



Διάλεξη 10: Σχεσιακή Άλγεβρα και Σχεσιακός Λογισμός (Relational Algebra/Calculus) II

Στην ενότητα αυτή θα μελετηθούν τα εξής επιμέρους θέματα:
Εισαγωγή στις έννοιες:

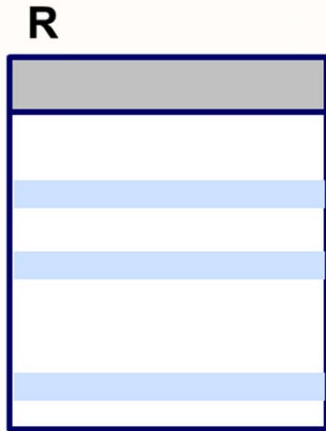
- Σχεσιακή Άλγεβρα
- Τελεστές Συνένωσης
- Διαίρεση
- Τελεστές Συνάθροισης και Ομαδοποίησης
- Άλλοι Τελεστές

Διδάσκων: Παναγιώτης Ανδρέου

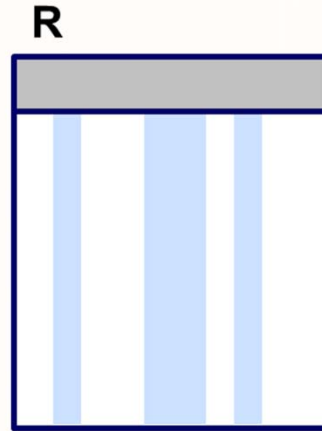
Τελεστές στην Σχεσιακή Άλγεβρα (Επανάληψη)

- **5 Βασικοί Τελεστές:** Επιλογή (Selection), Προβολή (Projection), Καρτεσιανό Γινόμενο (Cartesian Product), Ένωση (Union), Αφαίρεση (Division)
- Το σύνολο σχεσιακών τελεστών $\{\sigma, \pi, \cup, -, \rho, \times\}$ ονομάζεται **κλειστό σύνολο (*complete set*)** διότι κάθε έκφραση σχεσιακής άλγεβρας μπορεί να διατυπωθεί από τον συνδυασμό των πιο πάνω τελεστών:
- Π.χ., :
 - Τομή: $R \cap S = (R \cup S) - ((R - S) \cup (S - R))$
 - Συμμετρική Διαφορά: $R \oplus S = (R - S) \cup (S - R)$
- **Επιπρόσθετοι Τελεστές:** Δυαδικός Σχεσιακός Τελεστής (Join), και **Διαίρεση** (Division) που μπορούν να εκφραστούν με τους 5 βασικούς τελεστές

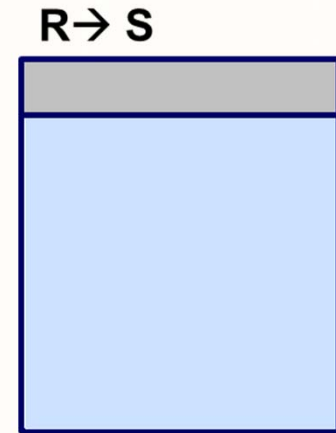
Σύνοψη Τελεστών/Πράξεων (Σχεσιακής Άλγεβρας)



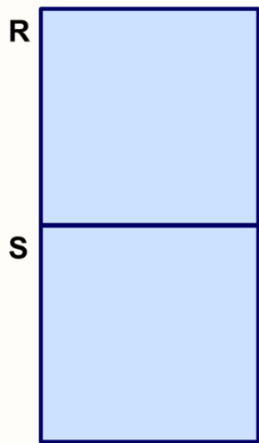
Επιλογή/Selection (σ)



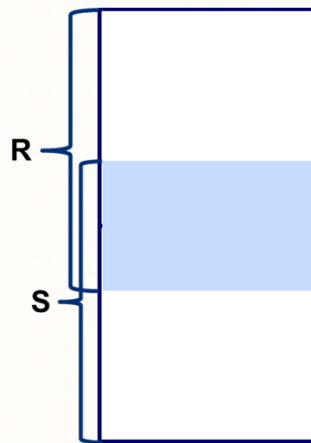
Προβολή/Projection (π)



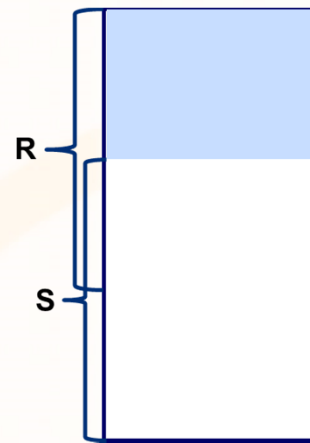
Μετονομασία/Rename (ρ)



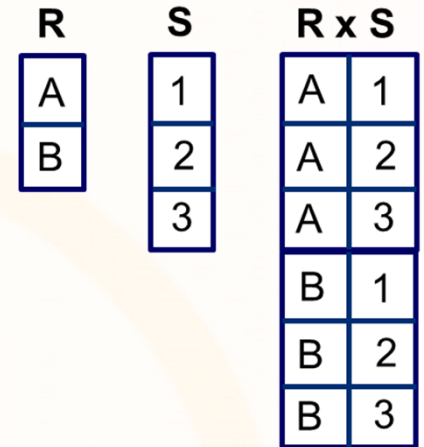
Ένωση/Union (\cup)



Τομή/Intersection (\cap)








Αφαίρεση/
Set Difference ($-$)



Καρτεσιανό Γινόμενο/
Cartesian Product (\times)

Τελεστές Συνένωσης (JOIN Operators)

- Είσοδος: Δύο σχέσεις
- Έξοδος: Μία νέα σχέση
- Η Σχεσιακή Άλγεβρα παρέχει τους ακόλουθους Τελεστές Συνένωσης
 - **Συνένωση Θ (Θ -Join):** 
 - Όταν το θ είναι συνθήκη ισότητας ονομάζεται (**Equi Join**)
 - Όταν τα γνωρίσματα των δύο σχέσεων έχουν το ίδιο όνομα τότε δεν χρειάζεται να αναφέρουμε τη συνθήκη θ (**Natural Join: ***)
 - **Ημι-Συνένωση (Semi Join):** 
 - **Αντι-Συνένωση (Anti Join):** 
 - **Αριστερή Εξ. Συνένωση/ Left (Outer) Join:** 
 - **Ολική Εξ. Συνένωση/ Full (Outer) Join:** 
- Επίσης υπάρχουν και πολλές άλλες παραλλαγές...

Σύνοψη Τελεστών Συνένωσης

R		S	
A	B	B	C
a	1	1	x
b	2	2	y
c	4	3	z

$$R \bowtie_{\Theta=B} S \quad (R.B=S.B)$$

A	B	C
a	1	x
b	2	y

Συνένωση Θ /
 Θ -Join (\bowtie_{Θ})

$$R * S$$

A	B	C
a	1	x
b	2	y

Φυσική Συνένωση/
Natural Join (*)

$$R \ltimes_B S$$

A	B
a	1
b	2

Ημι-Συνένωση /
Semi Join (\ltimes_{Θ})

$R \times S$

A	B	B	C
a	1	1	x
a	1	2	y
a	1	3	z
b	2	1	x
b	2	2	y
b	2	3	z
c	4	1	x
c	4	2	y
c	4	3	z

$$R \triangleright_B S$$

A	B
c	4

Αντι-Συνένωση /
Anti Join (\triangleright_{Θ})

$$R \ltimes_B S$$

A	B	C
a	1	x
b	2	y
c	4	

Αριστερή Εξ. Συνένωση/
Left (Outer) Join (\ltimes_{Θ})

$$R \bowtie_B S$$

A	B	C
a	1	x
b	2	y
c	4	
	3	z

Ολική Εξ. Συνένωση/
Full (Outer) Join (\bowtie_{Θ})

Τελεστής Συνένωσης (JOIN Operator)

- Ο Τελεστής Συνένωσης (\bowtie) επιτρέπει τη αναπαράσταση των συσχετιζόμενων πλειάδων μεταξύ δύο σχέσεων
- Π.χ., Δώστε τα ονόματα των ατόμων και τα τμήματα στα οποία ανήκουν.

PERSON		
PID	NAME	DID
1	Andreas	1
2	Kostas	1
3	Maria	1
4	Eleni	2
5	Nikos	2
6	Eleni	3

DEPARTMENT	
DNO	DNAME
1	ACC
2	HR
3	IT

PERSON x DEPARTMENT				
PID	NAME	DID	DNO	DNAME
1	Andreas	1	1	ACC
1	Andreas	1	2	HR
1	Andreas	1	3	IT
2	Kostas	1	1	ACC
2	Kostas	1	2	HR
2	Kostas	1	3	IT
...

- Είναι μία συντομογραφία της έκφρασης $\sigma_{\theta}(\text{PERSON} \times \text{DEPARTMENT})$

Τελεστής Συνένωσης (JOIN Operator)

$$R \bowtie_{\Theta} S$$

- **Είσοδος:** Δύο σχέσεις R και S, Συνθήκη συνένωσης (Θ)
- **Έξοδος:** Νέα σχέση η οποία συνενώνει τις R και S και περιλαμβάνει όλα τα γνωρίσματα που ανήκουν στην R και S και περιλαμβάνει όλες τις πλειάδες (γραμμές) που ικανοποιούν την συνθήκη Θ
- **Παράδειγμα:** $\text{PERSON} \bowtie_{\text{PERSON.DID}=\text{DEPARTMENT.DNO}} \text{DEPARTMENT}$

PERSON			DEPARTMENT		PERSON_DEPARTMENT				
PID	NAME	DID	DNO	DNAME	PID	NAME	DID	DNO	DNAME
1	Andreas	1	1	ACC	1	Andreas	1	1	ACC
2	Kostas	1	2	HR	2	Kostas	1	1	ACC
3	Maria	1	3	IT	3	Maria	1	1	ACC
4	Eleni	2			4	Eleni	2	2	HR
5	Nikos	2			5	Nikos	2	2	HR
6	Eleni	3			6	Eleni	3	3	IT

Τελεστής Συνένωσης (JOIN Operator)

- Σημειώστε ότι **οποιαδήποτε** λογική έκφραση μπορεί να μεταφραστεί σε **Συζευκτική Κανονική Μορφή (Conjunctive Normal Form - CNF)** με χρήση λογικών ισοδυναμιών
 - **CNF**: Μια σύζευξη όρων, όπου κάθε όρος είναι μια διάζευξη κριτηρίων (λεκτικών στοιχείων)
 - Π.χ., $\alpha \wedge \beta$, $\neg \alpha \vee \beta$, $\neg \alpha \wedge (\alpha \vee \beta)$
 - Π.χ., $(day < 8/9/94 \text{ OR } bid = 5 \text{ OR } sid = 3) \text{ AND } (rname = 'Paul' \text{ OR } bid = 5 \text{ OR } sid = 3)$
 - Όχι σε CNF: $\neg(\alpha \vee \beta)$, $(\alpha \wedge \beta) \vee \gamma$
 - Μετατροπή σε CNF: $\neg \alpha \wedge \neg \beta$, $(\alpha \vee \gamma) \wedge (\beta \vee \gamma)$
 - Μία λογική έκφραση δεν είναι πάντα ισότητα
 - Π.χ., $\bowtie_{DID > DNO}$

Συνένωση Ισότητας και Φυσική συνένωση

- **Συνένωση Ισότητας (EquiJoin):** Ειδική περίπτωση του τελεστή συνένωσης \bowtie_{Θ} όπου το Θ είναι ισοδυναμία
- Όπως και στον τελεστή συνένωσης η νέα σχέση έχει δύο γνωρίσματα με τα ίδια στοιχεία (δηλ., PERSON.DID, DEPARTMENT.DID)
- **Φυσική Συνένωση (Natural Join) *:** Ειδική περίπτωση του EquiJoin όπου τα ονόματα των γνωρισμάτων είναι όμοια.
 - Το κοινό γνώρισμα εμφανίζεται μόνο μία φορά

PERSON			*	DEPARTMENT	
PID	NAME	DID		DID	DNAME
1	Andreas	1		1	ACC
2	Kostas	1		2	HR
3	Maria	1		3	IT
4	Eleni	2			
5	Nikos	2			
6	Eleni	3			

PERSON * DEPARTMENT			
PID	NAME	DID	DNAME
1	Andreas	1	ACC
2	Kostas	1	ACC
3	Maria	1	ACC
4	Eleni	2	HR
5	Nikos	2	HR
6	Eleni	3	IT

Τελεστής Ημι-Συνένωσης (SemiJoin Operator)

$$R \bowtie_{\Theta} S$$

- **Είσοδος:** Δύο σχέσεις R και S, Συνθήκη συνένωσης (Θ)
- **Έξοδος:** Νέα σχέση η οποία συνενώνει τις R και S και περιλαμβάνει όλα τα γνωρίσματα που ανήκουν **μόνο στην R** και περιλαμβάνει όλες τις πλειάδες (γραμμές) που ικανοποιούν την συνθήκη Θ

- **Παράδειγμα:** $\text{PERSON} \bowtie_{P.DID=D.DNO} \text{DEPARTMENT}$

PERSON			DEPARTMENT		PERSON_DEPARTMENT		
PID	NAME	DID	DNO	DNAME	PID	NAME	DID
1	Andreas	1	1	ACC	1	Andreas	1
2	Kostas	1	2	HR	2	Kostas	1
3	Maria	1			3	Maria	1
4	Eleni	2			4	Eleni	2
5	Nikos	2			5	Nikos	2
6	Eleni	3					

Τελεστής Αντι-Συνένωσης (AntiJoin Operator)

$$R \triangleright_{\Theta} S$$

- **Είσοδος:** Δύο σχέσεις R και S, Συνθήκη συνένωσης (Θ)
- **Έξοδος:** Νέα σχέση η οποία συνενώνει τις R και S και περιλαμβάνει όλα τα γνωρίσματα που ανήκουν **μόνο στην R** και περιλαμβάνει όλες τις πλειάδες (γραμμές) που **ΔΕΝ** ικανοποιούν την συνθήκη Θ

- **Παράδειγμα:** $\text{PERSON} \triangleright_{P.DID=D.DNO} \text{DEPARTMENT}$

PERSON		
PID	NAME	DID
1	Andreas	1
2	Kostas	1
3	Maria	1
4	Eleni	2
5	Nikos	2
6	Eleni	3



DEPARTMENT	
DNO	DNAME
1	ACC
2	HR

PERSON_DEPARTMENT		
PID	NAME	DID
6	Eleni	3

Τελ. Αριστερ. Εξωτερικ. Συνένωσης (Left Outer Join)

$R \bowtie_{\Theta} S$ ή συμμετρικά (δεξιά) $S \bowtie_{\Theta} R$

- **Είσοδος:** Δύο σχέσεις R και S, Συνθήκη συνένωσης (Θ)
- **Έξοδος:** Νέα σχέση η οποία συνενώνει τις R και S και περιλαμβάνει όλα τα γνωρίσματα των R, S και **όλες τις πλειάδες της R (αριστερή)**. Αν μία πλειάδα της R δεν μπορεί να συνενωθεί με κάποια πλειάδα της S τότε τα γνωρίσματα της S στη νέα σχέση έχουν την τιμή **null**.
- **Παράδειγμα:** $PERSON \bowtie_{P.DID=D.DNO} DEPARTMENT$

PERSON			DEPARTMENT		PERSON_DEPARTMENT				
PID	NAME	DID	DNO	DNAME	PID	NAME	DID	DNO	DNAME
1	Andreas	1	1	ACC	1	Andreas	1	1	ACC
2	Kostas	1	2	HR	2	Kostas	1	1	ACC
3	Maria	1			3	Maria	1	1	ACC
4	Eleni	2			4	Eleni	2	2	HR
5	Nikos	2			5	Nikos	2	2	HR
6	Eleni	3			6	Eleni	3	null	null

Τελ. Ολικής Εξωτερικής Συνένωσης (Full Outer Join)

$R \bowtie_{\Theta} S$

- **Είσοδος:** Δύο σχέσεις R και S, Συνθήκη συνένωσης (Θ)
- **Έξοδος:** Νέα σχέση η οποία συνενώνει τις R και S και περιλαμβάνει όλα τα γνωρίσματα των R, S και όλες τις πλειάδες της R. Αν μία πλειάδα της R δεν μπορεί να συνενωθεί με κάποια πλειάδα της S τότε τα γνωρίσματα της S στη νέα σχέση έχουν την τιμή **null**
- **Παράδειγμα:** $\text{PERSON} \bowtie_{P.DID=D.DNO} \text{DEPARTMENT}$

PERSON			DEPARTMENT		PERSON_DEPARTMENT				
PID	NAME	DID	DNO	DNAME	PID	NAME	DID	DNO	DNAME
1	Andreas	1	1	ACC	1	Andreas	1	1	ACC
2	Kostas	1	2	HR	2	Kostas	1	1	ACC
3	Maria	1	4	IT	3	Maria	1	1	ACC
4	Eleni	2			4	Eleni	2	2	HR
5	Nikos	2			5	Nikos	2	2	HR
6	Eleni	3			6	Eleni	3	null	null
					null	null	null	4	IT

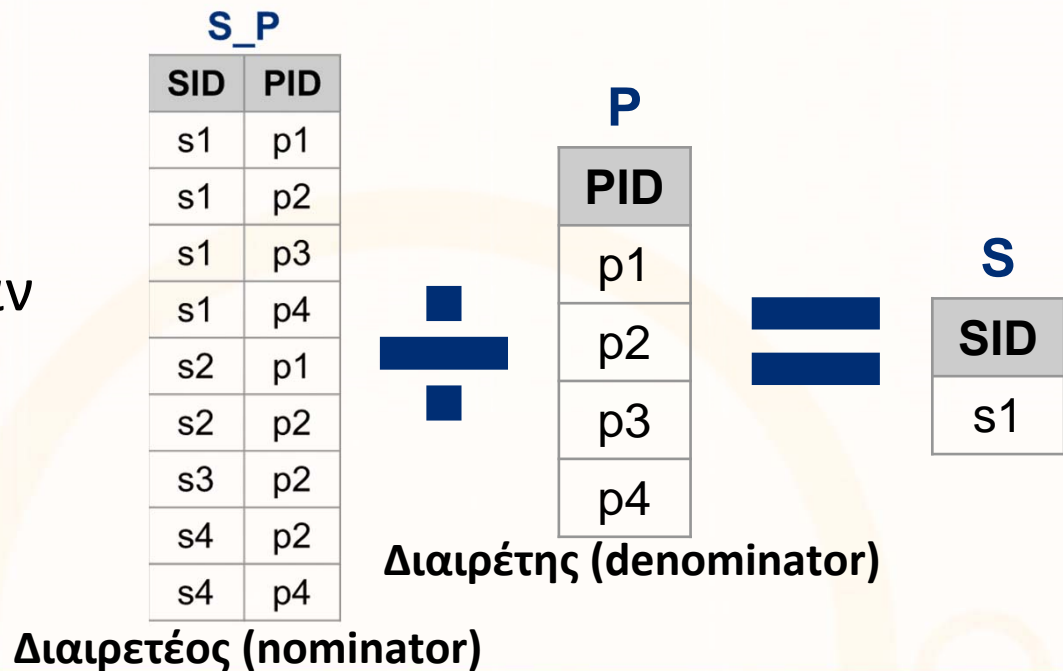
Τελεστής Διαίρεσης (Division Operator)

$$R \div S \quad R / S = \{ x \mid \exists(x,y) \in R \quad \forall y \in S \}$$

- **Είσοδος:** Δύο σχέσεις R και S
- **Έξοδος:** Νέα σχέση η οποία περιλαμβάνει τις πλειάδες της R οι οποίες μπορούν να συνενωθούν με κάθε πλειάδα της S σύμφωνα με τα κοινά τους γνωρίσματα.
- **Παράδειγμα:** Βρες τους Suppliers (S) που προσφέρουν ΌΛΑ τα Parts (P) Suppliers \div Parts

- Δεν είναι βασικός τελεστής: μπορεί να εκφραστεί και σαν

- $T_1 \leftarrow \pi_A (R)$
- $T_2 \leftarrow \pi_A ((T_1 \times S) - R)$
- $T \leftarrow T_2 - T_1$



Παραδείγματα Διαίρεσης

Διαίρεση (\div)

S_P	
SID	PID
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4



P
PID
p1
p2
p3
p4

P
PID
p1
p2

P
PID
p2



S
SID
s1

S
SID
s1
s2

S
SID
s1
s2
s3
s4

Τελεστής Διαίρεσης (Division Operator)

- R=SUPPLIERS_PARTS, S=PARTS

1. $T_1 \leftarrow \pi_A (R)$: Βρες τους μοναδικούς Suppliers

T_1

SID
s1
s2
s3
s4

2. $T_2 \leftarrow \pi_A ((T_1 \times S) - R)$

- Βρες όλους τους συνδυασμούς των μοναδικών Suppliers με τα parts της επερώτησης (όλα τα parts)
- Αφαίρεσε τους συνδυασμούς που υπάρχουν ώστε να παραμείνουν οι ανεπιθύμητοι συνδυασμοί
- Επέστρεψε λίστα με SID

$T_1 \times S$

SID	PID
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s2	p3
s2	p4
s3	p1
s3	p2
s3	p3
s3	p4
s4	p1
s4	p2
s4	p3
s4	p4

R

SID	PID
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s2	p2
s2	p2
s3	p2
s3	p2
s4	p2
s4	p4

$(T_1 \times S) - R$

SID	PID
s2	p3
s2	p4
s3	p1
s3	p3
s3	p4
s4	p1
s4	p3

T_2

SID
s2
s3
s4

3. $T \leftarrow T_1 - T_2$

- Αφαίρεσε από την αρχική λίστα τους ανεπιθύμητους

T

SID
s1

Τελεστής Διαίρεσης (Division Operator)

- Δεν υλοποιείται στις περισσότερες γλώσσες βάσεων δεδομένων αλλά μπορεί να εκφραστεί με πολλούς τρόπους στην SQL π.χ., EXISTS, EXCEPT, κτλ.
- Πρακτικά, εμφανίζεται αραιά σε ερωτήσεις
- Αναγνωρίζουμε όταν **...όλα, ...κάθε, ...συγκεκριμένα**
- Προτεινόμενη Στρατηγική Επίλυσης: Διαίρει και Βασίλευε
 - Βρες τους Suppliers (S) (σε κάποια συγκεκριμένη λίστα)
 - Π.χ., όλους, suppliers σε συγκεκριμένη πόλη → κριτήρια επιλογής
 - που προσφέρουν ΌΛΑ τα Parts (σε κάποια συγκεκριμένη λίστα)
 - Π.χ., όλα, parts που έχουν συγκεκριμένη τιμή, → κριτήρια επιλογής
- Ερώτηση: ***Βρες τους suppliers της Λευκωσίας που πουλάνε όλα τα parts με χρώμα κόκκινο***

Συναθροιστικές Συναρτήσεις και Ομαδοποίηση

- **Ερώτηση:** Πως μπορείτε να βρείτε το μέγιστο μισθό και μέση ηλικία υπαλλήλου ανά τμήμα;
- **Απάντηση:** Η αρχική έκδοση της σχεσιακής άλγεβρας δεν υποστήριζε συναθροιστικές συναρτήσεις/τελετές

$\langle \textit{grouping-list} \rangle \mathfrak{F} \langle \textit{function-list} \rangle (\mathbf{R})$

- function-list: Min, Max, Average, Count, ...
- grouping-list: 0 ή περισσότερα γνωρίσματα από τη σχέση (R)
- **Παράδειγμα:**

$\langle \mathbf{DID} \rangle \mathfrak{F} \langle \mathbf{MAX}(\textit{salary}), \mathbf{AVG}(\textit{age}) \rangle (\mathbf{EMPLOYEE})$

Συναθροιστικές Συναρτήσεις

- **COUNT:** καταμέτρηση πλειάδων

- Παράδειγμα: $\mathfrak{S}_{COUNT(PID)}(EMPLOYEE) = 6$
- Προσοχή: Δεν μετριοούνται τα null και τα διπλότυπα (τα πάντα είναι σχέσεις)

- Π.χ., $\mathfrak{S}_{COUNT(DNO)}(EMPLOYEE) = 3$, $\mathfrak{S}_{COUNT(SALARY)}(EMPLOYEE) = 5$

- **MIN/MAX:** Ελάχιστη/Μέγιστη Τιμή

- Παράδειγμα: $\mathfrak{S}_{MAX(SALARY)}(EMPLOYEE) = 1800$

- Άλλα παραδείγματα (T-SQL):

- CHECKSUM_AGG
- SUM
- STDEV
- VAR

EMPLOYEE			
PID	NAME	DNO	SALARY
1	Andreas	1	1000
2	Kostas	1	1200
3	Maria	1	1500
4	Eleni	2	1200
5	Nikos	2	1600
6	Eleni	3	1800

Ομαδοποίηση

- Συνεργασία (συνήθως) με συναθροιστικές συναρτήσεις για την παραγωγή πληροφοριών ανά ομάδα
- Π.χ., $\langle DNO \rangle \mathfrak{F} \langle MAX(salary), COUNT(PID) \rangle (EMPLOYEE)$

EMPLOYEE			
PID	NAME	DNO	SALARY
1	Andreas	1	1000
2	Kostas	1	1200
3	Maria	1	1500
4	Eleni	2	1200
5	Nikos	2	1600
6	Eleni	3	1800

???		
DNO	???	???
1	1500	3
2	1600	2
3	1800	1

- Είναι προτιμητέο να μετονομάζουμε τα αποτελέσματα ώστε να έχουν χρήσιμα ονόματα γνωρισμάτων
- Π.χ., $T_{(DNO,MSALARY,PCOUNT)} \leftarrow \langle DNO \rangle \mathfrak{F} \langle MAX(salary), COUNT(PID) \rangle (EMPLOYEE)$

Λογισμός Πλειάδων (Tuple Relational Calculus)

- Δηλωτική Έκφραση
 - Δηλώνει μία αίτηση ανάκτησης με μη διαδικαστικό τρόπο
- Κάθε ερώτηση της βασικής σχεσιακής άλγεβρας μπορεί να αναπαρασταθεί με το λογισμό πλειάδων
- Γενική σύνταξη του λογισμού πλειάδων
$$\{t_1.A_j, t_2.A_k, \dots, t_n.A_m \mid COND(t_1, t_2, \dots, t_n, t_{n+1}, \dots, t_{n+m})\}$$

t_i : μεταβλητές πλειάδων, A_i : γνωρίσματα, **COND**: συνθήκη
- Πρέπει να καθοριστεί:
 - Η σχέση μεταξύ των πλειάδων
 - Τα γνωρίσματα που πρέπει να ανακτηθούν
 - Η συνθήκη που πρέπει να ικανοποιούν
- Χρησιμοποιεί τους ποσοδείκτες (για κάθε \forall), (υπάρχει \exists)

Λογισμός Πεδίων Ορισμού (Domain Relational Calculus)

- Διαφορετικό από το λογισμό πλειάδων
 - Χρησιμοποιεί μοναδιαίες μεταβλητές από πεδία ορισμού των γνωρισμάτων
- Οι φόρμουλες αποτελούνται από atoms
 - Έξοδος: True ή False
 - Αποκαλούνται σαν οι truth values των atoms
- Παράδειγμα Γλώσσας: Query By Example (QBE)
- Χρησιμοποιεί επίσης τους ποσοδείκτες (για κάθε \forall), (υπάρχει \exists)
- Λογισμός Πλειάδων:
 - **$\{t.a, t.b, \dots \mid R(t) \text{ AND COND}(t.a, t.b) \}$**
- Λογισμός Πεδίου Ορισμού:
 - **$\{ab \mid R(ab\dots) \text{ AND COND}(t.a, t.b) \}$**

Σύνοψη Σχεσιακού Λογισμού

- Παράδειγμα

EMPLOYEE (E)			
PID	PNAME	DNO	SALARY
1	Andreas	1	1000
2	Kostas	1	1200
3	Maria	1	1500
4	Eleni	2	1200
5	Nikos	2	1600
6	Eleni	3	1800

DEPARTMENT (D)	
DID	DNAME
1	IT
2	HR
3	AC
4	CRM

- Σχεσιακή Άλγεβρα:

- $\pi_{PID, PNAME, DNAME} \sigma_{SALARY \geq 1500} (E \bowtie_{DNO=DID} D)$

- Λογισμός Πλειάδων:

- $\{t.PID, t.PNAME, d.DNAME \mid E(t) \text{ AND } t.SALARY \geq 1500 \text{ AND } D(d) \text{ AND } t.DNO=d.did\}$

- Λογισμός Πεδίου Ορισμού:

- $\{abf \mid (\exists c) (\exists d) (\exists e) E(abcd) \text{ AND } D(ef) \text{ AND } d \geq 1500 \text{ AND } c=e\}$

Αναδρομική Κλειστότητα

- Αναδρομική Κλειστότητα (Recursive or Transitive Closure Operations)
- Επεξήγηση με Παράδειγμα:
 - Supervisor(1) → 3
 - Supervisor(3) → 6
 - Supervisor(6) → NULL
- Ποιοί είναι οι Supervisor του (1) αναδρομικά;
 - Απάντηση A = {3, 6, NULL}

PID	PNAME	SID
1	Andreas	3
2	Kostas	3
3	Maria	6
4	Eleni	5
5	Nikos	6
6	Eleni	NULL

- Στη Σχεσιακή Άλγεβρα **δεν υπάρχει τελεστής** που να υποστηρίζει αυτή την πράξη.
- Μπορούμε ωστόσο να το υποστηρίξουμε με μια σειρά ενώσεων
- Σε SQL είναι δυνατό να υποστηριχθεί με κάποιο διαδικαστικό τρόπο, με χρήση επαναλήψεων, ή μέσω κάποιου εξειδικευμένου τελεστή.
- Μια ιδέα είναι η χρήση του αλγορίθμου Floyd-Warshall, το οποίο βρίσκει το ελάχιστο μονοπάτι μεταξύ οποιονδήποτε κορυφών (και κατ' επέκταση όλα τα δυνατά ζεύγη)

Παραδείγματα/Ασκήσεις

EMPLOYEE (E)			
PID	PNAME	DNO	SALARY
1	Andreas	1	1000
2	Kostas	1	1200
3	Maria	1	1500
4	Eleni	2	1200
5	Nikos	2	1600
6	Eleni	3	1800

DEPARTMENT (D)	
DID	DNAME
1	IT
2	HR
3	AC
4	CRM

- Τύπωσε το όνομα και το τμήμα (DNAME) του κάθε υπαλλήλου
- Τύπωσε το MAX PID από όλους τους υπαλλήλους του τμήματος 1
- Τύπωσε τα τμήματα που δεν έχουν υπαλλήλους
- Τύπωσε το όνομα του υπαλλήλου με το μέγιστο μισθό
- Τύπωσε για κάθε τμήμα τον αριθμό των υπαλλήλων του
- Βρες τους υπαλλήλους με τον ίδιο μισθό
- Βρες τα τμήματα με 2 ή περισσότερους υπαλλήλους