή στοιχείων

Όνοματεπώνυμο:	<input type="text" value="user1"/>
Όνομα Χρήστη:	<input type="text" value="user"/>
Κωδικός:	<input type="password" value="...."/>
Επιβεβαίωση Κωδικού:	<input type="password" value="...."/>
Επιστροφή	<input type="button" value="Αποθήκευση στοιχείων"/>

Εικόνα 5.2.2. Αλλαγή Στοιχείων Λογαριασμού

Εδώ ο χρήστης μπορεί να τροποποιήσει τα στοιχεία του λογαριασμού του. Μπορεί να αλλάξει τα:

- Όνοματεπώνυμο
- Όνομα Χρήστη
- Κωδικός
- Επιβεβαίωση Κωδικού

*Για την **σωστή αποθήκευση** των αλλαγών πρέπει οι δύο κωδικοί να είναι **ίδιοι**.

Μετά την συμπλήρωση των στοιχείων, επιλέγετε το πλήκτρο **Αποθήκευση στοιχείων**, για την καταχώρηση της δόσης, αλλιώς επιλέγετε το πλήκτρο **Επιστροφή**, για να επιστρέψετε στην προηγούμενη κατάσταση.

i. Αποθήκευση στοιχείων

Μετά την επιλογή του πλήκτρου **Αποθήκευση στοιχείων**, θα σας εμφανιστεί ένα μήνυμα αναφοράς. Οι περιπτώσεις είναι 2:

- a. Αν η αποθήκευση γίνει επιτυχώς :

[Τα στοιχεία αποθηκεύτηκαν με επιτυχία!]

- b. Αν υπάρχουν λάθη κατά την αποθήκευση δεδομένων :

[Κάποιο από τα πεδία είναι άδειο. Παρακαλώ εισάγετε όλα τα στοιχεία.]

ii. Επιστροφή

Μετά την επιλογή του πλήκτρου **Επιστροφή**, το πρόγραμμα θα επιστρέψει στην προηγούμενη κατάσταση που δείχνει η Εικόνα 5.2.1(Στοιχεία Λογαριασμού).

b) Επιστροφή

Το σύστημα επιστρέφει στην **Αρχική** σελίδα

5.2.2 – Πληρωμές

Η Αρχική σελίδα δείχνει στον χρήστη τα **έσοδα** και **έξοδα** ή τις δόσεις τους που πρέπει να πληρωθούν άμεσα επειδή η προθεσμία τους λήγει άμεσα.

Αν δεν υπάρχει κάποιο έσοδο, ή κάποιο έξοδο το οποίο επείγει, τότε εμφανίζεται το παρακάτω μήνυμα:

Έξοδα									
Α/α	Εταιρία	Είδος	Ποσό/Δόση	Προμήθ/Πελάτης	Αναδρομικότητα	Κάθε Πότε	Ημέρα Πληρωμής	Ποσό	Πληρώθηκε
Δεν υπάρχουν πληρωμές εξόδων που να εκκρεμούν για σήμερα									

Εικόνα 5.2.3. Άδεια Φόρμα Πληρωμών Τρέχουσας Ημέρας

Αν όμως υπάρχει κάποια πληρωμή, είτε είναι έσοδο είτε έξοδο, τότε φαίνεται στην αρχική σελίδα. Οι πληρωμές που εκκρεμούν για την σημερινή ημέρα μπορούν να είναι παραπάνω από μια.

A/A	Εταιρία	Είδος	Ποσό/ Δόση	Προμηθ/Πελάτης	Αναδρομικότητα	Κάθε Πότε	Ημέρα Πληρωμής	Ποσό	Πληρώθηκε
1	webconcept	dei	15.00	Παντελή Παντελής	<input checked="" type="checkbox"/>	1 ημέρα	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>
2	webconcept	dei	15.00	Παντελή Παντελής	<input checked="" type="checkbox"/>	1 ημέρα	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>
3	webconcept	dei	15.00	Παντελή Παντελής	<input checked="" type="checkbox"/>	1 ημέρα	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>
4	webconcept	dei	15.00	Παντελή Παντελής	<input checked="" type="checkbox"/>	1 ημέρα	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>
5	webconcept	dei	15.00	Παντελή Παντελής	<input checked="" type="checkbox"/>	1 ημέρα	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>
6	webconcept	dei	15.00	Παντελή Παντελής	<input checked="" type="checkbox"/>	1 ημέρα	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>
7	webconcept	Φ.Π.Α	100.23	test test	<input type="checkbox"/>	-	<input type="text"/>	<input type="text"/>	<input type="checkbox"/>

Εικόνα 5.2.4. Φόρμα Πληρωμών Τρέχουσας Ημέρας

Όπως φαίνεται και στο παραπάνω σχήμα εάν υπάρχουν εκκρεμείς πληρωμές στην Αρχική σελίδα, τότε μπορούμε να τις κατοχυρώσουμε σαν πληρωμένες εάν όντως έχουν ξεχρεωθεί. Αυτό μπορεί να γίνει είτε μαζικά είτε μεμονωμένα για κάθε πληρωμή.

Πως λειτουργεί:

Υπάρχουν δύο κουμπιά, το **Υποβολή** και το **Καθαρισμός**.

Υποβολή:

Το κουμπί **Υποβολή** κατοχυρώνει όποια ποσά από τις πληρωμές έχουν ξεκλειδώσει, δηλαδή αποθηκεύει τα δεδομένα μόνον όσων πλαισίων στο πεδίο **Πληρώθηκε** έχουν επιλεγεί.

- Αν το πλαίσιο έχει επιλεγεί τότε ξεκλειδώνουν τα 2 πεδία δίπλα από αυτό, δηλαδή τα πεδία **Ημέρα Πληρωμής** και **Ποσό**.
- Στο πεδίο **Ημέρα Πληρωμής** πρέπει να συμπληρωθεί σωστά η ημερομηνία που πληρώθηκε το ποσό ή η δόση του ποσού.
- Το πεδίο **Ημέρα Πληρωμής** πρέπει να συμπληρωθεί από χαρακτήρα **Ημερομηνίας**
- Στο πεδίο **Ποσό** συμπληρώνεται το χρηματικό ποσό που πληρώθηκε.
- Το πεδίο **Ποσό** στις αναδρομικές πληρωμές μπορεί να συμπληρωθεί από μέρος του ποσού της δόσης και όχι από ολόκληρο το ποσό. Στις μη αναδρομικές δόσεις όμως, το ποσό επιβάλλεται αυστηρώς να συμπληρωθεί από το ζητούμενο.
- Το πεδίο **Ποσό** πρέπει να συμπληρωθεί από **Αριθμούς**.
- Όλα τα κενά πεδία πρέπει να **συμπληρωθούν!**

Μετά την σωστή συμπλήρωση των ξεκλειδωμένων κελιών και το πάτημα του κουμπιού **Υποβολή** τότε πραγματοποιείται αποθήκευση των αλλαγών. Αν η αποθήκευση γίνει επιτυχώς τότε θα εμφανιστεί το αντίστοιχο μήνυμα:

[Η υποβολή έγινε επιτυχώς]

Αλλιώς:

- a. Αν κάποιο από τα πεδία παραμείνουν κενά:

[Σφάλμα κατά την υποβολή του αιτήματός σας. Παρακαλώ συμπληρώστε όλα τα πεδία για να προχωρήσετε.]

- b. Αν η καταχωρημένη ημερομηνία είναι συντακτικά λάθος γραμμένη:

[Σφάλμα κατά την υποβολή του αιτήματός σας. Παρακαλώ ελέγξτε για ορθές ημερομηνίες.]

- c. Αν το καταχωρημένο ποσό είναι συντακτικά λάθος γραμμένο:

[Σφάλμα κατά την υποβολή του αιτήματός σας. Παρακαλώ ελέγξτε για σωστό αριθμό ποσού.]

- d. Αν η πληρωμή είναι μη αναδρομική και το ποσό δεν είναι ακριβές με το ζητούμενο:

[Σε κάποιο από τα μη αναδρομικά έσοδα δεν έχει δοθεί σωστό ποσό και σταμάτησε η καταχώρηση. Το ποσό πρέπει να είναι ίδιο με το ποσό που αναγράφεται]

5.2.3 – Ενέργειες

Εδώ το σύστημα σας υπενθυμίζει τις ενέργειες που έχετε να κάνετε την τρέχουσα ημέρα, όπως κάποιο ραντεβού ή επαγγελματικά τηλέφωνα που έχετε να κάνετε.

Αν δεν υπάρχει κάποια ενέργεια η οποία είναι για την τρέχουσα ημέρα τότε εμφανίζεται το παρακάτω μήνυμα:

Ενέργειες				
A/a	Ενέργεια	Ημ. Έναρξης	Ημ. Λήξης	Σημειώσεις
1	thn allh vdomada parousiash	2011-06-13	2011-06-17	orws kai dhrote

Εικόνα 5.2.5. Φόρμα Ενεργειών Τρέχουσας Ημέρας

5.2.4 – Γενέθλια

Επίσης έχετε την δυνατότητα να δείτε ποιοι συνεργάτες, φίλοι και συγγενείς έχουν τα γενέθλια τους την τρέχουσα ημέρα.

Αν δεν υπάρχουν κάποια γενέθλια για την τρέχουσα ημέρα τότε εμφανίζεται το παρακάτω μήνυμα:

Γενέθλια		
A/a	Εορτάζοντες	Ημ. Γέννησης
Δεν υπάρχουν εορτάζοντες για σήμερα.		

Εικόνα 5.2.6. Φόρμα Γενεθλίων Τρέχουσας Ημέρας

5.3 – Εταιρίες

Στην σελίδα αυτή μπορείτε να οργανώσετε το παραστατικό της επιχείρησής σας. Μπορείτε να προβάλετε τις εταιρίες, τους πελάτες και τους προμηθευτές με τους οποίους συνεργάζεστε, ή να προσθέσετε κάτι νέο. Το ίδιο μπορείτε να κάνετε για τα έσοδα ή τα έξοδα σας, ή να προσθέσετε κάποιο νέο είδος.

5.3.1 – Εταιρίες

Εδώ μπορείτε να προβάλετε, να αλλάξετε ή και να διαγράψετε τα στοιχεία των ήδη συνεργαζόμενων εταιριών, ή να προσθέσετε μια νέα.

Εταιρίες	
Προσθήκη Νέας Εταιρίας Προσθήκη Νέας Εταιρίας	Προβολή Εταιριών Προβολή Εταιριών

Εικόνα 5.3.1. Εταιρίες

i. Προσθήκη Νέας Εταιρίας

Τα στοιχεία που μπορείτε να δώσετε σε μια εταιρία είναι:

- ✚ Επωνυμία
- ✚ Όνομα
- ✚ Εργασία

- ✚ Α.Φ.Μ.
- ✚ Δ.Ο.Υ.
- ✚ Διεύθυνση
- ✚ Τηλέφωνο/ΦΑΞ
- ✚ Ιστοσελίδα
- ✚ Ποσοστό

Προσθήκη Νέας Εταιρίας

Επωνυμία:

Όνομα:

Εργασία:

Α.Φ.Μ:

Δ.Ο.Υ.:

Διευθυνση:

ΦΑΞ:

Ιστοσελίδα:

Ποσοστό:

Εικόνα 5.3.2. Προσθήκη Νέας Εταιρίας

Μετά την συμπλήρωση των στοιχείων, επιλέγετε το πλήκτρο **Αποθήκευση**, εφόσον και το επιθυμείτε, αλλιώς επιλέγετε το πλήκτρο **Ακύρωση**, για να επιστρέψετε στην προηγούμενη κατάσταση.

Μετά την επιλογή του πλήκτρου **Αποθήκευση**, θα σας εμφανιστεί ένα μήνυμα αναφοράς. Οι περιπτώσεις είναι 2:

- a. Αν η αποθήκευση γίνει επιτυχώς :

Τα στοιχεία της εταιρίας καταχωρήθηκαν με επιτυχία.

[Επιστροφή](#) [Προβολή Εταιριών](#)

Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.1(Εταιρίες).

Αν επιλέξετε **Προβολή Εταιριών** τότε το πρόγραμμα θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.3(Προβολή Εταιριών).

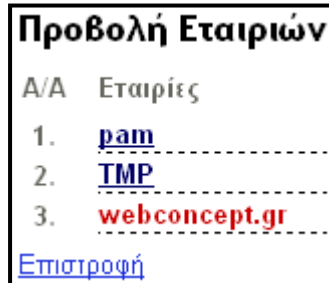
- b. Αν κατά την εισαγωγή στοιχείων, κάποιος τύπος δεδομένων δεν αναγνωρίζεται σε κάποιο από τα στοιχεία ή το πεδίο του παραμένει κενό :

Κάποιο από τα πεδία δεν έχει συμπληρωθεί. Επιστρέψτε στην προηγούμενη σελίδα για να διορθώσετε τα στοιχεία. [Επιστροφή](#)

Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.2(Προσθήκη Νέας Εταιρίας).

ii. Προβολή Εταιριών

Εδώ μπορείτε να δείτε όλες τις Επιχειρήσεις με τις οποίες συνεργάζεστε. Μπορείτε επίσης να διαγράψετε κάποια από τις εταιρίες με τις οποίες η συνεργασία σας έχει λάβει τέλος ή ακόμα και να αλλάξετε κάποια από τις πληροφορίες μιας εταιρίας.



Εικόνα 5.3.3. Προβολή Εταιριών

Τα πλήκτρα που συναντάτε εδώ είναι δυο,

- η **Επιλογή της Ονομασίας της Εταιρίας**
- και η **Επιστροφή**.

a. Επιλογή Ονομασίας Εταιρίας

Με την επιλογή κάποιας εταιρίας θα εμφανιστούν όλα τα στοιχεία που την αφορούν.



Εικόνα 5.3.4. Επισκόπηση Εταιρίας

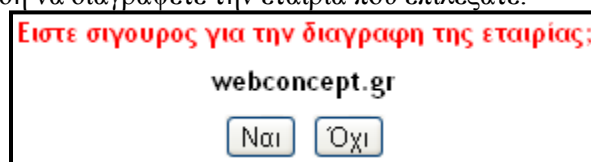
Τα πλήκτρα που συναντάτε είναι δυο, **Αλλαγή Στοιχείων** και **Διαγραφή**.

Αλλαγή Στοιχείων

Αυτή η περίπτωση δουλεύει όπως ακριβώς και η προσθήκη εταιριών, μόνο που αντί να προσθέεται νέα εταιρία, αλλάζουν τα δεδομένα κάποιας υπάρχουσας. Για περισσότερες λεπτομέρειες ανατρέξτε στο κεφάλαιο **Προσθήκη Εταιρίας**(σελ. βφδη).

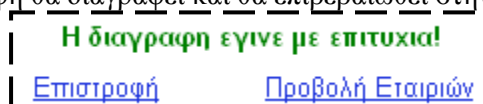
Διαγραφή

Με το πάτημα του κουμπιού θα σας ζητηθεί να επιβεβαιώσετε το αίτημα σας πριν την πραγματοποίησή του, δηλαδή να διαγράψετε την εταιρία που επιλέξατε:



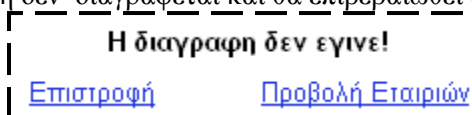
Εικόνα 5.3.5. Διαγραφή Εταιρίας

Αν επιλέξετε **ναι** τότε η επαφή θα διαγραφεί και θα επιβεβαιωθεί στην οθόνη σας με την ένδειξη:



Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.1(Εταιρίες).

Αν επιλέξετε **όχι** τότε η επαφή δεν διαγράφεται και θα επιβεβαιωθεί στην οθόνη σας με την ένδειξη:



Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.4(Επισκόπηση Εταιρίας).

Και στις δυο περιπτώσεις, αν επιλέξετε **Προβολή Εταιριών** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.1(Εταιρίες).

b. Επιστροφή

Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.1(Εταιρίες).

5.3.2 – Πελάτες/Προμηθευτές

Εδώ μπορείτε να προβάλετε, ή/και να διαγράψετε τους ήδη συνεργαζόμενους Πελάτες/Προμηθευτές, ή να προσθέσετε έναν νέο.

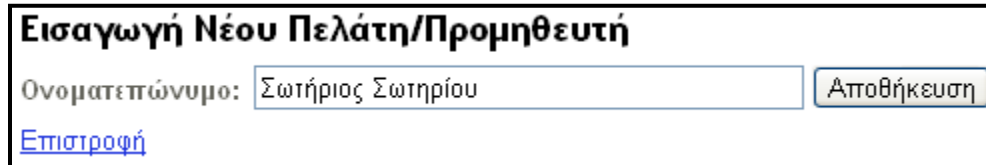


Εικόνα 5.3.6. Πελάτες/Προμηθευτές

i. Εισαγωγή Νέου Πελάτη/Προμηθευτή

Τα στοιχεία που μπορείτε να δώσετε σε ένα νέο Προμηθευτή ή Πελάτη είναι:

- ✚ Ονοματεπώνυμο



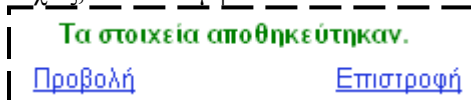
Εικόνα 5.3.7. Εισαγωγή Νέου Πελάτη/Προμηθευτή

Μετά την συμπλήρωση των στοιχείων, επιλέγετε το πλήκτρο **Αποθήκευση**, εφόσον και το επιθυμείτε, αλλιώς επιλέγετε το πλήκτρο **Επιστροφή**, για να επιστρέψετε στην προηγούμενη κατάσταση.

Μετά την επιλογή του πλήκτρου **Αποθήκευση**, θα σας εμφανιστεί ένα μήνυμα αναφοράς.

Οι περιπτώσεις είναι 2:

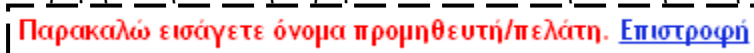
- a. Αν η αποθήκευση γίνει επιτυχώς, τότε θα εμφανιστεί το ακόλουθο μήνυμα :



Αν επιλέξετε **Προβολή** τότε το πρόγραμμα θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.8(Προβολή Προμηθευτών/Πελατών).

Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.6(Πελάτες/Προμηθευτές).

- b. Αν το πεδίο του Προμηθευτή/Πελάτη παραμείνει κενό κατά την αποθήκευση τότε:

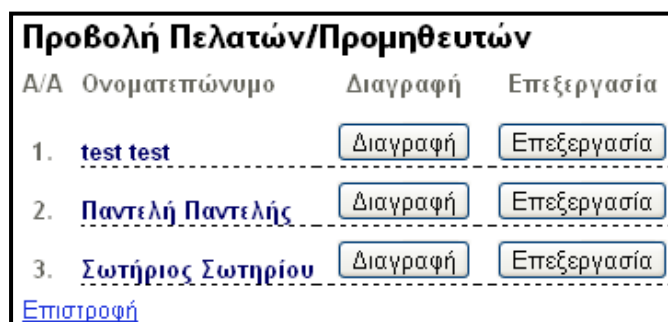


Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.7(Εισαγωγή Νέου Προμηθευτή/Πελάτη).

ii. Προβολή Πελατών/Προμηθευτών

Εδώ μπορείτε να δείτε όλα τα πρόσωπα με τα οποία συνεργάζεστε. Μπορείτε επίσης να διαγράψετε κάποιο από τα άτομα με τα όποια η συνεργασία σας έχει λάβει τέλος.

Τα πλήκτρα που συναντάτε είναι τρία, **Διαγραφή**, **Επεξεργασία** και **Επιστροφή**.

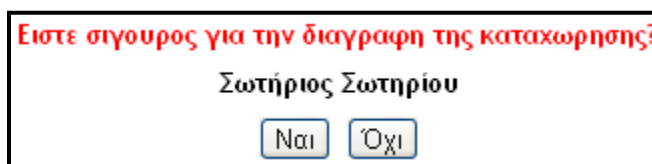


A/A	Ονοματεπώνυμο	Διαγραφή	Επεξεργασία
1.	test test	Διαγραφή	Επεξεργασία
2.	Παντελή Παντελής	Διαγραφή	Επεξεργασία
3.	Σωτήριος Σωτηρίου	Διαγραφή	Επεξεργασία

Εικόνα 5.3.8. Προβολή Πελατών/Προμηθευτών

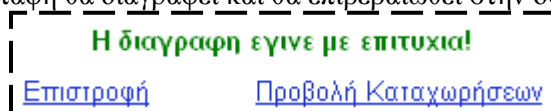
a. Διαγραφή

Με το πάτημα του κουμπιού θα σας ζητηθεί να επιβεβαιώσετε το αίτημα σας πριν την πραγματοποίησή του, δηλαδή να διαγράψετε τον πελάτη ή προμηθευτή που επιλέξατε

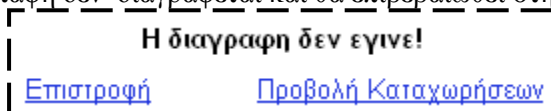


Εικόνα 5.3.9. Διαγραφή Πελάτη/Προμηθευτή

Αν επιλέξετε **ναι** τότε η επαφή θα διαγραφεί και θα επιβεβαιωθεί στην οθόνη σας με την ένδειξη:



Αν επιλέξετε **όχι** τότε η επαφή δεν διαγράφεται και θα επιβεβαιωθεί στην οθόνη σας με την ένδειξη:

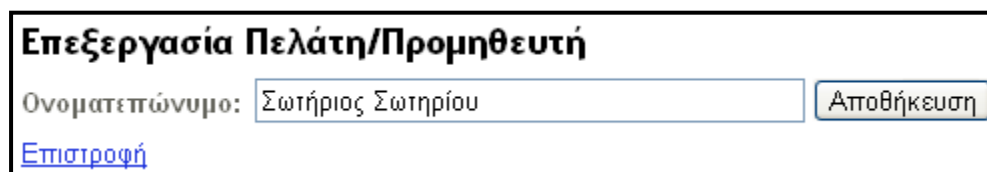


Αν επιλέξετε **Προβολή Καταχωρήσεων** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.8(Προβολή Πελατών/Προμηθευτών).

Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.6(Πελάτες/Προμηθευτές).

b. Επεξεργασία

Με το πάτημα του κουμπιού **Επεξεργασία** θα εμφανιστεί μια παρόμοια πλατφόρμα σαν αυτή της εισαγωγής Πελάτη/Προμηθευτή, μόνο που το πεδίο αυτή την φορά είναι συμπληρωμένο από τον Πελάτη/Προμηθευτή που επιλέξατε.



Εικόνα 5.3.10. Επεξεργασία Πελάτη/Προμηθευτή

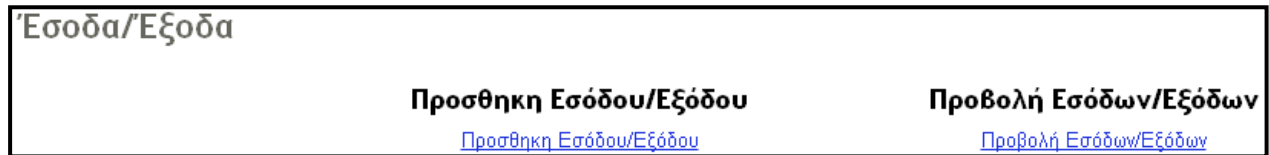
*Η αποθήκευση και τα υπόλοιπα κουμπιά λειτουργούν ακριβώς όπως στην εισαγωγή νέου Πελάτη/Προμηθευτή. Ανατρέξτε στο αντίστοιχο κεφάλαιο.

c. Επιστροφή

Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.6(Πελάτες/Προμηθευτές).

5.3.3 – Έσοδα/Εξοδα

Εδώ μπορείτε να προβάλετε τα είδη εσόδων/εξόδων, ή να προσθέσετε ένα νέο είδος.



Εικόνα 5.3.11. Έσοδα/Εξοδα

i. Εισαγωγή Εσόδου/Εξόδου

Τα στοιχεία που μπορείτε να δώσετε σε ένα είδος Εσόδου/Εξόδου είναι:

- ✚ Ονοματεπώνυμο

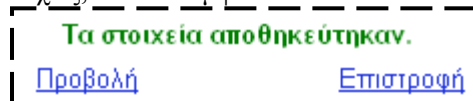


Εικόνα 5.3.12. Εισαγωγή Εσόδου/Εξόδου

Μετά την συμπλήρωση των στοιχείων, επιλέγετε το πλήκτρο **Αποθήκευση**, εφόσον και το επιθυμείτε, αλλιώς επιλέγετε το πλήκτρο **Επιστροφή**, για να επιστρέψετε στην προηγούμενη κατάσταση.

Μετά την επιλογή του πλήκτρου **Αποθήκευση**, θα σας εμφανιστεί ένα μήνυμα αναφοράς. Οι περιπτώσεις είναι 2:

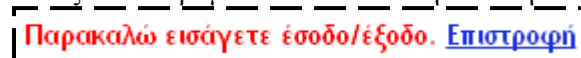
- a. Αν η αποθήκευση γίνει επιτυχώς, τότε θα εμφανιστεί το ακόλουθο μήνυμα :



Αν επιλέξετε **Προβολή** τότε το πρόγραμμα θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.13(Προβολή Εσόδων/Εξόδων).

Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.11(Εσοδα/Εξοδα).

- b. Αν το πεδίο των Εσόδων/Εξόδων παραμείνει κενό κατά την αποθήκευση τότε:



Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.12(Εισαγωγή Εσόδου/Εξόδου).

ii. Προβολή Εσόδων/Εξόδων

Εδώ μπορείτε να δείτε όλα τα είδη εσόδων ή εξόδων που χρησιμοποιείτε. Μπορείτε επίσης να διαγράψετε κάποιο από τα είδη που δεν χρησιμοποιούνται πλέον.

Προβολή Εσόδων/Εξόδων		
A/A	Έσοδο/Έξοδο	Διαγραφή
1.	CYTA	<input type="button" value="Διαγραφή"/>
2.	dei	<input type="button" value="Διαγραφή"/>
3.	Φ.Π.Α	<input type="button" value="Διαγραφή"/>
4.	ΦΠΑ	<input type="button" value="Διαγραφή"/>
Επιστροφή		

Εικόνα 5.3.13. Προβολή Εσόδων/Εξόδων

Τα πλήκτρα που συναντάτε είναι δυο, **Διαγραφή** και **Επιστροφή**.

a. Διαγραφή

Με το πάτημα του κουμπιού θα σας ζητηθεί να επιβεβαιώσετε το αίτημα σας πριν την πραγματοποίησή του, δηλαδή να διαγράψετε έσοδο ή έξοδο που επιλέξατε.

Είστε σίγουρος για την διαγραφή της καταχώρησης;

CYTA

Εικόνα 5.3.14. Διαγραφή Εσόδου/Εξόδου

Αν επιλέξετε **ναι** τότε το έσοδο/έξοδο θα διαγραφεί και θα επιβεβαιωθεί στην οθόνη σας με την ένδειξη:

Η διαγραφή δεν έγινε!

[Επιστροφή](#) [Προβολή Καταχωρήσεων](#)

Αν επιλέξετε **Προβολή Καταχωρήσεων** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.13(Προβολή Εσόδων/Εξόδων).

Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.11(Εσοδα/Εξοδα).

Αν επιλέξετε **όχι** τότε το έσοδο/έξοδο δεν διαγράφεται και επιβεβαιώνεται στην οθόνη με την ένδειξη:

Η διαγραφή έγινε με επιτυχία!

[Επιστροφή](#) [Προβολή Καταχωρήσεων](#)

Αν επιλέξετε **Προβολή Καταχωρήσεων** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.13(Προβολή Εσόδων/Εξόδων).

Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.11(Εσοδα/Εξοδα).

b. Επιστροφή

Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.3.11(Εσοδα/Εξοδα).

5.4 – Ημερολόγιο

Η σελίδα αυτή χρησιμοποιείται σαν ηλεκτρονική ατζέντα,. Δηλαδή υπενθυμίζει τα γενέθλια των φίλων ή συνεργατών σας, και κρατάει ιστορικό των συναντήσεων ή άλλων ενεργειών που γίνανε ή που πρόκειται να γίνουν.

5.4.1 – Γενέθλια

Σε αυτό το σκέλος της σελίδας, τα **Γενέθλια**, δεν υπάρχει καμία πρόσβαση ώστε να μπορεί ο χρήστης να κάνει κάποια αλλαγή. Δουλεύει μόνο για προβολή γενεθλίων όσων επαφών έχουν εισαχθεί στην βάση. Το **πρωτόρημα** της συγκεκριμένης υπηρεσίας, είναι ότι δεν προβάλλει τα γενέθλια όχι μόνο της τρέχουσας ημέρας, αλλά και των επόμενων πέντε. Αυτό συμβαίνει για χρηστικούς λόγους , για την περίπτωση δηλαδή που ο χρήστης δεν μπορείει κάποια ημέρα να χρησιμοποιήσει την εφαρμογή για προσωπικούς του λόγους.

Εορτάζοντες
Μαρία Μαρκοπούλου today
Παρασκευάς Γεωργίου today
Peter Brown Πέμπτη
Παντελάκης Παντελή Παρασκευή

Εικόνα 5.4.1. Γενέθλια

Στην παραπάνω εικόνα διακρίνονται τα πρόσωπα και η ημέρα που έχουν γενέθλια. Δηλαδή :

- Με **μπλε** χρώμα, το **Όνομα** και **Επίθετο** της επαφής
- Με **γκρι** χρώμα, η ημέρα που έχει **γενέθλια** η επαφή

5.4.2 – Ενέργειες

Εδώ μπορείτε να προβάλετε, να αλλάξετε ή και να διαγράψετε τα στοιχεία των υπάρχουσών ενεργειών , ή να προσθέσετε μια νέα.

Ενέργεια	Ημ. Έναρξης	Ημ. Λήξης	Σημειώσεις		
Παρουσίαση πτυχιακής	2010-07-17	2011-07-17	Έλεγχος ημερομηνιών	<input type="button" value="Διαγραφή"/>	<input type="button" value="Επεξεργασία"/>

Εικόνα 5.4.2. Ενέργειες

Οι ενέργειες οι οποίες έχουν δραστηριότητα την τρέχουσα ημερομηνία προβάλλονται συνεχώς.

a. Προσθήκη Νέας Ενέργειας

Τα στοιχεία που μπορείτε να δώσετε σε μια ενέργεια είναι:

- ✚ Ονομασία Ενέργειας
- ✚ Ημερομηνία Έναρξης
- ✚ Ημερομηνία Λήξης
- ✚ Σημειώσεις

Για να ξεκινήσετε μια νέα εισαγωγή ενέργειας δεν έχετε παρά να πατήσετε το πλήκτρο **Προσθήκη Νέας Ενέργειας**.

Στην συνέχεια συμπληρώνετε τα στοιχεία της κάθε νέας επαφής που θέλετε να προσθέσετε στον τηλεφωνικό σας κατάλογο.

Εισαγωγή Νέας Ενέργειας					
Ενέργεια	Ημ. Εναρξης	Ημ. Λήξης	Σημειώσεις		
Παράδοση Site	2011-01-04	2011-04-04	Εύρεση JQuery basket	Αποθήκευση	Ακύρωση

April 2011

Mo	Tu	We	Th	Fr	Sa	Su
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

Εικόνα 5.4.3. Προσθήκη Νέας Ενέργειας

Προσοχή! Το πεδίο της ημερομηνίας χρειάζεται να συμπληρωθεί με τη σωστό τρόπο. Για να αποθηκευτεί σωστά η καταχώρηση πρέπει να δώσετε τα στοιχεία με την σειρά. Δηλαδή:

Έτος-Μήνας-Ημέρα(χρειάζονται και οι παύλες), για παράδειγμα αν κάποιος ενέργειας, η προθεσμία λήγει την 1^η Ιανουαρίου του 2000, θα γραφόταν : 2000-01-01. Για την πιο εύκολη χρήση, υπάρχει ο πίνακας με το ημερολόγιο, έτσι, απλά επιλέγετε την ημερομηνία που ζητάτε με ένα κλικ, και η ημερομηνία αποθηκεύεται αυτόματα με τον σωστό τρόπο.

Μετά την συμπλήρωση των στοιχείων, επιλέγετε το πλήκτρο **Αποθήκευση**, εφόσον και το επιθυμείτε, αλλιώς επιλέγετε το πλήκτρο **Ακύρωση**, για να επιστρέψετε στην προηγούμενη κατάσταση.

Μετά την επιλογή του πλήκτρου **Αποθήκευση**, θα σας εμφανιστεί ένα μήνυμα αναφοράς. Οι περιπτώσεις είναι 3:

- a. Αν η αποθήκευση γίνει επιτυχώς, τότε θα εμφανιστεί το ακόλουθο μήνυμα :

Τα στοιχεία αποθηκεύτηκαν με επιτυχία.

- b. Αν κάποιο από τα πεδία παραμείνουν κενά κατά την αποθήκευση τότε:

Παρουσιάστηκε σφάλμα στα στοιχεία σας. Επιστρέψτε στην προηγούμενη σελίδα για να διορθώσετε τα στοιχεία. Πρέπει να συμπληρώσετε όλα τα πεδία που έχουν αστερίσκο.

- c. Αν κάποια από τις ημερομηνίες είναι συντακτικά λάθος γραμμένες :

Πληκτρολογήσατε ημερομηνία που δεν ισχύει. Επιστρέψτε στην προηγούμενη σελίδα για να διορθώσετε τα στοιχεία.

b. Προβολή Όλων

Όπως αναγράφεται, το κουμπί χρησιμοποιείται για την εμφάνιση όλων των αποθηκευμένων ενεργειών. Των ενεργειών που δεν φαίνονται από την αρχή, επειδή η προθεσμία τους έχουν λήξει.

Προβολή Ενεργειών					
Ενέργεια	Ημ. Εναρξης	Ημ. Λήξης	Σημειώσεις		
Παρουσίαση πτυχιακής	2010-07-17	2011-07-17	Έλεγχος ημερομηνιών	Διαγραφή	Επεξεργασία
Παράδοση Site	2011-01-04	2011-04-04	Εύρεση JQuery basket	Διαγραφή	Επεξεργασία
Test	2011-05-24	2011-05-28	test	Διαγραφή	Επεξεργασία

[Επιστροφή](#)

Εικόνα 5.4.4. Προβολή Ενεργειών

Ακολουθούν 3 νέα κουμπιά:

i. Διαγραφή

Με το πάτημα του κουμπιού θα σας ζητηθεί να επιβεβαιώσετε το αίτημα σας πριν την πραγματοποίησή του, δηλαδή να διαγράψετε την ενέργεια που επιλέξατε.

Είστε σίγουρος για την διαγραφή της καταχώρησης;			
Ενέργεια	Ημ. Έναρξης	Ημ. Λήξης	Σημειώσεις
Παρουσίαση πτυχιακής	2010-07-17	2011-07-17	Έλεγχος ημερομηνιών
<input type="button" value="Ναι"/> <input type="button" value="Όχι"/>			

Εικόνα 5.4.5. Διαγραφή Ενέργειας

Αν επιλέξετε **ναι** τότε η ενέργεια θα διαγραφεί και θα επιβεβαιωθεί στην οθόνη σας με την ένδειξη:

Η διαγραφή έγινε με επιτυχία!

Αν επιλέξετε **όχι** τότε η ενέργεια δεν διαγράφεται και θα επιβεβαιωθεί στην οθόνη σας με την ένδειξη:

Η διαγραφή δεν έγινε!

ii. Επεξεργασία

Με το πάτημα του κουμπιού, επιλέγουμε κάποια ενέργεια προς τροποποίηση κάποιων στοιχείων της, που ίσως γράφτηκαν λάθος κατά την αποθήκευσή τους. Κάθε στοιχείο της ενέργειας μπορεί να γίνει επεξεργάσιμο και συνεπάγεται έτοιμο για αλλαγή.

Επεξεργασία Ενέργειας					
Ενέργεια	Ημ. Έναρξης	Ημ. Λήξης	Σημειώσεις		
Παρουσίαση πτυχιακής	2010-07-17	2011-07-17	Έλεγχος ημερομηνίας Λήξης	<input type="button" value="Αποθήκευση"/>	<input type="button" value="Ακυρωση"/>

Εικόνα 5.4.6. Επεξεργασία Ενέργειας

*Στο παράδειγμα βλέπετε πως οι σημειώσεις έχουν αλλάξει.

*Η αποθήκευση και τα υπόλοιπα κουμπιά λειτουργούν ακριβώς όπως στην Προσθήκη Νέας Ενέργειας. Ανατρέξτε στο αντίστοιχο κεφάλαιο.

iii. Επιστροφή

Αν επιλέξετε **Επιστροφή** τότε το πρόγραμμα σας θα ανατρέξει στην κατάσταση που δείχνει η Εικόνα 5.4.2(Ενέργειες).

5.5 – Τηλεφωνικός Κατάλογος

Στην σελίδα αυτή, μπορείτε να εισάγετε μια νέα επαφή, με επιπλέον στοιχεία εκτός ονόματος, επίθετου και αριθμού τηλεφώνου ή να κάνετε αναζήτηση του ατόμου που χρειάζεστε.

5.5.1 – Εισαγωγή νέας Επαφής

Τα στοιχεία που μπορείτε να δώσετε στο τηλεφωνικό κατάλογο για μια επαφή είναι:

- ✚ Επίθετο
- ✚ Όνομα
- ✚ Ημερομηνία Γέννησης
- ✚ E-mail
- ✚ Τηλέφωνο
- ✚ Σχόλια

Για να ξεκινήσετε μια νέα εγγραφή τηλεφώνου δεν έχετε παρά να πατήσετε το πλήκτρο **Προσθήκη**.

Όνομα	Επιθετο	Ημερομηνία Γεννησης	E-mail	Τηλεφωνο	Σχολια
<input type="button" value="Προσθήκη"/>					

Εικόνα 5.5.1. Φόρμα Προσθήκης Νέας Επαφής

Στην συνέχεια συμπληρώνετε τα στοιχεία της κάθε νέας επαφής που θέλετε να προσθέσετε στον τηλεφωνικό σας κατάλογο.

Όνομα	Επιθετο	Ημερομηνία Γεννησης	E-mail	Τηλεφωνο	Σχολια		
Παντελάκης *	Παντελή *	1984-10-24 <input type="text"/>	user1@app.teicrete.gr *	6991234567 *	Ξάδερφος	<input type="button" value="Αποθήκευση"/>	<input type="button" value="Ακύρωση"/>

Εικόνα 5.5.2. Εισαγωγή Νέας Επαφής

Προσοχή! Το πεδίο της ημερομηνίας χρειάζεται να συμπληρωθεί με τη σωστό τρόπο. Για να αποθηκευτεί σωστά η καταχώρηση πρέπει να δώσετε τα στοιχεία με την σειρά. Δηλαδή: Έτος-Μήνας-Ημέρα(χρειάζονται και οι παύλες), για παράδειγμα αν κάποιος γεννήθηκε την 1^η Ιανουαρίου του 2000, θα γραφόταν : 2000-01-01. Για την πιο εύκολη χρήση, υπάρχει ο πίνακας με το ημερολόγιο. Έτσι, απλά επιλέγετε την ημερομηνία που ζητάτε με ένα κλικ, και η ημερομηνία αποθηκεύεται αυτόματα με τον σωστό τρόπο.

Μετά την συμπλήρωση των στοιχείων, επιλέγετε το πλήκτρο **Αποθήκευση**, εφόσον και το επιθυμείτε, αλλιώς επιλέγετε το πλήκτρο **Ακύρωση**, για να επιστρέψετε στην προηγούμενη κατάσταση.

Μετά την επιλογή του πλήκτρου **Αποθήκευση**, θα σας εμφανιστεί ένα μήνυμα αναφοράς.

Οι περιπτώσεις είναι 3:

- a. Αν η αποθήκευση γίνει επιτυχώς, τότε θα εμφανιστεί το ακόλουθο μήνυμα :

| Η καταχώρηση ολοκληρώθηκε! |

- b. Αν κάποιο από τα υποχρεωτικά πεδία με αστερίσκο παραμείνουν κενά κατά την αποθήκευση τότε:

| Τα στοιχεία που δώσατε είναι λάθος! Πρέπει να συμπληρώσετε όλα τα πεδία με αστερίσκο. |

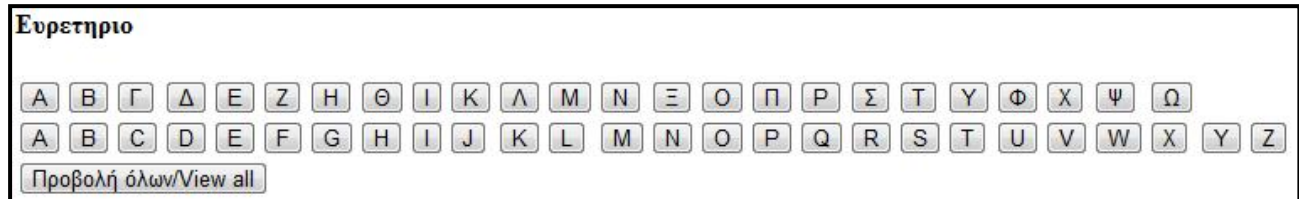
- c. Αν η ημερομηνία γέννησης είναι συντακτικά λάθος γραμμένη :

| Καταχωρήσατε λάθος ημερομηνία. Επιστρέψτε στην προηγούμενη σελίδα για να τη διορθώσετε! |

5.5.2 – Ευρετήριο

Το ευρετήριο είναι μια χρήσιμη εφαρμογή, που διευκολύνει τον χρήστη να δει τις επαφές του βάσει του αρχικού ονόματος τους. Εκτός από την απλή προβολή στοιχείων, στο συγκεκριμένο ευρετήριο έχετε το πλεονέκτημα της επεξεργασίας, της διαγραφής των επαφών αλλά και της εμφάνισης όλων μαζί με το πάτημα ενός κουμπιού.

Πως δουλεύει:



Εικόνα 5.5.3. Ευρετήριο

Με το πάτημα ενός από τα πλήκτρα, εμφανίζουμε όλα τα επίθετα που ξεκινάνε με το γράμμα που περιέχει το πλήκτρο. Αν δεν υπάρχει κάποιο καταχωρημένο επίθετο από το επιλεγμένο γράμμα τότε εμφανίζεται το ακόλουθο μήνυμα :



Εικόνα 5.5.4. Μήνυμα Ευρετηρίου Χωρίς Αποτελέσματα

Αλλιώς, εμφανίζει όλες τις επαφές που το επίθετο τους ξεκινάει από το επιλεγμένο γράμμα, είτε αγγλικό είτε ελληνικό. Για παράδειγμα, όταν πατήσουμε το πλήκτρο με το αγγλικό γράμμα **B**, μας εμφανίζει τα παρακάτω :

Αποτελεσματα Ευρετηρίου:						
Όνομα	Επίθετο	Ημερ. Γεννησης	E-mail	Τηλεφωνο	Σχολια	
Peter	Brown	1986-11-23	ionos@msn.com	2810261361	Computer Service	<input type="button" value="Διαγραφή"/> <input type="button" value="Επεξεργασία"/>

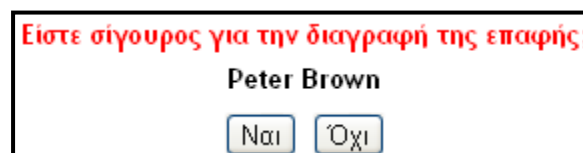
Εικόνα 5.5.5. Μήνυμα Ευρετηρίου Με Αποτελέσματα

Έτσι γίνεται για κάθε πλήκτρο εκτός του τελευταίου που γράφει **Προβολή όλων/View all**, το οποίο και εμφανίζει όλες τις επαφές που είναι αποθηκευμένη στην βάση.

Κατά την προβολή, εμφανίζονται δυο νέα πλήκτρα **Διαγραφή** και το **Επεξεργασία**. Κάθε επαφή που εμφανίζεται έχει τα δικά της πλήκτρα **Διαγραφή** και **Επεξεργασία** που αφορούν μόνο την ίδια και καμία άλλη επαφή.

a. Διαγραφή

Με το πάτημα του κουμπιού θα σας ζητηθεί να επιβεβαιώσετε το αίτημα σας πριν την πραγματοποίηση του, δηλαδή να διαγράψετε την επαφή που επιλέξατε



Εικόνα 5.5.6. Διαγραφή Επαφής

Αν επιλέξετε **ναι** τότε η επαφή θα διαγραφεί και θα επιβεβαιωθεί στην οθόνη σας με την ένδειξη:

Η διαγραφή έγινε με επιτυχία!

Αν επιλέξετε **όχι** τότε η επαφή δεν διαγράφεται και θα επιβεβαιωθεί στην οθόνη σας με την ένδειξη:

Η διαγραφή δεν έγινε!

b. Επεξεργασία

Με το πάτημα του κουμπιού, επιλέγουμε κάποια επαφή προς τροποποίηση κάποιων στοιχείων της, που ίσως γράφτηκαν λάθος κατά την αποθήκευσή τους. Κάθε στοιχείο της επαφής μπορεί να γίνει επεξεργασίμο και συνεπάγεται έτοιμο για αλλαγή.

Επεξεργασία						
Όνομα	Επίθετο	Ημερομηνία Γεννησης	E-mail	Τηλεφωνο	Σχολια	
Peter	Brown	1986-11-23	ionos@gmail.com*	2810261361	Computer Service	Αποθήκευση Ακύρωση

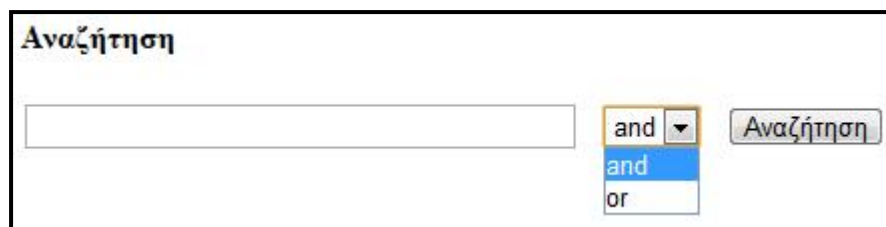
Εικόνα 5.5.7. Επεξεργασία Επαφής

*Στο παράδειγμα βλέπετε πως το e-mail έχει αλλάξει.

*Η αποθήκευση και τα υπόλοιπα κουμπιά λειτουργούν ακριβώς όπως στην Εισαγωγή Νέου/ας Τηλεφώνου/Επαφής. Ανατρέξτε στο αντίστοιχο κεφάλαιο.

5.5.3 – Αναζήτηση

Η αναζήτηση είναι σαν το ευρετήριο, δηλαδή χρησιμοποιείται για την εύρεση επαφών. Η διαφορά μεταξύ τους είναι πως στην αναζήτηση μπορείτε να εμφανίσετε μόνο το συγκεκριμένο επίθετο που ψάχνετε ή ακόμα και το όνομα.



Εικόνα 5.5.8. Αναζήτηση Επαφής

Υπάρχουν τρία πεδία,

- Το κουτί που γράφουμε το κείμενο που θέλουμε να αναζητήσουμε
- Το πεδίο and/or.
 - Το **and** βρίσκει αποτελέσματα τα οποία να έχουν όνομα και επίθετο τα οποία αρχίζουν με την λέξη αναζήτησης
 - Το **or** βρίσκει αποτελέσματα τα οποία να έχουν το όνομα ή το επίθετο το οποίο αρχίζει με την λέξη αναζήτησης
- Το κουμπί **Αναζήτηση**, όπου εμφανίζει τα αποτελέσματα.

Παραδείγματα:

→ Παράδειγμα 1

Αν θέλουμε να δούμε τα επίθετα που ξεκινάνε από **K** και αφήσουμε σαν επιλογή το **or** τότε:

Αναζήτηση

Αποτελέσματα Αναζήτησης:

Όνομα	Επίθετο	Ημερ. Γεννησης	E-mail	Τηλεφωνο	Σχολια		
Κώστας	Παντελή	1984-10-24	pantelispyla@gmail.com	6947775538	hhgd	<input type="button" value="Διαγραφή"/>	<input type="button" value="Επεξεργασία"/>
Κώστας	Κουτσόπουλος	2008-08-05	alcainta@alg.bomb	23498299	sto tei	<input type="button" value="Διαγραφή"/>	<input type="button" value="Επεξεργασία"/>
Νικόλαος	Καραγιάννης	1986-03-13	Kar_nick@in.gr	2610674598	Τεχνικός Πληροφορικής	<input type="button" value="Διαγραφή"/>	<input type="button" value="Επεξεργασία"/>
Κωνσταντίνα	Αλεξανδρή	1990-08-29	ntinaki004@hotmail.com	32442323432	dsad	<input type="button" value="Διαγραφή"/>	<input type="button" value="Επεξεργασία"/>

Εικόνα 5.5.9. Αποτελέσματα Αναζήτησης με Χρήση OR

Όπως βλέπετε, όταν το **or** είναι επιλεγμένο το αποτελέσματα δείχνουν μόνο τις επαφές που έχουν επίθετο ή/και όνομα, να αρχίζουν με την ζητούμενη λέξη, στην περίπτωση αυτή το Κ.

→ Παράδειγμα 2

Αν θέλουμε να δούμε τα επίθετα που ξεκινάνε από **Κ** και αφήσουμε σαν επιλογή το **and** τότε:

Αναζήτηση

Αποτελέσματα Αναζήτησης:

Όνομα	Επίθετο	Ημερ. Γεννησης	E-mail	Τηλεφωνο	Σχολια		
Κώστας	Κουτσόπουλος	2008-08-05	alcainta@alg.bomb	23498299	sto tei	<input type="button" value="Διαγραφή"/>	<input type="button" value="Επεξεργασία"/>

Εικόνα 5.5.10. Αποτελέσματα Αναζήτησης με Χρήση AND

Όπως βλέπετε, όταν το **and** είναι επιλεγμένο το αποτελέσματα δείχνουν μόνο τις επαφές που έχουν επίθετο αλλά και όνομα ΜΑΖΙ, να αρχίζουν με την ζητούμενη λέξη, στην περίπτωση αυτή το Κ.

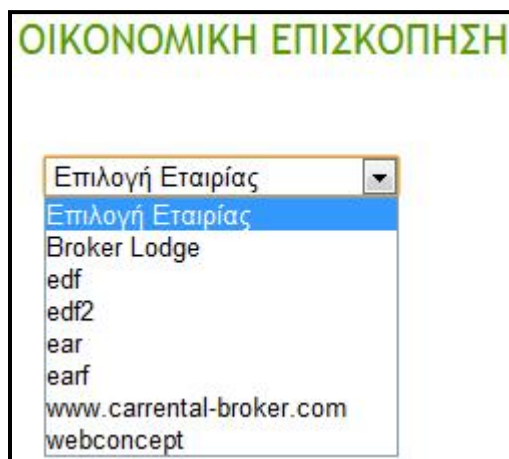
5.6 – Οικονομική Επισκόπηση

Στην σελίδα αυτή, μπορείτε να διαχειριστείτε όλες τις οικονομικές κινήσεις της επιχείρησής σας και όχι μόνο. Εδώ μπορείτε να δείτε τα έσοδα, τα έξοδα που λαμβάνετε ή δίνετε καθώς και το αν είναι επικερδής η συνεργασία με την κάθε επιχείρηση στο χρονικό διάστημα που εσείς επιλέξετε. Μπορείτε να κοιτάξετε το ιστορικό της οικονομικής διακίνησης με αναζήτηση και με άλλων μεθόδων εκτός από τις ημερομηνίες.

Εκτός αυτών το σύστημα σας υπενθυμίζει πότε πρέπει να δώσετε ή να λάβετε μια δόση ανάλογα με το επαναλαμβανόμενο διάστημα που εσείς θα δώσετε.

5.6.1 – Επιλογή Εταιρίας

Για να αρχίσουν όλα πρέπει πρώτα να διαλέξετε την εταιρία για την οποία θέλετε να εμφανίσετε, αλλάξτε, διαγράψτε τις πληροφορίες της. Για όλες τις εταιρίες οι δραστηριότητες που μπορείτε να χρησιμοποιήσετε είναι ίδιες. Οι εταιρίες είναι αυτές που έχετε εσείς εισάγει από την σελίδα **Εταιρίες**(σελ. ddd).



Εικόνα 5.6.1.Επιλογή Εταιρίας

Κατά την επιλογή μιας εταιρίας φορτώνονται 4 φόρμες οι οποίες θα μελετηθούν λεπτομερώς στις παρακάτω σελίδες. Αυτές είναι :

- ✚ Φόρμα Αναζήτησης
- ✚ Έσοδα
- ✚ Έξοδα
- ✚ Συνολικά

5.6.2 – Φόρμα Αναζήτησης

Για αρχή, σωστό θα ήταν να τονιστεί, ότι κατά την επιλογή εταιρίας εμφανίζονται αποτελέσματα που αφορούν όλο το τρέχον έτος, όπως και φαίνεται στις ημερομηνίες έναρξης και λήξης. Από εκεί και πέρα υπάρχουν 5 τρόποι αναζήτησης της οποίας τα αποτελέσματα εμφανίζονται μετά το πλήκτρο **Εμφάνιση**.

Εικόνα 5.6.2.Φόρμα Αναζήτησης

α) Κατά Ημερομηνία*

Ανάλογα τις ημερομηνίες που θα θέσετε το σύστημα θα εμφανίσει αποτελέσματα που αφορούν αυτές τις ημερομηνίες. Οι περιπτώσεις που υπάρχουν ώστε να εμφανιστεί κάποιο αποτέλεσμα είναι 4:

- Η ημερομηνία έναρξης της καταχώρησης να είναι πριν από την ημερομηνία έναρξης της αναζήτησης και η ημερομηνία λήξης της καταχώρησης να είναι πριν από την ημερομηνία λήξης και μετά από την ημερομηνία έναρξης της αναζήτησης.
- Η ημερομηνία έναρξης της καταχώρησης να είναι μετά από την ημερομηνία έναρξης και πριν από την ημερομηνία λήξης της αναζήτησης και η ημερομηνία λήξης της καταχώρησης να είναι μετά από την ημερομηνία λήξης.
- Η ημερομηνία έναρξης της καταχώρησης να είναι πριν από την ημερομηνία έναρξης της αναζήτησης και η ημερομηνία λήξης της καταχώρησης να είναι μετά από την ημερομηνία λήξης της αναζήτησης.

- Η ημερομηνία έναρξης της καταχώρησης να είναι μετά από την ημερομηνία έναρξης της αναζήτησης και η ημερομηνία λήξης της καταχώρησης να είναι πριν από την ημερομηνία λήξης.

b) Κατά Είδος Εσόδου/Εξόδου*

Ανάλογα το είδος που θα επιλέξετε το σύστημα θα εμφανίσει αποτελέσματα που έχουν σαν είδος εσόδου και εξόδου, αυτό που επιλέξατε.

c) Κατά Προμηθευτή/Πελάτη*

Ανάλογα τον προμηθευτή/πελάτη που θα επιλέξετε το σύστημα θα εμφανίσει αποτελέσματα που έχουν σαν προμηθευτή/πελάτη, αυτόν που επιλέξατε.

d) Κατά Κατάσταση Πληρωμής*

Ανάλογα την κατάσταση που θα επιλέξετε το σύστημα θα εμφανίσει αποτελέσματα όμοια με μια από τις τρεις επιλογές που υπάρχουν:

- Πληρωμένα
- Απλήρωτα
- Όλα

e) Εμφάνιση όλων των καταχωρήσεων της εταιρίας*

Αν επιλέξετε το *checkbox* δίπλα από το “Χωρίς Ημερομηνίες” και πατήστε **Εμφάνιση**, τότε θα δείτε όλες τις καταχωρήσεις της επιλεγμένης εταιρίας που υπάρχουν μέσα στην βάση ανεξαρτήτως ημερομηνιών.

*Εάν δεν υπάρχουν καταχωρήσεις που καλύπτουν την αναζήτηση τότε δεν επιστρέφουν αποτελέσματα

Εταιρία	Είδος	Ποσό	Ημ. Έναρξης	Ημ. Λήξης	Προμηθ./Πελάτης	Σημείωση	Αναδρ/τα	Κάθε Πότε	Συνολικά
Δεν βρέθηκαν αποτελέσματα.									
<input type="button" value="Εισαγωγή Νέου"/>									

Εικόνα 5.6.3. Άδεια Φόρμα Εσόδου ή Εξόδου

5.6.3 – Έσοδα

Εμφανίζονται όλα τα έσοδα της προεπιλεγμένης εταιρίας που πληρούν όλες τις συνθήκες της αναζήτησης. Για κάθε έσοδο παρουσιάζονται οι πιο χρήσιμες από τις πληροφορίες που αποτελεί το καθένα.

Αυτές είναι:

- ✚ **Εταιρία**
-Έσοδο για την συγκεκριμένη εταιρία
- ✚ **Είδος**
-Είδος εσόδου
- ✚ **Ποσό**
-Ποσό που χρειάζεται να πληρωθεί
- ✚ **Ημερομηνία Έναρξης**
-Ημερ/νία που ξεκινάει η συναλλαγή
- ✚ **Ημερομηνία Λήξης**
-Ημερ/νία που λήγει η συναλλαγή
- ✚ **Προμηθευτής/Πελάτης**
-Όνομα ή επωνυμία συνεργατών
- ✚ **Σημείωση**
-Σχόλια που δίνουν βοηθητικές πληροφορίες
- ✚ **Αναδρομικότητα**
-Δείχνει αν το ποσό είναι αναδρομικό ή όχι
- ✚ **Κάθε Πότε**
-Δείχνει την συχνότητα που πρέπει να καταβάλλεται το αναδρομικό ποσό
- ✚ **Συνολικά**
-Δείχνει το ποσό αναδρομικό ή μη, μέχρι το τέλος της ημέρας που λήγουν οι συναλλαγές

	Εταιρία	Είδος	Ποσό	Ημ. Έναρξης	Ημ. Λήξης	Προμηθ./Πελάτης	Σημείωση	Αναδρ/τα	Κάθε Πότε	Συνολικά
<input type="checkbox"/>	TMP	Φ.Π.Α	100.23	2011-06-01	2011-06-02	test test	ekssoftisi	<input type="checkbox"/>		100.23
<input type="checkbox"/>	TMP	dei	15.00	2011-05-26	2011-05-31	Παντελή Παντελής	ekssoftisi	<input checked="" type="checkbox"/>	1 ημέρα	90.00
<input type="checkbox"/>	TMP	dei	200.00	2010-11-01	2011-11-01	Σωτήριος Σωτηρίου	dfsdf	<input checked="" type="checkbox"/>	1 μήνα	2000.00
Σύνολο										2190.23

Εικόνα 5.6.4. Έσοδα

Η φόρμα όπου βρίσκονται τα έσοδα, αποτελείται από 4 πλήκτρα, αν υπάρχει έστω και ένα καταχωρημένο έσοδο στην επιλεγμένη εταιρία*(ακόμα και αν δεν εμφανίζεται κανένα λόγω συγκεκριμένης αναζήτησης χωρίς αποτέλεσμα)

Αυτά είναι:

- i. **Εισαγωγή Νέου**
- ii. **Περισσότερα**
- iii. **Διαγραφή**
- iv. **Επεξεργασία**
- v. **Ολοκληρώθηκε**

*Αν δεν υπάρχει έστω μια καταχώρηση στα έσοδα της επιλεγμένης εταιρίας τότε εμφανίζεται μόνο ένα πλήκτρο, αυτό της **Εισαγωγής Νέου**.

a) Εισαγωγή Νέου

Τα στοιχεία που μπορείτε να δώσετε σε ένα νέο έσοδο είναι:

- + Εταιρεία
- + Είδος
- + Ποσό
- + Ημερομηνία Έναρξης
- + Ημερομηνία Λήξης
- + Προμηθευτής/Πελάτης
- + Σημείωση
- + Αναδρομικότητα
 - Κάθε Πότε
- + Ολοκληρώθηκε(επιλέγεται όταν ολοκληρώνεται η πληρωμή του εσόδου πληρώνεται πλήρως)
 - Πότε(ημέρα που ολοκληρώθηκε η πληρωμή)
 - Διαφορά(διαφορά που ίσως υπάρξει στα χρήματα για κάποιο λόγο ,π.χ. καθυστέρηση)

Όσα κελιά είναι ενεργοποιημένα, πρέπει να συμπληρωθούν, με τον σωστό τύπο χαρακτήρα που παίρνει το καθένα ώστε να μπορέσει να γίνει η αποθήκευση τους επιτυχώς. **Κανένα** ενεργοποιημένο κελί δεν μπορεί να παραμείνει κενό. Στο τέλος του βιβλίου υπάρχει ένας βοηθητικός πίνακας που δείχνει τι είδους τιμές μπορεί να πάρει το κάθε στοιχείο.

Εισαγωγή Νέου						
Εταιρία	Είδος	Ποσό	Ημ. Έναρξης	Ημ. Λήξης	Προμηθ./Πελάτης	Σημείωση
TMP	Φ.Π.Α	20	2010-12-14	2011-06-20	Σωτήριος Σωτηρίου	laptop
Αναδρομικότητα	Κάθε Πότε	Ολοκληρώθηκε	Πότε	Διαφορά		
<input checked="" type="checkbox"/>	15 Ημέρες	<input type="checkbox"/>				

Εικόνα 5.6.5. Εισαγωγή Εσόδου

Μετά την συμπλήρωση των στοιχείων, επιλέγετε το πλήκτρο **Αποθήκευση**, για την καταχώρηση του εσόδου, αλλιώς επιλέγετε το πλήκτρο **Ακύρωση**, για να επιστρέψετε στην προηγούμενη κατάσταση.

i. Αποθήκευση

Μετά την επιλογή του πλήκτρου **Αποθήκευση**, θα σας εμφανιστεί ένα μήνυμα αναφοράς. Οι περιπτώσεις είναι 3:

a. Αν η αποθήκευση ολοκληρωθεί επιτυχώς, τότε :

Τα στοιχεία καταχωρήθηκαν με επιτυχία!

b. Αν κάποιο από τα ενεργοποιημένα πεδία παραμείνουν κενά μετά την αποθήκευση, τότε :

Τα στοιχεία που δώσατε είναι λάθος. Επιστρέψτε στην προηγούμενη σελίδα για να τα διορθώσετε.

c. Αν κάποια από τις ημερομηνίες είναι συντακτικά λάθος γραμμένες, τότε :

Τα στοιχεία που δώσατε είναι λάθος, παρακαλούμε επιστρέψτε στην προηγούμενη σελίδα και διορθώστε τα στοιχεία που εισαγάτε.

ii. Ακύρωση

Μετά την επιλογή του πλήκτρου **Ακύρωση**, το πρόγραμμα θα επιστρέψει στην προηγούμενη κατάσταση που δείχνει η Εικόνα 5.6.4(Εσοδα).

b) Περισσότερα

Στην επιλογή **Περισσότερα** ο χρήστης μπορεί να δει περαιτέρω στοιχεία που αφορούν το κάθε έσοδο, τα οποία δεν φαίνονται στην κεντρική σελίδα της **Οικονομικής Επισκόπησης**. Η συγκεκριμένη λειτουργία δουλεύει μεμονωμένα, δηλαδή μόνο για ένα έσοδο την κάθε φορά τσεκάροντας το πλαίσιο δίπλα από το έσοδο που θέλουμε να μελετήσουμε.

Προβολή Εσόδου						
Εταιρία	Είδος	Ποσό	Ημ. Εναρξης	Ημ. Λήξης	Προμηθ./Πελάτης	Σημείωση
TMP	dei	27.90	2011-05-10	2011-06-26	Παντελή Παντελής	φδοφο
Αναδρομικότητα	Κάθε Πότε	Ολοκληρώθηκε	Πότε	Διαφορά		
<input type="checkbox"/>		<input checked="" type="checkbox"/>	2011-06-15	0,00		

Εικόνα 5.6.6. Περισσότερα Εσόδου

Μετά την επιλογή της λειτουργίας φαίνονται παραπάνω πληροφορίες που αφορούν το συγκεκριμένο έσοδο. Αυτές είναι:

- ✚ **Ολοκληρώθηκε**(επιλέγεται όταν ολοκληρώνεται η πληρωμή του εσόδου πληρώνεται πλήρως)
 - **Πότε**(ημέρα που ολοκληρώθηκε η πληρωμή)
 - **Διαφορά**(διαφορά που ίσως υπάρξει στα χρήματα για κάποιο λόγο ,π.χ. καθυστέρηση)
- ✚ **Δόσεις**(παρουσιάζει αυτόματα τις δόσεις που χρειάζεται να πληρωθούν στα αναδρομικά έσοδα από την Ημερ/νία Έναρξης έως την τρέχουσα(σημερινή) Ημερ/νία)

Ολοκληρώθηκε

Μετά την επιλογή του πλαισίου, τα δύο επόμενα πεδία ξεκλειδώνουν, **Πότε** και **Διαφορά**. Στο πρώτο πεδίο καταχωρούμε την ημερομηνία που το συγκεκριμένο έσοδο ολοκληρώθηκε. Στο δεύτερο, αντίστοιχα, αν υπάρχει κάποιο επιπλέον ή λιγότερο χρηματικό ποσό για λόγους καθυστέρησης ή αντίστοιχα κάποιας έκπτωσης. Όλα αυτά μπορούν να πραγματοποιηθούν κατά την επιλογή της **Επεξεργασίας**, η οποία διατίθεται στην συγκεκριμένη σελίδα **Περισσότερα** του εσόδου, καθώς και άλλες χρήσιμες δυνατότητες .

Πιο συγκεκριμένα:

- **Διαγραφή**
- **Επεξεργασία**
- **Εισαγωγή Νέου**
- **Πίσω**

,τα οποία εξηγούνται λεπτομερώς στο τρέχον κεφάλαιο.

Δόσεις

Δόση Πληρωμένη	Δόση	Πότε	Προθεσμία Κατάθεσης	
<input type="checkbox"/>			2010-12-01	<input type="button" value="Επεξεργασία"/>
<input type="checkbox"/>			2011-01-01	<input type="button" value="Επεξεργασία"/>
<input type="checkbox"/>			2011-02-01	<input type="button" value="Επεξεργασία"/>
<input type="checkbox"/>			2011-03-01	<input type="button" value="Επεξεργασία"/>
<input type="checkbox"/>			2011-04-01	<input type="button" value="Επεξεργασία"/>
<input type="checkbox"/>			2011-05-01	<input type="button" value="Επεξεργασία"/>
<input type="checkbox"/>			2011-06-01	<input type="button" value="Επεξεργασία"/>
Σύνολο Πληρωμένων Δόσεων	.00			
Συνολικό Ποσό Εξόφλησης	2000.00			

Εικόνα 5.6.7. Δόσεις Εσόδου

Εδώ προβάλλονται, εισάγονται, επεξεργάζονται και διαγράφονται οι δόσεις που αφορούν το αναδρομικό έσοδο. Κατά την εισαγωγή ενός εσόδου **αναδρομικού** οι δόσεις δημιουργούνται **αυτόματα** από την ημερομηνία **έναρξης** του εσόδου έως και την τελευταία δόση που αφορά την **τρέχουσα**-σημερινή ημερομηνία. Αυτές οι δόσεις καταχωρούνται ως **απλήρωτες**.

Τα στοιχεία που αφορούν μια δόση είναι:

- Δόση
- Πότε
- Προθεσμία Κατάθεσης

Για να μπορέσει ο χρήστης να καταχωρήσει κάποιο δεδομένο ή να το αλλάξει, σε κάποια **ληχθείσα** δόση πρέπει να επιλέξει το κουμπί **Επεξεργασία**. Την ίδια ακριβώς διεργασία κάνει και το κουμπί **Εισαγωγή Δόσης**, το οποίο χρησιμοποιείται για την καταχώρησης της τρέχουσας δόσης που δεν έχει ξεπεράσει την ημερομηνία λήξεως της. Μετά από την εισαγωγή της τρέχουσας δόσης, δεν μπορεί να γίνει εισαγωγή άλλων που αφορούν τις επόμενες.

Επεξεργασία Δόσης Εσόδου			
Δόση Πληρωμένη	Δόση	Πότε	Προθεσμία Κατάθεσης
<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	2010-12-01

Εικόνα 5.6.8. Επεξεργασία/Εισαγωγή Δόσης Εσόδου

Για να μπορέσετε να εισάγετε δεδομένα σε μια δόση, πρέπει να τσεκάρετε το πλαίσιο του πεδίου **Δόση πληρωμένη**, όπου ξεκλειδώνει τα δυο επόμενα πεδία. Στο πεδίο **Δόση** συμπληρώνεται το ποσό που πληρώθηκε κατά την συγκεκριμένη δόση. Στο πεδίο **Πότε** συμπληρώνεται η ημερομηνία που πληρώθηκε επιτυχώς η συγκεκριμένη δόση. Το πεδίο **Προθεσμία Κατάθεσης** συμπληρώνεται αυτόματα και δείχνει την τελευταία ημερομηνία πληρωμής της δόσης πριν την λήξη.

Μετά την συμπλήρωση των στοιχείων, επιλέγετε το πλήκτρο **Αποθήκευση**, για την καταχώρηση της δόσης, αλλιώς επιλέγετε το πλήκτρο **Ακύρωση**, για να επιστρέψετε στην προηγούμενη κατάσταση.

iii. Αποθήκευση

Μετά την επιλογή του πλήκτρου **Αποθήκευση**, θα σας εμφανιστεί ένα μήνυμα αναφοράς. Οι περιπτώσεις είναι 4:

a. Αν η αποθήκευση ολοκληρωθεί επιτυχώς, τότε :

Τα στοιχεία αποθηκεύτηκαν με επιτυχία!

b. Αν κάποιο από τα ενεργοποιημένα πεδία παραμείνει κενό μετά την αποθήκευση, τότε :

Τα στοιχεία που δώσατε είναι λάθος. Παρακαλώ συμπληρώστε όλα τα πεδία.

c. Αν το πόσο της δόσης δεν αποτελείται από αριθμό, τότε :

Τα στοιχεία που δώσατε είναι λάθος. Παρακαλώ συμπληρώστε σωστό ποσό.

d. Αν η ημερομηνία είναι συντακτικά λάθος γραμμένη, τότε :

Τα στοιχεία που δώσατε είναι λάθος. Παρακαλώ συμπληρώστε σωστή ημερομηνία.

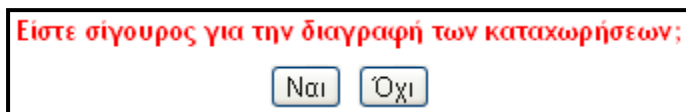
iv. Ακύρωση

Μετά την επιλογή του πλήκτρου **Ακύρωση**, το πρόγραμμα θα επιστρέψει στην προηγούμενη κατάσταση που δείχνει η Εικόνα 5.6.6(Περισσότερα Εσόδου).

c) Διαγραφή

Το πλεονέκτημα σε αυτή την περίπτωση, είναι ότι ο χρήστης μπορεί να κάνει μαζική διαγραφή, δηλαδή να διαγράψει περισσότερα από ένα έσοδα ταυτόχρονα. Για την επιτυχή διαγραφή ενός ή περισσότερων εσόδων, πρέπει να επιλέξετε(τσεκάρετε) το πλαίσιο που αντιστοιχεί στο καθένα από τα έσοδα που θέλετε να διαγραφούν.

Μετά την επιλογή του κουμπιού **Διαγραφή** θα σας ζητηθεί να επιβεβαιώσετε το αίτημα σας πριν την πραγματοποίησή του, δηλαδή να διαγράψετε το έσοδο που επιλέξατε:



Εικόνα 5.6.9. Διαγραφή Εσόδου

Αν επιλέξετε **ναι** τότε το έσοδο θα διαγραφεί και θα επιβεβαιωθεί στην οθόνη σας με την ένδειξη:

Η διαγραφή έγινε με επιτυχία!

Αν επιλέξετε **όχι** τότε το έσοδο δεν διαγράφεται και θα επιβεβαιωθεί στην οθόνη σας με την ένδειξη:

Η διαγραφή δεν έγινε!

d) Επεξεργασία

Όπως ονομάζεται και η λέξη, αυτή η επιλογή χρησιμοποιείται για την πραγματοποίηση τυχών αλλαγών που χρειάζεται κάποιο έσοδο μετά την δημιουργία του. Στην συγκεκριμένη περίπτωση, για την επεξεργασία κάποιου εσόδου, ο χρήστης πρέπει να τσεκάρει το πλαίσιο που αντιστοιχεί στο ανάλογο έσοδο. Αν δεν επιλεγεί κανένα, τότε τίποτα δεν θα γίνει. Ο χρήστης δεν μπορεί να κάνει αλλαγή των στοιχείων σε παραπάνω από ένα έσοδο την φορά.

Όσα κελιά είναι ενεργοποιημένα, πρέπει να συμπληρωθούν, με τον σωστό τύπο χαρακτήρα που παίρνει το καθένα ώστε να μπορέσει να γίνει η αποθήκευση τους επιτυχώς. **Κανένα** ενεργοποιημένο κελί δεν μπορεί να παραμείνει κενό. Στο τέλος του βιβλίου υπάρχει ένας βοηθητικός πίνακας που δείχνει τι είδους τιμές μπορεί να πάρει το κάθε στοιχείο.

Μετά την συμπλήρωση των στοιχείων, επιλέγετε το πλήκτρο **Αποθήκευση**, για την καταχώρηση του εσόδου, αλλιώς επιλέγετε το πλήκτρο **Ακύρωση**, για να επιστρέψετε στην προηγούμενη κατάσταση.

i. Αποθήκευση

Μετά την επιλογή του πλήκτρου **Αποθήκευση**, θα σας εμφανιστεί ένα μήνυμα αναφοράς.

Οι περιπτώσεις είναι 3:

a. Αν η αποθήκευση ολοκληρωθεί επιτυχώς, τότε :

Τα στοιχεία καταχωρήθηκαν με επιτυχία!

b. Αν κάποιο από τα ενεργοποιημένα πεδία παραμείνουν κενά μετά την αποθήκευση, τότε :

Τα στοιχεία που δώσατε είναι λάθος. Επιστρέψτε στην προηγούμενη σελίδα για να τα διορθώσετε.

c. Αν κάποια από τις ημερομηνίες είναι συντακτικά λάθος γραμμένες, τότε :

Τα στοιχεία που δώσατε είναι λάθος, παρακαλούμε επιστρέψτε στην προηγούμενη σελίδα και διορθώστε τα στοιχεία που εισαγάτε.

ii. Ακύρωση

Μετά την επιλογή του πλήκτρου **Ακύρωση**, το πρόγραμμα θα επιστρέψει στην προηγούμενη κατάσταση που δείχνει η Εικόνα 5.6.4(Εσοδα).

e) Ολοκληρώθηκε











Μια χρήσιμη λειτουργία, η οποία επιτρέπει στον χρήστη να μπορεί να καταχωρεί τα έσοδα του ως πληρωμένα, αναδρομικά ή μη. Το πλεονέκτημα της λειτουργίας αυτής, είναι πως μπορεί να χρησιμοποιηθεί όχι μόνο μεμονωμένα, δηλαδή για ένα μόνο έσοδο, αλλά και μαζικά. Ο χρήστης μπορεί συμπληρώνοντας τα πλαίσια δίπλα από κάθε έσοδο, και έπειτα πατώντας το κουμπί να τα δηλώνει σαν πληρωμένα ταυτόχρονα.

Το κουμπί αυτό χρησιμοποιείται όταν ένα έσοδο **πληρώνεται ολόκληρο**. Για τα έσοδα που δεν είναι αναδρομικά δεν υπάρχει καμία εξήγηση. Όταν ένα μη αναδρομικό έσοδο πληρώνεται, τότε δεν έχετε παρά να το επιλέξετε και να το δηλώσετε σαν **ολοκληρωμένο**. Όταν ένα έσοδο όμως είναι αναδρομικό χρειάζεται **προσοχή!** Πρέπει ο χρήστης να γνωρίζει ότι η επιλογή του κουμπιού θεωρεί την πληρωμή του εσόδου και **όχι** της **δόσης** του εσόδου. Γι' αυτό πρέπει, ο χρήστης να είναι σίγουρος ότι αν θέλει να χρησιμοποιήσει αυτή την λειτουργία ότι μιλάει για ολόκληρο το **έσοδο** και όχι για μια δόση του, γιατί μετά οι δόσεις θα **σταματήσουν** να μετράνε αυτόματα από το σύστημα.

5.6.4 – Έξοδα

Εμφανίζονται όλα τα έξοδα της προεπιλεγμένης εταιρίας που πληρούν όλες τις συνθήκες της αναζήτησης. Για κάθε έξοδο παρουσιάζονται οι πιο χρήσιμες από τις πληροφορίες που αποτελεί το καθένα.

Αυτές είναι:

-  **Εταιρία**
-Έξοδο για την συγκεκριμένη εταιρία
-  **Είδος**
-Είδος εξόδου
-  **Ποσό**
-Ποσό που χρειάζεται να πληρωθεί
-  **Ημερομηνία Έναρξης**
-Ημερ/νία που ξεκινάει η συναλλαγή
-  **Ημερομηνία Λήξης**
-Ημερ/νία που λήγει η συναλλαγή
-  **Προμηθευτής/Πελάτης**
-Όνομα ή επωνυμία συνεργατών
-  **Σημείωση**
-Σχόλια που δίνουν βοηθητικές πληροφορίες
-  **Αναδρομικότητα**
-Δείχνει αν το ποσό είναι αναδρομικό ή όχι
-  **Κάθε Πότε**
-Δείχνει την συχνότητα που πρέπει να καταβάλλεται το αναδρομικό ποσό
-  **Συνολικά**
-Δείχνει το ποσό αναδρομικό ή μη, μέχρι το τέλος της ημέρας που λήγουν οι συναλλαγές

	Εταιρία	Είδος	Ποσό	Ημ. Έναρξης	Ημ. Λήξης	Προμηθ./Πελάτης	Σημείωση	Αναδρ/τα	Κάθε Πότε	Συνολικά
<input type="checkbox"/>	TMP	dei	15.00	2011-02-01	2011-09-01	test test	cxfds	<input checked="" type="checkbox"/>	1 μήνα	105.00
<input type="checkbox"/>	TMP	dei	200.00	2011-03-01	2011-12-01	test test	dsaadsa	<input checked="" type="checkbox"/>	20 ημέρες	2600.00
Σύνολο										2705.00

Εικόνα 5.6.10. Έξοδα

Η φόρμα όπου βρίσκονται τα έξοδα, αποτελείται από **4** πλήκτρα, αν υπάρχει έστω και ένα καταχωρημένο έξοδο στην επιλεγμένη εταιρία*(*ακόμα και αν δεν εμφανίζεται κανένα λόγω συγκεκριμένης αναζήτησης χωρίς αποτέλεσμα*)

Αυτά είναι:

- i. **Εισαγωγή Νέου**
- ii. **Περισσότερα**
- iii. **Διαγραφή**
- iv. **Επεξεργασία**
- v. **Ολοκληρώθηκε**

*Αν δεν υπάρχει έστω μια καταχώρηση στα έξοδα της επιλεγμένης εταιρίας τότε εμφανίζεται μόνο ένα πλήκτρο, αυτό της **Εισαγωγής Νέου**.

a) Εισαγωγή Νέου

Τα στοιχεία που μπορείτε να δώσετε σε ένα νέο έξοδο είναι:

- ✚ Εταιρεία
- ✚ Είδος
- ✚ Ποσό
- ✚ Ημερομηνία Έναρξης
- ✚ Ημερομηνία Λήξης
- ✚ Προμηθευτής/Πελάτης
- ✚ Σημείωση
- ✚ Αναδρομικότητα
 - Κάθε Πότε
- ✚ Ολοκληρώθηκε(επιλέγεται όταν ολοκληρώνεται η πληρωμή του εξόδου)
 - Πότε(ημέρα που ολοκληρώθηκε η πληρωμή)
 - Διαφορά(διαφορά που ίσως υπάρξει στα χρήματα για κάποιο λόγο ,π.χ. καθυστέρηση)

Όσα κελιά είναι ενεργοποιημένα, πρέπει να συμπληρωθούν, με τον σωστό τύπο χαρακτήρα που παίρνει το καθένα ώστε να μπορέσει να γίνει η αποθήκευση τους επιτυχώς. Στο τέλος του βιβλίου υπάρχει ένας βοηθητικός πίνακας που δείχνει τι είδους τιμές μπορεί να πάρει το κάθε στοιχείο.

Μετά την συμπλήρωση των στοιχείων, επιλέγετε το πλήκτρο **Αποθήκευση**, για την καταχώρηση του εξόδου, αλλιώς επιλέγετε το πλήκτρο **Ακύρωση**, για να επιστρέψετε στην προηγούμενη κατάσταση.

Εισαγωγή Νέου						
Εταιρία	Είδος	Ποσό	Ημ. Έναρξης	Ημ. Λήξης	Προμηθ./Πελάτης	Σημείωση
TMP	Φ.Π.Α	40.99	2011-02-01	2011-06-30	Σωτήριος Σωτηρίου	paradigma
Αναδρομικότητα	Κάθε Πότε	Ολοκληρώθηκε	Πότε	Διαφορά		
<input checked="" type="checkbox"/>	1 Μήνες	<input type="checkbox"/>				

Εικόνα 5.6.11. Εισαγωγή Εξόδου

i. Αποθήκευση

Μετά την επιλογή του πλήκτρου **Αποθήκευση**, θα σας εμφανιστεί ένα μήνυμα αναφοράς. Οι περιπτώσεις είναι 3:

a. Αν η αποθήκευση ολοκληρωθεί επιτυχώς, τότε :

Τα στοιχεία καταχωρήθηκαν με επιτυχία!

b. Αν κάποιο από τα ενεργοποιημένα πεδία παραμείνουν κενά μετά την αποθήκευση, τότε :

Τα στοιχεία που δώσατε είναι λάθος. Επιστρέψτε στην προηγούμενη σελίδα για να τα διορθώσετε.

c. Αν κάποια από τις ημερομηνίες είναι συντακτικά λάθος γραμμένες, τότε :

Τα στοιχεία που δώσατε είναι λάθος, παρακαλούμε επιστρέψτε στην προηγούμενη σελίδα και διορθώστε τα στοιχεία που εισαγάτε.

ii. Ακύρωση

Μετά την επιλογή του πλήκτρου **Ακύρωση**, το πρόγραμμα θα επιστρέψει στην προηγούμενη κατάσταση που δείχνει η Εικόνα 5.6.10(Εξοδα).

b) Περισσότερα

Στην επιλογή **Περισσότερα** ο χρήστης μπορεί να δει περαιτέρω στοιχεία που αφορούν το κάθε έξοδο, τα οποία δεν φαίνονται στην κεντρική σελίδα της **Οικονομικής Επισκόπησης**. Η συγκεκριμένη λειτουργία δουλεύει μεμονωμένα, δηλαδή μόνο για ένα έξοδο την κάθε φορά τσεκάροντας το πλαίσιο δίπλα από το έξοδο που θέλουμε να μελετήσουμε.

Προβολή Εξόδου						
Εταιρία	Είδος	Ποσό	Ημ. Έναρξης	Ημ. Λήξης	Προμηθ./Πελάτης	Σημείωση
TMP	dei	200.00	2011-03-01	2011-12-01	test test	dsaadsa
Αναδρομικότητα	Κάθε Πότε	Ολοκληρώθηκε	Πότε	Διαφορά		
<input checked="" type="checkbox"/>	20 ημέρες	<input type="checkbox"/>				

Διαγραφή Επεξεργασία Πίσω
Εισαγωγή νέου

Εικόνα 5.6.12. Περισσότερα Εξόδου

Μετά την επιλογή της λειτουργίας φαίνονται παραπάνω πληροφορίες που αφορούν το συγκεκριμένο έξοδο. Αυτές είναι:

- ✚ **Ολοκληρώθηκε**(επιλέγεται όταν ολοκληρώνεται η πληρωμή του εξόδου πληρώνεται πλήρως)
 - **Πότε**(ημέρα που ολοκληρώθηκε η πληρωμή)
 - **Διαφορά**(διαφορά που ίσως υπάρξει στα χρήματα για κάποιο λόγο ,π.χ. καθυστέρηση)
- ✚ **Δόσεις**(παρουσιάζει αυτόματα τις δόσεις που χρειάζεται να πληρωθούν στα αναδρομικά έξοδα από την Ημερ/νία Έναρξης έως την τρέχουσα(σημερινή) Ημερ/νία)

Ολοκληρώθηκε

Μετά την επιλογή του πλαισίου, τα δύο επόμενα πεδία ξεκλειδώνουν, **Πότε** και **Διαφορά**. Στο πρώτο πεδίο καταχωρούμε την ημερομηνία που το συγκεκριμένο έξοδο ολοκληρώθηκε. Στο δεύτερο, αντίστοιχα, αν υπάρχει κάποιο επιπλέον ή λιγότερο χρηματικό ποσό για λόγους καθυστέρησης ή αντίστοιχα κάποιας έκπτωσης. Όλα αυτά μπορούν να πραγματοποιηθούν κατά την επιλογή της **Επεξεργασίας**, η οποία διατίθεται στην συγκεκριμένη σελίδα **Περισσότερα** του εσόδου, καθώς και άλλες χρήσιμες δυνατότητες .

Πιο συγκεκριμένα:

- **Διαγραφή**
- **Επεξεργασία**
- **Εισαγωγή Νέου**
- **Πίσω**

Δόσεις




Δόση Πληρωμένη	Δόση	Πότε	Προθεσμία Κατάθεσης	
<input type="checkbox"/>			2011-03-01	Επεξεργασία
<input type="checkbox"/>			2011-04-01	Επεξεργασία
<input type="checkbox"/>			2011-05-01	Επεξεργασία
<input type="checkbox"/>			2011-06-01	Επεξεργασία
Σύνολο Πληρωμένων Δόσεων	.00			
Συνολικό Ποσό Εξόφλησης	105.00			

Εισαγωγή Δόσης

Εικόνα 5.6.13. Δόσεις Εξόδου

Εδώ προβάλλονται, εισάγονται, επεξεργάζονται και διαγράφονται οι δόσεις που αφορούν το αναδρομικό έξοδο. Κατά την εισαγωγή ενός εξόδου **αναδρομικού** οι δόσεις δημιουργούνται **αυτόματα** από την ημερομηνία **έναρξης** του εξόδου έως και την τελευταία δόση που αφορά την **τρέχουσα**-σημερινή ημερομηνία. Αυτές οι δόσεις καταχωρούνται ως **απλήρωτες**.

Τα στοιχεία που αφορούν μια δόση είναι:

-  Δόση
-  Πότε
-  Προθεσμία Κατάθεσης

Για να μπορέσει ο χρήστης να καταχωρήσει κάποιο δεδομένο ή να το αλλάξει, σε κάποια **ληχθείσα** δόση πρέπει να επιλέξει το κουμπί **Επεξεργασία**. Την ίδια ακριβώς διεργασία κάνει και το κουμπί **Εισαγωγή Δόσης**, το οποίο χρησιμοποιείται για την καταχώρησης της τρέχουσας δόσης που δεν έχει ξεπεράσει την ημερομηνία λήξεως της. Μετά από την εισαγωγή της τρέχουσας δόσης, δεν μπορεί να γίνει εισαγωγή άλλων που αφορούν τις επόμενες.

Εισαγωγή Νέας Δόσης			
Δόση Πληρωμένη	Δόση	Πότε	Προθεσμία Κατάθεσης
<input checked="" type="checkbox"/>	<input type="text"/>	<input type="text"/>	2011-07-01
<input type="button" value="Αποθήκευση"/> <input type="button" value="Ακύρωση"/>			

Εικόνα 5.6.14. Εισαγωγή/ Επεξεργασία Δόσης Εξόδου

Για να μπορέσετε να εισάγετε δεδομένα σε μια δόση, πρέπει να τσεκάρετε το πλαίσιο του πεδίου **Δόση πληρωμένη**, όπου ξεκλειδώνει τα δυο επόμενα πεδία. Στο πεδίο **Δόση** συμπληρώνεται το ποσό που πληρώθηκε κατά την συγκεκριμένη δόση. Στο πεδίο **Πότε** συμπληρώνεται η ημερομηνία που πληρώθηκε επιτυχώς η συγκεκριμένη δόση. Το πεδίο **Προθεσμία Κατάθεσης** συμπληρώνεται αυτόματα και δείχνει την τελευταία ημερομηνία πληρωμής της δόσης πριν την λήξη.

Μετά την συμπλήρωση των στοιχείων, επιλέγετε το πλήκτρο **Αποθήκευση**, για την καταχώρηση της δόσης, αλλιώς επιλέγετε το πλήκτρο **Ακύρωση**, για να επιστρέψετε στην προηγούμενη κατάσταση.

i. Αποθήκευση

Μετά την επιλογή του πλήκτρου **Αποθήκευση**, θα σας εμφανιστεί ένα μήνυμα αναφοράς. Οι περιπτώσεις είναι 4:

- a. Αν η αποθήκευση ολοκληρωθεί επιτυχώς, τότε :

Τα στοιχεία αποθηκεύτηκαν με επιτυχία!

- b. Αν κάποιο από τα ενεργοποιημένα πεδία παραμείνει κενό μετά την αποθήκευση, τότε :

Τα στοιχεία που δώσατε είναι λάθος. Παρακαλώ συμπληρώστε όλα τα πεδία.

- c. Αν το πόσο της δόσης δεν αποτελείται από αριθμό, τότε :

Τα στοιχεία που δώσατε είναι λάθος. Παρακαλώ συμπληρώστε σωστό ποσό.

- d. Αν η ημερομηνία είναι συντακτικά λάθος γραμμένη, τότε :

Τα στοιχεία που δώσατε είναι λάθος. Παρακαλώ συμπληρώστε σωστή ημερομηνία.

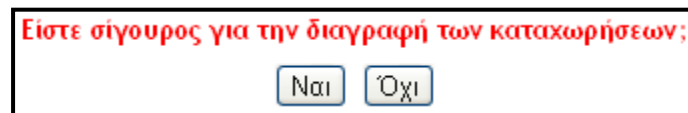
ii. Ακύρωση

Μετά την επιλογή του πλήκτρου **Ακύρωση**, το πρόγραμμα θα επιστρέψει στην προηγούμενη κατάσταση που δείχνει η Εικόνα 5.6.11(Περισσότερα Εξόδο).

c) Διαγραφή

Το πλεονέκτημα σε αυτή την περίπτωση, είναι ότι ο χρήστης μπορεί να κάνει μαζική διαγραφή, δηλαδή να διαγράψει περισσότερα από ένα έξοδα ταυτόχρονα. Για την επιτυχή διαγραφή ενός ή περισσότερων εσόδων, πρέπει να επιλέξετε(τσεκάρετε) το πλαίσιο που αντιστοιχεί στο καθένα από τα έξοδα που θέλετε να διαγραφούν.

Μετά την επιλογή του κουμπιού **Διαγραφή** θα σας ζητηθεί να επιβεβαιώσετε το αίτημα σας πριν την πραγματοποίηση του, δηλαδή να διαγράψετε το έξοδο που επιλέξατε:



Εικόνα 5.6.15. Διαγραφή Εξόδου

Αν επιλέξετε **ναι** τότε το έξοδο θα διαγραφεί και θα επιβεβαιωθεί στην οθόνη σας με την ένδειξη:

Η διαγραφή έγινε με επιτυχία!

Αν επιλέξετε **όχι** τότε το έξοδο δεν διαγράφεται και θα επιβεβαιωθεί στην οθόνη σας με την ένδειξη:

Η διαγραφή δεν έγινε!

d) Επεξεργασία

Όπως ονομάζεται και η λέξη, αυτή η επιλογή χρησιμοποιείται για την πραγματοποίηση τυχών αλλαγών που χρειάζεται κάποιο έξοδο μετά την δημιουργία του. Στην συγκεκριμένη περίπτωση, για την επεξεργασία κάποιου εξόδου, ο χρήστης πρέπει να τσεκάρει το πλαίσιο που αντιστοιχεί στο ανάλογο έξοδο. Αν δεν επιλεγεί κανένα, τότε τίποτα δεν θα γίνει. Ο χρήστης δεν μπορεί να κάνει αλλαγή των στοιχείων σε παραπάνω από ένα έξοδο την φορά.

Όσα κελιά είναι ενεργοποιημένα, πρέπει να συμπληρωθούν, με τον σωστό τύπο χαρακτήρα που παίρνει το καθένα ώστε να μπορέσει να γίνει η αποθήκευση τους επιτυχώς. **Κανένα** ενεργοποιημένο κελί δεν μπορεί να παραμείνει κενό. Στο τέλος του βιβλίου υπάρχει ένας βοηθητικός πίνακας που δείχνει τι είδους τιμές μπορεί να πάρει το κάθε στοιχείο.

Μετά την συμπλήρωση των στοιχείων, επιλέγετε το πλήκτρο **Αποθήκευση**, για την καταχώρηση του εξόδου, αλλιώς επιλέγετε το πλήκτρο **Ακύρωση**, για να επιστρέψετε στην προηγούμενη κατάσταση.

i. Αποθήκευση

Μετά την επιλογή του πλήκτρου **Αποθήκευση**, θα σας εμφανιστεί ένα μήνυμα αναφοράς. Οι περιπτώσεις είναι 3:

a. Αν η αποθήκευση ολοκληρωθεί επιτυχώς, τότε :

Τα στοιχεία καταχωρήθηκαν με επιτυχία!

b. Αν κάποιο από τα ενεργοποιημένα πεδία παραμείνουν κενά μετά την αποθήκευση, τότε :

Τα στοιχεία που δώσατε είναι λάθος. Επιστρέψτε στην προηγούμενη σελίδα για να τα διορθώσετε.

γ. Αν κάποια από τις ημερομηνίες είναι συντακτικά λάθος γραμμένες, τότε :

Τα στοιχεία που δώσατε είναι λάθος, παρακαλούμε επιστρέψτε στην προηγούμενη σελίδα και διορθώστε τα στοιχεία που εισαγάτε.

ii. Ακύρωση

Μετά την επιλογή του πλήκτρου **Ακύρωση**, το πρόγραμμα θα επιστρέψει στην προηγούμενη κατάσταση που δείχνει το Σχήμα 2_2(Εισαγωγή Νέου Προμηθευτή/Πελάτη).

e) Ολοκληρώθηκε

Μια χρήσιμη λειτουργία, η οποία επιτρέπει στον χρήστη να μπορεί να καταχωρεί τα έξοδα του ως πληρωμένα, αναδρομικά ή μη. Το πλεονέκτημα της λειτουργίας αυτής, είναι πως μπορεί να χρησιμοποιηθεί όχι μόνο μεμονωμένα, δηλαδή για ένα μόνο έξοδο, αλλά και μαζικά. Ο χρήστης μπορεί συμπληρώνοντας τα πλαίσια δίπλα από κάθε έσοδο, και έπειτα πατώντας το κουμπί να τα δηλώνει σαν πληρωμένα ταυτόχρονα.

Το κουμπί αυτό χρησιμοποιείται όταν ένα έξοδο **πληρώνεται ολόκληρο**. Για τα έξοδα που δεν είναι αναδρομικά δεν υπάρχει καμία εξήγηση. Όταν ένα μη αναδρομικό έξοδο πληρώνεται, τότε δεν έχετε παρά να το επιλέξετε και να το δηλώσετε σαν **ολοκληρωμένο**. Όταν ένα έξοδο όμως είναι αναδρομικό χρειάζεται **προσοχή!** Πρέπει ο χρήστης να γνωρίζει ότι η επιλογή του κουμπιού θεωρεί την πληρωμή του εξόδου και **όχι** της **δόσης** του εξόδου. Γι' αυτό πρέπει, ο χρήστης να είναι σίγουρος, ότι αν θέλει να χρησιμοποιήσει αυτή την λειτουργία, ότι μιλάει για ολόκληρο το **έξοδο** και όχι για μια δόση του, γιατί μετά οι δόσεις θα **σταματήσουν** να μετράνε αυτόματα από το σύστημα.

5.6.5 – Συνολικά

Εμφανίζει το σύνολο όλων των εσόδων και όλων των εξόδων μιας εταιρίας, και δείχνει αν υπάρχει κέρδος ή όχι ανάμεσα σε έσοδα και έξοδα. Στην φόρμα **Συνολικά** εμφανίζονται οι πιο χρήσιμες πληροφορίες που αφορούν το σύνολο.

Αυτές είναι:

- ✚ Εταιρεία
- ✚ Είδος
- ✚ Ημερομηνία Έναρξης
- ✚ Ημερομηνία Λήξης
- ✚ Προμηθευτής/Πελάτης
- ✚ Έσοδα
- ✚ Έξοδα
- ✚ Έσοδα – Έξοδα
- ✚ Σύνολο

Εταιρία	Είδος	Ημ. Έναρξης	Ημ. Λήξης	Προμηθ./Πελάτης	Εσοδα	Εξοδα	Έσοδα - Εξοδα	Σύνολο
TMP	Όλα	2011-01-01	2011-12-31	Όλοι	2190,23	2705,00	2190,23 - 2705,00	-514,77

Εικόνα 5.6.16. Συνολικά

Εταιρία

Δείχνει την επιλεγμένη εταιρία.

Είδος

Παίρνει στοιχεία ανεξάρτητα από το είδος.

Ημ. Έναρξης

Παίρνει την παλαιότερη Ημ. Έναρξης από τα στοιχεία που φαίνονται βάσει της φόρμας αναζήτησης.

Ημ. Λήξης

Παίρνει την τελευταία Ημ. Λήξης από τα στοιχεία που φαίνονται βάσει της φόρμας αναζήτησης.

Προμηθευτής/Πελάτης

Παίρνει στοιχεία ανεξάρτητα από το Προμηθευτή/Πελάτη.

Έσοδα

Δείχνει το άθροισμα όλων των εσόδων της εταιρίας.

Έξοδα

Δείχνει το άθροισμα όλων των εξόδων της εταιρίας.

Έσοδα - Έξοδα

Δείχνει την διαφορά των εσόδων μείον των εξόδων της εταιρίας.

Σύνολο

Δείχνει το κέρδος ή την ζημία που προκύπτει από το σύνολο εσόδων και εξόδων. Το **κέρδος** εμφανίζεται με **πράσινο** φόντο και η **ζημία** με **κόκκινο**.

5.7 – Έξοδος

Μετά την συγκεκριμένη επιλογή του μενού επιτυγχάνεται αποσύνδεση του χρήστη από την εφαρμογή. Για ασφάλεια του συστήματος, για να ξανασυνδεθεί ο χρήστης θα πρέπει να εισάγει τα στοιχεία λογαριασμού του, **Username** και **Password**.

Βιβλιογραφία

Εισαγωγή στον Προγραμματισμό. Βασιλάκος Ξενοφών
MySQL Connector/J. Matthews Mark
MySQL GUI Tools Manual
Εισαγωγή στο CSS. Κομβούτη-Βέρου Μιχαήλια
Εισαγωγή στην UML. Βεσκούκης Βασίλειος. Πανεπιστήμιο Πειραιά

Πηγές

<http://opensci.grnet.gr/software/mysql-administrator>
<http://www.15seconds.com/issue/050210.htm>
<http://www.javaguicodexample.com/installusejavanetbeans1.html>
<http://www.netbeans.org/kb/docs/web/mysql-webapp.html>
<http://opensci.grnet.gr/software>
http://en.wikipedia.org/wiki/Apache_Tomcat
<http://en.wikipedia.org/wiki/JavaScript>
<http://dide.flo.sch.gr>
<http://users.uom.gr/~kaklaman/book/Chapters/C7/Servlets%202.htm>
http://www.islab.demokritos.gr/gr/html/ptixiakes/kostas-ariss_ptyxiakh/Phtml/java.htm
<http://blogs.sch.gr/giannopk/files/2010/12/phpmysql.pdf>

Παράρτημα Α – Υπόδειξη Τιμών

Τύπος	Εξήγηση Τιμής	Default Τιμή
Integer	Ακέραιος αριθμός	0
Float	Δεκαδικός αριθμός	0.0f
Double		0.0d
String	Χαρακτήρας	null
Boolean	Τιμή που παίρνει true ή false	false
Date	Ημερομηνία	0000-00-00

Πίνακας Α1. Τύποι Συμπλήρωσης Πεδίων

Αρχική		
Κατηγορία	Πεδίο	Τύπος
Έσοδα/ Έξοδα	Ημέρα Πληρωμής	Date
	Ποσό	Float
Στοιχεία Λογαριασμού	Όνοματεπώνυμο	String
	Όνομα Χρήστη	String
	Κωδικός	String

Πίνακας Α2. Συμπλήρωση Πεδίων στην σελίδα Αρχική

Εταιρίες		
Κατηγορία	Πεδίο	Τύπος
Εταιρίες	Επωνυμία	String
	Όνομα	String
	Εργασία	String
	Α.Φ.Μ.	String
	Δ.Ο.Υ.	String
	Διεύθυνση	String
	ΦΑΞ	String
	Ιστοσελίδα	String
	Ποσοστό	String
Πελάτες/ Προμηθευτές	Όνοματεπώνυμο	String
Έσοδα/ Έξοδα	Έσοδο/Έξοδο	String

Πίνακας Α3. Συμπλήρωση Πεδίων στην σελίδα Εταιρίες

Ημερολόγιο		
Κατηγορία	Πεδίο	Τύπος
Ενέργειες (Εισαγωγή/ Επεξεργασία)	Ενέργεια	String
	Ημ. Έναρξης	Date
	Ημ. Λήξης	Date
	Σημειώσεις	String

Πίνακας Α4. Συμπλήρωση Πεδίων στην σελίδα Ημερολόγιο

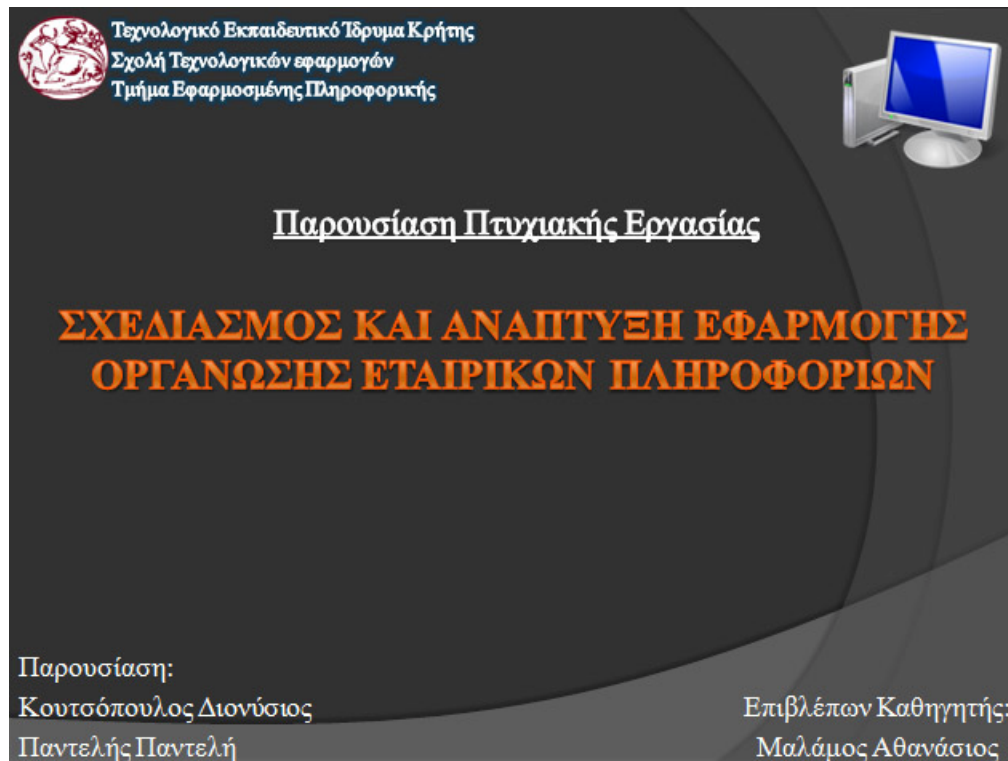
Τηλεφωνικός Κατάλογος		
Κατηγορία	Πεδίο	Τύπος
Επαφή (Εισαγωγή/ Επεξεργασία)	Όνομα	String
	Επίθετο	String
	Ημερ. Γέννησης	Date
	E-mail	String
	Τηλέφωνο	String
	Σχόλια	String

Πίνακας Α5. Συμπλήρωση Πεδίων στην σελίδα Τηλεφωνικός Κατάλογος

Οικονομική Επισκόπηση		
Κατηγορία	Πεδίο	Τύπος
Έσοδο/Εξοδο (Εισαγωγή/ Επεξεργασία)	Εταιρία	String
	Είδος	String
	Ημ. Έναρξης	Date
	Ημ. Λήξης	Date
	Προμηθ./Πελάτης	String
	Σημείωση	String
	Κάθε Πότε	String
	Πότε	Date
	Διαφορά	Float
Δόσεις (Εισαγωγή/ Επεξεργασία)	Δόση Πληρωμένη	Boolean
	Δόση	Float
	Πότε	Date
	Προθεσμία Κατάθεσης	Date

Πίνακας Α6. Συμπλήρωση Πεδίων στην σελίδα Οικονομική Επισκόπηση

Παράρτημα Β – Διαφάνειες Παρουσίασης Πτυχιακής Εργασίας



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης
Σχολή Τεχνολογικών εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής

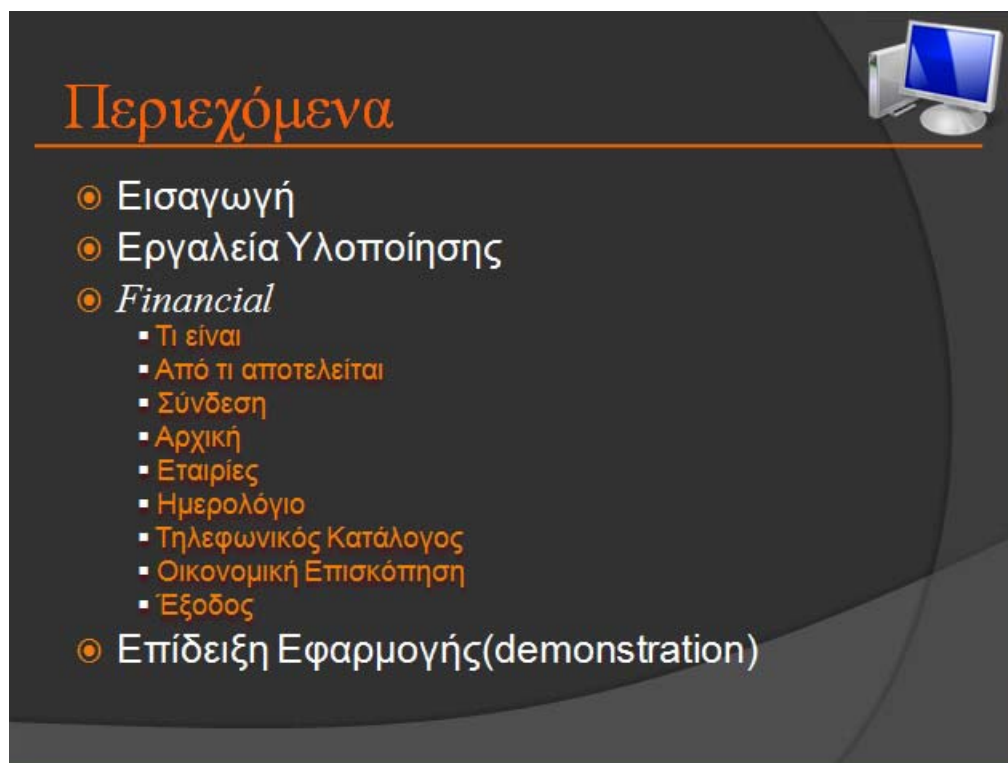
Παρουσίαση Πτυχιακής Εργασίας

**ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ
ΟΡΓΑΝΩΣΗΣ ΕΤΑΙΡΙΚΩΝ ΠΛΗΡΟΦΟΡΙΩΝ**

Παρουσίαση:
Κουτσόπουλος Διονύσιος
Παντελής Παντελής

Επιβλέπων Καθηγητής:
Μαλάμος Αθανάσιος

Εικόνα Β1. Εξώφυλλο Παρουσίασης




Περιεχόμενα

- Εισαγωγή
- Εργαλεία Υλοποίησης
- *Financial*
 - Τι είναι
 - Από τι αποτελείται
 - Σύνδεση
 - Αρχική
 - Εταιρίες
 - Ημερολόγιο
 - Τηλεφωνικός Κατάλογος
 - Οικονομική Επισκόπηση
 - Έξοδος
- Επίδειξη Εφαρμογής(demonstration)

Εικόνα Β2. Περιεχόμενα Παρουσίασης


Εισαγωγή



- Σκοπός
 - Να πραγματοποιηθεί και να παραδοθεί ένα Web Application που να οργανώνει τις υποχρεώσεις της εταιρίας ή του ιδιώτη. Υποχρεώσεις όπως οικονομικές συναλλαγές, τακτοποίηση ραντεβού και επαγγελματικών τηλεφώνων, αποθήκευση συνεργατών κ.α.
- Στόχος
 - Η κατανόηση στην σύνδεση μεταξύ των Γλωσσών Προγραμματισμού
 - Η έρευνα των σωστών όρων για την πραγματοποίηση της εφαρμογής.
 - Η ασφάλεια του συστήματος.
 - Η ανάπτυξη όλων των παραπάνω για το επιθυμητό αποτέλεσμα.

Εικόνα Β3. Εισαγωγή

Εργαλεία Υλοποίησης



- Γλώσσες Προγραμματισμού
 - Java
 - JSP
 - MySQL
 - JavaScript
 - CSS
 - HTML
- Προγράμματα
 - Netbeans IDE 6.9.1
 - MySQL Server 5.1
 - MySQL GUI Tools
 - Apache Tomcat Server 6.0
 - MySQL Connector/J

Εικόνα Β4. Εργαλεία Υλοποίησης

Financial



- Τι είναι:
 - > Είναι μια ψηφιακή **εφαρμογή** που χρησιμοποιείται από κάποιον **Web Browser**, η οποία οργανώνει και διαχειρίζεται υπηρεσίες που αφορούν μια επιχείρηση, μια εταιρία, ή ένα κατάστημα. Υπηρεσίες όπως την εισαγωγή και επεξεργασία χρηματικών συναλλαγών, την προσθήκη συνεργατών, υπενθύμιση υποχρεώσεων όπως ραντεβού ή τηλεφώνων, τηλεφωνικό κατάλογο κ.α.


Εικόνα Β5. Financial – Τι είναι

Financial



- Από τι αποτελείται:
 - > Από το κυρίως **μενού**, το οποίο ανάλογα την επιλογή φορτώνει την αντίστοιχη σελίδα.
 - > Από χρήσιμες πληροφορίες όπως:
 - Όνομα Εισαχθέντα Χρήστη
 - Ωρα
 - Ημερομηνία
 - > Από το περιεχόμενο της κάθε σελίδας. Οι σελίδες είναι:
 - Αρχική Σελίδα
 - Εταιρίες
 - Ημερολόγιο
 - Τηλεφωνικός Κατάλογος
 - Οικονομική Επισκόπηση
 - Έξοδος


Εικόνα Β6. Financial – Από τι Αποτελείται



Financial - Σύνδεση

- Ορισμός
 - Είναι η πρώτη σελίδα που εμφανίζεται κατά την φόρτωση της εφαρμογής.
 - Η σελίδα αυτή αποτελεί την είσοδο του χρήστη στα αποθηκευμένα δεδομένα του. Αποτελείται από δύο πεδία:
 - **Username** (Όνομα Χρήστη)
 - **Password** (Κωδικός Χρήστη)
- Πλεονεκτήματα
 - Ασφάλεια – Αποφυγή ανεπιθύμητων επιθέσεων.
 - Αποθήκευση δεδομένων ανά Χρήστη. Στην περίπτωση που υπάρχουν δυο ή παραπάνω χρήστες της εφαρμογής, τα προσωπικά δεδομένα φορτώνονται βάσει των στοιχείων εισαγωγής τους.


Εικόνα B7. Financial – Σύνδεση



Financial - Αρχική

- Ορισμός
 - Είναι η πρώτη σελίδα που εμφανίζεται μετά την σύνδεση του χρήστη. Εδώ φαίνονται όλες οι τρέχουσες υποχρεώσεις της ημέρας ή όλων των προηγούμενων οι οποίες δεν έχουν κανονιστεί. Οι υποχρεώσεις που υπενθυμίζει το σύστημα είναι:
 - Πληρωμές (έσοδα και έξοδα)
 - Ενέργειες
 - Γενέθλια
- Πλεονεκτήματα
 - Ταχύτερη προβολή υποχρεώσεων σε μια σελίδα χωρίς την ανάγκη της αναζήτησής τους.
 - Αυτόματη πληρωμή ενός ή πολλών μαζί ταυτόχρονα εσόδων ή εξόδων που εκκρεμούν.


Εικόνα B8. Financial – Αρχική



Financial – Εταιρίες(1/2)

- Ορισμός
 - Είναι η σελίδα όπου γίνεται η εισαγωγή και επεξεργασία όλων των στοιχείων που αναφέρονται στους συνεργάτες, και στο είδος συναλλαγής που αφορά των καθένα από αυτούς.
 - Αποτελείται από τρεις βασικές ενότητες:
 - Εταιρίες
 - Προμηθευτές/Πελάτες
 - Έσοδα/Έξοδα
- Εταιρίες
 - Σε αυτή την ενότητα γίνεται η εισαγωγή μιας νέας συνεργαζόμενης εταιρίας και των στοιχείων της ή η επεξεργασία και διαγραφή μιας παλιότερης. Τα στοιχεία που καλύπτουν μια συνεργαζόμενη εταιρία είναι:
 - Επωνυμία
 - Όνομα
 - Εργασία
 - Α.Φ.Μ.
 - Δ.Ο.Υ.
 - Διεύθυνση
 - ΦΑΞ
 - Ιστοσελίδα
 - Ποσοστό


Εικόνα Β9. Financial – Εταιρίες(Μέρος 1^ο)



Financial – Εταιρίες(2/2)

- Προμηθευτές/Πελάτες
 - Εδώ γίνεται η αποθήκευση όλων των ατόμων με τους οποίους συναναστρέφεται η επιχείρηση. Τέτοια πρόσωπα μπορεί να είναι είτε πελάτες, είτε προμηθευτές προϊόντων προς την εταιρία. Τα στοιχεία που καλύπτουν έναν προμηθευτή ή πελάτη είναι:
 - Ονοματεπώνυμο
- Έσοδα/Έξοδα
 - Σε αυτή την ενότητα, εισάγονται και αποθηκεύονται όλα τα είδη πληρωμής που καλύπτουν τα έσοδα και τα έξοδα της επιχείρησης. Τα στοιχεία που καλύπτουν ένα έσοδο ή έξοδο είναι:
 - Έσοδο/Έξοδο


Εικόνα Β10. Financial – Εταιρίες(Μέρος 2^ο)



Financial – Ημερολόγιο(1/2)

- Ορισμός
 - Στην σελίδα αυτή, γίνεται η υπενθύμιση γενεθλίων των αποθηκευμένων επαφών και η προβολή – εισαγωγή διαφόρων υποχρεώσεων, όπως επαγγελματικών ραντεβού ή τηλεφώνων.
 - Η σελίδα χωρίζεται σε δυο ενότητες:
 - Γενέθλια
 - Ενέργειες
- Γενέθλια
 - Αυτή η ενότητα υπάρχει μόνο για προβολή στοιχείων και δεν γίνεται καμία επεξεργασία. Δουλεύει για την υπενθύμιση γενεθλίων των αποθηκευμένων συνεργατών, συγγενών και φίλων.
 - Η μόνη διαφορά από τα άλλα ημερολόγια, και ταυτόχρονα το πλεονέκτημα της συγκεκριμένης ενότητας είναι ότι γίνεται η προβολή όλων των γενεθλίων όχι μόνο της τρέχουσας ημέρας, αλλά και των επόμενων 5.

Εικόνα B11. Financial – Ημερολόγιο(Μέρος 1°)



Financial – Ημερολόγιο(2/2)

- Ενέργειες
 - Εδώ, γίνεται η προσθήκη μιας νέας ενέργειας, αλλά και η προβολή όλων των τρεχουσών. Ύστερα από επιλογή, μπορεί κάποιος να δει και όλες τις προηγούμενες που η προθεσμία τους έχει λήξει. Τα στοιχεία που καλύπτουν μια ενέργεια είναι:
 - Ενέργεια
 - Ημερομηνία Έναρξης
 - Ημερομηνία Λήξης
 - Σημειώσεις


Εικόνα B12. Financial – Ημερολόγιο(Μέρος 2°)



Financial – Τηλεφωνικός Κατάλογος(1/2)

- Ορισμός
 - Στην σελίδα αυτή δημιουργείται ένας τηλεφωνικός κατάλογος. Εκτός από την αποθήκευση επαφών, υπάρχουν επιπλέον οι δυνατότητες του ευρετηρίου καθώς και η αναζήτηση ονόματος.
 - Η σελίδα χωρίζεται σε τρεις ενότητες:
 - Εισαγωγή Νέας Επαφής
 - Ευρετήριο
 - Αναζήτηση
- Εισαγωγή Νέας Επαφής
 - Σε αυτό το μέρος, γίνεται η αποθήκευση τηλεφώνων των συνεργατών, συγγενών και φίλων, καθώς και άλλων χρήσιμων πληροφοριών που τους αφορούν. Τα στοιχεία που καλύπτουν μια επαφή είναι:
 - Όνομα
 - Επίθετο
 - Ημερομηνία Γέννησης
 - E-mail
 - Τηλέφωνο
 - Σχόλια

Εικόνα B13. Financial – Τηλεφωνικός Κατάλογος(Μέρος 1^ο)



Financial – Τηλεφωνικός Κατάλογος(2/2)

- Ευρετήριο
 - Σε αυτή την ενότητα, βρίσκονται τα γράμματα του ελληνικού, καθώς και του αγγλικού αλφάβητου. Καθένα από τα γράμματα αποτελεί ένα κουμπί το οποίο, μετά από την επιλογή του, εμφανίζει όλα τα αποθηκευμένα επίθετα, καθώς και οι υπόλοιπες πληροφορίες που τα συνοδεύουν, τα οποία ξεκινούν από το επιλεγμένο γράμμα. Εάν ο χρήστης επιθυμεί να εμφανίσει όλες τις επαφές ανεξάρτητα από το γράμμα που ξεκινάει το επίθετο, υπάρχει το ανάλογο κουμπί.
- Αναζήτηση
 - Εδώ, βρίσκεται το πλεονέκτημα της αναζήτησης. Όταν ο χρήστης, γνωρίζει το ακριβές ή το ελλιπές όνομα της επαφής που χρειάζεται, τότε υπάρχει το επίπεδο της αναζήτησης. Συμφέρει από την χρήση του ευρετηρίου, επειδή επιστρέφει μόνο το αποτέλεσμα ή τα αποτελέσματα που συμπληρώνουν την απαιτούμενη λέξη της αναζήτησης.

Εικόνα B14. Financial – Τηλεφωνικός Κατάλογος(Μέρος 2^ο)

Financial – Οικονομική Επισκόπηση(1/3)

- Ορισμός
 - Είναι η σελίδα που διαχειρίζεται όλες τις οικονομικές πληροφορίες μιας επιχείρησης. Εδώ αποθηκεύονται, επεξεργάζονται και διαγράφονται τα έσοδα και τα έξοδα από και προς τις συνεργαζόμενες εταιρίες. Επίσης, υπάρχει η υπηρεσία αναζήτησης πληρωμών με την χρήση διάφορων παραμέτρων. Τέλος, βάσει των εμφανισμένων αποτελεσμάτων εμφανίζεται το τελικό σύνολο ζημίας ή κέρδους από την επιλεγμένη συνεργαζόμενη εταιρία.
 - Η σελίδα χωρίζεται σε τέσσερις ενότητες:
 - Επιλογή Εταιρίας
 - Φόρμα Αναζήτησης
 - Έσοδα/Έξοδα
 - Συνολικά

Εικόνα Β15. Financial – Οικονομική Επισκόπηση(Μέρος 1°)

Financial – Οικονομική Επισκόπηση(2/3)

- Επιλογή Εταιρίας
 - Σε πρώτο στάδιο, υπάρχει ένας κατάλογος όπου περιέχει όλες τις αποθηκευμένες συνεργαζόμενες εταιρίες. Αν δεν γίνει η επιλογή κάποιας, τότε η υπόλοιπη σελίδα παραμένει κενή. Εφόσον πραγματοποιηθεί η επιλογή κάποιας εταιρίας εμφανίζονται όλα τα υπόλοιπα στοιχεία που περιέχει η συγκεκριμένη σελίδα:
 - Φόρμα Αναζήτησης
 - Έσοδα/Έξοδα
 - Συνολικά
- Φόρμα Αναζήτησης
 - Μετά την επιλογή εταιρίας, η φόρμα αναζήτησης εμφανίζει αυτόματα τα έσοδα και τα έξοδα του **τρέχοντος έτους** για την επιλεγμένη εταιρία. Ύστερα από την αυτόματη εμφάνιση στοιχείων, η φόρμα μπορεί να επεξεργαστεί και να επιστρέψει αποτελέσματα βάσει των ζητούμενων παραμέτρων του χρήστη. Τέτοιες παράμετροι είναι:
 - Ημερομηνία
 - Προμηθευτής/Πελάτης
 - Είδος Εσόδου/Εξόδου
 - Κατάσταση Πληρωμής


Εικόνα Β16. Financial – Οικονομική Επισκόπηση(Μέρος 2°)



Financial – Οικονομική Επισκόπηση(3/3)

- Έσοδα/Έξοδα
 - Είναι η ενότητα όπου βρίσκονται τα έσοδα και τα έξοδα από μια εταιρία. Ο τρόπος που αποθηκεύεται, προβάλλεται και επεξεργάζεται ένα έσοδο ή ένα έξοδο είναι ο ίδιος. Τα στοιχεία που καλύπτουν ένα **έσοδο** ή ένα **έξοδο** είναι:
 - Εταιρία
 - Είδος
 - Ποσό
 - Ημ. Έναρξης
 - Ημ. Λήξης
 - Προμηθευτής/Πελάτης
 - Σημείωση
 - Αναδρομικότητα
 - Κάθε Πότε
 - Ολοκληρώθηκε
 - Πότε
 - Διαφορά
- Συνολικά
 - Εδώ, είναι το σημείο όπου το σύστημα καταλήγει στο αποτέλεσμα **ζημίας**, αν τα έξοδα είναι περισσότερα, ή **κέρδους**, αν τα έσοδα είναι περισσότερα, από την επιλεγμένη εταιρία τηρώντας τις απαιτούμενες παραμέτρους αναζήτησης. Τα στοιχεία που προβάλλονται στην φόρμα του **συνόλου** είναι:
 - Εταιρία
 - Είδος
 - Ημ. Έναρξης
 - Ημ. Λήξης
 - Προμηθευτής/Πελάτης
 - Έσοδα
 - Έξοδα
 - Έσοδα - Έξοδα
 - Σύνολο

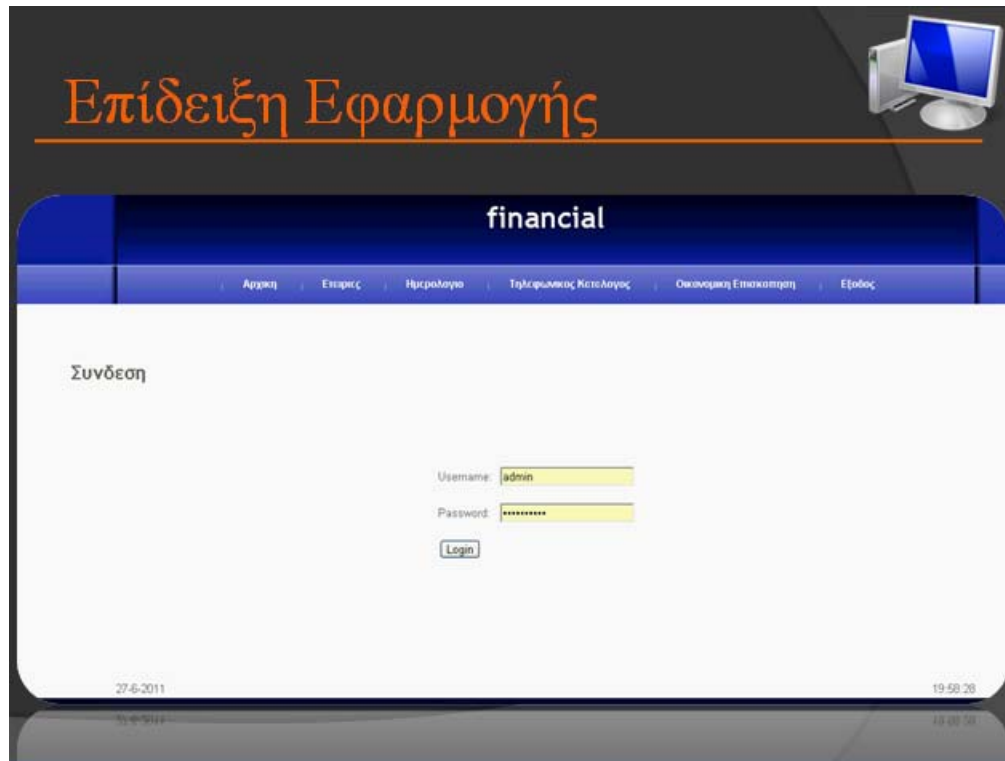
Εικόνα B17. Financial – Οικονομική Επισκόπηση(Μέρος 3°)



Financial – Έξοδος

- Ορισμός
 - Είναι η σελίδα που ο χρήστης αποσυνδέεται από το σύστημα. Για να γίνει αυτό καταστρέφεται το session που κρατάει τις πληροφορίες που χρειάζεται για να μπορούμε να αναγνωρίζουμε ποιος από τους χρήστες ήταν συνδεδεμένος.

Εικόνα B18. Financial – Έξοδος



Εικόνα Β19. Επίδειξη Εφαρμογής



Εικόνα Β20. Τέλος Παρουσίασης Πτυχιακής Εργασίας

Παράρτημα Γ – Περιεχόμενα CD

Μαζί με το βιβλίο της πτυχιακής εργασίας παρέχεται και ένα **CD** το οποίο περιέχει χρήσιμα αρχεία για την βοήθεια του κατασκευαστή. Πιο συγκεκριμένα, μέσα στο CD βρίσκονται τα παρακάτω:

- Ένας φάκελος που ονομάζεται **Προγράμματα** και περιέχει τα:
 - JDK 6 update 25
 - Netbeans IDE 6.9.1
 - MySQL Administrator 5.1
 - MySQL GUI Tools
 - Apache Tomcat 6.0
 - MySQL Connector/J
- Ένας φάκελος που ονομάζεται **Αρχεία** και περιέχει τα:
 - Το ολοκληρωμένο web application που χρησιμοποιείται στον Netbeans με την ονομασία **Financial**.
 - Την MySQL βάση με όλους τους πίνακες που θα χρειαστούν με την ονομασία **financial**.
- Ένας φάκελος που ονομάζεται **Έγγραφα** και περιέχει τα:
 - Το έγγραφο της πτυχιακής εργασίας σε αρχείο Word 2007 με την ονομασία **Πτυχιακή Εργασία**.
 - Το έγγραφο της πτυχιακής εργασίας σε αρχείο Word 2003 με την ονομασία **Πτυχιακή Εργασία 93**.
 - Το έγγραφο της πτυχιακής εργασίας σε αρχείο PDF με την ονομασία **Πτυχιακή Εργασία**.
 - Η παρουσίαση της πτυχιακής εργασίας σε αρχείο PowerPoint 2007 με την ονομασία **Παρουσίαση Πτυχιακής**.
 - Η παρουσίαση της πτυχιακής εργασίας σε αρχείο PowerPoint 2003 με την ονομασία **Παρουσίαση Πτυχιακής 93**.
 - Ο κώδικας όλης της εφαρμογής σε αρχείο PDF με την ονομασία **Κώδικας**.