

Εφαρμογές Πληροφορικής & Νέες Τεχνολογίες Στη Γεωργία

ΛΥΣΗ ΕΡΓΑΣΙΑΣ
ΑΝΑΠΤΥΞΗ ΕΡΩΤΗΜΑΤΩΝ

ΕΡΩΤΗΜΑ 1 (σχεδίαση)

Να δημιουργήσετε ερώτημα το οποίο να εμφανίζει τα περιεχόμενα του πίνακα έξοδα καθώς και το άθροισμα των ποσών που έχουμε δαπανήσει για κάθε περιγραφή εξόδου.

Ερώτημα2

The diagram shows two tables: Δαπάνες and Εξοδα. Δαπάνες has fields: κωδικος_καλλιεργειας, κωδικος_εξοδων, ημερομηνια, ποσό. Εξοδα has fields: κωδικος, περιγραφη. A relationship line connects them with cardinalities ∞ and 1.

Πεδίο:	κωδικος	περιγραφη	Άθροισμα: ποσό		
Συγκεντρωτικά στοιχεία:	Ομαδοποίηση κατά	Ομαδοποίηση κατά	Άθροισμα		
Ταξινόμηση:					
Εμφάνιση:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Κριτήρια:					
ή:					

ΕΡΩΤΗΜΑ 1 (Εκτέλεση)

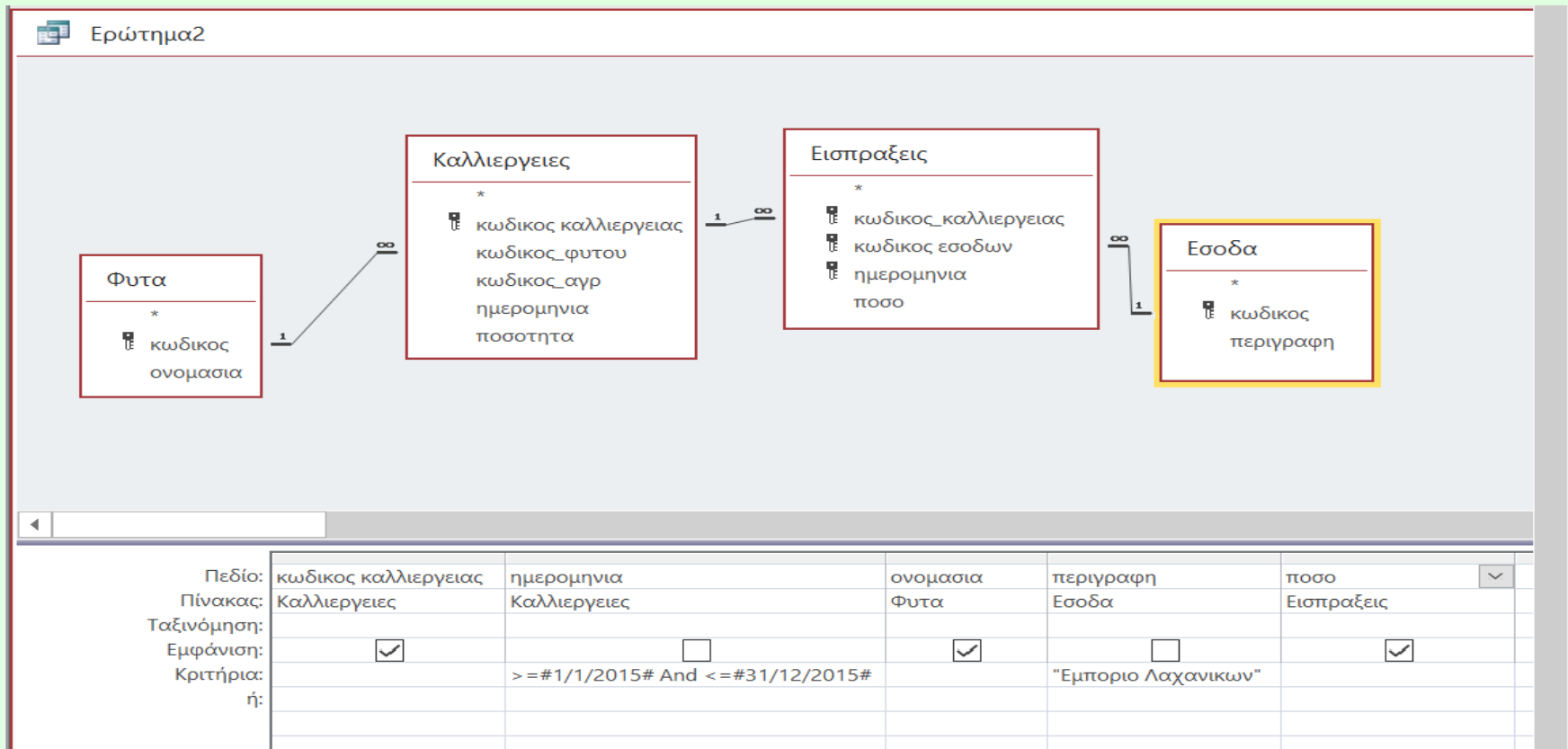


Ερώτημα

	κωδικος ▾	περιγραφη ▾	Αθροισμα ▾	
	1	Φυτοφαρμακα	1100	
	3	Ρευμα	530	
	4	Εγκαταστασεις	489	
	5	Φυτα	2569	
	8	Καυσιμα	599	

ΕΡΩΤΗΜΑ 2 (σχεδίαση)

Να δημιουργήσετε ερώτημα το οποίο να εμφανίζει το κωδικό των καλλιεργειών, τις ονομασίες των φυτών και τα ποσά για τις καλλιέργειες έτους 2015 με περιγραφή εσόδων Εμποριο Λαχανικων



ΕΡΩΤΗΜΑ 2 (Εκτέλεση)



Ερώτημα2

κωδικος καλί	ονομασια	ποσο	
10	Σπανακι	50	
22	Φασολακια	200	
22	Φασολακια	456	
10	Σπανακι	260	
*			

ΕΡΩΤΗΜΑ 3 (σχεδίαση)

Να δημιουργήσετε ερώτημα στο οποίο θα χρησιμοποιήσετε τους πίνακες Αγροκτήματα, Καλλιέργειες φυτά. Το ερώτημα θα εμφανίζει τα στοιχεία των αγροκτημάτων και τις ονομασίες φυτών που έχουν καλλιεργηθεί σε κάθε αγρόκτημα.

Ερώτημα2

```
graph LR; A[Αγροκτηματα] ---|1-∞| B[Καλλιέργειες]; B ---|∞-1| C[Φυτα];
```

The diagram shows three tables: Αγροκτηματα, Καλλιέργειες, and Φυτα. Αγροκτηματα has fields: κωδικος, περιοχη, οικισμος. Καλλιέργειες has fields: κωδικος καλλιερειας, κωδικος_φυτου, κωδικος_αγρ, ημερομηνια, ποσοτητα. Φυτα has fields: κωδικος, ονομασια.

Πεδίο:	κωδικος	περιοχη	οικισμος	ονομασια		
Πίνακας:	Αγροκτηματα	Αγροκτηματα	Αγροκτηματα	Φυτα		
Ταξινόμηση:						
Εμφάνιση:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Κριτήρια:						
ή:						

ΕΡΩΤΗΜΑ 3 (Εκτέλεση)

Ερώτημα

κωδικος	περιοχη	οικισμος	ονομασια
3	κορακοβούνι	Ηράκλειο	Πατατες
3	κορακοβούνι	Ηράκλειο	Κρεμμυδια
5	Πλατανιας	Ρεθυμνο	Πατατες
8	Αδελιανος Καμπος	Ρεθυμνο	Κρεμμυδια
3	κορακοβούνι	Ηράκλειο	Ντοματα
7	Αγάκου μετόχι	Ηράκλειο	Λαχανα
4	Κνωσσός	Ηράκλειο	Λαχανα
2	Μέση	Μαρουλας	Πιπερια
8	Αδελιανος Καμπος	Ρεθυμνο	Πιπερια
7	Αγάκου μετόχι	Ηράκλειο	Σπανακι
7	Αγάκου μετόχι	Ηράκλειο	Σπανακι
6	Πριναρές	Μαρουλάς	Σπανακι
3	κορακοβούνι	Ηράκλειο	Φασολακια
8	Αδελιανος Καμπος	Ρεθυμνο	Πατατες
2	Μέση	Μαρουλας	Φασολακια
5	Πλατανιας	Ρεθυμνο	Ντοματα
1	Μεζάρια	Μαρουλάς	Ντοματα
*			

ΕΡΩΤΗΜΑ 4

Να δημιουργήσετε το ίδιο ερώτημα με το προηγούμενο, αλλά χωρίς τον πίνακα καλλιέργειες. Υπάρχει διαφορά μεταξύ των αποτελεσμάτων που εμφανίζουν τα 2 ερωτήματα? Και αν ναι, μπορείτε να εξηγήσετε που οφείλετε?

ΕΡΓ_ΕΡΩΤ_ΛΥΣΗ_4

Αγροκτηματα

*

- κωδικος
- περιοχη
- οικισμος

Φυτα

*

- κωδικος
- ονομασια

Πεδίο:	κωδικος	περιοχη	οικισμος	ονομασια	
Πίνακας:	Αγροκτηματα	Αγροκτηματα	Αγροκτηματα	Φυτα	
Ταξινόμηση:					
Εμφάνιση:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Κριτήρια:					
ή:					

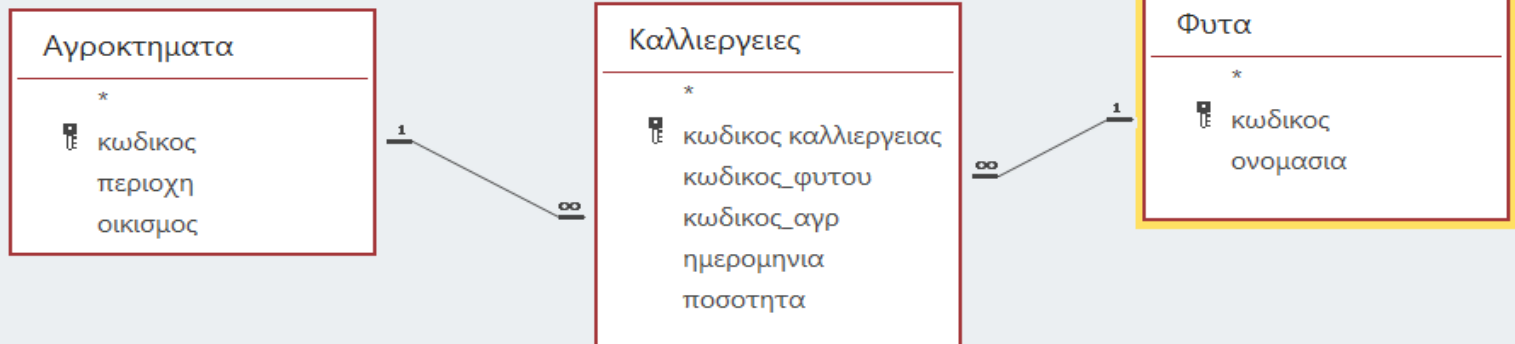
Κλείδωμα κερ

Στο ερώτημα 3 οι πίνακες Αγροκτήματα, Φυτά συνδέονται μέσω του πίνακα καλλιέργειες. Αφαιρώντας τον πίνακα καλλιέργειες οι πίνακες Αγροκτήματα, Φυτά δεν συνδέονται. Άρα δημιουργείται καρτεσιανό γινόμενο Αγροκτήματα X Φυτά

ΕΡΩΤΗΜΑ 5 (σχεδίαση)

Μετατρέψτε το ερώτημα 3 να λειτουργεί με παράμετρο για την περιοχή του αγροκτήματος και την ονομασία του φυτού

ΕΡΓ_ΕΡΩΤ_ΛΥΣΗ_5



Πεδίο:	κωδικος	περιοχη	οικισμος	ονομασια	
Πίνακας:	Αγροκτηματα	Αγροκτηματα	Αγροκτηματα	Φυτα	
Ταξινόμηση:					
Εμφάνιση:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Κριτήρια:		[ΔΩΣΕ ΠΕΡΙΟΧΗ]		[ΔΩΣΕ ΟΝΟΜΑΣΙΑ ΦΥΤΟΥ]	
ή:					

Εφαρμογές Πληροφορικής & Νέες Τεχνολογίες Στη Γεωργία

POSTGRESQL.
Γλώσσα Ορισμού Δεδομένων.
Ερωτήματα Επιλογής.

POSTGRESQL

Η PostgreSQL είναι ένα αντικείμενο-σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων (object relational database management system - ORDBMS) και έχει τα παρακάτω χαρακτηριστικά :

- Λογισμικό ανοικτού κώδικα.
- Τρέχει σε όλα τα βασικά λειτουργικά συστήματα (unix/linux windows macos).
- Προσφέρει
 - Ειδικούς τύπους δεδομένων για την αποθήκευση απλών γεωμετρικών οντοτήτων,
 - Μηχανισμούς δεικτοδότησης χωρικών δεδομένων όπως τετραδικά και R-δένδρα.
 - Περιορισμένο αριθμό γεωμετρικών τελεστών και συναρτήσεων για την διατύπωση χωρικών ή συνδυασμένων ερωτημάτων.
- Η PostgreSQL επιτρέπει τη συνεργασία με πληθώρα επιπρόσθετων σπονδύλων λογισμικού (addons) μέσω των οποίων γίνεται η εκτέλεση συγκεκριμένων και εξειδικευμένων λειτουργιών.
- Η επέκταση χωρικών λειτουργιών επιτυγχάνεται σε συνεργασία με το λογισμικό ανοικτού κώδικα **PostGIS** το οποίο ακολουθεί το πρότυπο OGC για τον ορισμό γεωγραφικών στοιχείων σε περιβάλλον SQL.
- Η PostgreSQL με την επέκταση λειτουργιών PostGIS αποτελεί ένα σύστημα κατάλληλο για ανάπτυξη και διαχείριση χωρικών βάσεων δεδομένων.

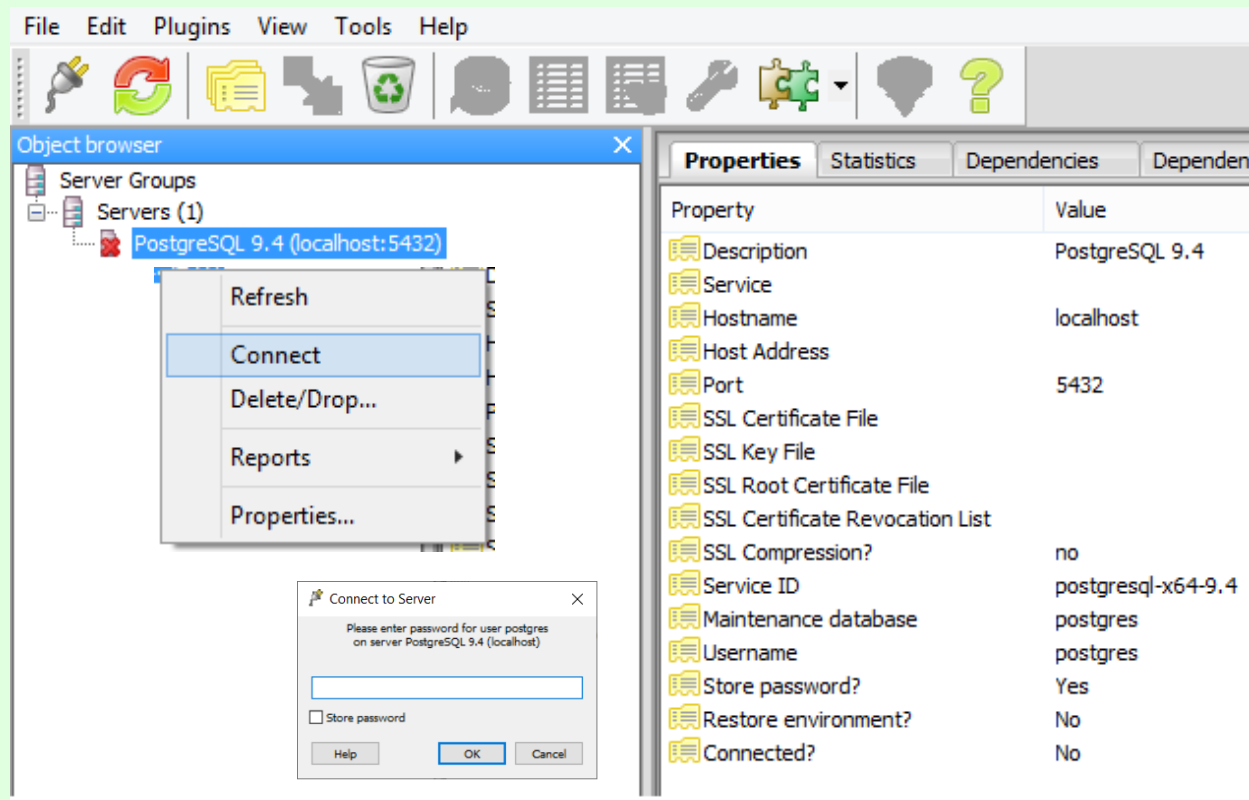


Η PostgreSQL δεν παρέχει ή ίδια κάποιο γραφικό περιβάλλον διεπαφής με τον χρήστη. Για την αλληλεπίδραση με την βάση δεδομένων χρησιμοποιείται ένα πρόγραμμα client το pgAdminn III . Παρέχεται και εγκαθίσταται μαζί με την PostgreSQL. Το pgAdmin III και έχει τα παρακάτω χαρακτηριστικά :

- Είναι ένα ολοκληρωμένο σύστημα σχεδιασμού και διαχείρισης βάσεων δεδομένων και αποτελεί το μέσο αλληλεπίδρασης της βάσης δεδομένων με το χρήστη.
- Είναι γραμμένο σε C++ και λειτουργεί σε περιβάλλοντα Linux, Mac και Windows.
- Προσφέρει ένα απλό γραφικό περιβάλλον για την ανάπτυξη πολύπλοκων βάσεων δεδομένων μέσω της διατύπωσης ερωτημάτων σε SQL, με στόχο την απλούστευση των διαδικασιών για το χρήστη.
- Είναι ελεύθερο λογισμικό και δεν απαιτεί επιπλέον προγράμματα για την επικοινωνία με τον διακομιστή της βάσης δεδομένων.







Εκκίνηση

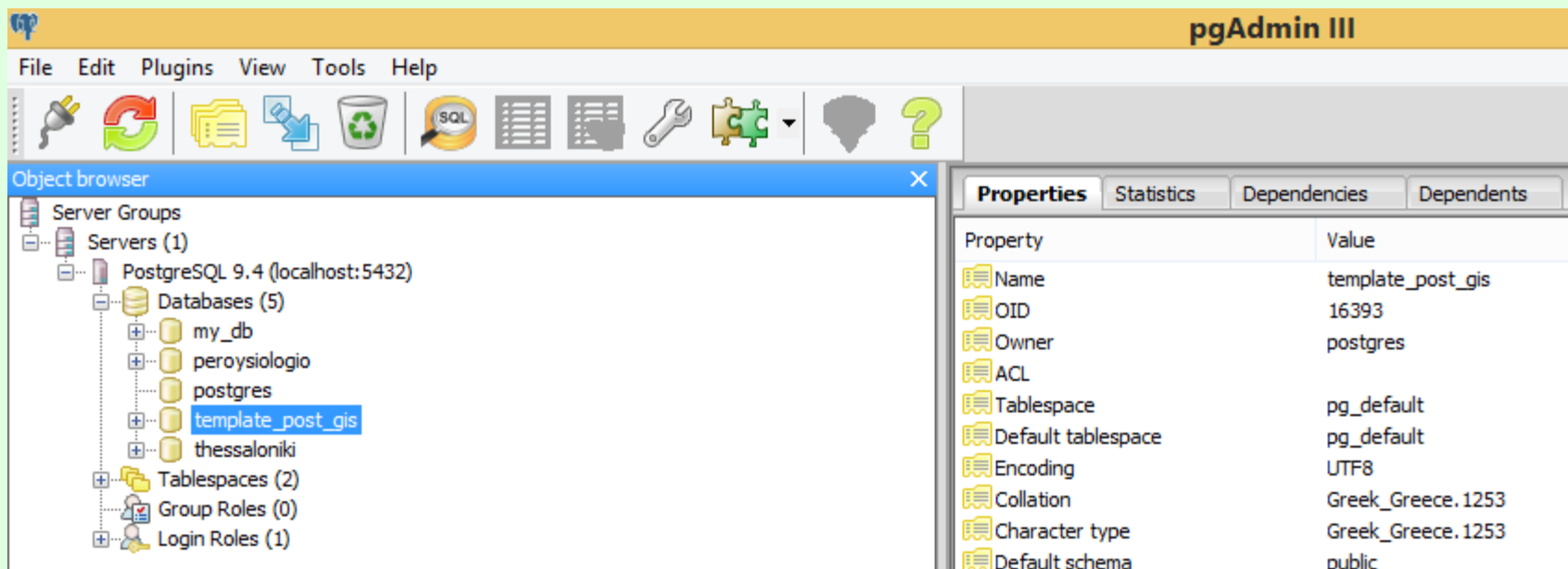
Με την εκκίνηση του client pgAdmin III εμφανίζεται το γραφικό περιβάλλον διεπαφής με την βάση postgresql. Με δεξί κλικ πάνω στο όνομα του SERVER (postgresql 9.4) εμφανίζεται μενού από μπορούμε να συνδεθούμε χρησιμοποιώντας τα συνθηματικά που έχουμε ορίσει κατά την εγκατάσταση.



Κεντρική Σελίδα

Όταν συνδεθούμε στον server βλέπουμε ότι έχουν δημιουργηθεί 5 βάσεις δεδομένων. Επίσης εμφανίζεται το μενού και κάποια εργαλεία όπως στην εικόνα που ακολουθεί:

-  Σύνδεση σε απομακρυσμένο Server
-  Ενημερώνει το σύστημα για τις αλλαγές που έχουν γίνει στο επιλεγμένο αντικείμενο
-  Εμφανίζει τα χαρακτηριστικά του αντικειμένου που έχουμε επιλέξει
-  Για δημιουργία αντικειμένου ίδιας κατηγορίας με αυτό που έχουμε επιλέξει
-  Για διαγραφή επιλεγμένων αντικειμένων
-  Για ανάπτυξη και εκτέλεση ερωτημάτων




The screenshot shows the pgAdmin III interface. The top menu bar includes File, Edit, Plugins, View, Tools, and Help. Below the menu is a toolbar with icons for connecting, refreshing, displaying properties, creating objects, deleting objects, and running SQL queries. The Object browser on the left shows a tree view of the database structure. The Properties window on the right displays the following details for the selected database:

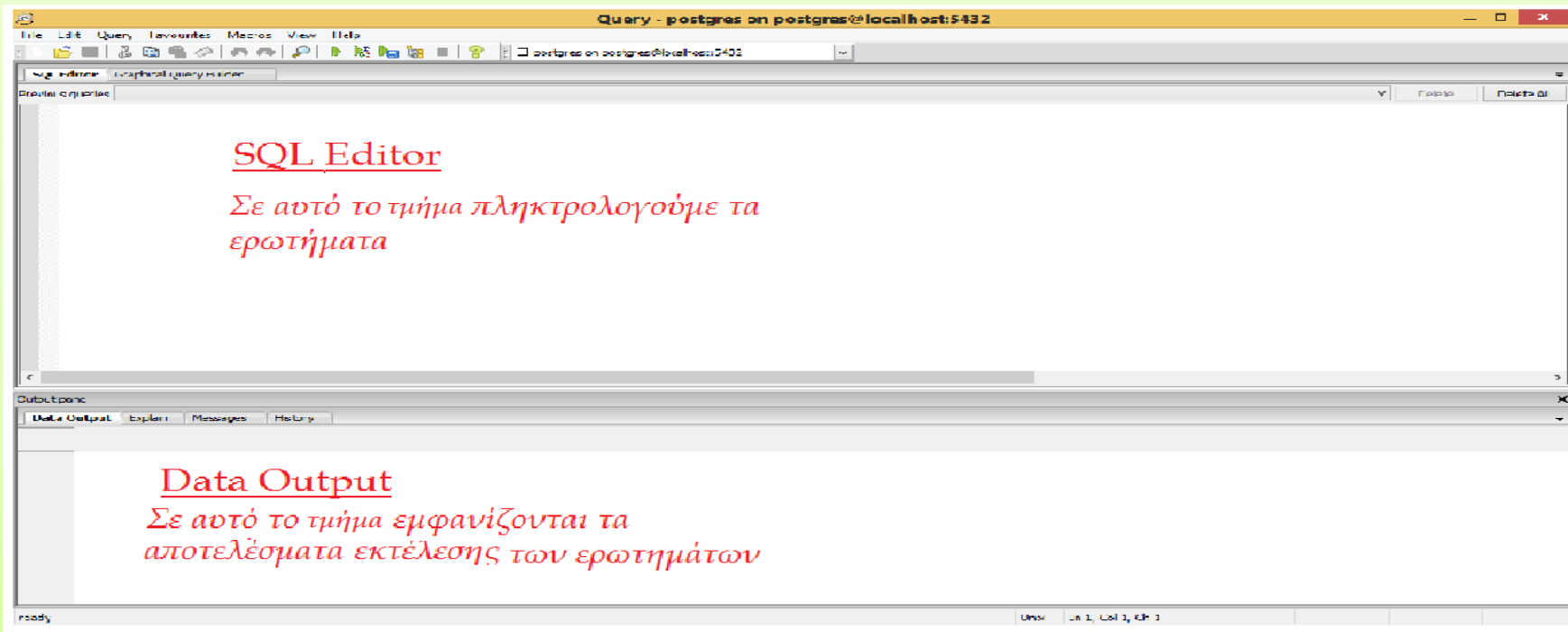
Property	Value
Name	template_post_gis
OID	16393
Owner	postgres
ACL	
Tablespace	pg_default
Default tablespace	pg_default
Encoding	UTF8
Collation	Greek_Greece.1253
Character type	Greek_Greece.1253
Default schema	public

Structure Query Language (SQL)

- Η **SQL** είναι μία **δομημένη γλώσσα ερωτημάτων** και χρησιμοποιείται για τη διαχείριση των δεδομένων της βάσης . Η διαχείριση γίνεται με την υποβολή ερωτημάτων από τους χρήστες με σκοπό τη δημιουργία, την ενημέρωση , την διαγραφή καθώς και την επιλογή δεδομένων.
- Αποτελεί την κοινή γλώσσα όλων των Σχεσιακών ΣΔΒΔ (standard)
- Η ανάπτυξη των σχεσιακών πράξεων στην SQL είναι πιο απλή από ότι στην σχεσιακή άλγεβρα
- Η SQL είναι μια πλήρης γλώσσα που περιλαμβάνει εντολές για
 - Γλώσσα ορισμού δεδομένων (DDL Data Definition Language)
 - Γλώσσα ενημέρωσης δεδομένων (DML Data Manipulation Language)
 - ερωτήσεις (Query Language)
- Πολλές γλώσσες προγραμματισμού έχουν επεκταθεί ώστε να υποστηρίζουν διατύπωση ερωτημάτων σε SQL π.χ. Java, C/C++, Pascal/Delphi, Visual Basic, php,...

Περιβάλλον ανάπτυξης ερωτημάτων SQL

Η postgresql διαθέτει ένα περιβάλλον ανάπτυξης εκτέλεσης αποθήκευσης ερωτημάτων SQL. Το εν λόγω περιβάλλον αποτελείται από ένα παράθυρο χωρισμένο σε 2 τμήματα: Το τμήμα του **SQL editor** όπου πληκτρολογούμε τα ερωτήματα και το τμήμα του **Data Output** όπου εμφανίζονται τα αποτελέσματα των ερωτημάτων. Για να ανοίξουμε αυτό το παράθυρο πατάμε το εργαλείο  ή επιλέγουμε **Tools\Query tool**. Για να εκτελέσουμε ερωτήματα επιλέγουμε **Query\Execute**. Τα δεδομένα των ερωτημάτων προέρχονται από την βάση που έχουμε ανοίξει. Τα αποτελέσματα ερωτημάτων ενημέρωσης ενημερώνουν την εν λόγω βάση. Για να ανοίξουμε μία βάση δεδομένων Κάνουμε διπλό κλικ πάνω στο όνομα της.



Βασικά ορίσματα για τη δημιουργία βάσεων Δεδομένων(1)

Για να δημιουργήσουμε μία βάση δεδομένων πρέπει να ορίσουμε τα παρακάτω:

- Το όνομα της βάσης. Δεν πρέπει να υπάρχει άλλη βάση με το ίδιο όνομα. Το διασφαλίζει η postgresql.
- Τον ιδιοκτήτη(owner) της βάσης ο οποίος θα διαθέτει πλήρη δικαιώματα στην βάση. Προεπιλογή: postgres
- Την κωδικοποίηση των χαρακτήρων η οποία ορίζει ένα κώδικα μέσω του οποίου οι χαρακτήρες συσχετίζονται σε ένα άλλο σύνολο χαρακτήρων, αριθμών, σημάτων κατάλληλο για την αποθήκευση, επεξεργασία, μετάδοση, κ.λπ. Ουσιαστικά είναι μια αντιστρέψιμη διαδικασία (ENCODE-DECODE). Προεπιλογή: UTF-8. Το UTF-8 (8-bit Unicode Transformation Format) είναι ένας κωδικοποιητής χαρακτήρων μεταβλητού πλάτους ικανός να κωδικοποιεί όλα τα 1.112.064 έγκυρα σημεία κωδικού χαρακτήρα στο Unicode χρησιμοποιώντας 1 έως 4 byte (8-bit). Σχεδιάστηκε για συμβατότητα με ASCII και έτσι μπορεί να χρησιμοποιηθεί στις περισσότερες γλώσσες προγραμματισμού.

Βασικά ορίσματα για τη δημιουργία βάσεων Δεδομένων(2)

•Συνέχεια:

- **_Connection Limit**(max_concurrent_connection). Ορίζει τις μέγιστες ταυτόχρονες συνδέσεις με τη νέα βάση δεδομένων. Η προεπιλογή είναι -1, δηλαδή, απεριόριστη.
- **LC_Collate** Ορίζει τη σειρά ταξινόμησης των συμβολοσειρών που επηρεάζουν το αποτέλεσμα του όρου ORDER BY.
- **LC_Ctype** Ορίζει την ταξινόμηση χαρακτήρων ανάλογα με την κατηγορία ψηφία, μικρά-κεφαλαία γράμματα π.χ. GREEK-1253
- Το **tablespace** ορίζει τον χώρο όπου αποθηκεύονται τα διάφορα αντικείμενα της βάσης π.χ. Πίνακες, δείκτες κ.λπ. Το σύμπλεγμα βάσεων δεδομένων ή σύμπλεγμα καταλόγων (initdb) είναι μια περιοχή αποθήκευσης βάσεων δεδομένων η οποία αρχικά περιέχει μια βάση δεδομένων την postgres. Δύο πίνακες χώρων δημιουργούνται αυτόματα από το initdb:
 - Το **pg_global** tablespace χρησιμοποιείται για κοινόχρηστους καταλόγους συστήματος.
 - Το **pg_default** είναι ένα προεπιλεγμένος χώρος. Σαν προεπιλεγμένος χώρος μπορεί να χρησιμοποιείται και από άλλες βάσεις δεδομένων.

Μπορούμε να ορίσουμε διαφορετικό χώρο αποθήκευσης για κάποιο πίνακα. Αυτό γίνεται σε περιπτώσεις που ο χώρος που φιλοξενεί τη βάση εξαντλείται ή όταν απαιτείται μεγαλύτερη ταχύτητα π.χ. κάποιοι πίνακες που χρησιμοποιούνται συχνά τοποθετούνται σε χώρο δίσκου στερεάς κατάστασης που έχει μεγάλη ταχύτητα σε σχέση με το μαγνητικό δίσκο.

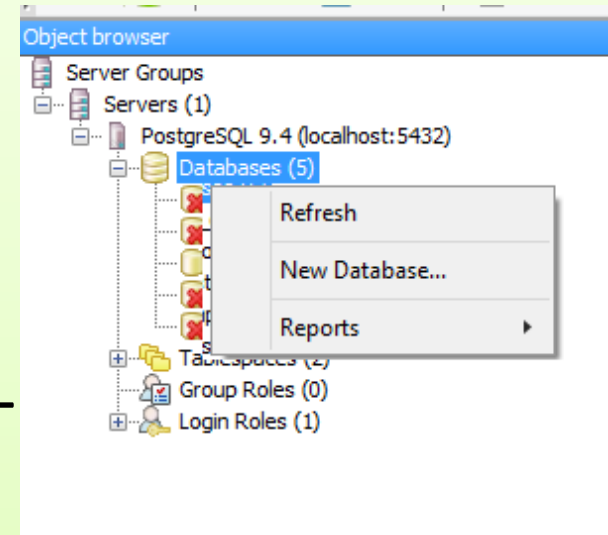
Παράδειγμα

```
CREATE TABLESPACE fastspace LOCATION '/mnt/sda1/postgresql/data';  
CREATE TABLE Pin1(.....) TABLESPACE fastspace;
```

Δημιουργία βάσης Δεδομένων

Για να δημιουργήσουμε μία βάση δεδομένων υπάρχουν 2 τρόποι:

Με δεξί κλικ στο εικονίδιο Databases εμφανίζεται μενού όπου κλικάρουμε New Database συμπληρώνουμε τα απαραίτητα ορίσματα για την δημιουργία της βάσης.



Με ερώτημα SQL. Δημιουργία βάσης peroyysiologio.

CREATE DATABASE ονομα βάσης WITH
ENCODING=κωδικοποιητής χαρακτήρων
OWNER = ιδιοκτήτης TABLESPACE =

Διαγραφή βάσης: DROP DATABASE όνομα_βάσης.

ΓΛΩΣΣΑ ΟΡΙΣΜΟΥ ΔΕΔΟΜΕΝΩΝ

Η γλώσσα ορισμού δεδομένων δίνει την δυνατότητα να μεταβάλλουμε την δομή της βάσης δεδομένων. Συγκεκριμένα επιτρέπει τις παρακάτω ενέργειες:

- Να δημιουργήσουμε Πίνακα
- Να διαγράψουμε πίνακα.
- Να μεταβάλλουμε την δομή Πίνακα.
- Να δημιουργήσουμε ευρετήρια
- Να διαγράψουμε ευρετήρια.

ΕΡΩΤΗΜΑΤΑ ΟΡΙΣΜΟΥ ΔΕΔΟΜΕΝΩΝ ΠΙΝΑΚΩΝ

Δημιουργία Πίνακα.

```
CREATE TABLE όνομα πίνακα (στήλη1 data type [NOT NULL]
[DEFAULT αρχική τιμή], στήλη2 data type [NOT NULL] [DEFAULT
αρχική τιμή], ... στήληN ..., PERIORISMOS_AKAIREOTHTAS_1,
PERIORISMOS_AKAIREOTHTAS_2, ...
PERIORISMOS_AKAIREOTHTAS_N, );
```

Διαγραφή Πίνακα.

```
DROP TABLE όνομα πίνακα.
```

Μεταβολή δομής Πίνακα (εισαγωγή-διαγραφή στήλης, προσθήκη περιορισμού).

```
ALTER TABLE όνομα πίνακα ADD COLUMN στήλη data type );
```

```
ALTER TABLE όνομα πίνακα DROP COLUMN στήλη;
```

```
ALTER TABLE όνομα πίνακα ADD CONSTRAINT περιορισμός
ακεραιότητας;
```

ΕΡΩΤΗΜΑΤΑ ΟΡΙΣΜΟΥ ΔΕΔΟΜΕΝΩΝ ΕΥΡΕΤΗΡΙΩΝ

Ευρετήρια είναι μικρά αρχεία που δημιουργούνται σύμφωνα με τα περιεχόμενα των τιμών ενός ή περισσότερων στηλών ενός πίνακα. Τα ευρετήρια αποτυπώνουν την αντιστοίχιση των τιμών που περιέχουν στην θέση των εγγραφών που περιέχονται. Προσφέρουν γρήγορη αναζήτηση των εγγραφών που περιέχουν κάποια τιμή του ευρετηρίου, καθώς και γρήγορη ταξινόμηση των εγγραφών σύμφωνα με τις τιμές που περιέχονται στα Ευρετήρια. Σε κάθε πίνακα δημιουργείται αυτόματα από το σύστημα ένα ευρετήριο σύμφωνα με τις τιμές του πρωτεύοντος κλειδιού το οποίο ονομάζεται πρωτεύων ευρετήριο.

Δημιουργία Ευρετηρίου.

```
CREATE INDEX όνομα ευρετηρίου ON όνομα πίνακα USING  
(στήλη1, στηλη2 ..., );
```

Διαγραφή Ευρετηρίου.

```
DROP INDEX όνομα ευρετηρίου.
```

Τύποι Δεδομένων

- Για συμβολσειρές `char(n)` σταθερού μήκους, και `varchar(n)` ή `varying(n)` μεταβλητού μήκους.
- Για ακεραίους `integer(ή int)` `smallint`
- Αριθμοί κινητής υποδιαστολής
`float[ακρίβεια]`, `real`, `double precision`
- `numeric(p, d)` ή `decimal(p, d)`
(p ψηφία συνολικά από τα οποία d είναι δεκαδικά)
- Δυαδικές τιμές `bit` `bit(n)`
- Ημερομηνία-ώρα
`date` (yyyy-mm-dd) `time` (hh:mm:ss)
`timestamp[ακρίβεια]` yyyy-mm-dd hh:mm:ss. _ _ _

Ορισμοί Περιορισμών Ακεραιότητας

Επιβολή ακεραιότητας σε επίπεδο στηλών:

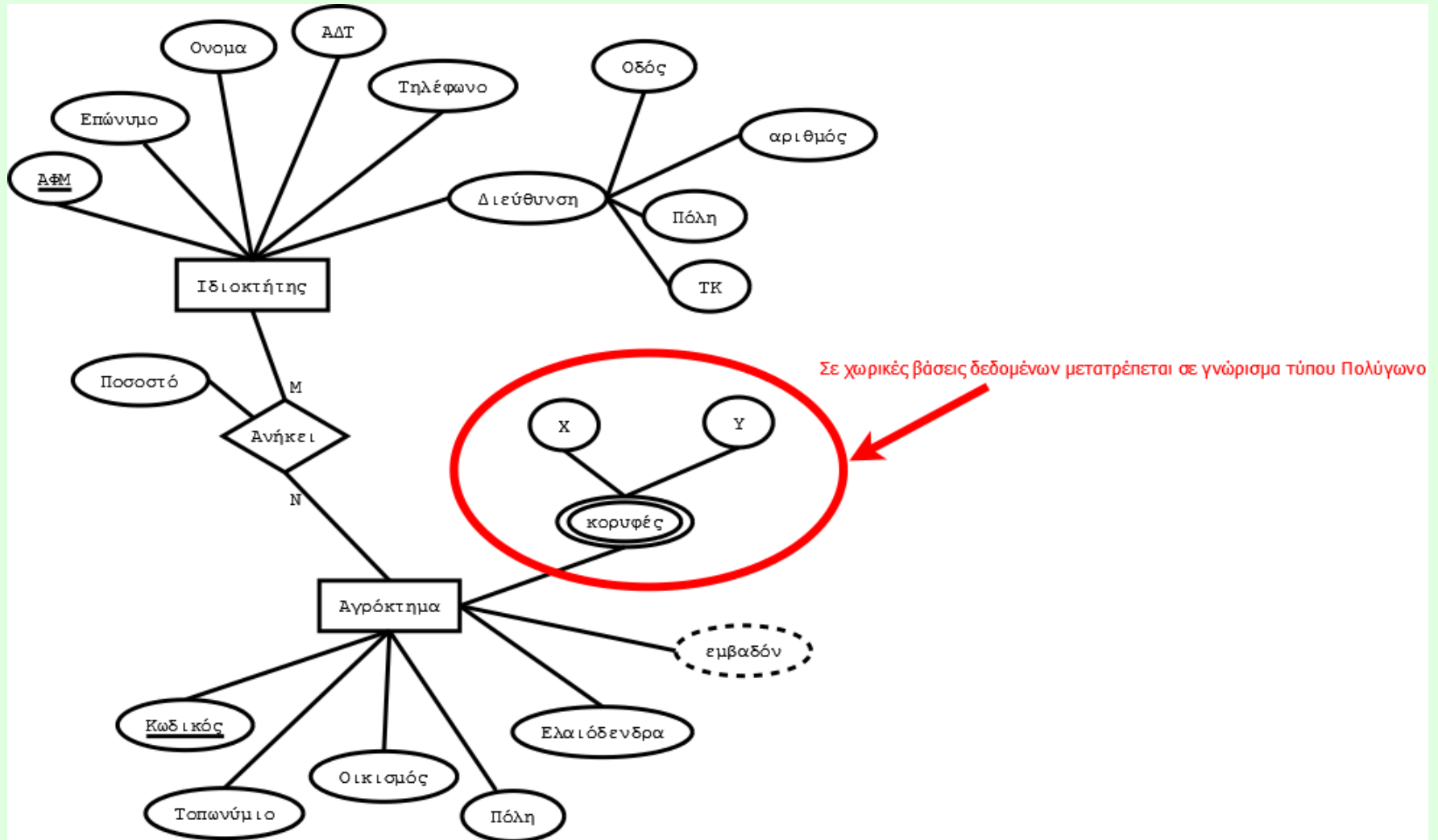
- Default Value τιμή, Ορισμός αρχικής τιμής.
 - NOT NULL, Δεν επιτρέπεται κενό περιεχόμενο.
 - Unique(Columns), Υποψήφια κλειδιά.
 - Primary Key(Columns), Πρωτεύων κλειδί.
 - Foreign key column references table(column)
Ακεραιότητα αναφοράς (ξένα κλειδιά).
 - CHECK (YEAR>2000) Σημασιολογικοί περιορισμοί.
- Επιβολή ακεραιότητας Foreign key :
 - On delete cascade ή restrict ή set null ή set default
 - On update cascade ή restrict ή set null ή set default

ΠΕΡΟΥΣΙΟΛΟΓΙΟ

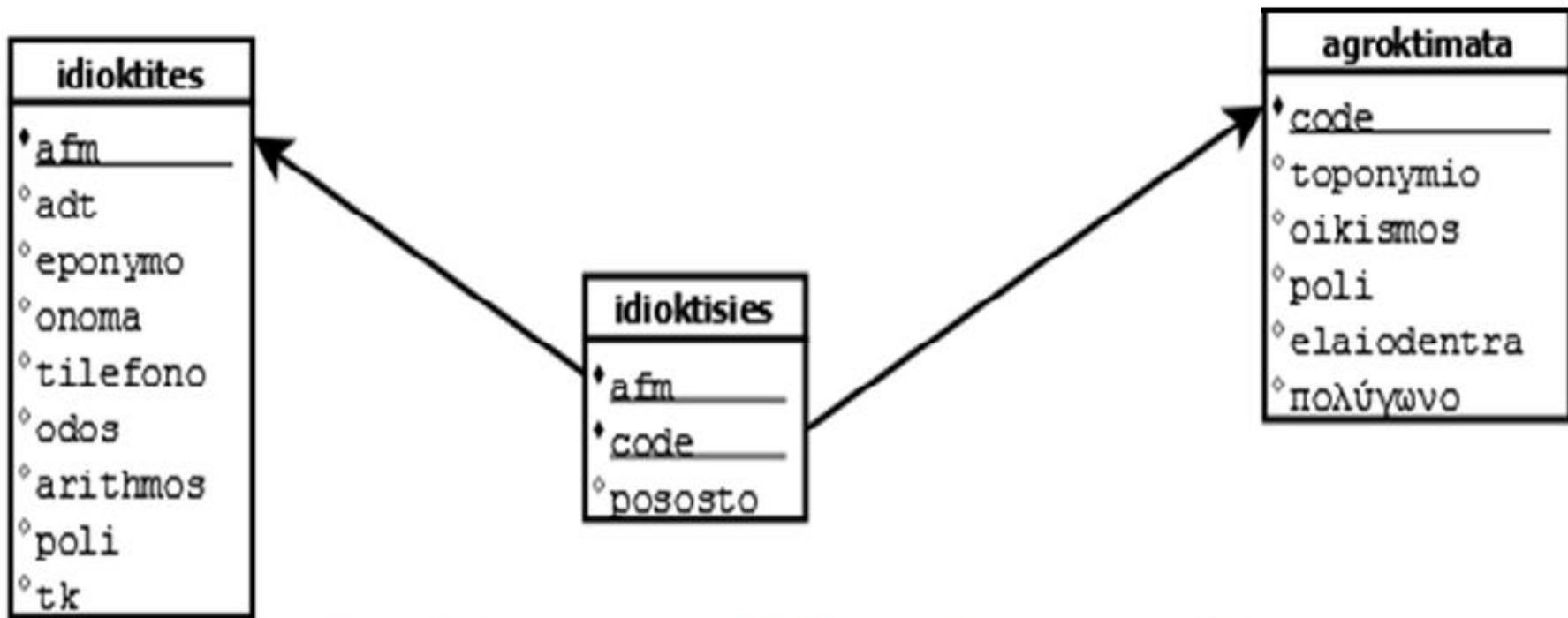
Θέλουμε να σχεδιάσουμε μία χωρική βάση δεδομένων στην οποία θα καταχωρούμε τους ιδιοκτήτες ελαιώνων και τους ελαιώνες (Ελαιοκομικό Μητρώο). Αναλυτικότερα θα περιέχει τα παρακάτω:

- Για τους ιδιοκτήτες θέλουμε να γνωρίζουμε τα παρακάτω στοιχεία: Αρ. Φορ. Μητρώου, Αρ. Δελτίου Ταυτότητας, Επώνυμο, Όνομα, Τηλέφωνο, Διεύθυνση(οδός, αριθμός, πόλη, ΤΚ). Θεωρείται γνωστό ότι ο Αρ.Φορ.Μητρώου καθώς και ο Αρ.Δελτίου Ταυτότητας έχουν την ιδιότητα της μοναδικότητας.
- Για τα αγροκτήματα θέλουμε να γνωρίζουμε τον κωδικό , το τοπωνύμιο, τον οικισμό, και την πόλη που βρίσκεται καθώς και το πλήθος των ελαιοδέντρων που περιέχει. Επίσης θέλουμε να γνωρίζουμε τις γεωγραφικές συντεταγμένες των κορυφών του και το εμβαδόν του.
- Κάθε ιδιοκτήτης κατέχει ένα ή περισσότερα αγροκτήματα και ένα αγρόκτημα ανήκει σε ένα ή περισσότερους ιδιοκτήτες. Για κάθε ιδιοκτησία πρέπει να γνωρίζουμε το ποσοστό που κατέχει ο ιδιοκτήτης. Το ποσοστό παίρνει τιμές στο διάστημα $(0,100]$ (ή αλλιώς >0 AND ≤ 100).

ΠΕΡΟΥΣΙΟΛΟΓΙΟ (Διάγραμμα οντοτήτων συσχετίσεων)



ΠΕΡΟΥΣΙΟΛΟΓΙΟ (σχεσιακό σχήμα)



Το πολύγωνο και το εμβαδόν του θα τα παρουσιάσουμε στην διάλεξη που αφορά τα χωρικά δεδομένα

ΟΡΙΣΜΟΣ ΔΕΔΟΜΕΝΩΝ ΠΕΡΟΥΣΙΟΛΟΓΙΟΥ

Το πρώτο βήμα είναι να δημιουργήσουμε την βάση δεδομένων :

```
CREATE DATABASE perouysiologio WITH OWNER = postgres ENCODING = 'UTF8'  
TABLESPACE = pg_default LC_COLLATE = 'Greek_Greece.1253'  
LC_CTYPE = 'Greek_Greece.1253' CONNECTION LIMIT = -1;
```

Το παραπάνω Ερώτημα δημιουργεί μια βάση δεδομένων με όνομα perouysiologio. Η κωδικοποίηση των χαρακτήρων ορίζεται από τον UTF-8. Ο χώρος αποθήκευσης των αντικειμένων της βάσης είναι ο προεπιλεγμένος χώρος. Η σειρά ταξινόμησης των χαρακτήρων ορίζεται από τον κώδικα Greece.1253. Δεν υπάρχει περιορισμός ως προς τον αριθμό των χρηστών που θα συνδεθούν στην βάση.

Όταν η βάση δεδομένων περιέχει μεγάλο αριθμό αντικειμένων(πίνακες, όψεις, ...) απαιτείται να γίνει οργάνωση του χώρου αποθήκευσης ώστε να είναι εύκολη η αναζήτηση τους. Για την εν λόγω οργάνωση τα στοιχεία της βάσης ομαδοποιούνται σε σχήματα. Η κάθε βάση περιέχει ένα σχήμα με όνομα **public** το οποίο δημιουργείται με την δημιουργία της.

Η δημιουργία σχήματος γίνεται με το ερώτημα : **CREATE SCHEMA όνομα_σχήματος ;**

Η διαγραφή σχήματος γίνεται με το ερώτημα : **DROP SCHEMA όνομα_σχήματος;**

ΔΗΜΙΟΥΡΓΙΑ ΠΙΝΑΚΑ ΑΓΡΟΚΤΗΜΑΤΩΝ.

```
CREATE TABLE agroktimata
(  
code VARCHAR(10) NOT NULL,  
toponymio VARCHAR(30),  
oikismos VARCHAR(20),  
poli VARCHAR(20),  
elaiodentra INTEGER,  
CONSTRAINT pr_key_agro  
PRIMARY KEY (code)  
);
```

agroktimata	
•	<u>code</u>
◊	toponymio
◊	oikismos
◊	poli
◊	elaiodentra
◊	πολύγωνο

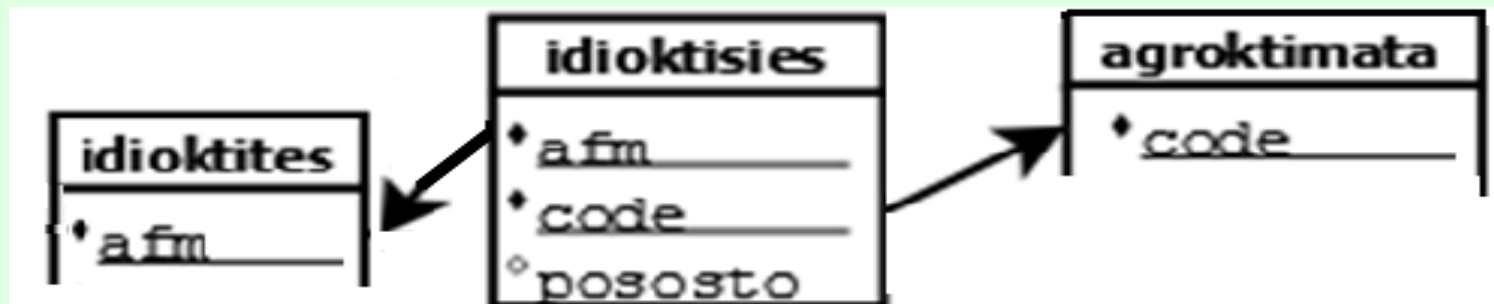
ΔΗΜΙΟΥΡΓΙΑ ΠΙΝΑΚΑ ΙΔΙΟΚΤΗΤΩΝ.

```
CREATE TABLE idioktites
(
  afm VARCHAR(12) NOT NULL,
  adt VARCHAR (8) NOT NULL,
  eponymo VARCHAR(20) NOT NULL,
  onoma VARCHAR(15),
  telefono VARCHAR(15),
  Odos VARCHAR(20),
  arithmos INTEGER,
  poli VARCHAR(20),
  tk VARCHAR(10),
  CONSTRAINT pr_key_idiokt
  PRIMARY KEY(AFM), CONSTRAINT
  uniq_idiokt UNIQUE (ADT)
);
```

idioktites
* <u>afm</u>
◊ adt
◊ eponymo
◊ onoma
◊ telefono
◊ odos
◊ arithmos
◊ poli
◊ tk

ΔΗΜΙΟΥΡΓΙΑ ΠΙΝΑΚΑ ΙΔΙΟΚΤΗΣΙΩΝ

```
CREATE TABLE idioktisies( afm VARCHAR(12) NOT NULL,  
code VARCHAR(10) NOT NULL, pososto smallint,  
CONSTRAINT pr_key_idioktis PRIMARY KEY(AFM,code),  
CONSTRAINT check_psosto CHECK(pososto<=100 AND  
pososto>0), CONSTRAINT forkey_afm FOREIGN KEY (afm)  
REFERENCES idioktites(afm) MATCH SIMPLE ON UPDATE  
RESTRICT ON DELETE RESTRICT, CONSTRAINT forkey_code  
FOREIGN KEY(code) REFERENCES agroktimata(code) MATCH  
SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT);
```



Γλώσσα Επεξεργασίας Δεδομένων Data Manipulation Language(DML)

Η γλώσσα επεξεργασίας δεδομένων παρέχει την δυνατότητα τροποποίησης του περιεχομένου που είναι αποθηκευμένο στη βάση Δεδομένων.

Συγκεκριμένα μπορούμε να εκτελέσουμε τις παρακάτω ενέργειες:

- Εισαγωγή νέων εγγραφών.
- Τροποποίηση ή ενημέρωση των εγγραφών.
- Διαγραφή εγγραφών.

Εισαγωγή νέων εγγραφών.

1^{ος} τρόπος: INSERT INTO όνομα_πίνακα VALUES (value1, value2, ...),(...),...;

2^{ος} τρόπος: INSERT INTO όνομα_πίνακα (col1, col2, col3) VALUES(value_1, value_2, value_3);

Και με τους 2 τρόπους μπορούμε να εισάγουμε όσες εγγραφές θέλουμε μέσα σε ζευγάρια παρενθέσεων μετά το Values.

Η διαφορά των 2 τρόπων είναι στο αριθμό των ορισμάτων μέσα στις παρενθέσεις:

Στον 1^ο τρόπο το πλήθος των ορισμάτων στις παρενθέσεις είναι όσες είναι οι στήλες του πίνακα που ενημερώνεται.

Στον 2^ο τρόπο το πλήθος των ορισμάτων στις παρενθέσεις είναι όσες είναι οι στήλες που αναφέρονται στις παρενθέσεις πριν το Values.

Εισαγωγή εγγραφών στην βάση perouysiologio.

Εισαγωγή εγγραφών στον πίνακα ιδιοκτήτες.

```
INSERT INTO idioktites VALUES('123456789', 'ΖΛ123456',  
'Παπαδάκης', 'Γιάννης', '2345678', 'Καλοκαιρινού',  
100, 'Ηράκλειο', '71410');
```

```
INSERT INTO idioktites (afm, adt, eponymo, onoma, poli)  
VALUES ('1234562000', 'ΖΗ123456', 'Τσαγκάρης', 'Νίκος',  
'Ηράκλειο');
```

Εισαγωγή εγγραφών στον πίνακα αγροκτήματα.

```
INSERT INTO agroktimata VALUES  
( 'Σ1', 'Αποσαμι Μετοχι', 'Σκαλανι', 'Ηράκλειο', '33'),  
( 'Κ1', 'Κορακοβούνι', 'Αμφιθέα', 'Ηράκλειο', '2');
```

Ενημέρωση και διαγραφή εγγραφών.

Ενημέρωση εγγραφών

```
UPDATE ονομα_πινακα set  
col_1=value1, ..., col_N=value_N  
where συνθήκη;
```

Διαγραφή εγγραφών

```
DELETE From ονομα_πινακα where  
συνθήκη;
```

Επεξεργασία Δεδομένων Περιουσιολογίου

- **Ενημέρωση Δεδομένων**

```
UPDATE idioktites SET eponymo='Τσαγκαράκης',  
poli='Χανιά' WHERE adt='ZH123456';
```

```
UPDATE agroktimata set poli='Ηράκλειο' where  
poli='Herakleion' or poli='Hraklio'
```

- **Διαγραφή Δεδομένων**

```
DELETE FROM idioktites WHERE afm='1234562000';
```

```
DELETE FROM idioktites WHERE afm not in (select  
afm from idioktisies);
```

Ερωτήματα Επιλογής Δεδομένων

Εμφάνιση όλων των στοιχείων των ιδιοκτητών από το Ηράκλειο.

```
SELECT * FROM idioktites WHERE poli='Ηράκλειο';
```

Εμφάνιση των ΑΔΤ, ΕΠΩΝΥΜΩΝ και ΟΝΟΜΑΤΩΝ ιδιοκτητών από το Ηράκλειο με ταξινόμηση ως προς το ΕΠΩΝΥΜΟ και το ΟΝΟΜΑ.

```
SELECT adt, eponymo , onoma FROM idioktites WHERE poli="Ηράκλειο" order by eponymo,onoma;
```

Εμφάνιση όλων των αγροκτημάτων που έχουν περισσότερα από 5 ελαιόδενδρα και βρίσκονται στο Ηράκλειο ή στην Ιεράπετρα.

```
select * from agroktimata where poli in ('Ηράκλειο','Ιεράπετρα') and elaiodentra>5;
```

Εμφάνιση των ΕΠΩΝΥΜΩΝ και των ΟΝΟΜΑΤΩΝ όλων των ιδιοκτητών. Σε περιπτώσεις συνωνυμίας(ίδιο επώνυμο και όνομα) το αποτέλεσμα να εμφανίζεται μία φορά.

```
SELECT DISTINCT eponymo , onoma FROM idioktites;
```

Παραδείγματα Επιλογής Δεδομένων από πολλούς πίνακες

Εμφάνιση των αγροτεμαχίων του ιδιοκτήτη με ΑΦΜ 3000

```
select idioktisies.code, toponymio, oikismos, poli, elaiodentra,  
pososto from idioktisies join agroktimata on  
idioktisies.code=agroktimata.code where afm='3000';
```

Εμφάνιση όλων των αγροκτημάτων και των κατόχων τους

```
select idioktisies.code, toponymio, oikismos, agroktimata.poli,  
elaiodentra,pososto,idioktites.afm,eponymio,onoma from  
(idioktisies join agroktimata on  
idioktisies.code=agroktimata.code) join idioktites on  
idioktites.afm=idioktisies.afm;
```

Παραδείγματα ομαδοποίησης αποτελεσμάτων

Εμφάνιση των στοιχείων όλων των ιδιοκτητών καθώς και τον συνολικό αριθμό ελαιοδέντρων που κατέχει ο καθένας

```
select idioktites.afm,adt,eponymo,onoma,  
sum(elaiodentra*pososto/100) as plithos_elaiodentron  
from (idioktisies join agroktimata on  
idioktisies.code=agroktimata.code) join idioktites on  
idioktites.afm=idioktisies.afm group by idioktites.afm;
```

Εμφάνιση των στοιχείων όλων των αγροτεμαχίων που το συνολικό ποσοστό ιδιοκτησίας των κατόχων του είναι διάφορο του 100.

```
select agroktimata.code, toponymio, oikismos,  
agroktimata.poli, elaiodentra,sum(pososto) from idioktisies  
join agroktimata on idioktisies.code=agroktimata.code  
group by agroktimata.code having sum(pososto)<>100;
```

Πράξεις Συνόλων

Ένωση συνόλων	Select Union select ...
Τομή συνόλων	Select intersect select
Διαφορά συνόλων	Select except select

Παρατήρηση Τα ερωτήματα που μετέχουν στις πράξεις συνόλων πρέπει να δίνουν αποτελέσματα με τον ίδιο αριθμό στηλών και οι στήλες που αντιστοιχούν κατά σειρά να είναι του ίδιου τύπου ώστε το αποτέλεσμα να αποτελεί πίνακα.

Παράδειγμα: Εμφάνιση των ιδιοκτητών που έχουν αγροκτήματα οπουδήποτε εκτός από το Ηράκλειο.

```
select idioktites.afm, adt, eponymo, onoma, telefono  
from (idioktisies join agroktimata on idioktisies.code=agroktimata.code)  
join idioktites on idioktites.afm=idioktisies.afm where  
agroktimata.poli<>'Ηράκλειο'
```

EXCEPT

```
select idioktites.afm, adt, eponymo, onoma, telefono  
from (idioktisies join agroktimata on idioktisies.code=agroktimata.code)  
join idioktites on idioktites.afm=idioktisies.afm where  
agroktimata.poli='Ηράκλειο';
```

Backup-Restore

Backup

- Επιλέγουμε τη βάση ή το στοιχείο της βάσης που θέλουμε να γίνει Backup
- Από τα tools κλικάρουμε Backup
- Στο παράθυρο που εμφανίζεται ορίζουμε το ονομα και τη θέση που θα αποθηκευτεί το backup.
- Και πατάμε το πλήκτρο backup.

• Restore

- Δημιουργούμε μια κενή βάση δεδομένων ή το αντικείμενο που θέλουμε να κάνουμε restore.
- Επιλέγουμε το αντικείμενο που μόλις δημιουργήσαμε και από τα tools κλικάρουμε restore