



Εισαγωγή στις Βάσεις Δεδομένων

Δρ. Ακουμιανάκης Δημοσθένης

Ατζέντα

✓ Εισαγωγικές έννοιες

- Δεδομένα και πληροφορία

✓ Εννοιολογικός σχεδιασμός βάσης δεδομένων

- Το μοντέλο Οντοτήτων – Συσχετίσεων (βασική και εκτεταμένη έκδοση)

✓ Λογική σχεδίαση

- Το σχεσιακό μοντέλο
- Μετατροπές (από ΟΣ σε σχεσιακό σε αντίστροφα)

❖ Σχεσιακές γλώσσες

- Ορισμός & επεξεργασία δεδομένων
 - Υποβολή ερωτήσεων



Η γλώσσα SQL

Η γλώσσα SQL

- ❖ Η “standard” γλώσσα για σχεσιακές βάσεις δεδομένων
- ❖ Αρχικά *Sequel* στην IBM ως μέρος του έργου System R
- ❖ Σήμερα γνωστή με τον όρο SQL (Structured Query Language)
 - Εκδόσεις
 - SQL89
 - SQL92
- ❖ SQL99

Τμήματα της SQL

- ✓ *Γλώσσα Ορισμού Δεδομένων (ΓΟΔ)*
 - Δημιουργία, τροποποίηση και διαγραφή σχήματος
- ✓ **Γλώσσα Χειρισμού Δεδομένων (ΓΧΔ)**
 - Δημιουργία, τροποποίηση, διαγραφή και επιλογή δεδομένων (γλώσσα ερωτήσεων)
- ✓ **Ορισμό Όψεων**
 - ❖ *Ενσωματωμένη Γλώσσα Χειρισμού Δεδομένων*
 - ❖ *Εξουσιοδότηση (authentication)*
 - ❖ *Ακεραιότητα*
 - ❖ *Έλεγχο Συναλλαγών*



Γλώσσα ορισμού δεδομένων (ΓΟΔ)

Δημιουργία σχεσιακού σχήματος, πεδία ορισμού
στηλών / γνωρισμάτων, τύποι δεδομένων,
ακεραιότητα γνωρισμάτων και πινάκων

Γλώσσα ορισμού δεδομένων – Τι είναι

- ❖ Είναι ένα υποσύνολο της SQL που επιτρέπει τον ορισμό
 - σχήματος και πινάκων μιας βάσης
 - πεδίου τιμών
 - περιορισμών ακεραιότητας

CREATE SCHEMA

- ❖ Δημιουργεί ένα νέο schema (με μοναδικό όνομα) στη βάση δεδομένων
- ❖ Ένα schema είναι ουσιαστικά ένα namespace

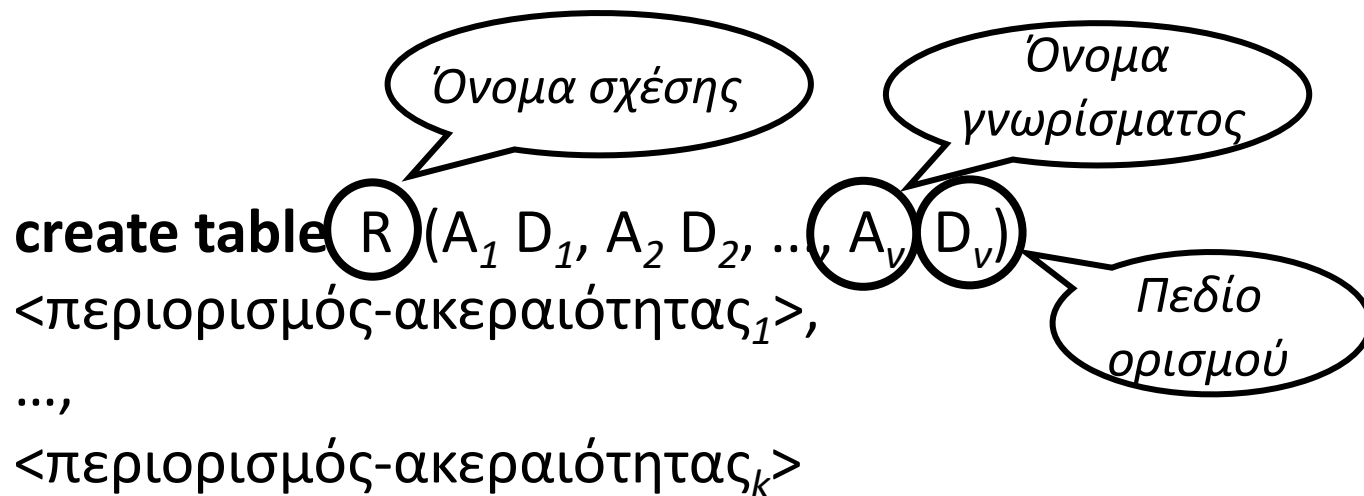
CREATE SCHEMA *νέο-σχήμα*
AUTHORIZATION *όνομα-χρήστη*

- ❖ Μια βάση δεδομένων μπορεί να περιλαμβάνει ένα ή και περισσότερα schemas
 - Ένα schema ανήκει *μόνο σε μια* βάση δεδομένων
 - Η postgres δημιουργεί ένα *public* schema για κάθε νέα βάση δεδομένων

Παράδειγμα ορισμού σχέσης

❖ Ορισμός νέου πίνακα στην SQL

- Ο πίνακας είναι αρχικά άδειος



Τύποι πεδίων ορισμού

- ❖ Διαθέσιμοι (built-in) τύποι δεδομένων
 - **char(n)** (σταθερού μήκους)
 - **varchar(n)**
 - **int**
 - **smallint**
 - **numeric(p, d)** (d από τα p ψηφία είναι στα δεξιά της υποδιαστολής)
 - **real, double precision**
 - **float(n)**
 - **date** (ημερομηνία)
 - **time** (ώρα)

Ορισμός πεδίου τιμών

- ❖ Στην SQL επιτρέπεται η δημιουργία πεδίου τιμών

```
create domain <name> as <type-description>
```

```
create domain Όνομα-Προσώπου char(20)
```

- ❖ Ορισμός πεδίου τιμών

- **create domain person-name char(20);**

- ✓ Ορίζει το **person-name** σαν ένα πεδίο τιμών αποτελούμενο από strings μήκους 20.

- Το **person-name** μπορεί να χρησιμοποιηθεί σαν τύπος γνωρισμάτων άλλων σχέσεων
- Η κενή τιμή (**null**) ανήκει σε κάθε πεδίο τιμών

Παράδειγμα

❖ Δημιουργία του DOMAIN MYDATA

```
CREATE DOMAIN MYDATA VARCHAR(100);
```

❖ /* Δημιουργία πίνακα {ΞΕΝΟΔΟΧΕΙΟ} */

```
CREATE TABLE "ΞΕΝΟΔΟΧΕΙΟ"  
("ΚΩΔ_Ξ" INT NOT NULL,  
"ΟΝΟΜΑ_Ξ" MYDATA NOT NULL,  
"ΚΑΤΗΓ_Ξ" INT,  
"ΔΙΕΥΘΥΝΣΗ" MYDATA,  
"ΠΟΛΗ" VARCHAR(50),  
"ΠΕΡΙΟΧΗ" MYDATA NOT NULL,  
"ΧΩΡΑ" VARCHAR(50),  
PRIMARY KEY("ΚΩΔ_Ξ"));
```

Περιορισμοί ακεραιότητας

❖ Δύο κατηγορίες

- Περιορισμοί που αφορούν **γνωρίσματα** ενός πίνακα
 - ✓ **not null**
 - ✓ **unique** $A_{j1}, A_{j2}, \dots, A_{jn}$
 - δεν επιτρέπονται επαναλαμβανόμενες τιμές στη στήλη
 - ✓ **check P**
 - Έλεγχος συνθήκης
- Περιορισμοί που αφορούν τον **πίνακα**
 - ✓ **primary key** $A_{j1}, A_{j2}, \dots, A_{jn}$
 - δεν επιτρέπονται επαναλαμβανόμενες τιμές και NULL τιμές
 - ✓ **foreign key (A_i) references A_j**
 - Αναφορική ακεραιότητα

Παράδειγμα

❖ Ο περιορισμός CHECK στον πίνακα ΞΕΝΟΔΟΧΕΙΟ

```
/* Δημιουργία πίνακα {ΞΕΝΟΔΟΧΕΙΟ} */  
CREATE TABLE "ΞΕΝΟΔΟΧΕΙΟ"  
(  
  "ΚΩΔ_Ξ" INT NOT NULL,  
  "ΟΝΟΜΑ_Ξ" MYDATA NOT NULL,  
  "ΚΑΤΗΓ_Ξ" INT,  
  "ΔΙΕΥΘΥΝΣΗ" MYDATA,  
  "ΠΟΛΗ" VARCHAR(50),  
  "ΠΕΡΙΟΧΗ" MYDATA NOT NULL,  
  "ΧΩΡΑ" VARCHAR(50) CHECK ("ΧΩΡΑ" = 'Ελλάδα' OR  
                              "ΧΩΡΑ" = 'Ηνωμένο Βασίλειο' OR  
                              "ΧΩΡΑ" = 'Γαλλική Πολυνησία'),  
  PRIMARY KEY("ΚΩΔ_Ξ"));
```



Τύποι δεδομένων

(Παρουσιάζεται ένα υποσύνολο)

Boolean data type

❖ Τύπος δεδομένων που επιτρέπει *true* ή *false* τιμές

```
83 CREATE TABLE "Μονόκλινο business σουίτα"  
84 ("ΚΩΔ_Ξ" INT NOT NULL,  
85 "ΑΠΙΘ_Δ" INT NOT NULL,  
86 AVAILABILITY BOOLEAN,  
87 FOREIGN KEY ("ΚΩΔ_Ξ") REFERENCES "ΞΕΝΟΔΟΧΕΙΟ" ("ΚΩΔ_Ξ"),  
88 PRIMARY KEY("ΚΩΔ_Ξ", "ΑΠΙΘ_Δ"));  
89  
90 -- 11, 'Μονόκλινο business σουίτα'  
91 INSERT INTO "Μονόκλινο business σουίτα" VALUES (5, 204, TRUE);  
92 INSERT INTO "Μονόκλινο business σουίτα" VALUES (4, 101, FALSE);
```

	ΚΩΔ_Ξ [PK] integer	ΑΠΙΘ_Δ [PK] integer	availability boolean
1	5	204	true
2	4	101	false

DATE

❖ Τύπος δεδομένων που αποθηκεύει την ημερομηνία σε μορφή *yyyy-mm-dd*

```
/* Δημιουργία πίνακα {ΚΡΑΤΗΣΗ} */
```

```
CREATE TABLE "ΚΡΑΤΗΣΗ"  
( "ΚΩΔ_ΠΕ" INT NOT NULL,  
  "ΚΩΔ_Ξ" INT NOT NULL,  
  "ΑΡΙΘ_Δ" INT NOT NULL,  
  "ΚΩΔ_ΠΡ" INT NOT NULL,  
  "ΗΜΕΡ_ΑΦΙΞΗΣ" DATE NOT NULL DEFAULT CURRENT_DATE,  
  "ΔΙΑΝΥΚΤΕΡΕΥΣΕΣ" INT NOT NULL,  
  "ΤΙΜΗ" INT,  
  FOREIGN KEY ("ΚΩΔ_ΠΕ") REFERENCES "ΠΕΛΑΤΗΣ" ("ΚΩΔ_ΠΕ"),  
  FOREIGN KEY ("ΚΩΔ_Ξ", "ΑΡΙΘ_Δ") REFERENCES "ΔΩΜΑΤΙΟ" ("ΚΩΔ_Ξ", "ΑΡΙΘ_Δ"),  
  FOREIGN KEY ("ΚΩΔ_ΠΡ") REFERENCES "ΠΡΑΚΤΟΡΕΙΟ" ("ΚΩΔ_ΠΡ"),  
  PRIMARY KEY ("ΚΩΔ_ΠΕ", "ΚΩΔ_Ξ", "ΑΡΙΘ_Δ", "ΗΜΕΡ_ΑΦΙΞΗΣ"));
```

ΚΩΔ_ΠΕ [PK] integer	ΚΩΔ_Ξ [PK] integer	ΑΡΙΘ_Δ [PK] integer	ΚΩΔ_ΠΡ integer	ΗΜΕΡ_ΑΦΙΞΙ [PK] date
10001	1	111	1001	2022-12-24
10001	1	111	1004	2023-01-20
10005	1	112	1001	2022-12-24

```
INSERT INTO "ΚΡΑΤΗΣΗ" VALUES (10002, 5, 203, 1004,  
TO_DATE('1/1/2023','DD/MM/YYYY'), 2, 240);
```

TIME

❖ Τύπος δεδομένων που αποθηκεύει το χρόνο
χωρίς time zones

```
1 CREATE TABLE Officeshifts (  
2     Shiftid serial PRIMARY KEY,  
3     starttime TIME NOT NULL,  
4     Endtime TIME NOT NULL  
5 );  
6  
7 INSERT INTO officeshifts (starttime, Endtime)  
8 VALUES ('06:00:00', '15:00:00'),  
9         ('13:00:00', '22:00:00'),  
10        ('22:00:00', '06:00:00');  
11  
12 Select * from officeshifts
```

	shiftid [PK] integer		starttime time without time zone		endtime time without time zone	
1		1	06:00:00		15:00:00	
2		2	13:00:00		22:00:00	
3		3	22:00:00		06:00:00	

TIMESTAMP

❖ Το timestamp data type είναι συνδυασμός date και time

1	CREATE TABLE sampletable(	
2	Entrytime TIMESTAMP	
3	);	
4	Insert into sampletable	
5	values('2020-01-12 15:00:00-00');	
6		
7	select * from sampletable	
	entrytime	
	timestamp without time zone	
1	2020-01-12 15:00:00	

Network address type

- ❖ Ο τύπος δεδομένων αποθηκεύει Internet Protocol addresses

```
CREATE TABLE DemoINET (  
    address INET  
);  
INSERT INTO DemoINET (address) VALUES  
('198.24.10.0/24'),  
('198.24.10.0'),  
('198.10/8');  
  
Select * from DemoINET;
```

	address inet	
1	198.24.10.0/24	
2	198.24.10.0	
3	198.10.0.0/8	



Εισαγωγή δεδομένων

(σε πίνακες που έχουν
δημιουργηθεί)

INSERT

❖ Η δήλωση INSERT δημιουργεί μια πλειάδα σε ένα πίνακα

- Νέες πλειάδες μπορεί να δημιουργηθούν με εναλλακτικούς τρόπους

- ✓ Εισαγωγή πλειάδας με βάση τον ορισμό του πίνακα

INSERT INTO XYZ VALUES ('Τιμή 1', 'Τιμή 2', ..., 'Τιμή ν');

- ✓ Εισαγωγή πλειάδας με στήλες που κατονομάζονται

INSERT INTO XYZ (col 1, col 2) VALUES ('Τιμή 1', 'Τιμή 2');

Στήλες που δεν ονομάζονται λαμβάνουν default τιμή

- ✓ Εισαγωγή πλειάδας με default τιμές σε στήλες

INSERT INTO XYZ VALUES ('Τιμή 1', DEFAULT, ..., 'Τιμή ν');

- ✓ Με τη χρήση SELECT – FROM – WHERE



Γλώσσας Χειρισμού Δεδομένων - Βασική σύνταξη

Βασική δομή

- ❖ Μια χαρακτηριστική ερώτηση σε SQL έχει την εξής μορφή:

ονόματα γνωρισμάτων

select $A_1, A_2, \dots, A_n$

ονόματα σχέσεων

from $R_1, R_2, \dots, R_m$

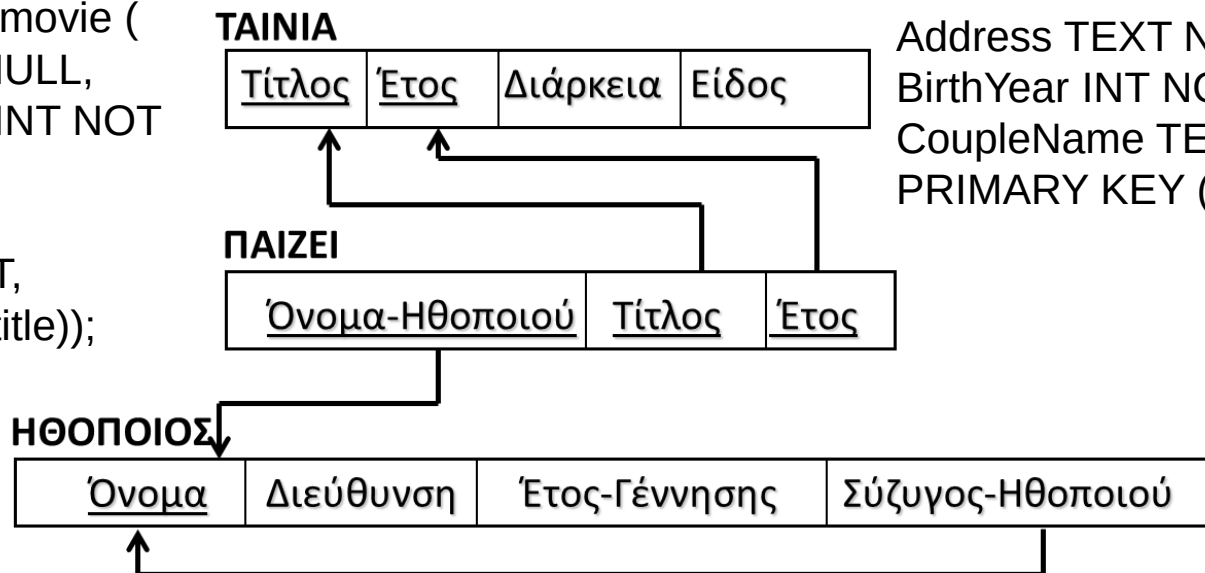
συνθήκη

where P

Παράδειγμα σχήματος

❖ Ας θεωρήσουμε το σχήμα

```
CREATE TABLE movie (  
  title TEXT NOT NULL,  
  production_year INT NOT  
  NULL,  
  duration INT,  
  colour_type TEXT,  
  PRIMARY KEY (title));
```



```
CREATE TABLE actor (  
  ActorName TEXT NOT NULL,  
  Address TEXT NOT NULL,  
  BirthYear INT NOT NULL,  
  CoupleName TEXT,  
  PRIMARY KEY (ActorName));
```

```
CREATE TABLE plays (  
  Actor TEXT NOT NULL,  
  Movie TEXT NOT NULL,  
  Etos INT NOT NULL,  
  PRIMARY KEY (Actor, Movie),  
  FOREIGN KEY (Actor) REFERENCES actor(ActorName),  
  FOREIGN KEY (Movie) REFERENCES movie(title));
```

Παράδειγμα ερώτησης

- ❖ Ονόματα όλων των ηθοποιών που έχουν παίξει σε ταινίες

```
select Actor  
from plays;
```

actor text	
Sharlto Copley	
Sharlto Copley	
Bruce Willis	
Jason Cope	
Jason Cope	

Παράδειγμα ερώτησης

- ❖ Ονόματα ηθοποιών που παίζουν στην ταινία 'V for Vendetta';

```
select Actor  
from plays  
where Movie = 'V for Vendetta';
```

actor	
text	
Bruce Willis	
Jason Cope	

Παράδειγμα ερώτησης

- ❖ Ονόματα όλων των ηθοποιών που έχουν παίξει σε ασπρόμαυρες ταινίες

```
select Actor
from   plays, movie
where  plays.Movie = movie.title AND
       movie.colour_type = 'Black and White' AND
       movie.production_year= plays.Etos;
```

actor	
text	
Sharlto Copley	
Jason Cope	

Απαλοιφή διπλών εμφανίσεων

- ❖ Η SQL επιτρέπει πολλαπλές εμφανίσεις της ίδιας πλειάδας σε μια σχέση

```
99 select Actor
100 from plays;
101
```

	actor text
1	Sharlto Copley
2	Sharlto Copley
3	Bruce Willis
4	Jason Cope
5	Jason Cope

- ❖ Για την απαλοιφή διπλών εμφανίσεων μπορούμε να δηλώσουμε

```
99 select distinct Actor
.00 from plays;
.01
```

	actor text
1	Bruce Willis
2	Jason Cope
3	Sharlto Copley

Επιλογή όλων των γνωρισμάτων

- ❖ Σε περίπτωση που δεν απαιτείται προβολή επιμέρους γνωρισμάτων

```
.01  
.02 select * From plays;
```

	actor [PK] text	movie [PK] text	etos integer
1	Sharlto Copley	District 9	2009
2	Sharlto Copley	Moon	2009
3	Bruce Willis	V for Vendetta	2009
4	Jason Cope	V for Vendetta	2005
5	Jason Cope	Surrogates	2009

Αριθμητικές πράξεις

- ❖ Το ερώτημα παρακάτω ερώτημα επιστρέφει μια σχέση ίδια με τη σχέση `movie` μόνο που το γνώρισμα διάρκεια μας δίνει τις ώρες (έχει διαιρεθεί με το 60)

```
104 select title, production_year, duration / 60, colour_type
105 from movie;
```

	title [PK] text	production_year integer	?column? integer	colour_type text
1	District 9	2009	1	Colour
2	Moon	2009	1	Black and White
3	Surrogates	2009	1	Black and White
4	Star Trek	2009	2	Colour
5	V for Vendetta	2005	2	Colour
6	The Truman Show	1998	1	Colour
7	Star Wars	1977	2	Colour
8	Blade Runner	1982	1	Colour
9	Terminator 2: Judgment Day	1991	2	Colour

Η συνθήκη του where

❖ Η συνθήκη επιλογής μπορεί να περιλαμβάνει

- Λογικούς τελεστές

- ✓ and, or, not

- Τελεστές σύγκρισης

- ✓ <, <=, >, >=, =, <>, between, not between ανάμεσα σε αριθμητικές εκφράσεις, συμβολοσειρές (strings), και ειδικούς τύπους

Επιλογή - where

- ❖ Το ερώτημα «Εμφάνισε τον τίτλο όλων των ταινιών που γυρίστηκαν μετά το 1000 και είναι ασπρόμαυρες» δηλώνεται ως εξής

```
107 select title
108 from movie
109 where production_year > 1000 AND
110        colour_type = 'Black and White';
```

Data Output		Messages	Notifications
        			
	title [PK] text 		
1	Moon		
2	Surrogates		

BETWEEN

❖ Χρήση του between

```
select title
from movie
where production_year between 2000 and 2020;
```

Data Output		Messages	Notifications
        			
	title [PK] text 		
1	District 9		
2	Moon		
3	Surrogates		
4	Star Trek		
5	V for Vendetta		

❖ Ισοδύναμο του

```
select title
from movie
where production_year >= 2000 and
      production_year <= 2020;
```

Data Output		Messages	Notifications
        			
	title [PK] text 		
1	District 9		
2	Moon		
3	Surrogates		
4	Star Trek		
5	V for Vendetta		

Διαχωρισμός γνωρισμάτων

- ❖ Όταν το ίδιο γνώρισμα εμφανίζεται στο σχήμα περισσότερων από μια σχέσεων, τότε διάκριση βάση του συμβολισμού:

<όνομα-σχέσης>.<όνομα-γνωρίσματος>

- ❖ Βρες τους ηθοποιούς που παίζουν σε ασπρόμαυρες ταινίες

```
select distinct Actor
from plays, movie
where plays.movie = movie.title and
      plays.Etos = movie.production_year and
      colour_type = 'Black and White';
```

Data Output		Messages	Notifications
	actor text		
1	Jason Cope		
2	Sharlto Copley		

Πράξεις με συμβολοσειρές

- ❖ Η πιο συνηθισμένη πράξη είναι ταίριασμα προτύπων:
 - % ταιριάζει οποιαδήποτε συμβολοσειρά
 - _ ταιριάζει οποιοδήποτε χαρακτήρα
- ❖ Γίνεται διάκριση ανάμεσα σε κεφαλαία / μικρά
- ❖ Σύγκριση χρησιμοποιώντας το like, not like

Παράδειγμα χρήσης του %

- ❖ Βρες τους τίτλους όλων των ταινιών που περιέχουν τη λέξη 'The'

```
128 select distinct title
129 from movie
130 where title like '%The%'; |
```

Data Output		Messages	Notifications
     			 
	title [PK] text 		
1	The Truman Show		

Διάταξη πλειάδων

- ❖ Χρήση του **order by** ώστε οι πλειάδες στο αποτέλεσμα να είναι ταξινομημένες με βάση το αντίστοιχο γνώρισμα

```
select distinct Movie, Etos
from plays
where Actor = 'Bruce Willis'
order by Etos;
```

Data Output		Messages		Notifications				
								
	movie text		etos integer					
1	V for Vendetta		2009					
2	District 9		2010					

Διάταξη πλειάδων (συν.)

- ❖ Το **default** είναι η αύξουσα διάταξη
- ❖ Είναι όμως δυνατόν να οριστεί η επιθυμητή διάταξη χρησιμοποιώντας το **asc** (αύξουσα) ή το **desc** (φθίνουσα)

```
select *  
from movie  
order by production_year desc, title asc;
```

Data Output Messages Notifications				
	title [PK] text	production_year integer	duration integer	colour_type text
1	District 9	2009	112	Colour
2	Moon	2009	97	Black and White
3	Star Trek	2009	127	Colour
4	Surrogates	2009	88	Black and White
5	V for Vendetta	2005	132	Colour
6	The Truman Show	1998	103	Colour
7	Terminator 2: Judgment Day	1991	137	Colour
8	Blade Runner	1982	117	Colour
9	Star Wars	1977	121	Colour

Αλλαγή ονόματος γνωρίσματος

❖ Τα ονόματα των γνωρισμάτων στο αποτέλεσμα είναι αυτά που ορίστηκαν στην ερώτηση

❖ Υπάρχει η δυνατότητα να ορίσουμε ένα alias για το γνωρίσμα

<Τ

■ Το **as** μι

❖ Παράδειγμα

	title [PK] text	production_year integer	Ωρες integer	colour_type text
1	District 9	2009	1	Colour
2	Moon	2009	1	Black and White
3	Surrogates	2009	1	Black and White
4	Star Trek	2009	2	Colour
5	V for Vendetta	2005	2	Colour
6	The Truman Show	1998	1	Colour
7	Star Wars	1977	2	Colour
8	Blade Runner	1982	1	Colour
9	Terminator 2: Judgment Day	1991	2	Colour

ς

:

om

```
select title, production_year, duration / 60 as Ωρες, colour_type  
from movie;
```


Μεταβλητές πλειάδων

- ❖ Μια μεταβλητή πλειάδας μπορεί να οριστεί στο from χρησιμοποιώντας το as

```
145 select distinct Actor
146 from plays as P, movie as M
147 where P.movie = M.title and
148        P.Etos = M.production_year and
149        colour_type = 'Black and White';
```

Data Output			Messages	Notifications
        				
	actor	text		
1	Jason Cope			
2	Sharlto Copley			

Πράξεις συνόλων

❖ Η γενική σύνταξη είναι

(select from where)
union/intersection/except
(select from where)

❖ Απαιτείται συμβατότητα των σχέσεων

Πράξεις συνόλων – Σχήμα

❖ Έστω το σχήμα



Πράξεις συνόλων - Παραδείγματα

❖ Βρες τα ονόματα όλων των πελατών που έχουν καταθέσεις ή/και έχουν πάρει δάνειο

(select Όνομα-Πελάτη
from Καταθέτης)

union

(select Όνομα-Πελάτη
from Δανειζόμενος)

Πράξεις συνόλων - Παραδείγματα

- ❖ Βρες τα ονόματα όλων των πελατών που έχουν καταθέσεις και έχουν πάρει δάνειο

(select Όνομα-Πελάτη
from Καταθέτης)

intersect

(select Όνομα-Πελάτη
from Δανειζόμενος)



Εμφωλευμένες υποερωτήσεις

Εξωτερικό ερώτημα και υποερώτημα

❖ Η SQL επιτρέπει το φώλιασμα υποερωτήσεων

- Μια υποερώτηση είναι μια έκφραση select-from-where που χρησιμοποιείται μέσα σε μια άλλη ερώτηση
- Υπολογίζεται η υποερώτηση για κάθε γραμμή (πλειάδα) της εξωτερικής ερώτησης

select ...

from ...

where

(select ...

from ...

where ...);



Εξωτερική ερώτηση

***Φωλιασμένη
υποερώτηση***

Εξωτερικό ερώτημα και υποερώτημα

- ❖ Ο τελεστής **in (not in)** ελέγχει αν μια πλειάδα ανήκει (δεν ανήκει) σε ένα σύνολο πλειάδων που προκύπτουν από έκφραση select-from-where
- ❖ Παράδειγμα: Τα ονόματα όλων των πελατών που έχουν πάρει δάνειο και έχουν καταθέσεις
select distinct Όνομα-Πελάτη
from Δανειζόμενος
where Όνομα-Πελάτη **in**
(select Όνομα-Πελάτη from Καταθέτης)

Παραπάνω από δύο γνωρίσματα

❖ Παράδειγμα: Τα ονόματα όλων των πελατών που έχουν πάρει δάνειο και έχουν καταθέσεις στο υποκατάστημα Ψηλά-Αλώνια

```
select distinct Όνομα-Πελάτη  
from Δανειζόμενος, Δάνειο  
where Δανειζόμενος.Αριθμός-Δανείου = Δάνειο.Αριθμός.Δανείου  
and Όνομα-Υποκαταστήματος = "Ψηλά-Αλώνια"  
and (Όνομα-Υποκαταστήματος, Όνομα-Πελάτη) in
```

```
(select Όνομα-Υποκαταστήματος, Όνομα-Πελάτη  
from Καταθέτης, Λογαριασμός  
where Καταθέτης.Αριθμός-Λογαριασμού =  
Λογαριασμός.Αριθμός-Λογαριασμού)
```

Χρήση σε διατεταγμένα σύνολα

- ❖ Παράδειγμα: Τα ονόματα όλων των πελατών που έχουν πάρει δάνειο και δεν λέγονται “Παπαδόπουλος” ή “Πέτρου”

select distinct Όνομα-Πελάτη

from Δανειζόμενος

where Όνομα-Πελάτη **not in**

(“Παπαδόπουλος” , “Πέτρου”)

Διατεταγμένο
σύνολο



Σύγκριση συνόλων

- ❖ Ο τελεστής **some (any)** έχει τη σημασία του **τουλάχιστον ένα** από ένα σύνολο
- ❖ Βρες τα ονόματα όλων των υποκαταστημάτων που έχουν καταθέσεις μεγαλύτερες από τις καταθέσεις ενός τουλάχιστον υποκαταστήματος στο Ηράκλειο

```
select distinct Όνομα-Υποκαταστήματος  
from Υποκατάστημα  
where Ποσό > some (select Ποσό  
                    from Υποκατάστημα  
                    where Πόλη = “Ηράκλειο”)
```

Ο τελεστής all

- ❖ Ο τελεστής **all** συγκρίνει με όλα τα στοιχεία ενός συνόλου
- ❖ Βρες τα ονόματα των υποκαταστημάτων που έχουν καταθέσεις μεγαλύτερες από τις καταθέσεις των υποκαταστημάτων στο Ηράκλειο

```
select distinct Όνομα-Υποκαταστήματος  
from Υποκατάστημα  
where Ποσό > all (select Ποσό  
                  from Υποκατάστημα  
                  where Πόλη = “Ηράκλειο”)
```

Ο τελεστής all

❖ Έστω το σχήμα

Ταινία (Τίτλος, Έτος, Διάρκεια, Είδος)

Παίζει (Όνομα, Τίτλος, Έτος)

Ηθοποιός (Όνομα, Διεύθυνση, Έτος-Γέννησης, Σύζ-Ηθοποιού)

❖ Βρες τον πιο ηλικιωμένο ηθοποιό που έπαιξε στην ταινία Μανταλένα

select distinct Όνομα

from Ηθοποιός

where Έτος-Γέννησης **<= all** **in** (**select** Έτος-Γέννησης

from Παίζει, Ηθοποιός

where Παίζει.Όνομα = Ηθοποιός.Όνομα

and Τίτλος = «Μανταλένα»

< all
<= all
>= all,
= all,
< > all $\cong$ not in

Ο τελεστής exists

- ❖ Ο τελεστής **exists** επιστρέφει true αν η υποερώτηση δεν είναι κενή
- ❖ Βρες τους πελάτες που έχουν καταθέσεις και έχουν πάρει δάνειο

```
select Όνομα-Πελάτη  
from Δανειζόμενος  
where exists  
  (select *  
   from Καταθέτης  
   where Καταθέτης.Όνομα-Πελάτη =  
         Δανειζόμενος.Όνομα-Πελάτη)
```

Ο τελεστής not exists

- ❖ Ο τελεστής **not exists** μπορεί να χρησιμοποιηθεί για έλεγχο αν η σχέση A περιέχει τη σχέση B
- ❖ Βρες τους ηθοποιούς που έχουν παίξει σε όλες τις ταινίες της Βουγιουκλάκη

```
select distinct S.Όνομα  
from Παίζει as S  
where not exists
```

Υπολογισμός
για κάθε S

```
((select Τίτλος, Έτος  
from Παίζει  
where Όνομα = "Βουγιουκλάκη")  
except  
(select Τίτλος, Έτος  
from Παίζει as R  
where R.Όνομα = S.Όνομα))
```

Ο τελεστής not exists

❖ Βρείτε τους πελάτες που έχουν καταθέσεις σε όλα τα υποκαταστήματα της Πάτρα

```
select distinct S.Όνομα-Πελάτη
from Καταθέτης as S
where not exists ((select Όνομα-Υποκαταστήματος
                    from Υποκατάστημα
                    where Πόλη = "Πάτρα"))
except
(select R.Όνομα-Υποκαταστήματος
 from Καταθέτης as T, Λογαριασμός as R
 where T.Όνομα-Πελάτη = S.Όνομα-Πελάτη
      and T.Αριθμός-Λογαριασμού =
           R.Αριθμός-Λογαριασμού ))
```


Έλεγχος για διπλές εμφανίσεις

- ❖ Ο τελεστής **unique** επιστρέφει true αν η υποερώτηση δεν έχει πολλαπλές όμοιες πλειάδες
- ❖ Έστω το σχήμα
 - Ταινία (Τίτλος, Έτος, Διάρκεια, Είδος)
 - Παίζει(Όνομα, Τίτλος, Έτος)
 - Ηθοποιός(Όνομα, Διεύθυνση, Έτος-Γέννησης, Σύζ-Ηθοποιού)

Έλεγχος για διπλές εμφανίσεις

- ❖ Οι πελάτες που έχουν ακριβώς μια κατάθεση στο υποκατάστημα “Ψηλά Αλώνια”

```
select T.Όνομα-Πελάτη
from Καταθέτης as T
where unique (select R.Όνομα-Πελάτη
              from Λογαριασμός, Καταθέτης as R
              where T.Όνομα-Πελάτη = R.Όνομα-Πελάτη
              and R. Αριθμός-Λογαριασμού =
                  Λογαριασμός. Αριθμός- Λογαριασμού
              and Λογαριασμός.Όνομα-Υποκαταστήματος =
                  “Ψηλά Αλώνια”)
```

Συναθροιστηκές συναρτήσεις

❖ Οι βασικές είναι

- Μέσος όρος: $\text{avg}(A)$ (μόνο σε αριθμούς) A γνώρισμα
- Ελάχιστο: $\text{min}(A)$
- Μέγιστο: $\text{max}(A)$
- Άθροισμα: $\text{sum}(A)$ (μόνο σε αριθμούς)
- Πλήθος: $\text{count}(A)$

Συναθροιστικές συναρτήσεις

- ❖ Μέσο ποσό όλων των λογαριασμών στο υποκατάστημα XYZ

```
select avg(Ποσό)
from Λογαριασμός
where Όνομα-Υποκαταστήματος = "XYZ"
```

- ❖ Μέγιστο ποσό όλων των λογαριασμών στο υποκατάστημα XYZ και τον αριθμό του λογαριασμού!!

```
select Αριθμός-Λογαριασμού, max(Ποσό)
from Λογαριασμός
where Όνομα-Υποκαταστήματος = "XYZ"
```

Τέλος για σήμερα

