

Εγχειρίδιο Χρήσης Universal Customization Language (UCL)

Version 3

Πίνακας Περιεχομένων

1 ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ	ΣΦΑΛΜΑ! ΔΕΝ ΕΧΕΙ ΟΡΙΣΤΕΙ ΣΕΛΙΔΟΔΕΙΚΤΗΣ.
2 ΕΙΣΑΓΩΓΗ	2
2.1 Επίπεδα χρηστών	2
2.2 Φόρτωμα	2
2.3 Γενικές Μεταβλητές	2
2.4 Οδηγίες ανάγνωσης των scripts στο παρόν έγγραφο	2
▪ 2.4.1 Χειρισμός των Controls -----	2
3 ΔΥΝΑΤΟΤΗΤΕΣ	3
3.1 System	3
▪ 3.1.1 Date Format-----	3
▪ 3.1.2 Custom SQL -----	3
3.2 Event	3
▪ 3.2.1 OnConnection -----	4
3.3 Tasks	3
▪ 3.3.1 Παραδείγματα με Tasks-----	4
3.4 Selector	6
3.5 Forms	7
▪ 3.5.1 OnNewRecord -----	7
▪ 3.5.2 OnChangeField -----	7
▪ 3.5.3 UCL συναρτήσεις-----	8
▪ 3.5.4 Browser -----	10
▪ 3.5.5 StoredProcedures -----	13
3.5.5.1 OnBeforeUpdate	11
3.5.5.2 OnAfterUpdate	11
3.5.5.3 OnBeforeInsert.....	11
3.5.5.4 OnAfterInsert.....	11
3.5.5.5 OnBeforeDelete	11
3.5.5.6 OnInsertWarning	11
3.5.5.7 OnUpdateWarning	12
3.5.5.8 OnDeleteWarning	12
▪ 3.5.6 Commands -----	13
▪ 3.5.7 Required-----	13
▪ 3.5.8 Memory Tables-----	16
3.6 Menu	17

3.7 Crystal Reports..... 17

Εισαγωγή

Η γλώσσα UCL ενεργοποιεί κάποιες επιπρόσθετες δυνατότητες σε συγκεκριμένα σημεία της εφαρμογής Atlantis.

1.1 Επίπεδα χρηστών

Οι νέες αυτές δυνατότητες αποδίδονται σε τρία επίπεδα:

- σε όλους
- σε ομάδες χρηστών
- σε συγκεκριμένο χρήστη

Η προτεραιότητα απόδοσης των δυνατοτήτων αυξάνεται καθώς ο χρήστης προχωρεί από το επίπεδο Α έως το επίπεδο Γ. Δηλαδή, εάν αποδοθεί μία δυνατότητα κάποιου τύπου σε όλους τους χρήστες και μία άλλη δυνατότητα του ίδιου τύπου σε έναν συγκεκριμένο χρήστη, τότε η ενεργή δυνατότητα στον χρήστη θα είναι αυτή που έχει αποδοθεί σε αυτόν και όχι αυτή που έχει σχεδιαστεί για όλους τους χρήστες. Επίσης, πρέπει να επισημανθεί ότι οι δυνατότητες κάποιου επιπέδου κληρονομούνται στα χαμηλότερα επίπεδα. Δηλαδή, για κάποιο συγκεκριμένο χρήστη θα ισχύουν οι δυνατότητες που έχουν σχεδιαστεί για όλους τους χρήστες, για την ομάδα χρηστών στην οποία ανήκει και για τον ίδιο μόνο.

1.2 Φόρτωμα

Αρχικά, ο χρήστης επιλέγει σε ποιο επίπεδο θα αποδώσει νέες δυνατότητες και ανάλογα με την κατηγορία που έχει επιλέξει προσδιορίζει τίποτα, συγκεκριμένη ομάδα χρηστών ή συγκεκριμένο χρήστη αντίστοιχα. Στην συνέχεια επιλέγει από το menu **Commands**→**Φόρτωμα**.

Στο σημείο αυτό μπορούμε να δούμε τις δυνατότητες που έχουν ήδη καταχωρηθεί, καθώς και να αποδώσουμε νέες.

1.3 Γενικές Μεταβλητές

Οι μεταβλητές αυτές μπορούν να χρησιμοποιηθούν οπουδήποτε. Αυτές είναι:

Γενική Μεταβλητή	Σημασία
@COMPANYID	Κωδικός εταιρίας
@ BRANCHID	Υποκατάστημα με το οποίο έχει γίνει login
@USERID	Χρήστης με τον οποίο έχει γίνει login
@LOGINDATE	Ημερομηνία με την οποία έχει γίνει login
@FYEID	Το έτος (χρήση) της login ημερομηνίας
@FIPID	Η περίοδος της login ημερομηνίας

1.4 Οδηγίες ανάγνωσης των scripts στο παρόν έγγραφο

Κατά την περιγραφή των scripts, ότι είναι γραμμένο σε bold είναι υποχρεωτικό να γραφεί όπως ακριβώς φαίνεται. Επιπλέον, ότι είναι γραμμένο μέσα σε brackets [] είναι προαιρετικό, ενώ ότι είναι μέσα σε αγκύλες {} μπορεί να επαναληφθεί από μηδέν ή περισσότερες φορές.

Όλα τα scripts αποτελούνται από μία ή περισσότερες γραμμές. Οι γραμμές εισάγονται μία-μία. Κάθε γραμμή αποτελείται από δύο μέρη, το "κλειδί" και την "περιγραφή του κλειδιού". Στα παραδείγματα των scripts ανάμεσα στο κλειδί και την περιγραφή υπάρχει το σύμβολο '='. Αυτό το σύμβολο δεν εισάγεται από τον χρήστη αλλά από την εφαρμογή. Αν χρειάζεται να χρησιμοποιηθεί το ίδιο κλειδί περισσότερο από μία φορά, διαχωρίζεται με έναν αριθμό πχ.

ΚλειδίA,1 ΚλειδίA,2 ΚλειδίA,3 κτλ

Χειρισμός των Controls

- Button Εισαγωγή:** Για την εισαγωγή ενός νέου κλειδιού.
- Button Διόρθωση:** Για την διόρθωση / επαναπροσδιορισμό ενός επιλεγμένου κλειδιού.
- Button Διαγραφή:** Για τη διαγραφή ενός κλειδιού.
- Buttons ↑ και ↓ :** Για την μετακίνηση από κλειδί σε κλειδί.
- Βελάκι ↑ :** Για τη διαγραφή ή τη μετακίνηση σε Πρόσθετα Πεδία.
- Βελάκι ↓ :** Για τη δημιουργία ή τη μετακίνηση σε Πρόσθετα Πεδία.
- Ctrl + Delete :** Για τη διαγραφή ενός επιλεγμένου Πρόσθετου Πεδίου.

Δυνατότητες

Η UCL έχει χωριστεί σε διάφορες υποκατηγορίες οι οποίες αντιστοιχούν σε ποικίλες δομές και λειτουργικότητες της εφαρμογής Atlantis. Παρακάτω περιγράφονται αναλυτικά:

1.5 System

Date Format

Χρησιμοποιείται για να μπορεί να αλλαχθεί το format της ημερομηνίας του συστήματος.

Κλειδί	Script κλειδιού
VALUE	'format ημερομηνίας'

Πχ. VALUE='DD-MON-YYYY'

Custom SQL

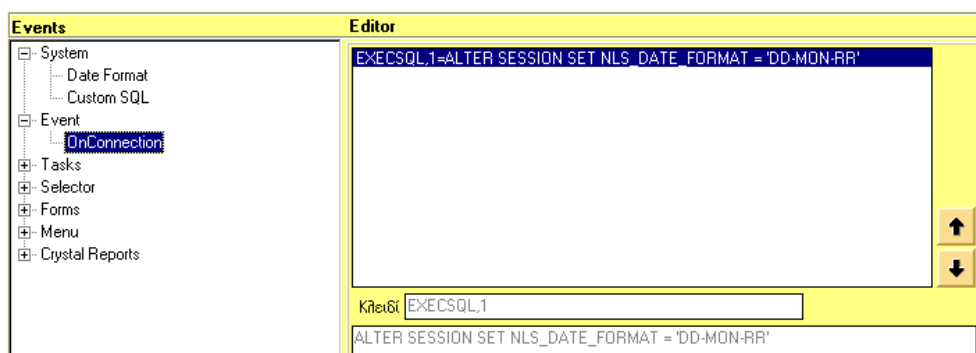
Αποκλειστικά για εσωτερική χρήση.

1.6 Event

Σε αυτό το μέρος μπορούν να εκτελεστούν διάφορα queries πριν ή μετά από κάποια events. Το μόνο event αυτή τη στιγμή που είναι διαθέσιμο είναι το On Connection.

On Connection

Εκτελείται κάποιο query κατά τη σύνδεση της εφαρμογής με τη βάση. Στο παράδειγμα της παρακάτω οθόνης το κλειδί δηλώνει μία κλήση σε sql ενώ στην περιγραφή του κλειδιού γράφουμε το sql statement όπως ακριβώς θα το γράφαμε σε ένα sql plus session.



1.7 Tasks

Με την δυνατότητα αυτή μπορεί ο χρήστης να ορίσει εργασίες που αποδίδονται σε νέες επιλογές menu. Ο ορισμός των νέων επιλογών menu, καθώς και η απόδοση των νέων εργασιών σε αυτές γίνεται με την δυνατότητα "menu" που θα περιγραφεί παρακάτω.

Με αυτήν την επιλογή η εφαρμογή μπορεί να καλέσει ένα:

- **Crystal Report**
- **exe** μίας άλλης εφαρμογής, πχ, Oracle developer (fmx files), Oracle reports (rep files), excel, word κτλ.
- **Import txt file**
- **DII** του Delphi (σε πολύ ειδικές περιπτώσεις και μετά από συνεννόηση με την Unisoft)

Παραδείγματα με Tasks

Παράδειγμα για Crystal

Για τις νέες εργασίες που αφορούν την εκτέλεση **crystal reports**, πρέπει να σημειωθεί ότι το προσχεδιασμένο report δεν πρέπει να έχει δεδομένα (τα crystal reports δίνουν την δυνατότητα δημιουργίας report files με δεδομένα). Ο χρήστης μπορεί να προσδιορίσει τιμές σε κάποιες από τις παραμέτρους των crystal reports. Οι τιμές που θα πάρουν αυτές οι παράμετροι επικαλύπτουν τις τιμές των παραμέτρων που πιθανόν να έχουν αποδοθεί με την δυνατότητα «crystal reports» της UCL που θα δούμε παρακάτω.

Περιγραφή script για την κλήση ενός Crystal Report

Κλειδί	Script κλειδιού
TYPE	CRYSTAL
FILENAME	Το όνομα του αρχείου crystal μαζί με το path
PARAMSTITLE	ο τίτλος που θα εμφανίζεται στην φόρμα των παραμέτρων
[{όνομα παραμέτρου crystal	0 ή 1}]
[{@όνομα παραμέτρου query crystal	edit; date ή string ή integer ή float; Πλάτος; Περιγραφή λεκτικού[;αρχική τιμή]}}

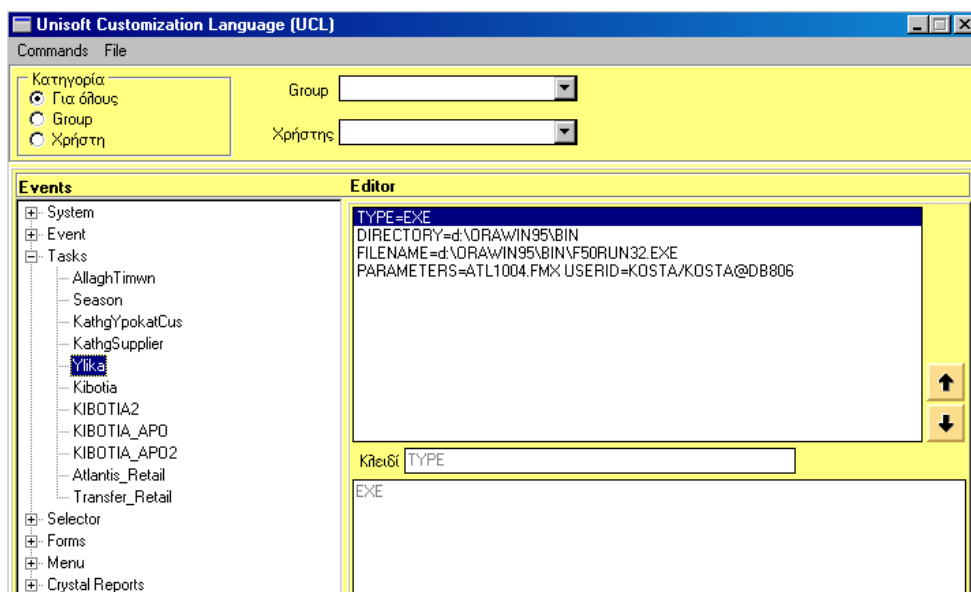
Στην περιγραφή αρχικών τιμών μπορούν να χρησιμοποιηθούν οι γενικές μεταβλητές.

Παράδειγμα για Oracle Developer

Μια πιο απλή προσέγγιση δίνεται για τις περιπτώσεις που θέλουμε να καλέσουμε Oracle Developer ή Oracle Reports

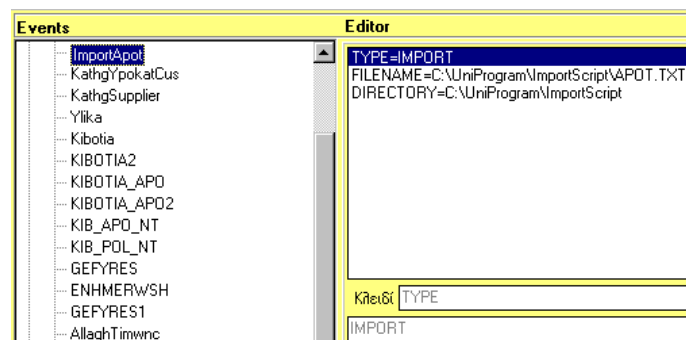
<i>Κλειδί</i>	<i>Script κλειδιού</i>
TYPE	EXE
DIRECTORY	Το path στο οποίο βρίσκεται το executable του Oracle Developer
FILENAME	Το path και το executable του Oracle Developer
PARAMETERS	Το fmx αρχείο που θα καλέσει ο Oracle Developer καθώς και το connection string με τη βάση
<p><i>Παρατηρείται ότι ουσιαστικά εδώ προσπαθεί ο χρήστης να προσομοιώσει την κλήση του Oracle Developer όπως θα έκανε αν δημιουργούσε ένα shortcut ή όπως θα καλούσε ένα αρχείο από το Command line πχ.</i></p> <p><i>D:\orawin95\bin\F50RUN32.exe Myfile.fmx USERID=mydbuser\mypassword@mydbalias</i></p>	

Με παρόμοιο τρόπο καλούνται και τα Oracle Reports:



Παράδειγμα για Import

<i>Κλειδί</i>	<i>Script κλειδιού</i>
TYPE	IMPORT
DIRECTORY	Path του txt file που περιγράφει το import
FILENAME	Path και Όνομα του txt file που περιγράφει το import



1.8 Selector

Με την δυνατότητα αυτή μπορεί ο χρήστης να αλλάξει την συμπεριφορά κάποιου από τους επιλογείς της εφαρμογής. (Οι επιλογείς χρησιμοποιούνται κατά την αναζήτηση πεδίων.) Συγκεκριμένα, μπορεί να εισαγάγει νέα φίλτρα ή/και νέες στήλες. Τα νέα φίλτρα εισάγονται στη θέση **Προτεινόμενη Επιλογή (where)** ή **Δεσμευτική Επιλογή (where)** και οι νέες στήλες στα **πρόσθετα πεδία**.

Η διαφορά των "**Προτεινόμενη Επιλογή (where)**" ή "**Δεσμευτική Επιλογή (where)**" είναι ότι η τελευταία επηρεάζει μόνιμα τα φίλτρα του επιλογέα, ενώ η προτεινόμενη μόνο την πρώτη φορά που ενεργοποιείται ο επιλογέας (δηλαδή με ανοιχτό τον επιλογέα, εάν ζητηθεί εκ νέου "ενεργοποίηση" των φίλτρων δεν εφαρμόζονται τα νέα φίλτρα).

Τα πρόσθετα πεδία προσθέτονται σε αυτά που έρχονται by default από το Atlantis και τα οποία μπορεί κανείς να βρει μέσω του sql monitor του Go server.

Πεδίο	Τίτλος	Πλάτος
KOSTA_EFORIA	Εφορία KOSTA	15
KOSTA_KATCODE	Κωδ. Κατηγορίας	3

Στο control της οθόνης, το οποίο αναπαριστά τον selector, το Atlantis, φέρνει by default δύο πεδία, τα οποία συνήθως είναι ο κωδικός και η περιγραφή. Αν θέλει ο χρήστης να το αλλάξει αυτό θα πρέπει να ορίσει τα **Πεδίο(1)** και **Πεδίο(2)**. Θα πρέπει όμως να επιλεγούν δύο πεδία unique ώστε να μην εμφανιστούν πανομοιότυπες γραμμές. Δίνεται επίσης η δυνατότητα να καθοριστεί το **πλάτος** του Πεδίου (1) σε χαρακτήρες.

Αν θέλει ο χρήστης να εμφανιστεί στον selector κάποιο πεδίο lookup ή κάποιο που δεν ανήκει στον πίνακα που αφορά ο selector, τότε θα πρέπει να δημιουργήσει ένα view με αυτό το πεδίο και να κάνει reference σε αυτό στη θέση **Εναλλακτικό Table/View**. Εάν είναι συμπληρωμένη η θέση αυτή, τότε ο selector διαβάζει από αυτό το table ή view. Όλα τα παραπάνω ισχύουν όπως έχουν αλλά για το Εναλλακτικό table/view.

Προσοχή!

Θα πρέπει στο Εναλλακτικό table/View να περιέχονται με τη σειρά και με τα ίδια ονόματα τα πεδία που έχουν οριστεί by default από το Atlantis για τον συγκεκριμένο selector.

1.9 Forms

Με την δυνατότητα αυτή μπορεί ο χρήστης να αλλάξει την συμπεριφορά φορμών της εφαρμογής. Η δυνατότητα αυτή περιορίζεται στις φόρμες που ελέγχουν master αρχεία και στις φόρμες των συναλλαγών. Οι νέες δυνατότητες διακρίνονται σε πέντε κατηγορίες:

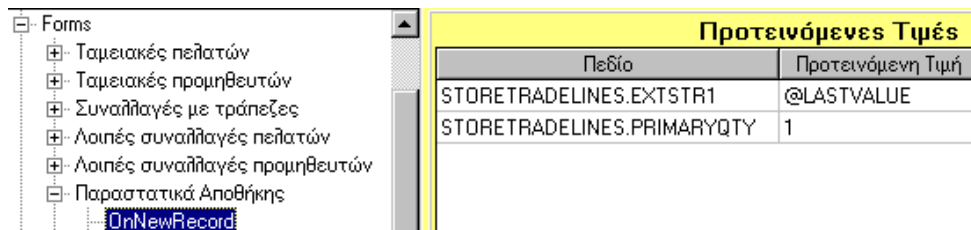
OnNewRecord

Με την δυνατότητα αυτή μπορεί ο χρήστης να δώσει αρχικές τιμές σε πεδία του master πίνακα που σχετίζεται με την φόρμα και μόνο στην εργασία "νέα εγγραφή". Για να δοθεί αρχική τιμή σε ένα πεδίο, αρκεί να χρησιμοποιηθεί το όνομα του πεδίου σαν κλειδί.

Περιγραφή script

Κλειδί	Script κλειδιού
{όνομα πίνακα βάσης, όνομα πεδίου	έκφραση της οποίας το αποτέλεσμα θα δοθεί στο πεδίο}

Στο script των κλειδιών μπορούν να χρησιμοποιηθούν οι γενικές μεταβλητές και επιπλέον η μεταβλητή **@LASTVALUE**, που εκφράζει την προηγούμενη τιμή που είχε το πεδίο σε προηγούμενη χρήση των εργασιών "νέα εγγραφή" ή "μεταβολή". Η μεταβλητή LASTVALUE παίρνει αρχική τιμή μόλις δοθεί για πρώτη φορά τιμή στο σχετικό πεδίο κλειδί.



OnChangeField

Με την δυνατότητα αυτή μπορεί ο χρήστης να δώσει τιμές σε πεδία των πινάκων που αναγνωρίζονται από την φόρμα εάν αλλάξει η τιμή κάποιου πεδίου. Για να πραγματοποιηθεί αυτό, αρκεί να χρησιμοποιηθεί σαν κλειδί το όνομα του πεδίου του οποίου ελέγχεται η αλλαγή τιμής.

- Στο τμήμα ONCHANGEFIELD μπορεί ο χρήστης να ορίσει μέσα σε αγκύλες {} διάφορες εντολές από την εσωτερική Language Script of Atlantis.

Περιγραφή script

Κλειδί	Script κλειδιού
{όνομα πίνακα βάσης,όνομα πεδίου	όνομα πίνακα βάσης,όνομα πεδίου= έκφραση της οποίας το αποτέλεσμα θα δοθεί στο πεδίο}

Στο script των κλειδιών μπορούν να χρησιμοποιηθεί κάποια από τις γενικές μεταβλητές καθώς και η μεταβλητή "LASTVALUE".

UCL συναρτήσεις

Στην έκφραση στο script κλειδιού των δυνατοτήτων *OnNewRecord* και *OnChangeField*, μπορούν να χρησιμοποιηθούν μερικές συναρτήσεις (όπου αναφέρεται πεδίο, μπορεί να χρησιμοποιηθεί οποιοδήποτε από τα διαθέσιμα πεδία της φόρμας και όχι μόνο των πινάκων που γίνονται editing από την φόρμα):

Σύνταξη συνάρτησης	Παράδειγμα
!DBFIELD('όνομα πίνακα βάσης','όνομα πεδίου')	CUSTOMER.MYEFORIA=!DBFIELD('CUSTOMER','DOYID')
<i>Φέρνει το πεδίο του συγκεκριμένου πίνακα της βάση, Αν θέλει ο χρήστης να πάρει την τιμή ενός πεδίου με το όνομα που αναγνωρίζεται από την φόρμα θα πρέπει να χρησιμοποιήσουμε την αντίστοιχη !FLD που συντάσσεται παρόμοια.</i>	

! έκφραση	MATERIAL.DESCR= !'THIS IS A '+'TEST'
<i>Φέρνει το αποτέλεσμα της έκφρασης</i>	

!FLDSQL('select πεδίο ..[where] {[πεδίοA:=1] and [πεδίοB := 2]}', {όνομα παραμέτρου 1, όνομα παραμέτρου 2})

Επιστέφει ένα πεδίο, ως αποτέλεσμα ενός SQL query. Το where clause μπορεί να δεχτεί παραμέτρους. Το join γίνεται με το σύμβολο αντικατάστασης := ακολουθούμενο από τον α/α της παραμέτρου. Τα ονόματα των παραμέτρων παρατίθενται σύμφωνα με τον α/α διαχωριζόμενα με κόμμα.

!SP('όνομα stored procedure', 'ονόματα IN παραμέτρων διαχωριζόμενα με ελληνικό ερωτηματικό', 'όνομα OUT παραμέτρου', τιμές παραμέτρων διαχωριζόμενες με κόμμα)

Επιστέφει ένα πεδίο, ως αποτέλεσμα μιας stored procedure η οποία μπορεί να δεχτεί παραμέτρους. Τα ονόματα των παραμέτρων πρέπει να είναι στα αγγλικά και χωρίς κενά, ενώ θα πρέπει να επιστρέφεται μόνο μία OUT παράμετρος από την stored procedure.

#=!ISP('όνομα stored procedure', 'ονόματα IN παραμέτρων διαχωριζόμενα με ελληνικό ερωτηματικό', 'τιμές παραμέτρων διαχωριζόμενες με κόμμα)

Ίδια όπως και η απλή SP, με τη διαφορά ότι δεν επιστρέφεται πίσω καμία τιμή. Η out παράμετρος εδώ αντικαθιστάται από το κενό. Είναι χρήσιμη όταν θέλουμε να εκτελείται μία stored procedure όταν αλλάζει η τιμή ενός πεδίου της φόρμας.

#=!SCROLLTABLE('όνομα πίνακα στη φόρμα', 'όνομα πίνακα στη βάση', 'όνομα detail πεδίου που καθορίζει το scrolling')

Αν ο χρήστης αναφέρεται σε μία διάταξη master detail και θέλουμε για μία αλλαγή ενός master πεδίου να αλλάξει (επανυπολογιστεί) η τιμή για μία στήλη στο detail και όχι μόνο η πρώτη γραμμή του, τότε χρησιμοποιούμε τη SCROLLTABLE.

Στο παράδειγμά υπάρχουν τα παραστατικά πωλήσεων (fintrade) και τις γραμμές τους (storedtradelines). Στην 4^η και 5^η γραμμή του script δηλώνεται ότι κάθε φορά που θα αλλάξει το STORETRADELINES.ITEID θα υπολογίζεται η αξία(storedtradelines.price) και η έκπτωση (storedtradelines .prcdesc1) της 1^{ης} γραμμής του παραστατικού από δύο stored procedures. Για να επαναληφθεί αυτή η ενημέρωση για όλες τις γραμμές του detail χρησιμοποιούμε τη SCROLLTABLE.στην 8^η γραμμή του script όπου ζητάμε για κάθε αλλαγή στον κωδικό πελάτη να εκτελούνται οι γραμμές που αναφέρονται στο ITEID δηλαδή τις γραμμές 4 και 5 του script.

!UPPERCASE('πεδίο') ACCOUNT.FOREIGNDESCR=!UPPERCASE(DBFIELD('ACCOUNT','FOREIGNDESCR'))
*Επιστρέφει το πεδίο σε upper case. Αντίστοιχα συντάσσεται και η **!LOWERCASE***

#=!FieldChange('όνομα πίνακα', 'όνομα πεδίου')

Χρησιμοποιείται όταν θέλουμε να εκτελείται ένα OnchangeField script με την αλλαγή περισσότερων του ενός πεδίων της φόρμας. Είναι χρήσιμο σε περιπτώσεις upgrade, όπου ο customizer δεν θέλει να επαναλαμβάνει το ίδιο script για διαφορετικά κλειδιά. Διαλέγουμε ένα κλειδί σαν βάση, και στα επόμενα κάνουμε αναφορές σε αυτό. Στο παράδειγμα, όταν αλλάζει η χώρα, καλείται το ίδιο script με αυτό που καλείται όταν αλλάζει η υποκατηγορία του πελάτη.

Προσοχή:

- Θα πρέπει να χρησιμοποιείται μόνο ένα ! στο script του κλειδιού. (Το σύμβολο αυτό ξεκινάει το parsing)
- Το # σημαίνει ότι δεν επιστρέφεται καμία τιμή, απλά εκτελείται αυτό που ακολουθεί.

Browser

Με την δυνατότητα αυτή μπορεί ο χρήστης να αλλάξει την συμπεριφορά των browsers της εφαρμογής. Συγκεκριμένα, μπορεί να εισάγει νέα φίλτρα, ή/και νέες στήλες ή ακόμη να αλλάξει μερικώς το σχετικό SQL. Τα νέα φίλτρα εισάγονται στη θέση **Προτεινόμενη επιλογή where** ή **Δεσμευτική επιλογή where** και οι νέες στήλες στη θέση **Πεδία**. Η διαφορά μεταξύ Προτεινόμενη επιλογή where και Δεσμευτική επιλογή where είναι ότι το τελευταίο επηρεάζει μόνιμα τα φίλτρα του browser, ενώ το πρώτο μόνο την πρώτη φορά που θα ανοίξει η φόρμα.

Πρέπει κανείς να προσέξει τα εξής:

- Στο **Εναλλακτικό sql statement** θα πρέπει οπωσδήποτε να περιέχονται τα default πεδία που ορίζονται για τον browser από το Atlantis. Κατά αυτόν τον τρόπο μπορεί ο χρήστης μόνο να "αυξήσει" το query και όχι να το "μειώσει". Τα default πεδία φαίνονται από το sql monitor του Goserver
- Τα πεδία στο εναλλακτικό sql statement θα πρέπει να έχουν μοναδικά ονόματα ή aliases.
- Όπου γίνεται αναφορά σε πεδία εκτός από το εναλλακτικό sql statement, δεν θα πρέπει να χρησιμοποιείται το όνομα του table που τα αφορά.
- Κάθε αλλαγή στον browser επιβραδύνει την διαδικασία εμφάνισής του στην οθόνη.
- Μπορεί ο χρήστης να "εξαφανίσει" πεδία από το browser στην οθόνη, αν στη λίστα του πατήσει δεξί κλικ και επιλέξει "Καθορισμός στηλών".

Πεδίο	Τίτλος	Πλάτος
#SMNCODE	ΚΩΔΙΚΟΣ Π.	100

ΣΗΜΕΙΩΣΕΙΣ:

Εάν στα "πρόσθετα πεδία" ο χρήστης βάλει πεδίο που υπάρχει στο "εναλλακτικό Sql" τότε πρέπει πριν από αυτό να βάλουμε τον χαρακτήρα # (βάζοντας αυτόν τον χαρακτήρα δεν θα καλέσει για δεύτερη φορά το πεδίο).



ΠΕΡΙΟΡΙΣΜΟΣ

Δεν μπορεί να εισαχθεί πεδίο στο τμήμα "πρόσθετα πεδία" εάν για αυτό απαιτείται join στο αρχικό query του browser. π.χ.: Εάν το εναλλακτικό SQL είναι `SELECT A.*,SMN.CODE FROM FULLLCUSTSTOTRADE A, SALESMAN SMN WHERE COLIDSALESMAN=SMN.ID(+)` τότε δεν μπορεί το πεδίο SMNCODE να καταχωρηθεί σαν πρόσθετο πεδίο.

1.9.1.1 OnBeforeUpdate

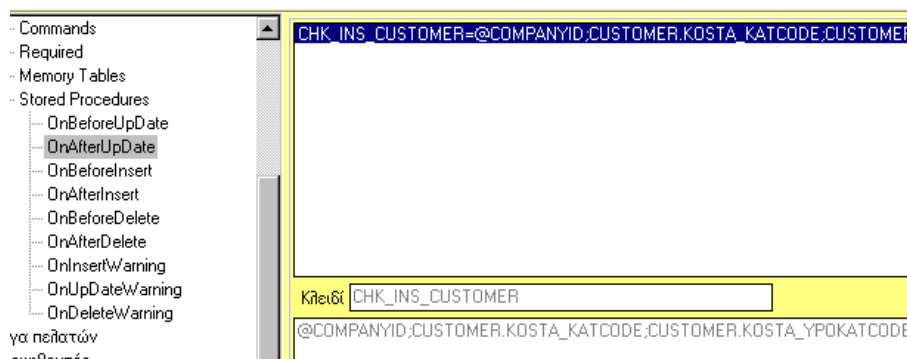
Με την δυνατότητα αυτή μπορεί ο χρήστης να εκτελέσει μία stored procedure πριν ξεκινήσει του transaction της φόρμας, κατά την εργασία τροποποίησης. Για το σκοπό αυτό αρκεί να χρησιμοποιηθεί το όνομα της stored procedure σαν κλειδί. Στο script του κλειδιού περιγράφονται οι παράμετροι της stored procedure.

Περιγραφή script

Κλειδί	Script κλειδιού
όνομα stored procedure	[όνομα πίνακα.όνομα πεδίου {;όνομα πίνακα.όνομα πεδίου}]

1.9.1.2 OnAfterUpdate

Με την δυνατότητα αυτή μπορεί ο χρήστης να εκτελέσει μία stored procedure πριν το commit του transaction, κατά την εργασία τροποποίησης.



Όπου CHK_INS_CUSTOMER είναι ως εξής

```
(P_COMID IN NUMBER,
                                     P_KATHG IN NUMBER,
                                     P_YPOKAT IN NUMBER
) AS
begin
  IF SUBSTR(P_YPOKAT,1,LENGTH(P_KATHG)) != P_KATHG THEN
    raise_APPLICATION_ERROR(-20000,'Η κατηγορία δεν συμφωνεί με την υποκατηγορία!');
  END IF;
END;
```

1.9.1.3 OnBeforeInsert

Με την δυνατότητα αυτή μπορεί ο χρήστης να εκτελέσει μία stored procedure πριν ξεκινήσει το transaction, κατά την εργασία εισαγωγής.

1.9.1.4 OnAfterInsert

Με την δυνατότητα αυτή μπορεί ο χρήστης να εκτελέσει μία stored procedure πριν το commit του transaction, κατά την εργασία εισαγωγής.

1.9.1.5 OnBeforeDelete

Με την δυνατότητα αυτή μπορεί ο χρήστης να εκτελέσει μία stored procedure πριν ξεκινήσει το transaction, στην διαδικασία διαγραφής.

1.9.1.6 OnInsertWarning

Με την δυνατότητα αυτή μπορεί ο χρήστης να εμφανίσει κάποιο μήνυμα (μέσω κάποιας stored procedure) πριν το commit του transaction, κατά την διαδικασία εισαγωγής.

1.9.1.7 OnUpdateWarning

Με την δυνατότητα αυτή μπορεί ο χρήστης να εμφανίσει κάποιο μήνυμα (μέσω κάποιας stored procedure) πριν το commit του transaction, κατά την διαδικασία τροποποίησης.

1.9.1.8 OnDeleteWarning

Με την δυνατότητα αυτή μπορεί ο χρήστης να εμφανίσει κάποιο μήνυμα (μέσω κάποιας stored procedure) πριν το commit του transaction, κατά την διαδικασία διαγραφής.

- Πρέπει να σημειωθεί ότι από εάν κάποια stored procedure προκαλέσει exception, τότε προκαλείται **διακοπή** του transaction εκτός από τις τρεις τελευταίες περιπτώσεις που χρησιμοποιούνται για την εμφάνιση μηνυμάτων.
- Στα Warnings, το μήνυμα που θα εμφανιστεί πρέπει να δηλωθεί στην καλούμενη stored procedure σαν μια επιπλέον παράμετρος (στο τέλος) τύπου string. Όμως στο UCL script του κλειδιού ΔΕΝ δηλώνουμε OUT παράμετρο.

Παραδείγματα

Παράδειγμα stored procedure (σε Oracle) που μπορεί να χρησιμοποιηθεί στην δυνατότητα StoredProcedures-OnInsertWarning

```
/*Η παρακάτω stored procedure κάνει έλεγχο υπολοίπου σε γραμμή παραστατικού

PROCEDURE CHECKBATCHBALANCE1 (COMPANYID IN NUMBER, BRANCHID IN NUMBER,
FINTRADEID IN NUMBER, INFOTEXT OUT VARCHAR2) AS

CURSOR FINLINES (CCOMPANYID NUMBER, CFINTRADEID NUMBER) IS
SELECT STL.COMID, STL.FTRID, STL.PARID, STL.PRIMARYQTY,
ESTQTY1, INPUTQTY1, OUTPUTQTY1, INITQTY1, PARTITION.CODE
FROM STORETRADELINES STL, PARTITION
WHERE STL.COMID = CCOMPANYID
AND STL.FTRID = CFINTRADEID
AND STL.PARID = PARTITION.ID;

CHECKLIMIT NUMBER := 3000; /* 3000 LIT */
MYCOUNTER NUMBER := 0;
BEGIN
INFOTEXT:='';
FOR ee IN FINLINES (COMPANYID, FINTRADEID)
LOOP
IF ( ( ee.InitQty1 + ee.InputQty1) - ee.OutputQty1 ) < CHECKLIMIT /*ee.PRIMARYQTY*/ )
THEN
BEGIN
MYCOUNTER := MYCOUNTER + 1;
IF MYCOUNTER < 5 THEN /* MAXIMUM 4 MESSAGES */
INFOTEXT := INFOTEXT || 'Υπόλ.Διασάφισης ' || ee.CODE || ' (' || TO_CHAR(ee.InitQty1 +
ee.InputQty1 - ee.OutputQty1) || ') < Ποσότητα.γραμμής' ||CHR(13);
END IF;
END;
END IF;
END LOOP;
/*IF MYCOUNTER > 4 THEN
INFOTEXT := INFOTEXT || ' (Σ.ΜΗΝ.= ' || TO_CHAR(MYCOUNTER) || ')';
END IF;*/

END;
```

Παράδειγμα stored procedure (σε SQL-Server) που μπορεί να χρησιμοποιηθεί στην δυνατότητα StoredProcedures- OnAfterUpdate

```
/*Η παρακάτω stored procedure κάνει έλεγχο στην ημερομηνία παραστατικού και ενεργοποιεί ή όχι exception */

CREATE PROCEDURE CHECKDOC
@ID1 INTEGER
AS
DECLARE @ARES1 INTEGER

SELECT @ARES1=DATEDIFF (DAY, FTRDATE, GETDATE())
FROM FINTRADE WHERE ID=@ID1
IF ABS(@ARES1)>5
RAISERROR('Data documentului este in afara datelor limita!',16,-1)
```

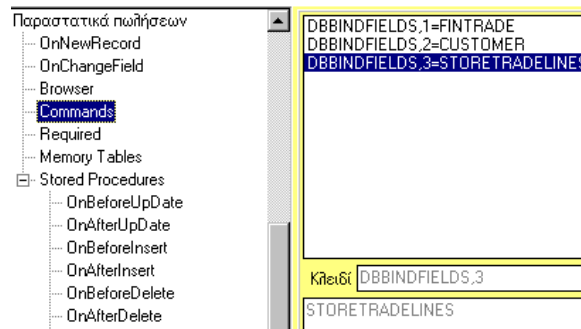
Commands

Είναι κυρίως για εσωτερική χρήση. Παρόλα αυτά όταν θέλουμε να γίνονται αναφορές σε πεδία της βάσης και να διαβάζουμε από τη βάση και όχι από τα πεδία της φόρμας τότε μπορούμε να δώσουμε:

Περιγραφή script

Κλειδί	Script κλειδιού
DBBINDFIELDS ,n (όπου n αριθμός)	όνομα πίνακα
DYNAMICFILTER ,n (όπου n αριθμός)	Τύπος, Περιγραφή, subquery, Πεδίο

Παράδειγμα

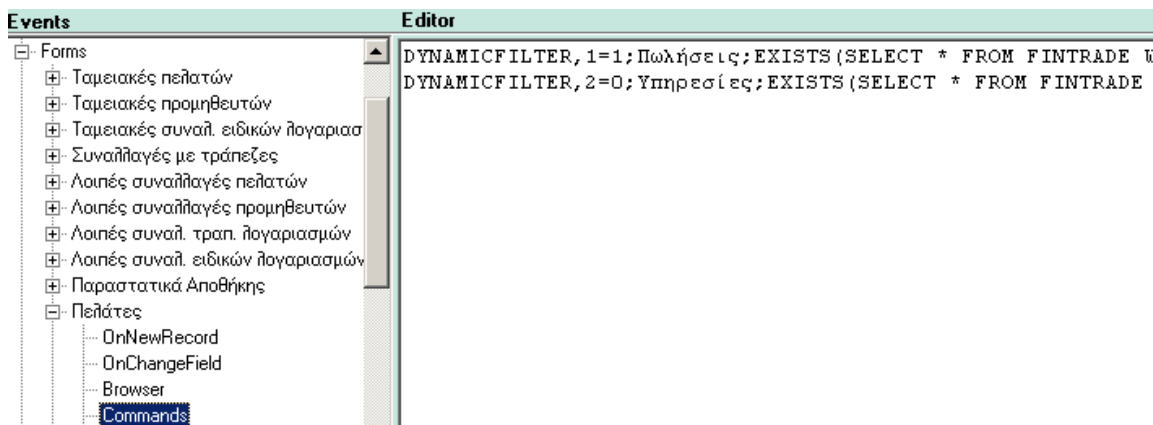


DYNAMICFILTER

Η εντολή αυτή επιτρέπει στο χρήστη να δημιουργεί δυναμικά πρόσθετα φίλτρα τα οποία θα χρησιμοποιεί στη συνέχεια για την αναζήτηση εγγραφών. Ουσιαστικά, με τη παραπάνω εντολή έχουμε τη δυνατότητα καθορισμού subquery, που θα συμμετέχει στην αναζήτηση των επιθυμητών εγγραφών.

1.10 Παράδειγμα

Στο παρακάτω παράδειγμα έχουν δημιουργηθεί δύο δυναμικά πρόσθετα φίλτρα στη φόρμα αναζήτησης εγγραφών πελατών.



1. **DYNAMICFILTER,1=1;Πωλήσεις;EXISTS(SELECT * FROM FINTRADE WHERE FINTRADE.SOURCE=5 AND FINTRADE.CUSID=CUSTOMER.ID AND FINTRADE.CUSID IN (%s)).**

The screenshot shows a software window titled "Διαχείριση πελατών [Αναζήτηση]". It is divided into two main sections: "Κριτήρια" (Criteria) and "Ταξινόμηση" (Sorting).
In the "Κριτήρια" section, there is a list of available fields ("Διαθέσιμα Πεδία") including ΑΦΜ, Διεύθυνση, ΤΚ, Τηλέφωνο, Υποκ/μα, Ενεργός, Κατηγορία ABC Analysis, Πωλήσεις (highlighted), and Υπηρεσίες. To the right is a text input field labeled "Λίστα (Διαχωριστικό ',')".
In the "Ταξινόμηση" section, there is another list of available fields ("Διαθέσιμα Πεδία") including Επωνυμία, ΑΦΜ, Διεύθυνση, ΤΚ, Τηλέφωνο, and Υποκ/μα. To the right is a list of sorting options ("Ταξινόμησης") with a checked checkbox for "Αύξουσα" (Ascending) and "Κωδικός" (Code) selected. There are also up and down arrow buttons next to the sorting list.
At the bottom of the window are two buttons: "Εφαρμογή" (Apply) and "Ακύρωση" (Cancel).

Η εκτέλεση του παραπάνω subquery, θα έχει σαν αποτέλεσμα την αναζήτηση στοιχείων των πελατών από τα παραστατικά πωλήσεων όπου:

- DYNAMICFILTER,1=1; έχει σαν αποτέλεσμα την εμφάνιση του παρακάτω πεδίου στο οποίο μπορώ να δώσω τιμές π.χ (1,2,3,,,100)

A screenshot of the text input field from the previous image, showing the label "Λίστα (Διαχωριστικό ',')" and an empty text box below it.

- Πωλήσεις: είναι το λεκτικό με το οποίο θα εμφανίζετε το φίλτρο στη λίστα με τα διαθέσιμα πεδία.

2. **DYNAMICFILTER,2=0;Υπηρεσίες;EXISTS(SELECT * FROM FINTRADE WHERE FINTRADE.SOURCE=16 AND FINTRADE.CUSID=CUSTOMER.ID);CUSTOMER.CODE**

Διαχείριση πελατών [Αναζήτηση]

Κριτήρια

Διαθέσιμα Πεδία

- ΑΦΜ
- Διεύθυνση
- TK
- Τηλέφωνο
- Υποκ/μα
- Ενεργός
- Κατηγορία ABC Analysis
- Πωλήσεις
- Υπηρεσίες**

Ισο/Μάσκα

Μεγαλύτερο (=)

Μικρότερο (=)

Ταξινόμηση

Διαθέσιμα Πεδία

- Επωνυμία**
- ΑΦΜ
- Διεύθυνση
- TK
- Τηλέφωνο
- Υποκ/μα

Ταξινόμησης (☑ = Αύξουσα)

- Κωδικός

Εφαρμογή Ακύρωση

Η εκτέλεση του παραπάνω subquery θα έχει σαν αποτέλεσμα την αναζήτηση στοιχείων των πελατών από τα παραστατικά παροχής υπηρεσιών όπου:

- DYNAMICFILTER,2=0; έχει σαν αποτέλεσμα την εμφάνιση φόρμας στην οποία πρέπει, να δώσουμε το κωδικό βάση του οποίου θα γίνει η επιλογή.

Ισο/Μάσκα

Μεγαλύτερο (=)

Μικρότερο (=)

- Υπηρεσίες: είναι το λεκτικό με το οποίο θα εμφανίζετε το φίλτρο στη λίστα με τα διαθέσιμα πεδία.

Παρακάτω βλέπουμε πως έχει διαμορφωθεί το query, που εκτελείτε για την αναζήτηση των επιθυμητών εγγραφών.

```
SELECT COMID, ID, CODE, NAME, AFM, STREET1, ZIPCODE1, PHONE11,CURID, BRAID,
ISACTIVE,
          ABCDDID
FROM
CUSTOMER
WHERE
ROWNUM<=50
AND
(COMID=:wrm1)
AND
*
EXISTS(SELECT
FROM
FINTRADE
WHERE
CUSTOMER.CODE LIKE '10%'
AND
FINTRADE.SOURCE=16
AND
FINTRADE.CUSID=CUSTOMER.ID)
ORDER BY COMID,CODE
```

Required

Με την δυνατότητα αυτή μπορεί ο χρήστης να ζητήσει την υποχρεωτική (required) συμπλήρωση κάποιου συγκεκριμένου πεδίου που ανήκει στους πίνακες που ελέγχει η φόρμα. Για να πραγματοποιηθεί αυτό, αρκεί στο script του κλειδιού να χρησιμοποιηθεί το όνομα του πεδίου του οποίου η συμπλήρωση έχει ζητηθεί να γίνει υποχρεωτική. Σαν κλειδί πρέπει να χρησιμοποιηθεί ένα μοναδικό αναγνωριστικό όπως περιγράφεται παρακάτω.

Περιγραφή script

Κλειδί	Script κλειδιού
FIELD[,number]	όνομα πίνακα βάσης, όνομα πεδίου, λεκτικό που εμφανίζεται εάν δεν συμπληρωθεί το πεδίο}

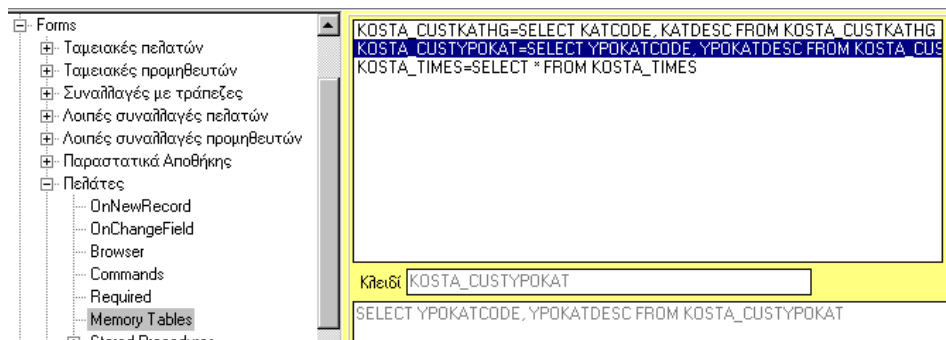
Παράδειγμα

FIELD,1= FINTRADE.PTRID, Πρέπει να συμπληρώσετε τον τρόπο πληρωμής

Memory Tables

Χρησιμοποιούνται όταν από μία φόρμα θέλει ο χρήστης να εμφανίσει custom πεδία είτε από custom πίνακες, είτε custom πεδία που κάνουν Lookup σε custom πίνακες. Εδώ, οι custom πίνακες φορτώνονται στη μνήμη κάθε φορά που ανοίγει η φόρμα στην οποία έχουν δηλωθεί.

*Δυνατότητα για Update και χειρισμό αυτών των tables δεν έχει ακόμη δοθεί. Μπορεί πάντως έμμεσα από την χρήση των tasks να πραγματοποιηθεί μέσω κάποιου άλλου exe.

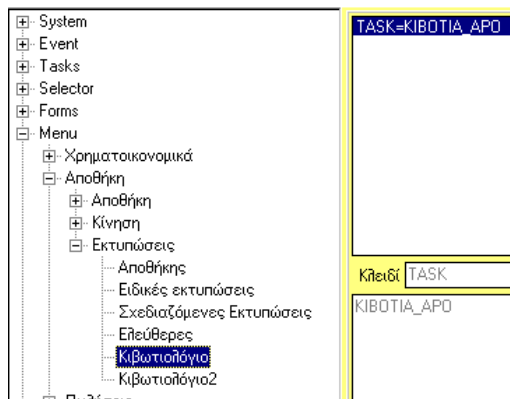


Περιγραφή script

Κλειδί	Script κλειδιού
Όνομα του custom πίνακα	Ένα select statement, το οποίο προσδιορίζει τα πεδία που θα φορτωθούν στη μνήμη από τον custom πίνακα

1.11 Menu

Με την δυνατότητα αυτή μπορεί ο χρήστης να δημιουργήσει (δεξί κλικ + create) νέες επιλογές στο menu της εφαρμογής και να αποδώσει σε αυτές κάποιες εργασίες (δυνατότητα «tasks»). Η απόδοση των εργασιών γίνεται με το κλειδί "task". Ο χρήστης μπορεί να προσθέσει επιλογές στο menu σε επίπεδο το πολύ δύο.



Περιγραφή script

Κλειδί	Script κλειδιού
TASK	το όνομα του task όπως έχει δοθεί μέσω της δυνατότητας «task»

1.12 Crystal Reports

Με την δυνατότητα αυτή μπορεί ο χρήστης να προσδιορίσει με ποια βάση δεδομένων θα συνδεθεί η εφαρμογή των crystal reports, καθώς και να δώσει τιμές σε κάποιες από τις παραμέτρους των crystal reports.

Περιγραφή script

Κλειδί	Script κλειδιού
SERVER NAME	το alias name του server της database
USERNAME	το όνομα χρήστη της database
PASSWORD	το password της database
DATABASE NAME (μόνο για SQL-Server)	το όνομα της database του SQL-Server
[{όνομα παραμέτρου crystal	0 ή 1}]

Παράδειγμα

```
SERVER NAME=JETSERVER.WORLD
USER NAME=JETOIL
PASSWORD=JETOIL
AllowDrillDown=1
PrintSetupBtn=1
RefreshBtn=1
SearchBtn=1
GroupTree=1
```