

ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

Χειμερινό Εξάμηνο 2013-2014



SQL – Structured Query Language

Δρ. Βαγγελιώ Καβακλή

ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ,
ΤΜΗΜΑ ΠΟΛΙΤΙΣΜΙΚΗΣ ΤΕΧΝΟΛΟΓΙΑΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΑΣ

1

Πράξεις της σχεσιακής άλγεβρας

ΠΡΑΞΗ	ΣΚΟΠΟΣ	ΣΥΜΒΟΛΙΣΜΟΣ
ΕΠΙΛΟΓΗ	Επιλέγει όλες τις πλειάδες, από μια σχέση R, που ικανοποιούν τη συνθήκη επιλογής.	σ <συνθήκη επιλογής> (R)
ΠΡΟΒΟΛΗ	Παράγει μια νέα σχέση με μερικά μόνο γνωρίσματα της R.	π <λίστα γνωρισμάτων> (R)
ΘΗΤΑ ΣΥΝΕΝΩΣΗ	Παράγει όλους τους συνδυασμούς πλειάδων από τις R ₁ και R ₂ που ικανοποιούν τη συνθήκη συνένωσης.	$R_1 \bowtie R_2$ <συνθήκη συνένωσης>
ΣΥΝΕΝΩΣΗ ΙΣΟΤΗΤΑΣ	Παράγει όλους τους συνδυασμούς πλειάδων από τις R ₁ και R ₂ που ικανοποιούν τη συνθήκη συνένωσης με συγκρίσεις ισότητας μόνο.	$R_1 \bowtie_{1>} R_2$ <γνωρίσματα συνένωσης 1> <γνωρίσματα συνένωσης 2>
ΦΥΣΙΚΗ ΣΥΝΕΝΩΣΗ	Ίδια με τη ΣΥΝΕΝΩΣΗ ΙΣΟΤΗΤΑΣ εκτός από το ότι τα γνωρίσματα συνένωσης της R ₂ δεν περιλαμβάνονται στο αποτέλεσμα. Αν τα γνωρίσματα της συνένωσης έχουν τα ίδια ονόματα δεν χρειάζεται να προσδιοριστούν.	$R_1 * R_2$ <συνθήκη συνένωσης> R ₂ ή $R_1 * R_2$
ΕΝΩΣΗ	Παράγει μια σχέση που περιέχει όλες τις πλειάδες που βρίσκονται στην R ₁ , ή στην R ₂ , ή και στην R ₁ και στην R ₂ .	$R_1 \cup R_2$
ΤΟΜΗ	Παράγει μια σχέση που περιέχει τις κοινές πλειάδες των R ₁ και R ₂ .	$R_1 \cap R_2$
ΔΙΑΦΟΡΑ	Παράγει μια σχέση που περιέχει τις πλειάδες της R ₁ που δεν βρίσκονται στην R ₂ .	$R_1 - R_2$
ΚΑΡΤΕΣΙΑΝΟ ΓΙΝΟΜΕΝΟ	Παράγει μια σχέση που περιέχει τα γνωρίσματα των R ₁ και R ₂ και περιλαμβάνει ως πλειάδες όλους τους δυνατούς συνδυασμούς πλειάδων των R ₁ και R ₂	$R_1 \times R_2$

2

SQL

- SQL: Structured Query Language ή Δομημένη Γλώσσα Ερωτήσεων
 - αρχικά ονομαζόταν SEQUEL (Structured English Query Language)
 - υλοποιήθηκε στο ερευνητικό τμήμα της IBM (Chamberlin & Boyce 1976)
 - περιλαμβάνει χαρακτηριστικά της σχεσιακής άλγεβρας αλλά με σύνταξη πιο φιλική στο χρήστη
- Αποτελεί το πρότυπο των σχεσιακών βάσεων δεδομένων
 - θεωρείται ένας από τους πιο σημαντικούς λόγους για την επιτυχία των σχεσιακών βάσεων δεδομένων
 - έχει υλοποιηθεί σε μεγάλο αριθμό εμπορικών ΣΔΒΔ (DB2, ORACLE, INGRES, INFORMIX, Sybase, MS Access, MS SQL Server)
 - η μεταφορά μεταξύ διαφορετικών ΣΔΒΔ δεν είναι δαπανηρή εφόσον και τα δύο ακολουθούν το ίδιο πρότυπο SQL
 - σε ένα πρόγραμμα εφαρμογής ΒΔ οι ίδιες εντολές χρησιμοποιούνται για την προσπέλαση δεδομένων που αποθηκεύονται σε δύο ή περισσότερα σχεσιακά ΣΔΒΔ
- Κοινή προσπάθεια των οργανισμών ANSI και ISO οδήγησαν σε διαδοχικές τυποποιημένες εκδόσεις της SQL
 - SQL-86, SQL-92 (SQL2), SQL-99

3

SQL: Χαρακτηριστικά

- Αποτελεί πλήρης γλώσσα βάσεων δεδομένων
 - είναι ταυτόχρονα:
 - Γλώσσα Ορισμού Δεδομένων (ΓΟΔ) – ορισμός, δημιουργία και εξέλιξη σχήματος
 - Γλώσσα Χειρισμού Δεδομένων (ΓΧΔ) – ορισμός, δημιουργία, τροποποίηση, διαγραφή και επιλογή δεδομένων (ερωτήσεις και ενημερώσεις)
- Επιτρέπει
 - ορισμό περιορισμών ακεραιότητας
 - ορισμό όψεων στη βάση δεδομένων
 - προσδιορισμό της ασφάλειας και των δικαιοδοσιών (χρήστες και δικαιώματα χρηστών)
 - έλεγχο των δοσοληψιών
- Περιέχει κανόνες για εμφύτευση εντολών SQL σε μια γενικού σκοπού γλώσσα προγραμματισμού όπως η C ή η PASCAL

4

SQL: Συνοπτική παρουσίαση

- Αντιστοιχία ορολογίας Σχεσιακής Άλγεβρας - SQL:
 - Σχέση (relation) – Πίνακας (table)
 - Πλειάδα (tuple) – Γραμμή (row)
 - Κατηγορημα (attribute) – Πεδίο (field) ή Στήλη (column)
- Βασικές ΓΟΔ εντολές της SQL:
 - CREATE TABLE
 - DROP TABLE
 - ALTER TABLE
- Βασικές ΓΧΔ εντολές της SQL:
 - INSERT
 - DELETE
 - UPDATE
 - SELECT
- Στην SQL ένας πίνακας είναι δυνατό να περιλαμβάνει την ίδια γραμμή περισσότερες από μία φορές (πολυσύνολο - multiset)

5

Δημιουργία Πίνακα

- **CREATE TABLE** όνομα_πίνακα(
Όνομα στήλης τύπος δεδομένων [NULL|NOT NULL] [,...])
 - όπου τύπος δεδομένων
 - INTEGER
 - SMALLINT
 - FLOAT(p)
 - DECIMAL(p, q)
 - CHAR(n)
 - VARCHAR(n)
 - DATE
 - TIME

6

Παράδειγμα Σχήματος

ΟΔΗΓΟΣ

<u>ΑρΑδείας</u>	Επώνυμο	Όνομα	Ημερ'Έκδοσης
-----------------	---------	-------	--------------

ΑΥΤΟΚΙΝΗΤΟ

<u>ΑρΚυκλοφορίας</u>	Μάρκα	Μοντέλο	Οδηγός
----------------------	-------	---------	--------



7

Πρωτεύον κλειδί – PRIMARY KEY

CREATE TABLE ΟΔΗΓΟΣ(

```
ΑρΑδείας INTEGER NOT NULL,  
Επώνυμο VARCHAR (50),  
Όνομα VARCHAR(50),  
Ημερ'Έκδοσης DATE,  
PRIMARY KEY(ΑρΔιπλώματος) )
```

8

Ξένο Κλειδί – FOREIGN KEY

```
CREATE TABLE ΑΥΤΟΚΙΝΗΤΟ(  
  ΑρΚυκλοφορίας INTEGER NOT NULL,  
  Μάρκα VARCHAR (20),  
  Μοντέλο VARCHAR(15),  
  PRIMARY KEY(ΑρΚυκλοφορίας)  
  FOREIGN KEY (Οδηγός) REFERENCES ΟΔΗΓΟΣ(ΑρΑδείας)  
  ON DELETE SET NULL)  
  
  ON DELETE|UPDATE {CASCADE|NO ACTION|SET NULL}
```

9

Εισαγωγή – Διαγραφή - Τροποποίηση

```
INSERT INTO ΟΔΗΓΟΣ VALUES(  
  967654,  
  'ΓΕΩΡΓΙΑΔΗΣ',  
  'ΑΝΤΩΝΙΟΣ',  
  '4/5/1990')
```

```
DELETE ΟΔΗΓΟΣ  
WHERE ΑρΑδείας = 967654
```

```
UPDATE ΟΔΗΓΟΣ  
SET Όνομα = 'ΓΕΩΡΓΙΟΣ'  
WHERE ΑρΑδείας = 967654
```

10

Ερωτήσεις στην SQL: Η εντολή SELECT

Βασική συντακτική δομή:

```
SELECT <λίστα πεδίων>  
FROM <λίστα πινάκων>  
[WHERE <σύνθετη συνθήκη>]
```

Παράδειγμα

Πίνακας 1: **Εργαζόμενος** (ΑΦΜ, Όνομα, Επώνυμο, Μισθός, Τμήμα)

```
SELECT Μισθός  
FROM Εργαζόμενος  
WHERE ΑΦΜ=079593828
```

Στη σχεσιακή άλγεβρα: $\pi_{\text{Μισθός}} (\sigma_{\text{ΑΦΜ}=079593828}(\text{Εργαζόμενος}))$

11

Η δομή της SELECT

- **select** αντιστοιχεί στην πράξη της προβολής της σχεσιακής άλγεβρας
 - ποια γνωρίσματα θέλουμε να υπάρχουν στο αποτέλεσμα της ερώτησης
- **from** αντιστοιχεί στην πράξη του καρτεσιανού γινομένου της σχεσιακής άλγεβρας
 - ποιες σχέσεις θα χρησιμοποιηθούν για τον υπολογισμό του αποτελέσματος
- **where** αντιστοιχεί στη συνθήκη της πράξης της επιλογής στη σχεσιακή άλγεβρα
 - η **συνθήκη** περιλαμβάνει γνωρίσματα των σχέσεων που εμφανίζονται στο from

12

Τελεστές συνθήκης

Λογικοί τελεστές: **and, or, not**

Τελεστές σύγκρισης: **<, <=, >, >=, =, <>, between, not between**

ανάμεσα σε αριθμητικές εκφράσεις, συμβολοσειρές (strings), και ειδικούς τύπους.

Πράξεις με Συμβολοσειρές

Η πιο συνηθισμένη πράξη είναι ταίριασμα προτύπων:

% ταιριάζει οποιαδήποτε συμβολοσειρά

_ ταιριάζει οποιοδήποτε χαρακτήρα

Γίνεται διάκριση ανάμεσα σε κεφαλαία και μικρά

Σύγκριση χρησιμοποιώντας το **like, not like**

13

Πράξεις με Συμβολοσειρές

Ο Τελεστής **LIKE**

Πίνακας 1: Εργαζόμενος (AΦM, Όνομα, Επώνυμο, Μισθός, Τμήμα)

(A) Εμφάνισε τα ονόματα όλων των εργαζομένων που έχουν την κατάληξη -όπουλος

```
SELECT DISTINCT Όνομα, Επώνυμο
FROM Εργαζόμενος
WHERE Επώνυμο LIKE "%όπουλος"
```

(B) Εμφάνισε όλους τους εργαζόμενους που το επώνυμό τους αρχίζει από "Π"

```
SELECT *
FROM Εργαζόμενος
WHERE Επώνυμο LIKE "Π%"
```

14

Join πινάκων

Πίνακας 1: Εργαζόμενος (ΑΦΜ, Όνομα, Επώνυμο, Μισθός, Τμήμα)

Πίνακας 2: Τμήμα (Κωδικός, Ονομασία, Περιοχή)

Εμφάνιση του ονοματεπώνυμου των εργαζομένων και της ονομασίας του τμήματος όπου εργάζονται

```
SELECT Όνομα, Επώνυμο, Ονομασία
FROM Εργαζόμενος, Τμήμα
WHERE Τμήμα = Κωδικός
```

Στη σχεσιακή άλγεβρα:

$\pi_{\text{Όνομα, Επώνυμο, Ονομασία}}(\text{Εργαζόμενος} \bowtie_{\text{Τμήμα = Κωδικός}} \text{Τμήμα})$

15

Άλλες ιδιότητες της εντολής SELECT

- Η πρόταση WHERE είναι **προαιρετική**
- Όταν η λίστα πινάκων περιέχει δύο ή περισσότερους πίνακες, χωρίς να υπάρχει πρόταση WHERE, τότε η πράξη ισοδυναμεί με το καρτεσιανό γινόμενο των πινάκων
- Για να εμφανιστούν όλα τα πεδία ενός πίνακα:

```
SELECT *
FROM Εργαζόμενος
```
- Για να παραληφθούν οι όμοιες πολλαπλές γραμμές και να εμφανιστεί μόνο μία εξ' αυτών:

```
SELECT DISTINCT Μισθός
FROM Εργαζόμενος
```
- Είναι δυνατές πράξεις στα πεδία του τμήματος SELECT

```
SELECT Επώνυμο, Όνομα, Μισθός*1,1
FROM Εργαζόμενος
```

(εμφανίζει τον μισθό των εργαζομένων αυξημένο κατά 10%)

16

Άλλες ιδιότητες της εντολής SELECT

- Σύγκριση με την ειδική τιμή **NULL**

```
SELECT *  
FROM Εργαζόμενος  
WHERE Τμήμα IS NOT NULL
```

(εμφανίζει τους εργαζόμενους που έχουν ενταχθεί σε τμήμα)

```
SELECT *  
FROM Εργαζόμενος  
WHERE Μισθός IS NULL
```

(εμφανίζει τους εργαζόμενους που δεν έχει ορισθεί ο μισθός τους)

17

Συναθροιστικές Συναρτήσεις

- Παραδείγματα συναθροιστικών συναρτήσεων της SQL:
 - Μέσος όρος: **avg(A)** (μόνο σε αριθμούς) A γνώρισμα
 - Ελάχιστο: **min(A)**
 - Μέγιστο: **max(A)**
 - Άθροισμα: **sum(A)** (μόνο σε αριθμούς)
 - Πλήθος: **count(A)**

18

Ομαδοποίηση εγγραφών: group by

- Ομαδοποιεί τις εγγραφές με βάση την τιμή που έχουν σε ένα ή περισσότερα πεδία.
- Οι προς ομαδοποίηση εγγραφές υπολογίζονται από τα τμήματα FROM και WHERE της εντολής SELECT.
- Οι ομάδες υπολογίζονται από το τμήμα **GROUP BY** της εντολής SELECT. Το αποτέλεσμα της εντολής αποτελείται από μία εγγραφή για κάθε ομάδα.
- Τα πεδία που περιλαμβάνονται στο τμήμα SELECT πρέπει είτε να περιλαμβάνονται και στο τμήμα GROUP BY είτε να είναι συναθροιστικές συναρτήσεις.

19

Παράδειγμα: group by

Πίνακας 1: **Εργαζόμενος** (ΑΦΜ, Όνομα, Επώνυμο, Μισθός, Τμήμα)

Υπολογίστε το μέσο όρο του μισθού των εργαζομένων κάθε τμήματος

```
SELECT Τμήμα, AVG(Μισθός)  
FROM Εργαζόμενος  
GROUP BY Τμήμα
```

Για κάθε τμήμα να βρεθεί ο κωδικός του τμήματος το πλήθος των εργαζομένων στο τμήμα και ο μέσος μισθός τους

```
SELECT Τμήμα, COUNT(*), AVG(Μισθός)  
FROM Εργαζόμενος  
GROUP BY Τμήμα
```

20

Επιβολή περιορισμών: group by – having

Περιορισμοί στις ομάδες του αποτελέσματος επιβάλλονται από το τμήμα **HAVING** της εντολής SELECT.

Παράδειγμα

Πίνακας 1: **Εργαζόμενος** (ΑΦΜ, Όνομα, Επώνυμο, Μισθός, Τμήμα)

Πίνακας 2: **Τμήμα** (Κωδικός, Ονομασία, Περιοχή)

Εμφανίστε τους κωδικούς και τις ονομασίες των Τμημάτων με περισσότερους από 2 εργαζομένους

```
SELECT Κωδικός, Ονομασία
FROM Εργαζόμενος, Τμήμα
WHERE Τμήμα = Κωδικός
GROUP BY Κωδικός, Ονομασία
HAVING count(ΑΦΜ) >= 3
```

21

Παράδειγμα: group by - having

ΠΟΛΗ

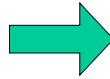
A/A	ΟΝΟΜΑ	ΝΟΜΟΣ
90	ΜΥΤΙΛΗΝΗ	ΛΕΣΒΟΥ
91	ΒΡΟΝΤΑΔΟΣ	ΧΙΟΥ
92	ΣΑΜΟΣ	ΣΑΜΟΥ
93	ΑΓΙΑΣΟΣ	ΛΕΣΒΟΥ
94	ΠΥΡΓΙ	ΧΙΟΥ
95	ΚΑΛΛΟΝΗ	ΛΕΣΒΟΥ

Πόσες πόλεις υπάρχουν για κάθε νομό;

22

ΠΟΛΗ

A/A	ΟΝΟΜΑ	ΝΟΜΟΣ
90	ΜΥΤΙΛΗΝΗ	ΛΕΣΒΟΥ
91	ΒΡΟΝΤΑΔΟΣ	ΧΙΟΥ
92	ΣΑΜΟΣ	ΣΑΜΟΥ
93	ΑΓΙΑΣΟΣ	ΛΕΣΒΟΥ
94	ΠΥΡΓΙ	ΧΙΟΥ
95	ΚΑΛΛΟΝΗ	ΛΕΣΒΟΥ



GROUP BY ΝΟΜΟΣ

ΝΟΜΟΣ	COUNT
ΛΕΣΒΟΥ	3
ΧΙΟΥ	2
ΣΑΜΟΥ	1



```
SELECT ΝΟΜΟΣ, COUNT(A/A)
FROM ΠΟΛΗ
GROUP BY ΝΟΜΟΣ
HAVING COUNT(A/A) >= 2
```

HAVING COUNT(A/A) >= 2

ΝΟΜΟΣ	COUNT
ΛΕΣΒΟΥ	3
ΧΙΟΥ	2

23

Εντολή SELECT



Πλήρης συντακτική δομή:

```
SELECT <λίστα πεδίων>
FROM <λίστα πινάκων>
[WHERE <σύνθετη συνθήκη>]
[GROUP BY <λίστα πεδίων>]
[HAVING <σύνθετη συνθήκη>]
[ORDER BY <λίστα πεδίων>]
```

24

Παράδειγμα: Επιχείρηση μεταποίησης αγροτικών προϊόντων

Για την διαχείριση των δεδομένων μιας επιχείρησης μεταποίησης αγροτικών προϊόντων, κατασκευάστηκε η ακόλουθη σχεσιακή βάση δεδομένων:

ergazom (afm, onoma, eponymo, misthos, proistam, tmima)

tmima (kodikos, onomasia, perioxi)

proion (kodikos, onomasia, mon_met)

paragogh (kod_proion, kod_tmima, hmer_parag)

Τα ξένα κλειδιά που υπάρχουν είναι:

Πίνακας	Ξένο κλειδί	Πίνακας που αναφέρεται
Ergazom	tmima	tmima
Ergazom	proistam	ergazom
Paragogh	kod_proion	proion
Paragogh	kod_tmima	tmima

25

Παραδείγματα συναθροιστικών συναρτήσεων (1)

Να βρεθεί το άθροισμα των μισθών όλων των εργαζομένων, καθώς και ο μέγιστος, ο ελάχιστος και ο μέσος μισθός.

```
SELECT SUM(misthos), MAX(misthos),  
        MIN(misthos), AVG(misthos)  
FROM   ergazom;
```

Να βρεθεί το άθροισμα των μισθών όλων των εργαζομένων στο τμήμα 'ΠΑΡΑΓΩΓΗ', καθώς και ο μέγιστος, ο ελάχιστος μισθός στο τμήμα αυτό.

```
SELECT SUM(misthos), MAX(misthos),  
        MIN(misthos), AVG(misthos)  
FROM   ergazom, tmima  
WHERE  e_tmima=kodikos AND onomasia='ΠΑΡΑΓΩΓΗ';
```

26

Παραδείγματα συναθροιστικών συναρτήσεων (2)

Να βρεθεί ο συνολικός αριθμός των εργαζομένων στην επιχείρηση και ο αριθμός των εργαζομένων στο τμήμα 'ΠΑΡΑΓΩΓΗ' .

```
SELECT COUNT (*)  
FROM   ergazom;
```

```
SELECT COUNT (*)  
FROM   ergazom, tmima  
WHERE  e_tmima=kodikos AND onomasia='ΠΑΡΑΓΩΓΗ';
```

27

Παραδείγματα συναθροιστικών συναρτήσεων (3)

Να βρεθεί το πλήθος των διαφορετικών μισθών στη βάση δεδομένων .

```
SELECT COUNT (DISTINCT misthos)  
FROM   ergazom;
```

Να ανακτηθούν τα ονόματα όλων των τμημάτων που έχουν περισσότερους από 10 εργαζόμενους .

```
SELECT tmima.onomasia  
FROM   tmima  
WHERE  (SELECT COUNT(*)  
        FROM ergazomenos  
        WHERE ergazomenos.e_tmima=tmima.kodikos) >=10 ;
```

Ερώτημα 1

Επιλέξτε τον ΑΦΜ, το επώνυμο και το όνομα των εργαζομένων που εργάζονται στο τμήμα με κωδικό αριθμό 1, ταξινομημένα κατά αύξουσα σειρά του επωνύμου.



```
SELECT afm, eponymo, onoma  
FROM ergazom  
WHERE tmima=1  
ORDER BY eponymo
```

29

Ερώτημα 2

Εμφανίστε τον ΑΦΜ, το όνομα, το επώνυμο και τον μισθό κάθε εργαζόμενου, κατά φθίνουσα σειρά του μισθού, αρκεί να έχει μισθό άνω των τριακοσίων χιλιάδων δραχμών.

```
SELECT afm, eponymo, onoma, misthos  
FROM ergazom  
WHERE misthos > 300000  
ORDER BY misthos DESC
```



30

Ερώτημα 3,4

Υπολογίστε το καρτεσιανό γινόμενο των πινάκων `ergazom` και `tmima`, και εμφανίστε μόνο το επώνυμο και όνομα του εργαζομένου και την ονομασία του τμήματος.

```
SELECT eponymo, onoma, onomasia  
FROM ergazom, tmima
```

Εμφανίστε όλα τα χαρακτηριστικά όλων των προϊόντων της επιχείρησης.

```
SELECT *  
FROM proion
```



31