



Εισαγωγή στις Βάσεις Δεδομένων

Ατζέντα

❖ Σχεσιακές γλώσσες

- γλώσσες σχεδιασμένες για διαχείριση δεδομένων σε σχεσιακές βάσεις δεδομένων, όπου τα δεδομένα αποθηκεύονται σε πίνακες (σχέσεις) που συνδέονται μεταξύ τους

❖ Αρχικά, με το σχεσιακό μοντέλο προτάθηκαν

- Σχεσιακή Άλγεβρα
- Σχεσιακός Λογισμός (πλειάδων και πεδίου τιμών)

❖ Ακολούθησαν διάφορες εμπορικές γλώσσες με την πλέον επικρατούσα να είναι η SQL



Σχεσιακή άλγεβρα

Βασικοί σχεσιακοί τελεστές

❖ Βασικοί τελεστές

- Ένωση (union)
- Τομή (intersection)
- Διαφορά (difference)
- Καρτεσιανό Γινόμενο (Cartesian product)

Τελεστής Σχεσιακής Άλγεβρας	Σύμβολο
Selection	σ
Projection	Π
Cartesian Product	\times
Join	\bowtie
Union	\cup
Intersection	\cap
Difference	$-$
Logical AND	\wedge
Logical OR	\vee
Logical NOT	\sim
Renaming	ρ

Ειδικοί σχεσιακοί τελεστές

❖ Ειδικοί σχεσιακοί τελεστές

- Select – επιλογή
- Project – προβολή
- Cartesian product
- Join – σύνδεση
- Natural join – φυσική σύνδεση
- Divide (quotient)
- Semi-join

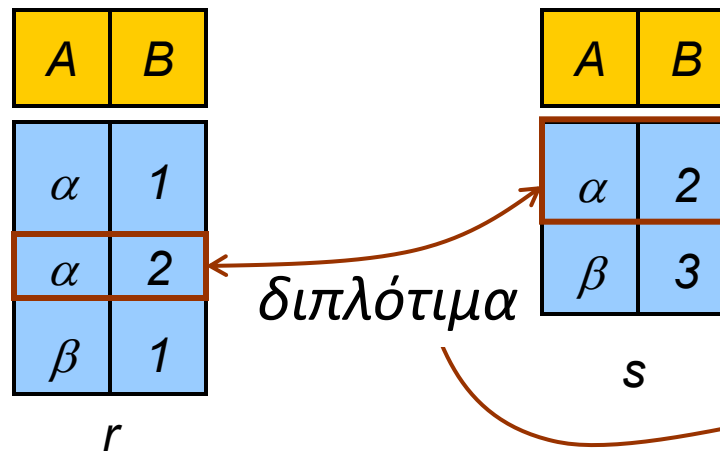
Τελεστής Σχεσιακής Άλγεβρας	Σύμβολο
Selection	σ
Projection	Π
Cartesian Product	\times
Join	\bowtie
Union	\cup
Intersection	\cap
Difference	$-$
Logical AND	\wedge
Logical OR	\vee
Logical NOT	\sim
Renaming	ρ

Η ένωση (\cup)

- ❖ Η ένωση (union) δύο ή παραπάνω σχέσεων κτίζει μια νέα σχέση η οποία αποτελείται από όλες τις διαφορετικές πλειάδες των καθορισμένων σχέσεων
- ❖ Η σχέση που προκύπτει έχει τα ίδια ονόματα με την πρώτη σχέση
- ❖ Απαλοιφή διπλότιμων

Παράδειγμα

❖ Ας θεωρήσουμε δύο σχέσεις r , s



❖ Ο τελεστής $r \cup s$ παράγει τη σχέση

A	B
α	1
α	2
β	1
β	3

Ορισμός ένωσης

❖ Συμβολισμός: $r \cup s$

❖ Ορίζεται ως εξής:

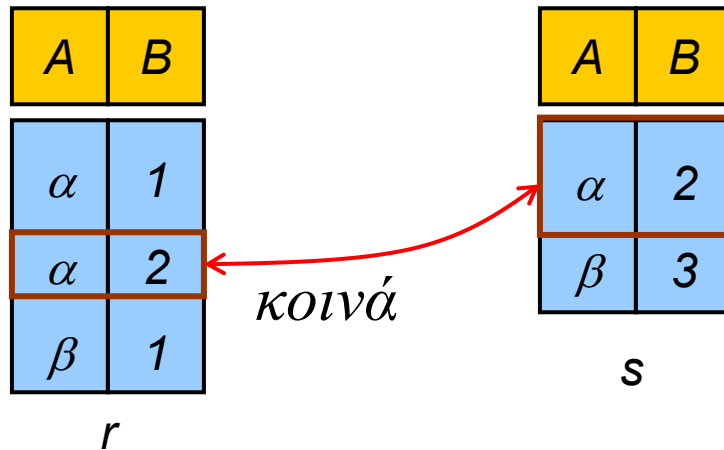
$$r \cup s = \{t \mid t \in r \vee t \in s\}$$

❖ Περιορισμοί ακεραιότητας της ένωση $r \cup s$

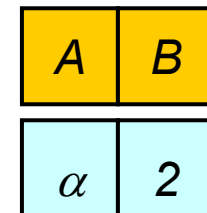
- r, s πρέπει να είναι του ίδιου βαθμού (να έχουν τον ίδιο αριθμό γνωρισμάτων)
- Τα πεδία ορισμού των γνωρισμάτων να είναι συμβατά (δηλ. η i στήλη της r έχει το ίδιο πεδίο ορισμού με την i στήλη της s)

Τομή (\cap)

- ❖ Δημιουργεί μια σχέση που αποτελείται από όλες τις πλειάδες που εμφανίζονται και στις δυο σχέσεις



- ❖ Ο τελεστής $r \cap s$ παράγει τη σχέση



Ορισμός τομής

❖ Συμβολισμός $r \cap s$

❖ Ορισμός:

$$r \cap s = \{t \mid t \in r \wedge t \in s\}$$

❖ Η τομή προϋποθέτει συμβατές σχέσεις

- r και s πρέπει να είναι του ιδίου βαθμού
- Πεδία ορισμού γνωρισμάτων της r και της s πρέπει να είναι συμβατά

Διαφορά (set difference)

- ❖ Δημιουργεί μια σχέση που αποτελείται από όλες τις πλειάδες που εμφανίζονται στην πρώτη και όχι στην δεύτερη από δυο ορισμένες σχέσεις

r

A	B
α	1
α	2
β	1

s

A	B
α	2
β	3

- ❖ Η έκφραση $r - s$ παράγει τη σχέση

A	B
α	1
β	1

Ορισμός διαφοράς

❖ Συμβολισμός $r - s$

❖ Ορισμός:

$$r - s = \{t \mid t \in r \wedge t \notin s\}$$

❖ Διαφορά προϋποθέτει συμβατές σχέσεις

- r και s πρέπει να είναι του ιδίου βαθμού
- Πεδία ορισμού γνωρισμάτων της r και της s πρέπει να είναι συμβατά

Καρτεσιανό Γινόμενο (X)

❖ Δημιουργεί μια σχέση από δυο ορισμένες σχέσεις που αποτελείται από όλους τους δυνατούς συνδυασμούς πλειάδων, μια από κάθε σχέση

■ $\mathbf{t} \in \mathbf{R} \times \mathbf{S}$ εάν

✓ Τα πρώτα $\mathbf{K1}$ στοιχεία του \mathbf{t} αποτελούν μια πλειάδα στο \mathbf{R}

✓ Τα υπόλοιπα $\mathbf{K2}$ στοιχεία του \mathbf{t} αποτελούν μια πλειάδα στο \mathbf{S}

– $\mathbf{K1}$ είναι ο βαθμός της σχέσης \mathbf{R}

– $\mathbf{K2}$ είναι ο βαθμός της σχέσης \mathbf{S}

Καρτεσιανό Γινόμενο – Παράδειγμα

❖ Έστω οι σχέσεις r και s

r	<table border="1"><tr><th>A</th><th>B</th></tr><tr><td>α</td><td>1</td></tr><tr><td>β</td><td>2</td></tr></table>	A	B	α	1	β	2	s	<table border="1"><tr><th>A</th></tr><tr><td>α</td></tr><tr><td>δ</td></tr></table>	A	α	δ
A	B											
α	1											
β	2											
A												
α												
δ												

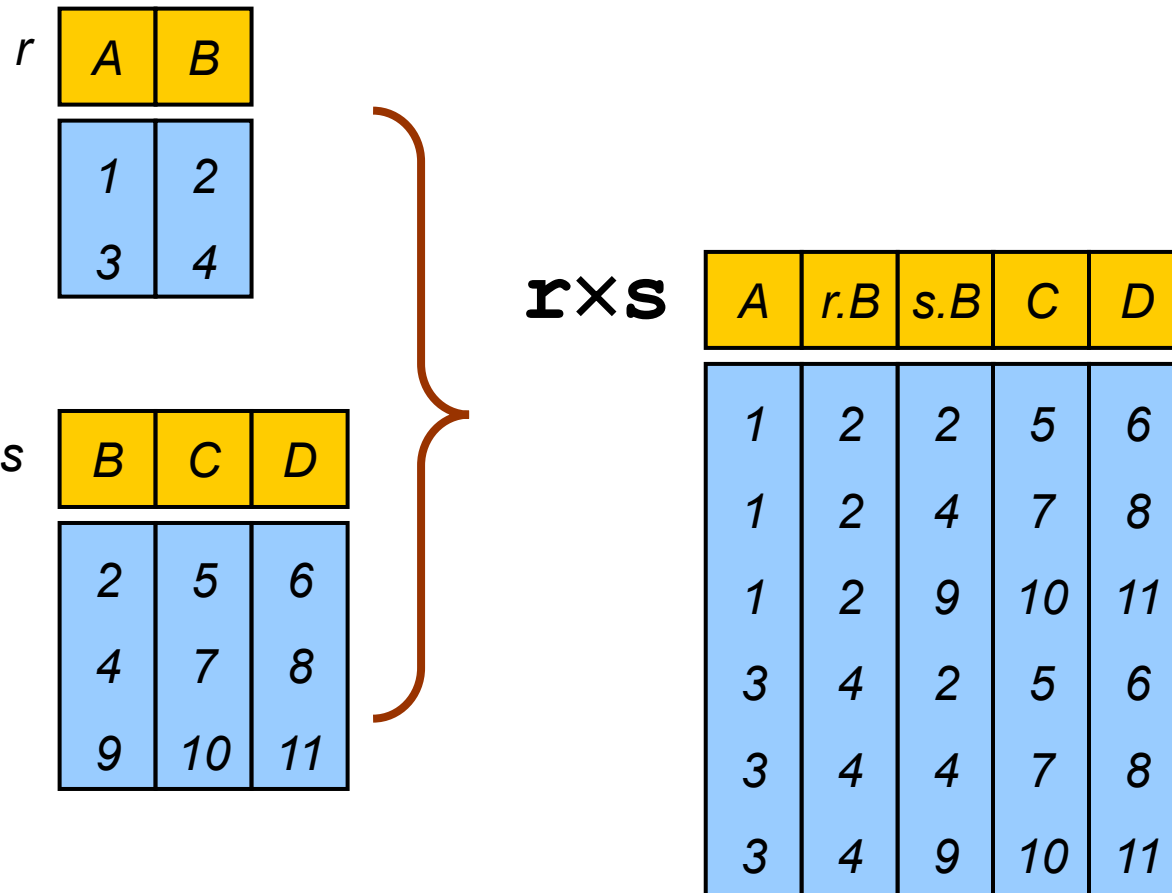
❖ Η έκφραση $r \times s$ παράγει τη σχέση

A	B	A
α	1	α
α	1	δ
β	2	α
β	2	δ

**ΔΕΝ ΥΠΑΡΧΟΥΝ
ΠΕΡΙΟΡΙΣΜΟΙ
ΣΥΜΒΑΤΟΤΗΤΑΣ**

Καρτεσιανό Γινόμενο – Παράδειγμα

❖ Υπολογίστε το καρτεσιανό γινόμενο $\mathbf{r} \times \mathbf{s}$



Ειδικοί σχεσιακοί τελεστές

❖ Είναι οι εξής:

- Selection
- Projection
- Natural join
- Divide (quotient)
- Join
- Semi-join

❖ Δεν προσφέρουν περισσότερη εκφραστική δύναμη απλά διευκολύνουν και απλοποιούν τις ερωτήσεις σε μια βάση δεδομένων

❖ Παράγονται από τους βασικούς τελεστές

Selection

- ❖ Επιλέγει συγκεκριμένες πλειάδες από μια σχέση βάσει συνθήκης. Με άλλα λόγια περιορίζει τη σχέση σε αυτές τις πλειάδες που ικανοποιούν μια συνθήκη φ
 - $t \in \sigma_{\varphi}(R)$ εάν $t \in R$ και $F(t)$
 - Όπου F είναι μια λογική παράσταση αποτελούμενη από
 - ✓ Τελεστές, σταθερές, ονόματα στηλών, σειρά όρου
 - ✓ Αριθμητικοί τελεστές συγκρίσεως = < > ≤ ≥
 - ✓ Λογικοί τελεστές $\neg \vee \wedge$

Παράδειγμα επιλογής

- ❖ Ας θεωρήσουμε το παρακάτω στιγμιότυπο της σχέσης r

A	B	C	D
α	α	1	7
α	β	5	7
β	β	12	3
β	β	23	10

- ❖ Υπολογίστε την έκφραση $\sigma_{A=B \wedge D > 5}(r)$

A	B	C	D
α	α	1	7
β	β	23	10

Ορισμός επιλογής

- ❖ Συμβολισμός: $\sigma_p(r)$
- ❖ p ονομάζεται τελεστής επιλογής και ορίζεται ως εξής
$$\sigma_p(r) = \{t \mid t \in r \wedge p(t)\}$$
- ❖ Όπου p είναι μια φόρμουλα προτασιακού λογισμού αποτελούμενη από όρους (**terms**) που συνδέονται με τα συνδετικά: \wedge (και), \vee (ή), \neg (άρνηση)
- ❖ Κάθε όρος είναι της μορφής:
 $\langle \text{γνώρισμα} \rangle \text{ op } \langle \text{γνώρισμα} \rangle$ ή $\langle \text{σταθερά} \rangle$ και op είναι ένα από τα $=, \neq, >, \geq, <, \leq$
- ❖ Παράδειγμα Example of selection:
$$\sigma_{\text{branch-name}=\text{"Perryridge"}}(\text{account})$$

Προβολή

❖ Ας θεωρήσουμε το παρακάτω στιγμιότυπο της σχέσης r

	A	B	C
α	10		1
α	20		1
β	30		1
β	40		2

❖ Η έκφραση $\Pi_{A,C}(r)$ παράγει σχέση

A	C
α	1
α	1
β	1
β	2

=

A	C
α	1
β	1
β	2

Ορισμός προβολής

- ❖ Συμβολισμός: $\Pi_{A_1, A_2, \dots, A_k}(r)$
όπου A_1, A_2 είναι γνωρίσματα και r είναι το όνομα του τύπου σχέσης
- ❖ Το αποτέλεσμα ορίζεται ως μια σχέση αποτελούμενη από k στήλες που προκύπτουν αφού διαγραφούν οι στήλες που δεν ονοματίζονται
- ❖ Όμοιες πλειάδες διαγράφονται

Σύνδεση

❖ Έστω οι σχέσεις r και s

r	<table border="1"><tr><th>A</th><th>B</th></tr><tr><td>α</td><td>1</td></tr><tr><td>β</td><td>2</td></tr></table>	A	B	α	1	β	2	<table border="1"><tr><th>K</th></tr><tr><td>α</td></tr><tr><td>δ</td></tr></table>	K	α	δ
A	B										
α	1										
β	2										
K											
α											
δ											

❖ Τι παράγει η πράξη $r \bowtie_{s.K=\delta} s$

$(r \times s)$	<table border="1"><thead><tr><th>A</th><th>B</th><th>K</th></tr></thead><tbody><tr><td>α</td><td>1</td><td>α</td></tr><tr><td>α</td><td>1</td><td>δ</td></tr><tr><td>β</td><td>2</td><td>α</td></tr><tr><td>β</td><td>2</td><td>δ</td></tr></tbody></table>	A	B	K	α	1	α	α	1	δ	β	2	α	β	2	δ	$\sigma_{s.K=\delta} (r \times s)$	<table border="1"><thead><tr><th>A</th><th>B</th><th>K</th></tr></thead><tbody><tr><td>α</td><td>1</td><td>α</td></tr><tr><td>α</td><td>1</td><td>δ</td></tr><tr><td>β</td><td>2</td><td>α</td></tr><tr><td>β</td><td>2</td><td>δ</td></tr></tbody></table>	A	B	K	α	1	α	α	1	δ	β	2	α	β	2	δ
A	B	K																															
α	1	α																															
α	1	δ																															
β	2	α																															
β	2	δ																															
A	B	K																															
α	1	α																															
α	1	δ																															
β	2	α																															
β	2	δ																															

Φυσική σύνδεση

❖ Έστω οι σχέσεις r και s

r	<table border="1"><tr><td>A</td><td>B</td></tr><tr><td>α</td><td>1</td></tr><tr><td>β</td><td>2</td></tr></table>	A	B	α	1	β	2	s	<table border="1"><tr><td>A</td></tr><tr><td>α</td></tr><tr><td>δ</td></tr></table>	A	α	δ
A	B											
α	1											
β	2											
A												
α												
δ												

❖ Τι παράγει η πράξη $r \triangleright \triangleleft s$

A	B
α	1

✓ Η πράξη προϋποθέτει κοινά γνωρίσματα

Φυσική σύνδεση – Παράδειγμα

❖ Ας θεωρήσουμε δύο σχέσεις r και s

r	A	B	C	D
α	1	α	a	
β	2	γ	a	
γ	4	β	b	
α	1	γ	a	
δ	2	β	b	

s	B	D	E
1	a	α	
3	a	β	
1	a	γ	
2	b	δ	
3	b	ϵ	

❖ Τι παράγει η πράξη $r \triangleright \triangleleft s$

	A	B	C	D	E
α	1	α	a	α	
α	1	α	a	γ	
α	1	γ	a	α	
α	1	γ	a	γ	
δ	2	β	b	δ	

Φυσική σύνδεση - Natural Join

❖ Συμβολισμός: $r \bowtie s$

❖ Ας θεωρήσουμε δύο σχέσεις r και s με σχήματα R και S αντίστοιχα. Τότε, $r \bowtie s$ είναι μια σχέση με σχήμα $R \cup S$ που παράγεται ως εξής:

- Για κάθε ζευγάρι πλειάδων t_r της r and t_s της s
- Εάν t_r και t_s έχουν την ίδια τιμή για κάθε γνώρισμα της $R \cap S$, προσθέτουμε μια πλειάδα t στο αποτέλεσμα, όπου
 - ✓ t έχει την ίδια με το t_r της r
 - ✓ t έχει την ίδια με το t_s της s

Φυσική σύνδεση - Παράδειγμα

❖ Ας θεωρήσουμε:

$$R = (A, B, C, D)$$

$$S = (E, B, D)$$

- Σχήμα αποτελέσματος = (A, B, C, D, E)
- $r \triangleright \triangleleft s$ ορίζεται ως εξής:

$$\prod_{r.A, r.B, r.C, r.D, s.E} (\sigma_{r.B=s.B \wedge r.D=s.D} (r \times s))$$

Διαίρεση

- ❖ Συμβολισμός $r \div s$
- ❖ Η χρήση του τελεστή ενδείκνυται για ερωτήσεις (queries) που περιλαμβάνουν τη φράση «για κάθε ένα» ή «για όλα»
- ❖ Ας θεωρήσουμε r και s δύο σχέσεις με σχήματα R και S αντίστοιχα όπου
 - $R = (A_1, \dots, A_m, B_1, \dots, B_n)$
 - $S = (B_1, \dots, B_n)$
- ❖ Το αποτέλεσμα του τελεστή $r \div s$ είναι μια νέα σχέση με σχήμα $R - S = (A_1, \dots, A_m)$
$$r \div s = \{ t \mid t \in \Pi_{R-S}(r) \wedge \forall u \in s (tu \in r) \}$$

Διαίρεση - Παράδειγμα

❖ Ας θεωρήσουμε τις σχέσεις r και s

r	A	B	s	B
	α	1		1
	α	2		2
	α	3		
	β	1		
	γ	1		
	δ	1		
	δ	3		
	δ	4		
	ϵ	6		
	ϵ	1		
	β	2		

❖ Τότε $r \div s$ παράγει τη σχέση

A
α
β

Διαίρεση - Παράδειγμα

❖ Ας θεωρήσουμε τις σχέσεις r και s

r	A	B	C	D	E
α	a	α	a	1	
α	a	γ	a	1	
α	a	γ	b	1	
β	a	γ	a	1	
β	a	γ	b	3	
γ	a	γ	a	1	
γ	a	γ	b	1	
γ	a	β	b	1	

s	D	E
a	1	
b	1	

❖ Τότε $r \div s$ παράγει τη σχέση

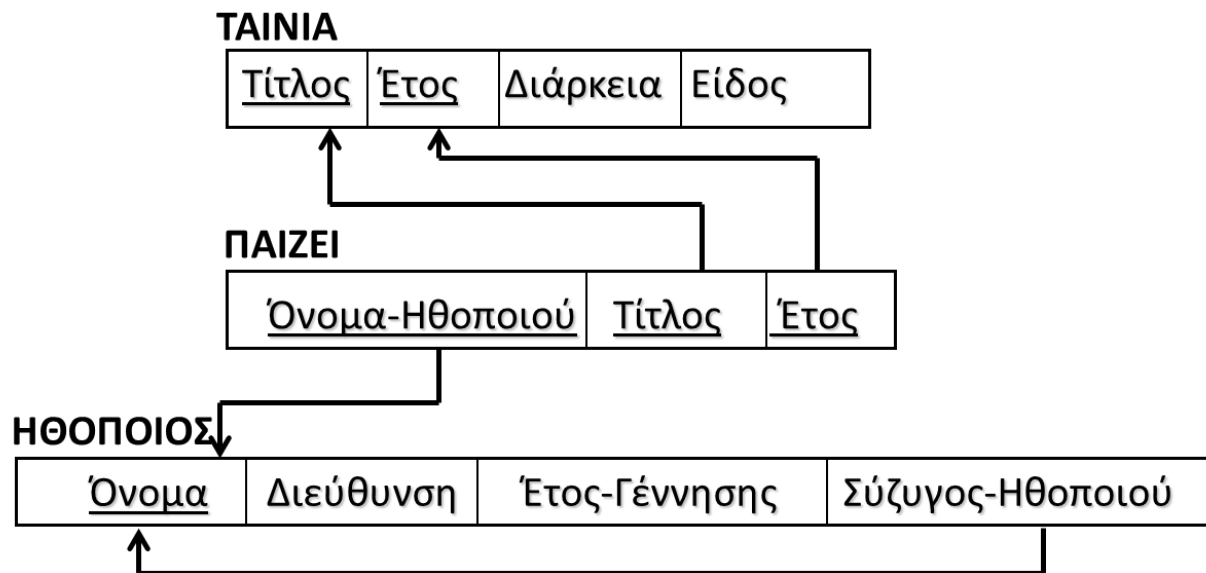
A	B	C
α	a	γ
γ	a	γ



Ασκήσεις στη σχεσιακή άλγεβρα

Παραδείγματα

❖ Ας θεωρήσουμε το σχεσιακό σχήμα

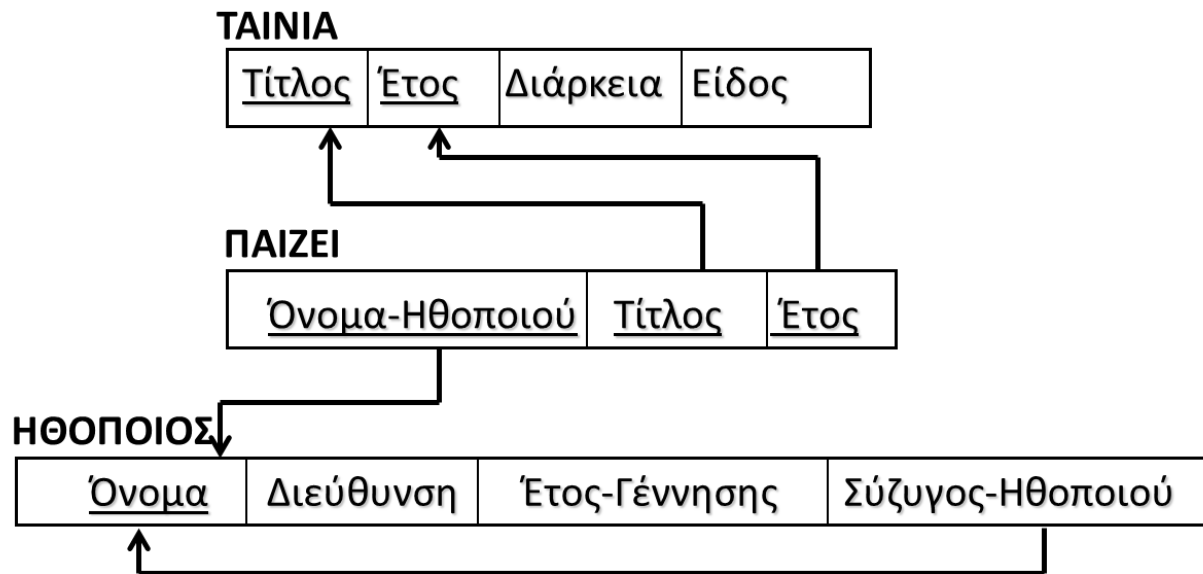


Ταινία	Τίτλος	Έτος	Διάρκεια	Είδος
	Παραμύθι	1990	90	Έγχρωμη
	Παραμύθι	1930	120	Ασπρόμαυρη

Παίζει	Όνομα-Ηθοποιού	Τίτλος	Έτος
	Αλίκη Παππά	Παραμύθι	1930
	Μαρία Γεωργίου	Παραμύθι	1990

Παράδειγμα έκφρασης

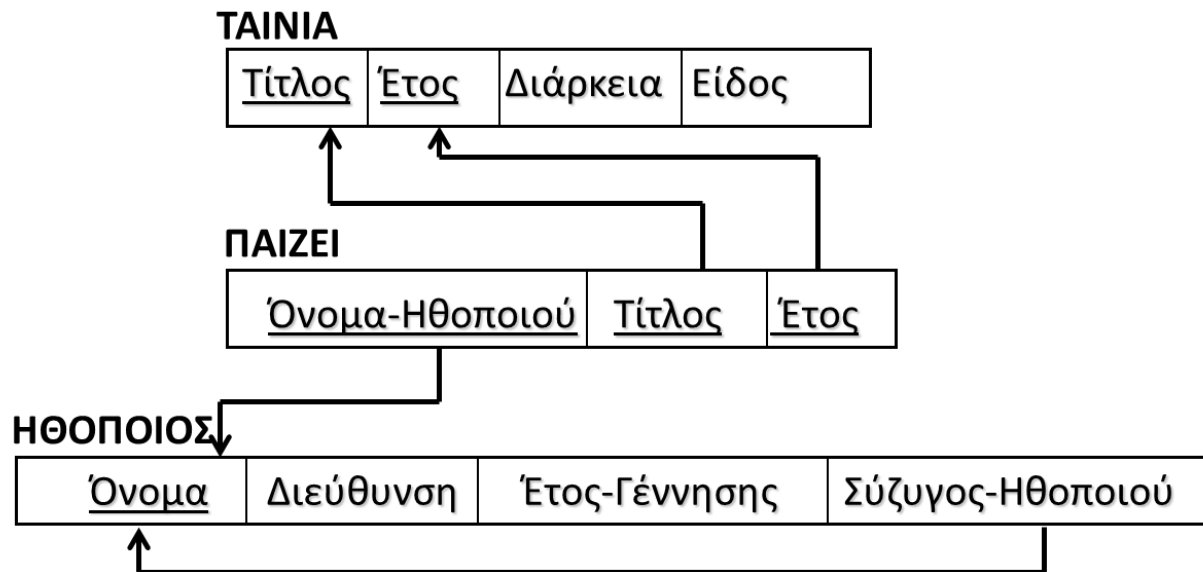
- ❖ Υπολογίστε το τίτλο και τη διάρκεια όλων των ταινιών



π τίτλος, διάρκεια (Ταινία)

Παράδειγμα σύνθετης έκφρασης

- ❖ Να υπολογιστεί για κάθε ηθοποιό το όνομα, ο τίτλος και το έτος για όλες τις έγχρωμες ταινίες στις οποίες παίζει



π γνωρίσματα $(\sigma$ κριτήριο (Παίζει x Ταινία))

Απάντηση

❖ Διάταξη των τελεστών της Σχεσιακής Άλγεβρας

Προβολή

Επιλογή

↓
 π όνομα, τίτλος, έτος (**σ** είδος = “έγχρωμη” AND Παίζει.τίτλος = Ταινία.τίτλος
AND Παίζει.έτος = Ταινία.έτος (**Παίζει x Ταινία**))

↑
Καρτεσιανό
γινόμενο

Γίνεται με φυσική σύνδεση;

❖ Ναι, οι δύο εκφράσεις είναι ισοδύναμες

π όνομα, τίτλος, έτος (σ είδος = “έγχρωμη” AND Παίζει.τίτλος = Ταινία.τίτλος
AND Παίζει.έτος = Ταινία.έτος (**Παίζει x Ταινία**))

π όνομα, τίτλος, έτος (σ είδος = “έγχρωμη” (**Ταινία r \triangleright \triangleleft Παίζει**))

Πράξεις Αλλαγής Δεδομένων

❖ Με τη Σχεσιακή Άλγεβρα το περιεχόμενο μιας βάσης δεδομένων μπορεί να αλλάξει χρησιμοποιώντας τους τελεστές

- INSERT
- DELETE
- UPDATE

INSERT (εισαγωγή)

- ❖ Για την εισαγωγή νέων δεδομένων σε μια σχέση αρκεί να
 - Προσδιορίσουμε μια πλειάδα για εισαγωγή
 - Συντάξουμε μια επερώτηση (query) που το αποτέλεσμα της να είναι ένα σύνολο πλειάδων προς εισαγωγή

- ❖ Με όρους σχεσιακής άλγεβρας η εισαγωγή νέων πλειάδων σε μια σχέση r ορίζεται ως εξής:

$$r \leftarrow r \cup E,$$

r είναι σχέση και E είναι ερώτηση (query) σχεσιακής άλγεβρας

- ❖ Η εισαγωγή μιας και μόνο πλειάδας προκύπτει από τον παραπάνω τύπο αν θεωρήσουμε το E ως σταθερή σχέση μιας και μόνο πλειάδας

Παραδείγματα χρήσης του INSERT

- ❖ Εισαγωγή του δεδομένου 'Ο Ακουμιανάκης έχει 1200 euro σε λογαριασμό με κωδικό A-973 στο παράρτημα του Ηρακλείου'

$$account \leftarrow account \cup \{("Ηράκλειο", A-973, 1200)\}$$
$$depositor \leftarrow depositor \cup \{("Ακουμιανάκης", A-973)\}$$

- ❖ Εισαγωγή του δεδομένου 'Δίδεται δώρο ένας αποταμιευτικός λογαριασμός (με κωδικό τον κωδικό του δανείου) με ποσό \$200 σε όλους τους δανειολήπτες του παρατήματος του Ηρακλείου'

$$r_1 \leftarrow (\sigma_{branch-name = "Ηράκλειο"}(borrower\ loan))$$
$$account \leftarrow account \cup \Pi_{branch-name, account-number, 200}(r_1)$$
$$depositor \leftarrow depositor \cup \Pi_{customer-name, loan-number}(r_1)$$

DELETE (Διαγραφή)

- ❖ Αντί να παρουσιάζει επιλεγμένες πλειάδες τις διαγράφει από τη βάση δεδομένων
- ❖ Δεν μπορούν να διαγραφούν τιμές συγκεκριμένων γνωρισμάτων
- ❖ Σε σχεσιακή άλγεβρα η διαγραφή ορίζεται ως εξής

$$r \leftarrow r - E$$

όπου r είναι μια σχέση και E είναι μια ερώτηση σχεσιακής άλγεβρας

Παραδείγματα χρήσης DELETE

- ❖ Διαγραφή εγγραφών στο παράρτημα του Ηρακλείου

$account \leftarrow account - \sigma_{branch-name = \text{“Ηράκλειο”}}(account)$

- ❖ Διαγραφή εγγραφών δανείου με ποσό 0 - 50

$loan \leftarrow loan - \sigma_{amount \geq 0 \text{ and } amount \leq 50}(loan)$

Περιορισμοί της σχεσιακής άλγεβρας

- ❖ Κάποιου είδους ερωτήσεις δεν μπορούν να απαντηθούν
 - Είναι βασισμένη στη λογική πρώτης τάξης, επομένως
 - ✓ Ανεπαρκές για αναδρομικές ερωτήσεις
 - Βρες όλους τους υπαλλήλους που διευθύνονται από τη «Σοφία»
 - ✓ Ποσοδεικτικές ερωτήσεις
 - Πόσες πλειάδες υπάρχουν στη σχέση R
 - Λύσεις
 - ✓ Host Γλώσσες (π.χ. C, Fortran)
 - ✓ Ειδικές κατασκευές (π.χ. Count(*))
 - ✓ Καινούργιες πιο ισχυρές γλώσσες υπερωτήσεων



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 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

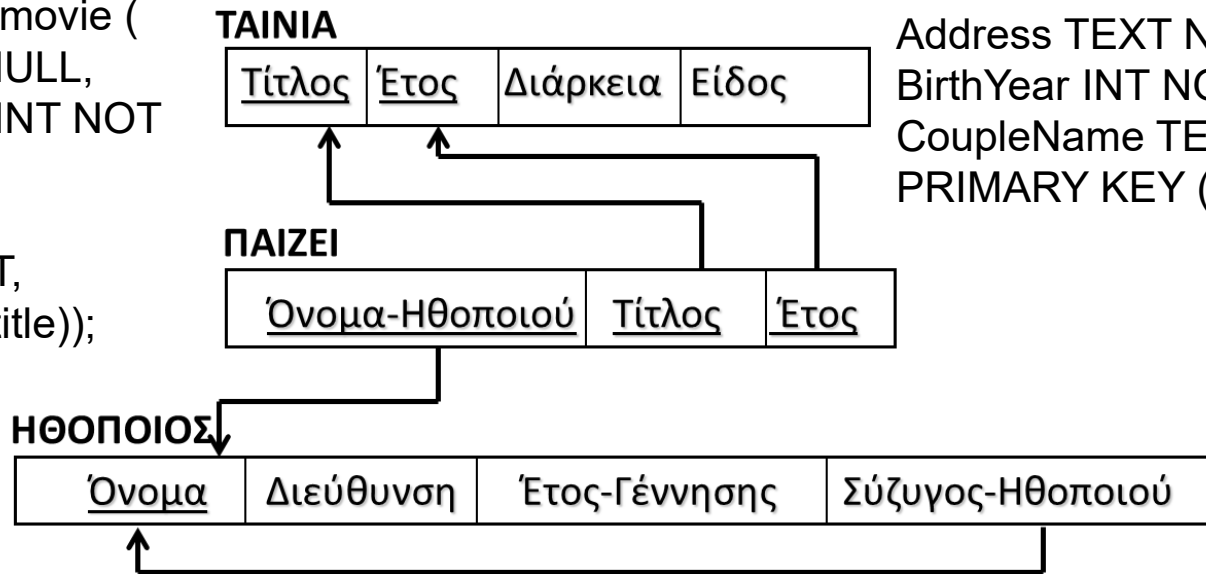
συνθήκη

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

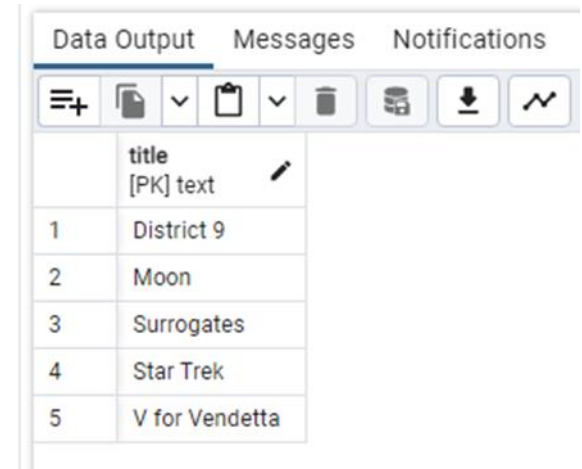
```
107 select title
108 from movie
109 where production_year > 1000 AND
110        colour_type = 'Black and White';|
```

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;
```

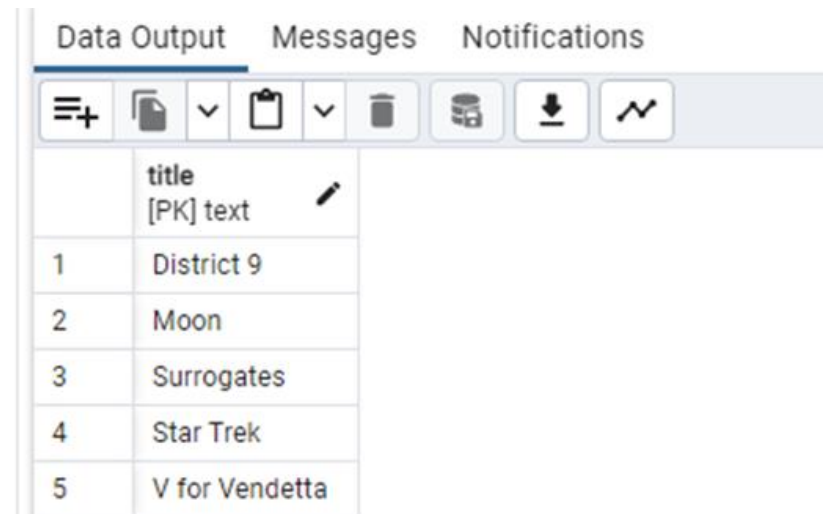


The screenshot shows a database interface with a table titled 'title [PK] text'. The table contains 5 rows of data:

	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;
```



The screenshot shows a database interface with a table titled 'title [PK] text'. The table contains 5 rows of data:

	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;
```

	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;
```

	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 ορίζονται στην SQL query

❖ Υπάρχει η δυνατότητα να ονομάζουμε τα γνωρίσματα με το όνομα που θέλουμε

■ Το **as** μιλάει για το όνομα που θέλουμε να δώσει στο γνωρίσμα

❖ Παράδειγμα

	title [PK] text	production_year integer	Ωρες integer	lock	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

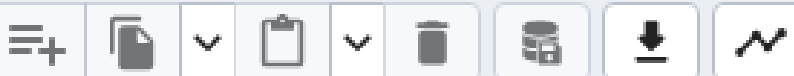
```
select title, production_year, duration / 60 as Ωρες, colour_type  
from movie;
```

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»
```

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
```

```
((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 ))
```

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"
```

- ❖ Λογαριασμός στο υποκατάστημα XYZ με το μέγιστο ποσό

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



Επεξεργασία ερωτήσεων

Θεματολογία

❖ Θα εξετάσουμε (συνοπτικά)

- Γενικές αρχές της επεξεργασίας ερωτήσεων σε σχεσιακά συστήματα
- Βελτιστοποίηση ερωτήσεων
 - ✓ Βελτιωμένη εγγραφή ερωτήσεων
 - ✓ Παραγωγή εναλλακτικών πλάνων
 - ✓ Αποτίμηση πλάνων
 - ✓ Πρόβλεψη μεγέθους / αποτελέσματος

Επιμέρους ερωτήματα

- ❖ Ειδικότερα, υπάρχουν τα ακόλουθα επιμέρους ζητήματα / ερωτήματα
 - Πως λειτουργεί το DBMS για να αποκωδικοποιήσει τη δήλωση του χρήστη;
 - Πως αποφασίζει ποιο πίνακα να ανακτήσει πρώτο, ποιο δεύτερο, κοκ.;
 - Πως συνθέτει το ενδιάμεσο ή/και το τελικό αποτέλεσμα;
 - Πως παρουσιάζει την απάντηση;

Κλασική Θεώρηση

❖ Optimize-then-execute

- Το ερώτημα SQL πριν εκτελεστεί βελτιστοποιείται πρώτα

❖ Τα βήματα είναι τα ακόλουθα

- **ΣΥΝΤΑΚΤΙΚΟΣ** έλεγχος του ερωτήματος σε δηλωτική μορφή
- Μετατροπή σε **ΛΟΓΙΚΟ** πλάνο εκτέλεσης
 - ✓ Συνήθως μια αλγεβρική αναπαράσταση του ερωτήματος
- Υπολογισμός **ΒΕΛΤΙΣΤΟΥ** πλάνου εκτέλεσης
 - ✓ Ισοδύναμο ερώτημα που απαιτεί λιγότερο χρόνο εκτέλεσης
- **ΦΥΣΙΚΟ** πλάνο εκτέλεσης
 - ✓ Δενδρική αναπαράσταση του τρόπου και της σειράς προσπέλασης των σχέσεων καθώς και των αλγορίθμων υλοποίησης των τελεστών
- **Εκτέλεση** φυσικού πλάνου

Κλασσική Θεώρηση (συν.)

❖ Optimize-then-execute

Βελτιστοποιητής

- Το ερώτημα SQL πριν εκτελεστεί βελτιστοποιείται πρώτα

❖ Τα βήματα είναι τα ακόλουθα

- *Συντακτικός* έλεγχος του ερωτήματος σε δηλωτική μορφή

- Μετατροπή σε *λογικό* πλάνο εκτέλεσης

- ✓ Συνήθως μια αλγεβρική αναπαράσταση του ερωτήματος

- Υπολογισμός *βέλτιστου* πλάνου εκτέλεσης

- ✓ Ισοδύναμο ερώτημα που απαιτεί λιγότερο χρόνο εκτέλεσης

- *Φυσικό* πλάνο εκτέλεσης

- ✓ Δενδρική αναπαράσταση του τρόπου και της σειράς προσπέλασης των σχέσεων καθώς και των αλγορίθμων υλοποίησης των τελεστών

- *Εκτέλεση* φυσικού πλάνου

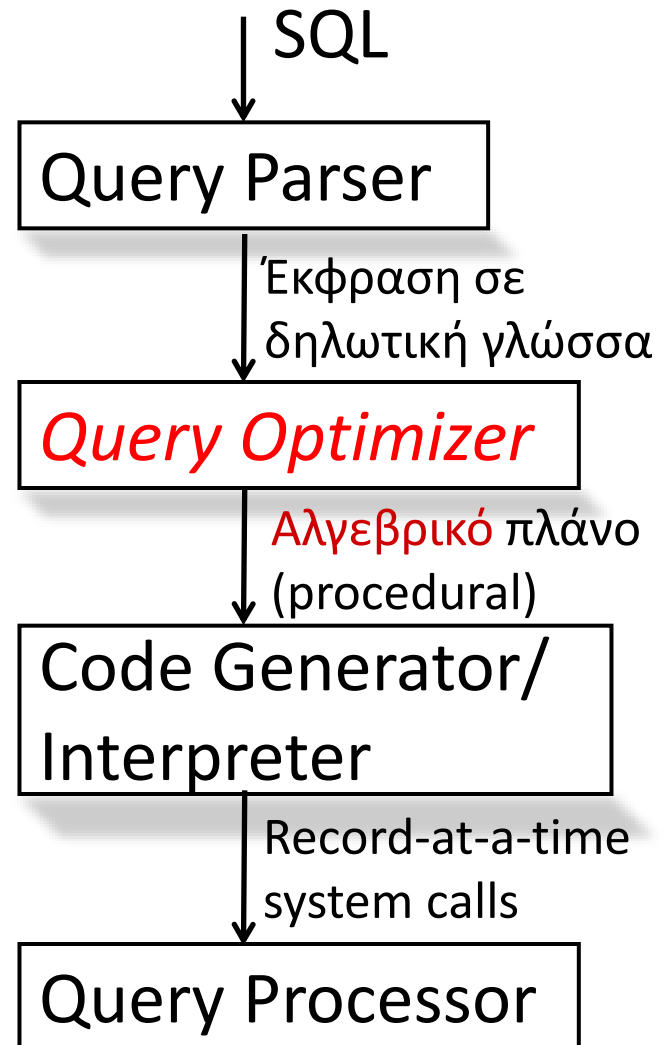
Λειτουργίες του DBMS

- ❖ Το DBMS εκτελεί τρεις βασικές λειτουργίες
 - Parsing
 - ✓ Έλεγχος λεκτικών συστατικών ερωτήματος
 - Ονόματα πινάκων, γνωρισμάτων, κλπ.
 - ✓ Έλεγχος συντακτικής ορθότητας
 - Η δήλωση πρέπει να είναι καλώς συντεταγμένη έκφραση
 - Σύνθεση βέλτιστου λογικού πλάνου / αλγεβρικού χώρου
 - ✓ Μετατροπές σε ισοδύναμες αλγεβρικές παραστάσεις
 - ✓ Εύρεση βέλτιστου πλάνου (με αντιμετάθεση τελεστών)
 - Εκτέλεση βέλτιστου φυσικού πλάνου και επιστροφή του αποτελέσματος

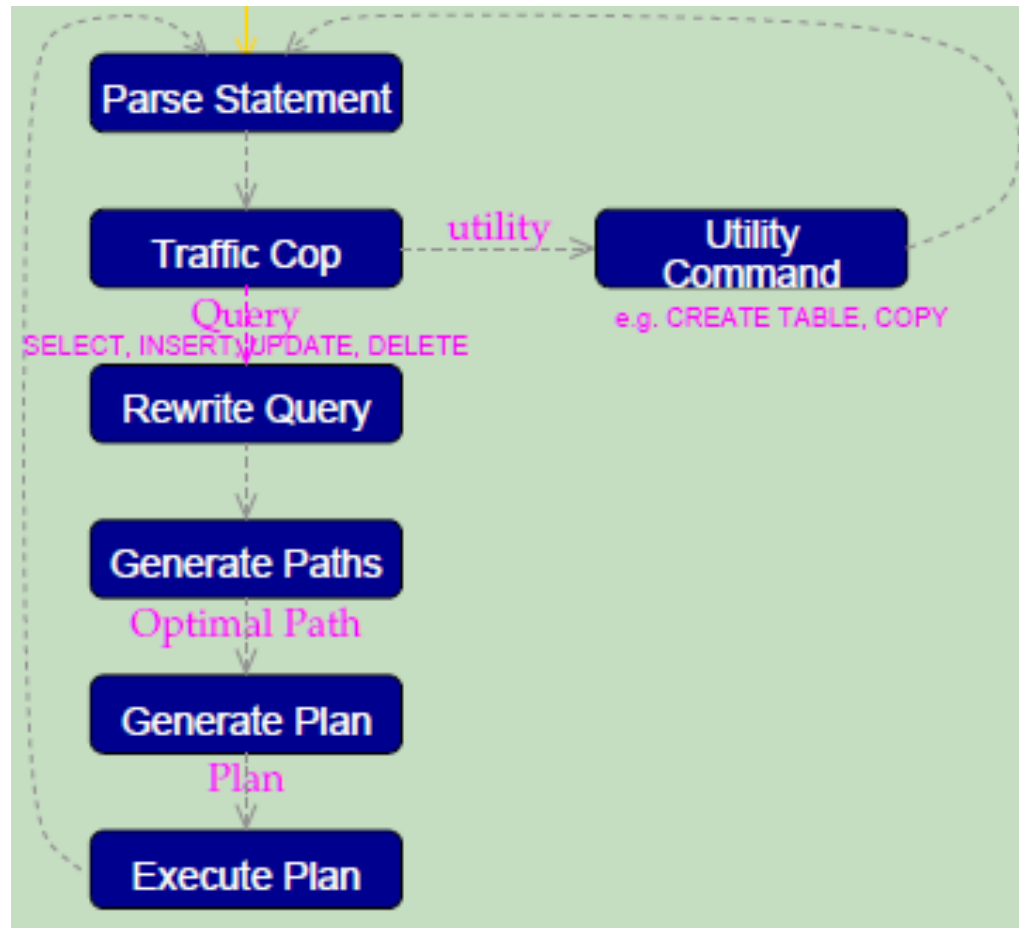
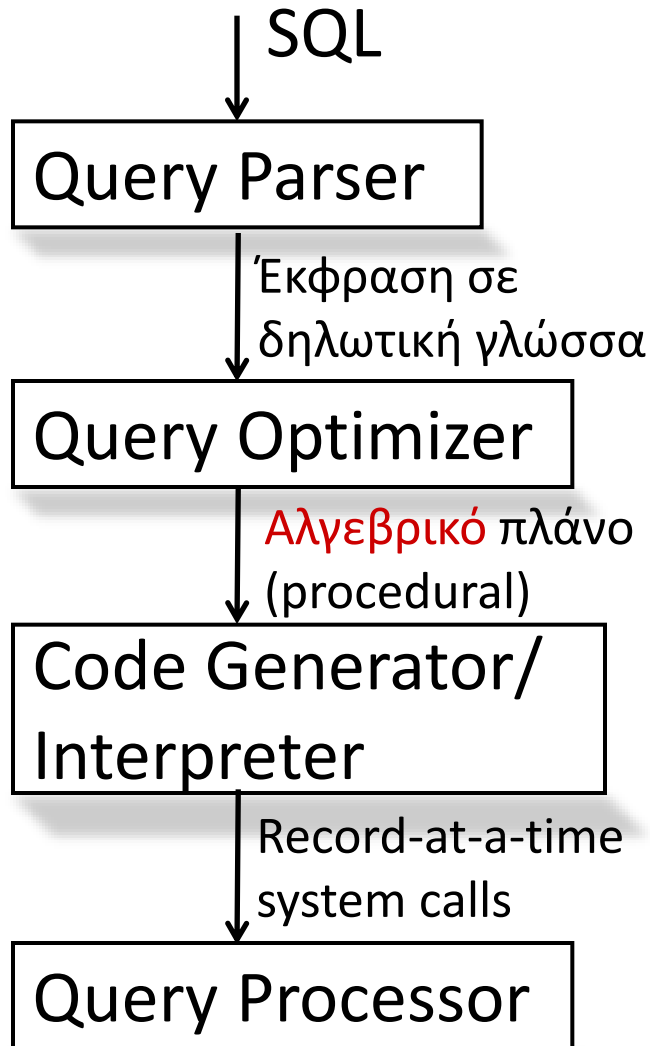
Στάδια επεξεργασίας

❖ Τέσσερα βασικά στάδια

- Μπορεί να διαφοροποιούνται από σύστημα σε σύστημα
- Ο *βελτιστοποιητής* είναι ο 'εγκέφαλος' του συστήματος



Τα στάδια στην Postgres



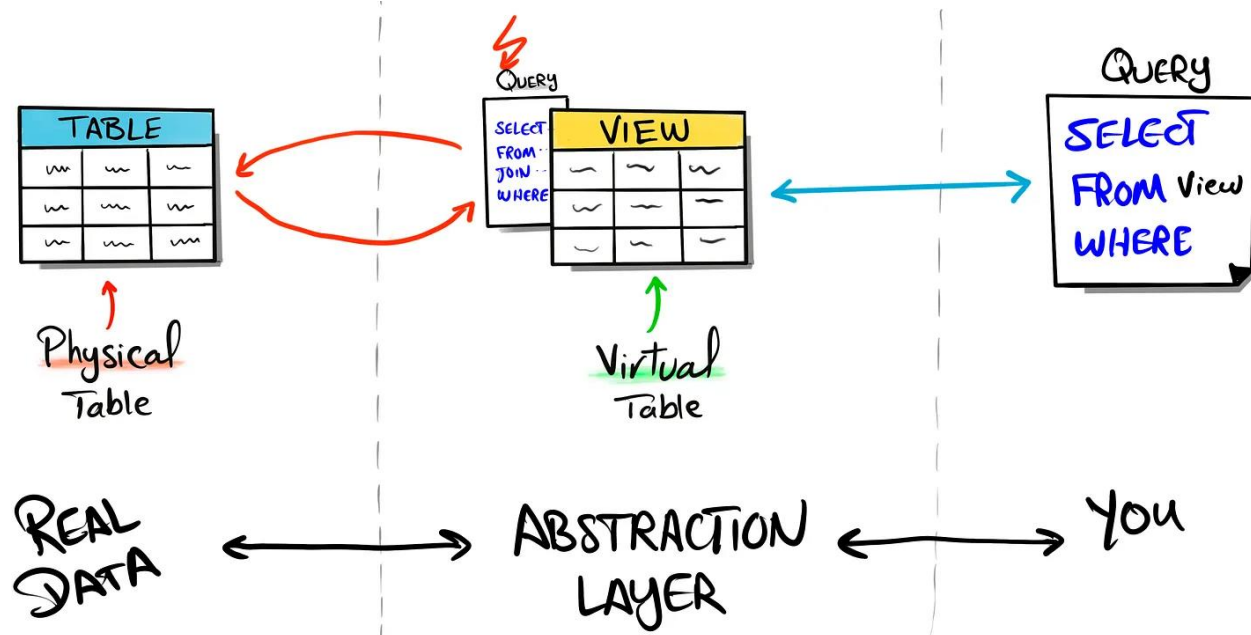


Όψεις

Τι είναι όψη

❖ Μία όψη είναι μία εικονική (virtual) ή αλλιώς παραγόμενη (derived) σχέση

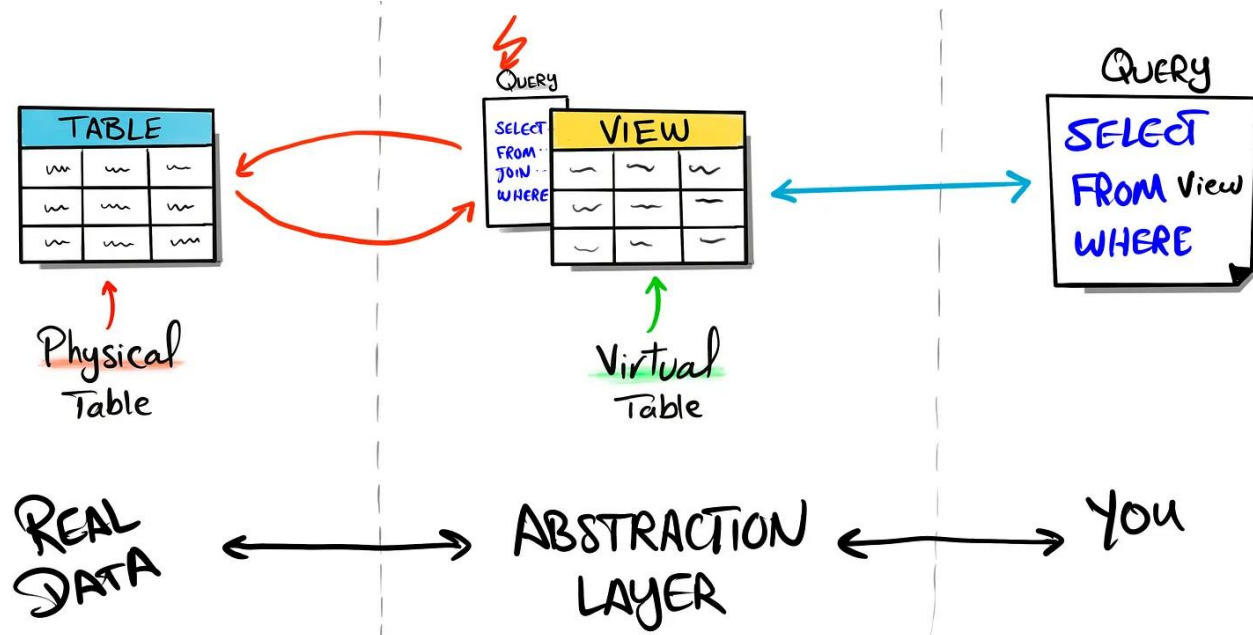
- Αυτό σημαίνει ότι σε αντίθεση με μία σχέση βάσης (base relation) μία όψη **δεν έχει φυσική υπόσταση**
- Δεν υπάρχουν πλειάδες που να ανήκουν φυσικά σε μια όψη
 - ✓ Οι πλειάδες της όψης ανήκουν σε μία ή περισσότερες σχέσεις βάσης



Που χρησιμεύουν οι όψεις

❖ Μία όψη είναι μία εικονική (virtual) ή αλλιώς παραγόμενη (derived) σχέση

- Αυτό σημαίνει ότι σε αντίθεση με μία σχέση βάσης (base relation) μία όψη **δεν έχει φυσική υπόσταση**
- Δεν υπάρχουν πλειάδες που να ανήκουν φυσικά σε μια όψη
 - ✓ Οι πλειάδες της όψης ανήκουν σε μία ή περισσότερες σχέσεις βάσης



Ορισμός όψης

- ❖ Ορισμός όψης χρησιμοποιώντας την εντολή:

view
definition

```
create view <όνομα--όψης> as <select-from-where ερώτηση>
```

- ❖ Επίσης, μπορούν να προσδιοριστούν τα ονόματα των γνωρισμάτων άμεσα

```
create view <όνομα--όψης> (<λίστα ονομάτων-γνωρισμάτων>)  
as <select-from-where ερώτηση>
```

Παράδειγμα δημιουργίας όψης

- ❖ Ορίστε μια όψη που να περιλαμβάνει τα ονόματα όλων των υποκαταστημάτων και το άθροισμα του ποσού των δανείων που έχουν γίνει από αυτά

Δάνειο

Όνομα-Υποκαταστήματος	<u>Αριθμός-Δανείου</u>	Ποσό
-----------------------	------------------------	------

```
create view Υποκατάστημα-Σύνολο-Δανείων  
(Σύνολο-Δανείων, Όνομα-Υποκαταστήματος) as  
select Όνομα-Υποκαταστήματος, sum(Ποσό)  
from Δάνειο  
group by Όνομα-Υποκαταστήματος
```

Όνομα όψης

Γνωρίσματα όψης

Σχόλια για τις όψεις

- ❖ Οι όψεις μπορεί να χρησιμοποιηθούν όπως οι σχέσεις βάσης
 - Δηλαδή τα ονόματα των όψεων μπορεί να χρησιμοποιηθούν οπουδήποτε μπορεί να χρησιμοποιηθεί το όνομα μιας σχέσης
- ❖ Ο ορισμός μιας όψης παραμένει στην βάση δεδομένων, εκτός αν σβηστεί:
drop view <όνομα-όψης>

Άσκηση στην τάξη

❖ Θυμηθείτε το ερώτημα 13 της εργασίας 10

- Να βρεθούν τα ονόματα και οι τύποι των πλοίων τα οποία έλαβαν μέρος σε τουλάχιστον δύο πολέμους

```
SELECT SHIP.Ship_Name, SHIP_TYPE.Name, COUNT(AT_WAR.Ship_Name)
FROM SHIP, SHIP_TYPE, AT_WAR
WHERE SHIP.Ship_Name = AT_WAR.Ship_Name AND
      SHIP.Ship_Type_ID = SHIP_TYPE.Ship_Type_ID
GROUP BY AT_WAR.Ship_Name, SHIP.Ship_Name, SHIP_TYPE.name
HAVING COUNT(AT_WAR.Ship_Name) >= 2;
```

Το αποτέλεσμα που υπολογίζει

❖ Με τα δεδομένα που σας δόθηκαν

```
SELECT SHIP.Ship_Name, SHIP_TYPE.Name, COUNT(AT_WAR.Ship_Name)
FROM SHIP, SHIP_TYPE, AT_WAR
WHERE SHIP.Ship_Name = AT_WAR.Ship_Name AND
      SHIP.Ship_Type_ID = SHIP_TYPE.Ship_Type_ID
GROUP BY AT_WAR.Ship_Name, SHIP.Ship_Name, SHIP_TYPE.name
HAVING COUNT(AT_WAR.Ship_Name) >= 2;
```

	Data Output	Explain	Messages	Notifications
	Όνομα Πλοίου text	🔒	Ονομασία Τύπου text	🔒
▲				Αριθμός Πολέμων bigint
1	Έλλη Ι		Καταδρομικό	2
2	Ψαρά		Θωρηκτό	2
3	Γεώργιος Αβέρωφ		Θωρηκτό	4
4	Τορπιλοβόλο-11		Τορπιλοβόλο	2

Εναλλακτική προσέγγιση (1/2)

- ❖ Δημιουργία όψης που θα καταγράφει το πλήθος των πολέμων στους οποίους συμμετείχε κάθε πλοίο

```
CREATE VIEW WARS_PER_SHIP AS
SELECT AT_WAR.Ship_Name, COUNT(AT_
FROM AT_WAR, SHIP
WHERE AT_WAR.Ship_Name = SHIP.Ship_
GROUP BY AT_WAR.Ship_Name;
```

	ship_name text	war_count bigint	
1	Λήμνος II	1	
2	Παπανικολής I	1	
3	Βαυαρία	1	
4	Ψαρά	2	
5	Καρτερία	1	
6	Βασίλισσα Όλγα	1	
7	Κατσώνης II	1	
8	Βασιλεύς Γεώρ...	1	
9	Γεώργιος Αβέρ...	4	
10	Τορπιλοβόλο-11	2	
11	Έλλη I	2	

Εναλλακτική προσέγγιση (2/2)

```
452  
453 SELECT SHIP.Ship_Name AS "Όνομα Πλοίου", SHIP_TYPE.Name AS "Τύπος Πλοίου", WARS_PER_SHIP.WAR_COUNT AS "  
454 FROM SHIP, SHIP_TYPE, WARS_PER_SHIP  
455 WHERE SHIP.Ship_Type_ID = SHIP_TYPE.Ship_Type_ID  
456 AND SHIP.Ship_Name = WARS_PER_SHIP.Ship_Name  
457 AND WARS_PER_SHIP.WAR_COUNT >= 2;  
458
```

Data Output

	Όνομα Πλοίου text	Τύπος Πλοίου text	Αριθμός Πολέμων bigint
461			
462			
463	1 Γεώργιος Αβέρωφ	Θωρηκτό	4
464	2 Έλλη Ι	Καταδρομικό	2
465	3 Τορπιλοβόλο-11	Τορπιλοβόλο	2
466	4 Ψαρά	Θωρηκτό	2
467			

GROUP BY AT_WAR.Ship_Name;

❖ Χρήση όψης για το τελικό αποτέλεσμα

```
SELECT SHIP.Ship_Name, SHIP_TYPE.Name, WARS_PER_SHIP.WAR_COUNT  
FROM SHIP, SHIP_TYPE, WARS_PER_SHIP  
WHERE SHIP.Ship_Type_ID = SHIP_TYPE.Ship_Type_ID AND  
SHIP.Ship_Name = WARS_PER_SHIP.Ship_Name AND  
WARS_PER_SHIP.WAR_COUNT >= 2;
```

Σύγκριση των δύο λύσεων

❖ Ίδια λογική, διαφορετική ακολουθία πράξεων

```
SELECT SHIP.Ship_Name,  
SHIP_TYPE.Name,  
COUNT(AT_WAR.Ship_Name)  
FROM SHIP, SHIP_TYPE, AT_WAR  
WHERE SHIP.Ship_Name =  
AT_WAR.Ship_Name AND  
SHIP.Ship_Type_ID =  
SHIP_TYPE.Ship_Type_ID  
GROUP BY AT_WAR.Ship_Name,  
SHIP.Ship_Name, SHIP_TYPE.name  
HAVING  
COUNT(AT_WAR.Ship_Name) >= 2;
```

```
CREATE VIEW WARS_PER_SHIP AS  
SELECT AT_WAR.Ship_Name,  
COUNT(AT_WAR.Ship_Name)  
FROM AT_WAR, SHIP  
WHERE AT_WAR.Ship_Name =  
SHIP.Ship_Name  
GROUP BY AT_WAR.Ship_Name;
```

```
SELECT SHIP.Ship_Name,  
SHIP_TYPE.Name,  
WARS_PER_SHIP.WAR_COUNT  
FROM SHIP, SHIP_TYPE,  
WARS_PER_SHIP  
WHERE SHIP.Ship_Type_ID =  
SHIP_TYPE.Ship_Type_ID AND  
SHIP.Ship_Name =  
WARS_PER_SHIP.Ship_Name AND  
WARS_PER_SHIP.WAR_COUNT >= 2;
```

Σύγκριση των δύο αποτελεσμάτων

	Data Output	Explain	Messages	Notifications
	Όνομα Πλοίου text	🔒	Ονομασία Τύπου text	🔒
▲				
1	Έλλη Ι		Καταδρομικό	2
2	Ψαρά		Θωρηκτό	2
3	Γεώργιος Αβέρωφ		Θωρηκτό	4
4	Τορπιλοβόλο-11		Τορπιλοβόλο	2

Το αποτέλεσμα της όψης



	Data Output			
	Όνομα Πλοίου text	🔒	Τύπος Πλοίου text	🔒
▲				
1	Γεώργιος Αβέρωφ		Θωρηκτό	4
2	Έλλη Ι		Καταδρομικό	2
3	Τορπιλοβόλο-11		Τορπιλοβόλο	2
4	Ψαρά		Θωρηκτό	2

Κατακλείδα

- ❖ Οι όψεις αποτελούν μια ιδιαιτέρως χρήσιμη δυνατότητα
- ❖ Συχνά είναι αναπόφευκτες !

Τέλος για σήμερα

