

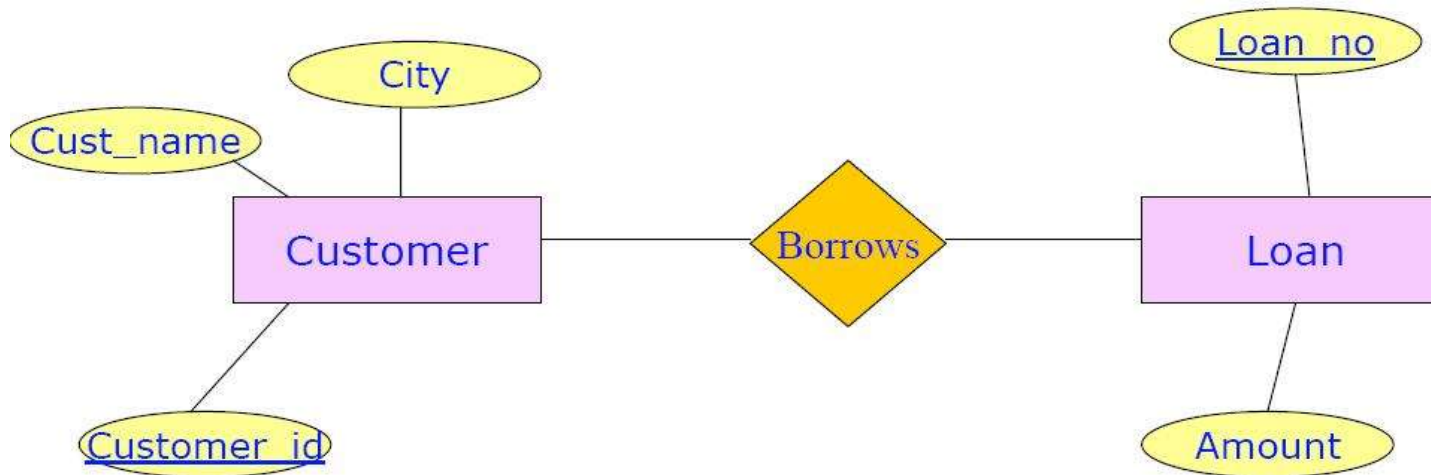
Βάσεις Δεδομένων III

Πληροφοριακά Συστήματα Διοίκησης

Διδάσκων: Δημόκας Νίκος, Ph.D.

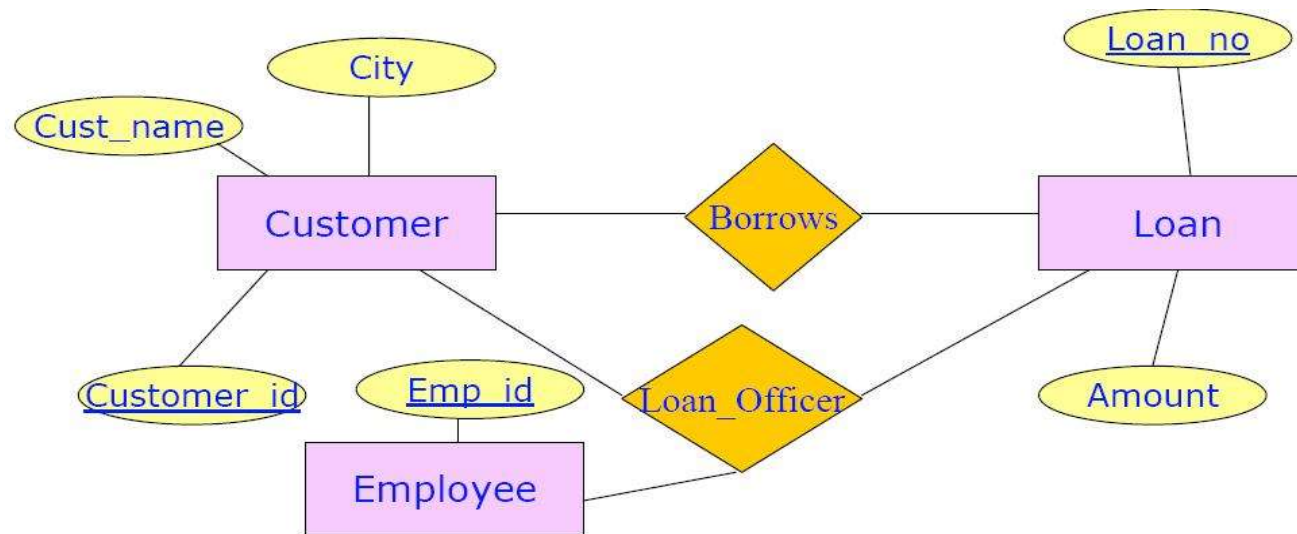
Μοντέλο Οντοτήτων Συσχετίσεων

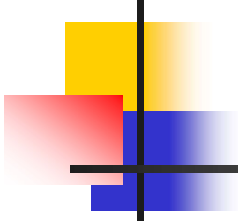
- Συνάθροιση (Aggregation)
- Ένας από τους περιορισμούς του μοντέλου οντοτήτων-σχέσεων είναι ότι δεν είναι δυνατός ο ορισμός σχέσεων μεταξύ σχέσεων
- Τέτοιες σχέσεις είναι συχνά απαραίτητες.
- Για παράδειγμα:



Μοντέλο Οντοτήτων Συσχετίσεων

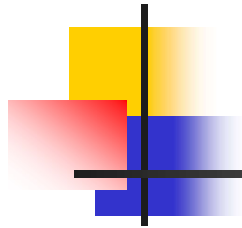
- Υποθέστε ότι σε κάθε ζεύγος πελάτη-δανείου αντιστοιχίζεται ένας υπάλληλος ως υπεύθυνος. Πως θα αναπαρασταθεί αυτή η σχέση?
- Ένας τρόπος θα ήταν να εισαχθεί μια καινούργια σχέση μεταξύ πελατών και δανείων.





Μοντέλο Οντοτήτων Συσχετίσεων

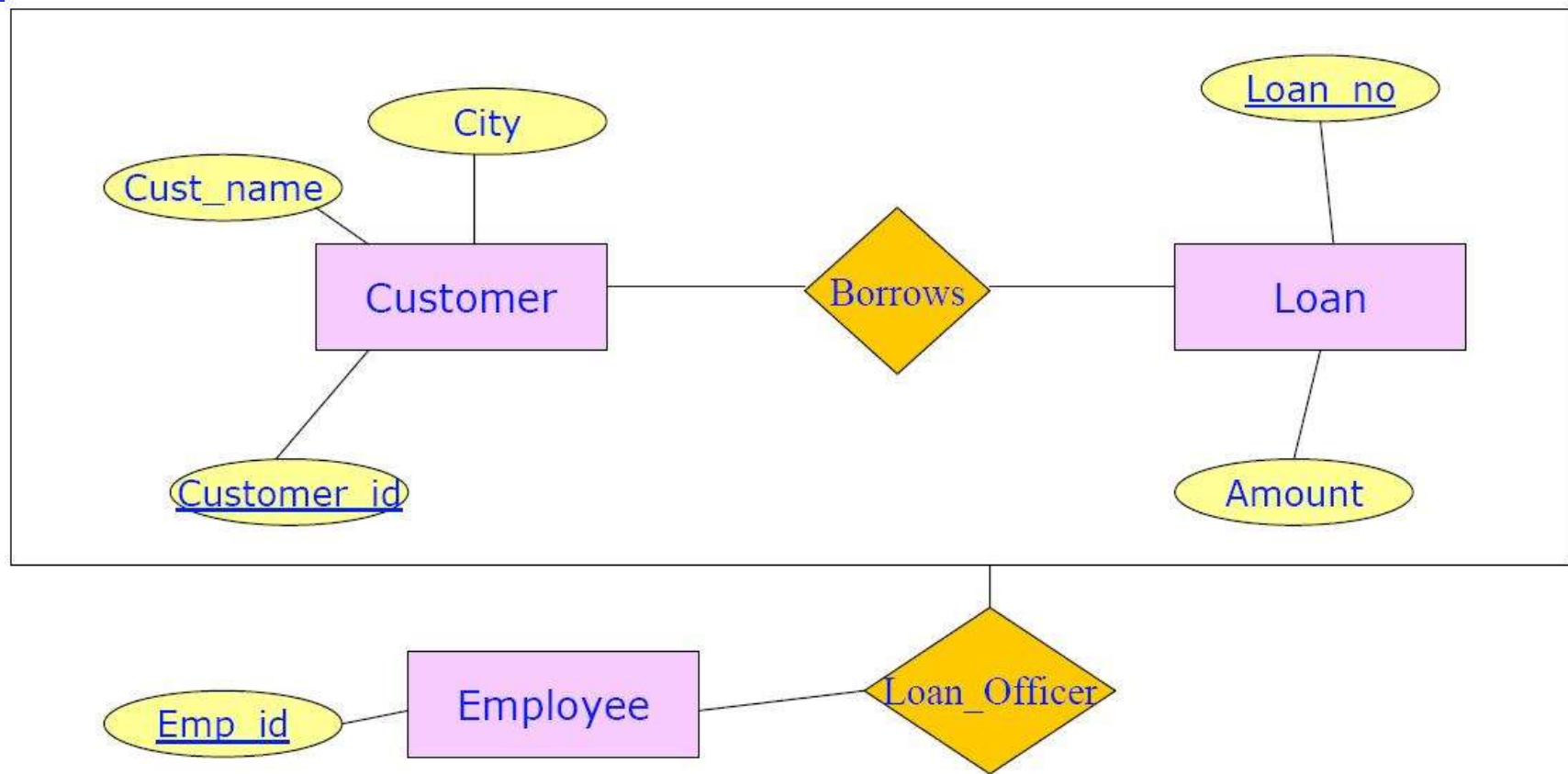
- Μπορούν οι σχέσεις Borrows και Loan_Officer να συνδυαστούν σε μία σχέση?
 - Αν συνδυαστούν, αυτό σημαίνει ότι ένας υπάλληλος πρέπει να αντιστοιχηθεί σε κάθε ζεύγος πελάτη-δανείου.
 - Υπάρχει επίσης πλεονάζουσα πληροφορία: κάθε ζεύγος πελάτη-δανείου στη σχέση Loan_Officer ανήκει και στη σχέση Borrows.
 - Εναλλακτικά, μπορεί να εισαχθεί το γνώρισμα loan_officer στη σχέση Borrows.
 - Πως όμως θα βρούμε τα ζεύγη πελατών δανείων για τα οποία ένας υπάλληλος είναι υπεύθυνος?

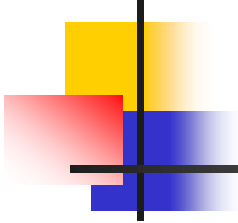


Μοντέλο Οντοτήτων Συσχετίσεων

- Το πρόβλημα μπορεί να επιλυθεί με τη χρήση του μηχανισμού της συνάθροισης, δηλαδή να θεωρήσουμε ότι η σχέση Borrows και οι σχετιζόμενες οντότητες αποτελούν μια οντότητα υψηλότερου επιπέδου.
- Μια τέτοια οντότητα συμπεριφέρεται όπως κάθε άλλη οντότητα. Ποιο θα είναι το αναγνωριστικό της?
 - Το αναγνωριστικό θα σχηματιστεί από τα αναγνωριστικά των οντοτήτων που συμμετέχουν στη σχέση Borrows
- Συμβολισμός: χρησιμοποιείται ένα παραλληλόγραμμο το οποίο περικλείει τις οντότητες και τη σχέση

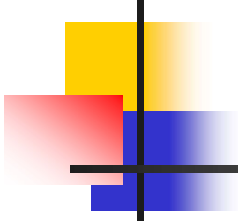
Μοντέλο Οντοτήτων Συσχετίσεων





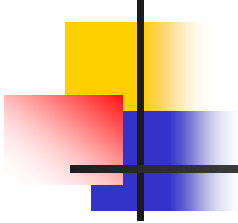
Σχεσιακό Μοντέλο Δεδομένων

- Συμβολισμός για την αναπαράσταση δεδομένων: πίνακες
- Κανόνες για τον καθορισμό της σωστής δομής των πινάκων
- Τελεστές για το χειρισμό πινάκων: σχεσιακή άλγεβρα (relational algebra)
- Η σχεσιακή άλγεβρα χρησιμοποιείται και ως γλώσσα ερωτήσεων (query language)
- SQL (structured query language): πρότυπο γλωσσών ερωτήσεων σχεσιακών βάσεων δεδομένων



Σχεσιακό Μοντέλο Δεδομένων

- Παράδειγμα: CAP database
 - A DB used by a wholesale business in order to keep track of its customers, the products it sells and the agents that place orders for products on behalf of customers.
 - Customers are retail businesses that order large quantities of products from the wholesale company.
 - Customers, products, agents and orders are distinguished by unique identifiers.



Σχεσιακό Μοντέλο Δεδομένων

- Ορολογία:
 - Πίνακες / σχέσεις (tables / relations)
 - Γραμμές / πλειάδες (rows / tuples)
 - Στήλες / γνωρίσματα (columns / attributes)
 - Μια ΒΔ είναι ένα σύνολο από σχέσεις με μοναδικά ονόματα
 - Π.χ., η ΒΔ CAP ορίζεται ως $CAP = \{Customers, Agents, Products, Orders\}$
 - Το σχήμα ή επικεφαλίδα (heading) ενός πίνακα είναι το σύνολο των ονομάτων των γνωρισμάτων της σχέσης που αναπαριστά
 - Π.χ. $Head(Customers) = \{cid, cname, city, discent\}$

Σχεσιακό Μοντέλο Δεδομένων

Entity-Relation

STUDENT	Name	StudentNumber	Class	Major
	Smith	17	1	CS
	Brown	8	2	CS

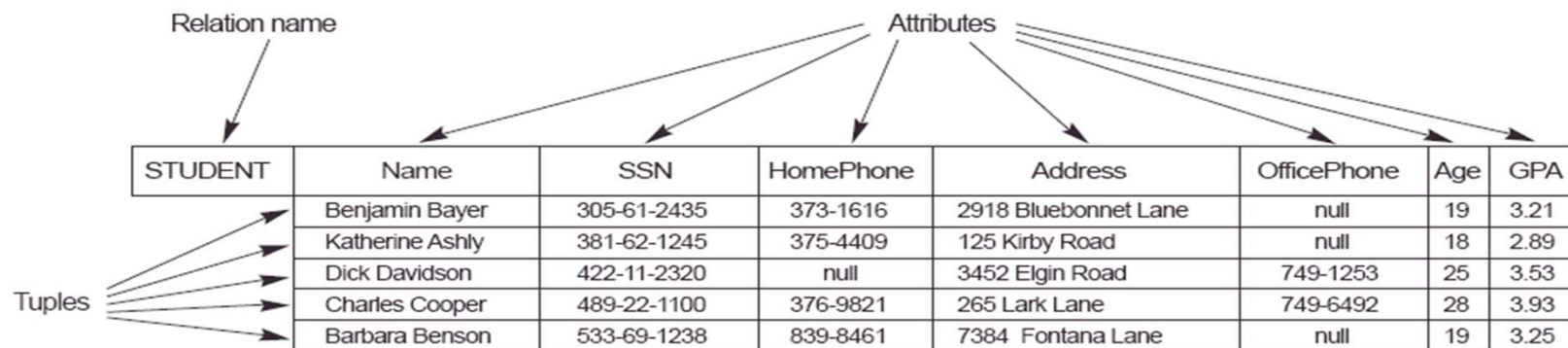
COURSE	CourseName	CourseNumber	CreditHours	Department
	Intro to Computer Science	CS1310	4	CS
	Data Structures	CS3320	4	CS
	Discrete Mathematics	MATH2410	3	MATH
	Database	CS3380	3	CS

SECTION	SectionIdentifier	CourseNumber	Semester	Year	Instructor
	85	MATH2410	Fall	98	King
	92	CS1310	Fall	98	Anderson
	102	CS3320	Spring	99	Knuth
	112	MATH2410	Fall	99	Chang
	119	CS1310	Fall	99	Anderson
	135	CS3380	Fall	99	Stone

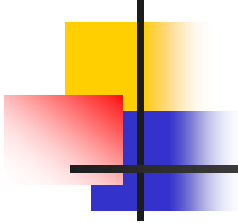
GRADE_REPORT	StudentNumber	SectionIdentifier	Grade
	17	112	B
	17	119	C
	8	85	A
	8	92	A
	8	102	B
	8	135	A

Σχεσιακό Μοντέλο Δεδομένων

- Η σειρά των tuples ΔΕΝ ΠΑΙΖΕΙ ΑΠΟΛΥΤΩΣ ΚΑΝΕΝΑ ΡΟΛΟ

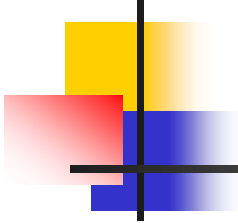


STUDENT	Name	SSN	HomePhone	Address	OfficePhone	Age	GPA
	Dick Davidson	422-11-2320	null	3452 Elgin Road	749-1253	25	3.53
	Barbara Benson	533-69-1238	839-8461	7384 Fontana Lane	null	19	3.25
	Charles Cooper	489-22-1100	376-9821	265 Lark Lane	749-6492	28	3.93
	Katherine Ashly	381-62-1245	375-4409	125 Kirby Road	null	18	2.89



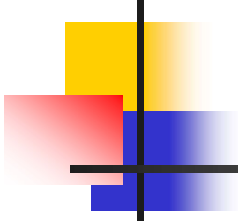
Σχεσιακό Μοντέλο Δεδομένων

- Το σύνολο των σχημάτων των σχέσεων ονομάζεται σχήμα της βάσης δεδομένων (database schema).
- Το σύνολο όλων των πλειάδων των σχέσεων λέγεται περιεχόμενο της βάσης δεδομένων (content, extension). Για μια σχέση R , το περιεχόμενο συμβολίζεται ως $\text{cont}(R)$
- Κάθε γνώρισμα μιας σχέσης έχει έναν τύπο (type). Τα γνωρίσματα παίρνουν τιμές από ένα πεδίο τιμών (domain). Το σύνολο αυτό περιέχει τις σταθερές οι οποίες μπορούν να εμφανίζονται ως τιμές ενός γνωρίσματος.



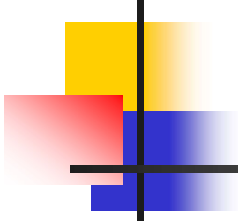
Σχεσιακό Μοντέλο Δεδομένων

- Το πεδίο τιμών ενός γνωρίσματος A συμβολίζεται ως $\text{domain}(A)$.
- Μια σχέση R με σχήμα $\{A_1, A_2, \dots, A_m\}$ είναι ένα υποσύνολο του $\text{domain}(A_1) \times \text{domain}(A_2) \times \dots \times \text{domain}(A_m)$



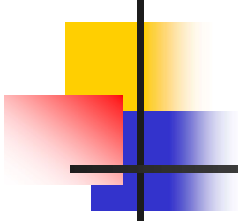
Σχεσιακό Μοντέλο Δεδομένων

- Παράδειγμα: CAP Database
 - $\text{head}(\text{Customers}) = \{\text{cid}, \text{cname}, \text{city}, \text{discent}\}$
 - $(\text{c001}, \text{TipTop}, \text{Paris}, 10.00) \in \text{cont}(\text{Customers})$
 - $(\text{c002}, \text{ACME}, \text{Kyoto}, 8.00) \in \text{domain}(\text{cid}) \times \text{domain}(\text{cname}) \times \text{domain}(\text{city}) \times \text{domain}(\text{discent})$
 - $(\text{c002}, \text{ACME}, \text{Kyoto}, 8.00) \notin \text{cont}(\text{Customers})$



Σχεσιακό Μοντέλο Δεδομένων

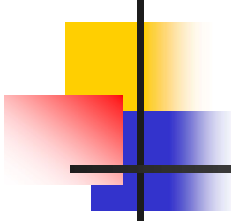
- Σχεσιακοί Κανόνες:
 - Ορίζουν επιλογές και περιορισμούς στη δομή των σχέσεων.
 - Έχουν σημαντικές επιπτώσεις στην «καλή συμπεριφορά» των δομών που χρησιμοποιούνται.
 - Υποδεικνύουν σημεία όπου είναι δυνατή η τυποποίηση.
 - Δεν ακολουθούνται πάντα από εμπορικά συστήματα.
- Κανόνας1 (Πρώτη Κανονική Μορφή)
 - Μια σχέση είναι σε Πρώτη Κανονική Μορφή (1NF) αν δεν έχει πλειότιμα γνωρίσματα



Σχεσιακό Μοντέλο Δεδομένων

- Κανόνας1 (Πρώτη Κανονική Μορφή)
 - Παράδειγμα: η παρακάτω σχέση (Employees) δεν είναι σε 1NF

Eid	Ename	Position	Dependents
E001	Smith, John	Agent	Michael J. Susan R.
E002	Andrews, David	Agent	David M.
E003	Jones, Franklin	Agent	Andrew K. Mark W. Lois M.

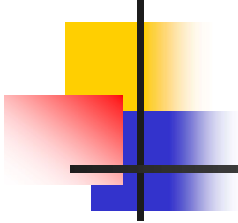


Σχεσιακό Μοντέλο Δεδομένων

- Κανόνας1 (Πρώτη Κανονική Μορφή)
 - Πρόβλημα: πως θα αναπαρασταθεί πληροφορία σχετικά με τα εξαρτώμενα πρόσωπα σε σχέσεις σε 1NF?
 - 1^η επιλογή: να αλλαχθεί το σχήμα ώστε να περιλάβει τόσα πρόσθετα γνωρίσματα όσα ο μέγιστος αριθμός εξαρτ. προσώπων

Eid	Ename	Position	Dependents1	Dependents2	Dependents3
E001	Smith, John	Agent	Michael J.	Susan R.	
E002	Andrews, David	Agent	David M.		
E003	Jones, Franklin	Agent	Andrew K.	Mark W.	Lois M.

- Μη αποδοτική λύση: σπατάλη χώρου, δυσκολία στις ερωτήσεις



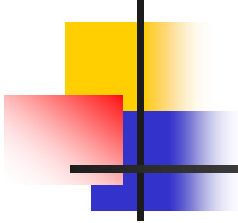
Σχεσιακό Μοντέλο Δεδομένων

- Κανόνας1 (Πρώτη Κανονική Μορφή)
 - 2^η επιλογή: να δημιουργηθεί μια νέα σχέση για τους υπαλλήλους και τα εξαρτώμενα από αυτούς πρόσωπα

eid	ename	position
e001	Smith, John	Agent
e002	Andrews, David	Superintendent
e003	Jones, Franklin	Agent

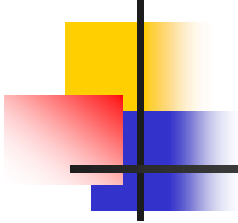
eid	dependent
e001	Michael J.
e001	Susan R.
e002	David M. Jr.
e003	Andrew K.
e003	Mark W.
e003	Lois M.

- Προτιμότερη λύση: η περιττή πληροφορία είναι ελάχιστη.



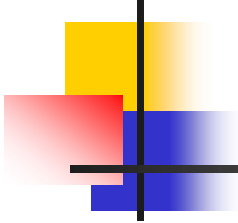
Σχεσιακό Μοντέλο Δεδομένων

- Υπάρχουν συστήματα (extended RDBMS) τα οποία επιτρέπουν τον ορισμό σχέσεων που δεν είναι σε 1NF.
- Κανόνας2: Πρόσβαση βάσει περιεχομένου (row access by content)
 - Οι πλειάδες μιας σχέσης συνιστούν ένα μη-διατεταγμένο σύνολο. Η πρόσβαση στις πλειάδες γίνεται μόνο βάσει του περιεχομένου τους
 - Ερωτήσεις της μορφής «ποια είναι η 3^η πλειάδα της σχέσης?» είναι μη-έγκυρες



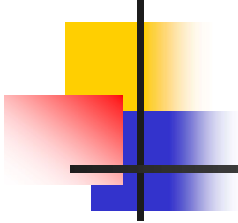
Σχεσιακό Μοντέλο Δεδομένων

- Κανόνας2: Πρόσβαση βάσει περιεχομένου (row access by content)
 - Οι ερωτήσεις πρέπει να βασίζονται στις τιμές των γνωρισμάτων
 - Πολλά εμπορικά συστήματα παραβιάζουν τον κανόνα αυτό και παρέχουν ένα εσωτερικό αναγνωριστικό πλειάδων (tupleid).
 - Πρόσβαση βάσει του tupleid ενδέχεται να έχει πλεονεκτήματα από την άποψη της απόδοσης του συστήματος.



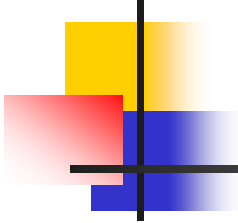
Σχεσιακό Μοντέλο Δεδομένων

- Κανόνας3: Μοναδικότητα πλειάδων (unique rows)
 - Μια σχέση αποτελείται από διακεκριμένες πλειάδες
 - Οι σχέσεις ορίζονται σαν σύνολα, επομένως δεν μπορούν να έχουν επαναλαμβανόμενα στοιχεία
 - Η εφαρμογή του κανόνα αυτού συνεπάγεται κόστος για τον έλεγχο της μοναδικότητας των πλειάδων
 - Πολλά συστήματα επιλέγουν τη μη-εφαρμογή του κανόνα αυτού για να αποφύγουν το επιπλέον κόστος



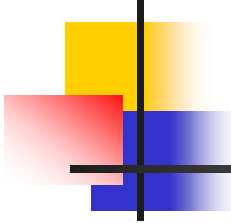
Σχεσιακό Μοντέλο Δεδομένων

- Κλειδιά και υπερ-κλειδιά (keys and superkeys)
 - Ο 3^{ος} σχεσιακός κανόνας επιβάλλει την απουσία πλειάδων οι οποίες έχουν τις ίδιες τιμές σε όλα τους τα γνωρίσματα
 - Άρα πρέπει να υπάρχει ένα υποσύνολο του συνόλου των γνωρισμάτων το οποίο μπορεί να χρησιμοποιηθεί προκειμένου να διακρίνουμε οποιεσδήποτε δύο πλειάδες μιας σχέσης



Σχεσιακό Μοντέλο Δεδομένων

- Κλειδιά και υπερ-κλειδιά (keys and superkeys)
 - Ένα σύνολο γνωρισμάτων μιας σχέσης αποτελεί ένα υπερ-κλειδί (superkey) της σχέσης αν οποιεσδήποτε δύο πλειάδες της σχέσης έχουν διακεκριμένες τιμές για το σύνολο αυτό των γνωρισμάτων
 - Ένα σύνολο γνωρισμάτων μιας σχέσης αποτελεί κλειδί (key) της σχέσης αν το σύνολο αυτό είναι ένα υπερ-κλειδί και κανένα υποσύνολο αυτού δεν έχει αυτή την ιδιότητα



Σχεσιακό Μοντέλο Δεδομένων

- Κλειδιά και υπερ-κλειδιά (keys and superkeys)
 - Η επιλογή των κλειδιών και υπερ-κλειδιών είναι θέμα της σχεδίασης μιας ΒΔ και δεν μπορεί να αποφασιστεί εξετάζοντας το περιεχόμενο της
 - Παράδειγμα:

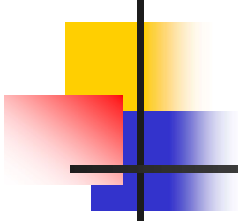
{pid, pname}: υπερ-κλειδί

{pid}: κλειδί

{pname}: δεν είναι κλειδί

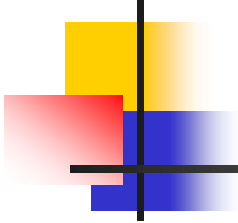
Products

pid	pname	city	quantity	price
p01	comb	Dallas	111400	0.50
p02	brush	Newark	203000	0.50
p03	razor	Duluth	150500	1.00
p04	pen	Duluth	125000	1.00
p05	pencil	Dallas	221400	1.00



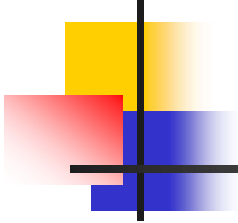
Σχεσιακό Μοντέλο Δεδομένων

- Ορισμός Έστω R μια σχέση με σχήμα $\text{head}(R) = \{A_1, A_2, \dots, A_n\}$ και έστω t μια πλειάδα της R . Αν $\{A_{i1}, A_{i2}, \dots, A_{ik}\}$ υποσύνολο του $\{A_1, A_2, \dots, A_n\}$, τότε ο περιορισμός (restriction) $t[A_{i1}, A_{i2}, \dots, A_{ik}]$ της t στα γνωρίσματα $\{A_{i1}, A_{i2}, \dots, A_{ik}\}$ είναι η πλειάδα τιμών της t στα γνωρίσματα $A_{i1}, A_{i2}, \dots, A_{ik}$
- Παράδειγμα: Αν $t = (p01, comb, Dallas, 111400, 0.50)$ τότε $t[\text{pid}, \text{city}, \text{price}] = (p01, Dallas, 0.50)$



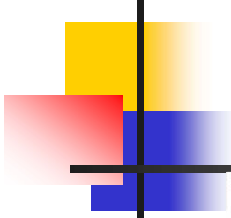
Σχεσιακό Μοντέλο Δεδομένων

- Ορισμός: Ένα κλειδί μιας σχέσης R με σχήμα $\text{head}(R) = \{A_1, A_2, \dots, A_n\}$ είναι ένα σύνολο γνωρισμάτων $K = \{A_{i1}, A_{i2}, \dots, A_{ik}\}$ για το οποίο:
 - Αν u, v είναι διακεκριμένες πλειάδες της R , τότε $u[K] \neq v[K]$
 - Δεν υπάρχει H υποσύνολο του K με την παραπάνω ιδιότητα



Σχεσιακό Μοντέλο Δεδομένων

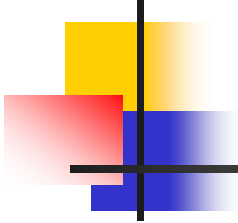
- Ένα σύνολο γνωρισμάτων το οποίο πληρεί την πρώτη ιδιότητα του προηγούμενου ορισμού λέγεται υπερ-κλειδί.
- Ένα κλειδί είναι ένα ελάχιστο υπερ-κλειδί.
- Το κενό σύνολο δεν μπορεί να είναι κλειδί.
- Μια σχέση μπορεί να έχει περισσότερα από ένα κλειδιά
 - Τα κλειδιά μιας σχέσης ονομάζονται υποψήφια κλειδιά
 - Ένα από τα υποψήφια κλειδιά επιλέγεται ως το πρωτεύον κλειδί
- Θεώρημα: Κάθε σχέση έχει τουλάχιστον ένα κλειδί



Σχεσιακό Μοντέλο Δεδομένων

CAR	<u>LicenseNumber</u>	EngineSerialNumber	Make	Model	Year
	Texas ABC-739	A69352	Ford	Mustang	96
	Florida TVP-347	B43696	Oldsmobile	Cutlass	99
	New York MPO-22	X83554	Oldsmobile	Delta	95
	California 432-TFY	C43742	Mercedes	190-D	93
	California RSK-629	Y82935	Toyota	Camry	98
	Texas RSK-629	U028365	Jaguar	XJS	98

- Superkey
 - {LicenseNumber, Make}
 - {LicenseNumber, EngineSerialNumber, make, model}
 - {LicenseNumber, make, model}
- Key (Candidate Key)
 - {LicenseNumber}, {EngineSerialNumber}
- Primary Key
 - Select one candidate key: {LicenseNumber}

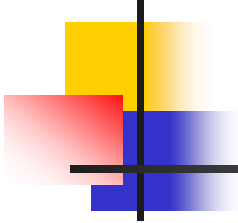


Σχεσιακό Μοντέλο Δεδομένων

- Κενές Τιμές
- Παράδειγμα: Έστω το παρακάτω στιγμιότυπο της σχέσης Products. Θέλουμε να εισάγουμε ένα καινούργιο προϊόν αλλά δε γνωρίζουμε όλη τη σχετική πληροφορία.

Products

pid	pname	city	quantity	price
p01	comb	Dallas	111400	0.50
p02	brush	Newark	203000	0.50
p03	razor	Duluth	150500	1.00
p04	pen	Duluth	125000	1.00
p05	pencil	Dallas	221400	1.00

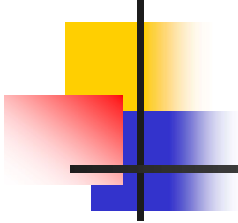


Σχεσιακό Μοντέλο Δεδομένων

- Κενές Τιμές
- Για τα γνωρίσματα των οποίων τις τιμές δεν γνωρίζουμε, θα χρησιμοποιήσουμε μια ειδική τιμή η οποία λέγεται κενή τιμή (null value)
- Η πλειάδα που εισάγεται έχει τη μορφή

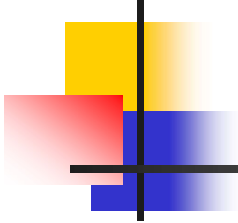
pid	pname	city	quantity	price
p07	stapler	null	null	3.50

- Οι «άγνωστες» τιμές μπορεί να γίνουν γνωστές σε κάποια επόμενη κατάσταση.



Σχεσιακό Μοντέλο Δεδομένων

- Προσοχή: ορισμένα γνωρίσματα δεν επιτρέπεται να δέχονται κενές τιμές
- 4^{ος} Σχεσιακός Κανόνας: Ακεραιότητα Οντοτήτων (Entity Integrity)
 - Γνωρίσματα τα οποία ανήκουν στο πρωτεύον κλειδί μιας σχέσης R δεν δέχονται κενές τιμές για καμία πλειάδα της R
 - Παραβίαση αυτού του κανόνα θα σήμαινε ότι δεν έχουμε τρόπο να ανακτήσουμε την πληροφορία που φυλάσσεται σε πλειάδες με κενές τιμές

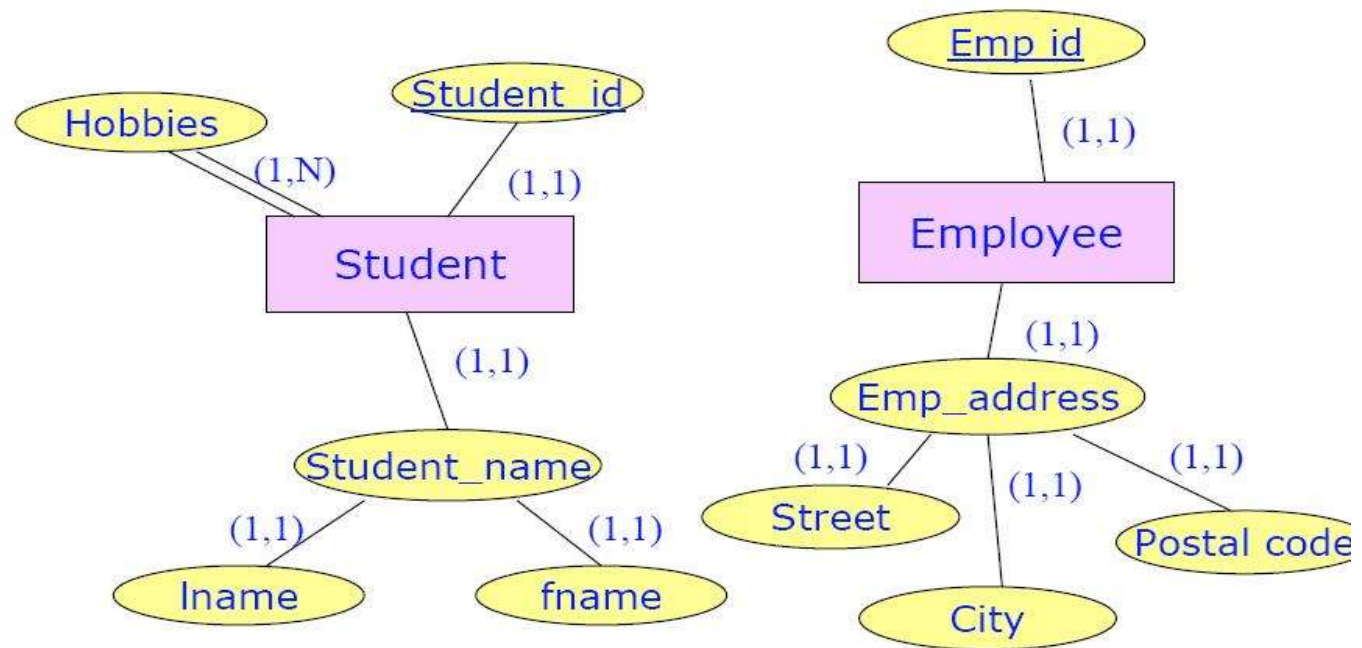


Μοντέλο Οντοτήτων-Σχέσεων → Σχεσιακό Μοντέλο

- Ένα εννοιολογικό μοντέλο που εκφράζεται με το συμβολισμό του μοντέλου Οντοτήτων – Σχέσεων μπορεί να μετατραπεί σε ένα σύνολο σχέσεων σύμφωνα με τους παρακάτω κανόνες:
 1. Κάθε οντότητα σε ένα διάγραμμα E-R απεικονίζεται σε μια σχέση με το ίδιο όνομα.
 - Όλα τα μονότιμα γνωρίσματα της οντότητας γίνονται γνωρίσματα της αντίστοιχης σχέσης.
 - Τα αναγνωριστικά γνωρίσματα της σχέσης σχηματίζουν ένα υποψήφιο κλειδί της σχέσης
 - Τα στιγμιότυπα της οντότητας γίνονται πλειάδες της σχέσης

Μοντέλο Οντοτήτων-Σχέσεων → Σχεσιακό Μοντέλο

- Παράδειγμα:

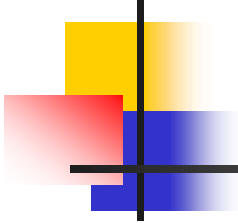


Student

sid	lname	fname
-----	-------	-------

Employee

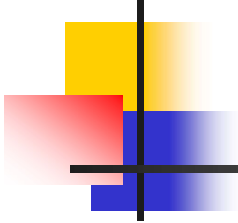
eid	street	city	postal_code
-----	--------	------	-------------



Μοντέλο Οντοτήτων-Σχέσεων → Σχεσιακό Μοντέλο

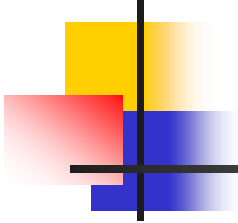
2. Μια οντότητα E με αναγνωριστικό p και ένα γνώρισμα πολλαπλών τιμών a απεικονίζεται σε μια σχέση με όνομα το όνομα του γνωρίσματος πολλαπλών τιμών.
- Το σχήμα της σχέσης περιλαμβάνει τα γνωρίσματα p και a και οι πλειάδες της είναι ζεύγη τιμών των p και a
 - Το πρωτεύον κλειδί της σχέσης είναι το σύνολο που περιλαμβάνει τα γνωρίσματα p και a .
 - Παράδειγμα:

Hobbies	
sid	hobby



Μοντέλο Οντοτήτων-Σχέσεων → Σχεσιακό Μοντέλο

3. Μια N-M σχέση r μεταξύ οντοτήτων E και F απεικονίζεται σε μια σχέση R το σχήμα της οποίας περιέχει όλα τα γνωρίσματα που ανήκουν στα πρωτεύοντα κλειδιά των σχέσεων που αντιστοιχούν στις οντότητες E και F . Ο συνδυασμός αυτός σχηματίζει το πρωτεύον κλειδί της R . Επίσης, το σχήμα της R περιέχει όλα τα γνωρίσματα της σχέσης r . Οι πλειάδες της R αντιστοιχούν στα στιγμιότυπα της r .



Μοντέλο Οντοτήτων-Σχέσεων → Σχεσιακό Μοντέλο

- Παράδειγμα:

Employee

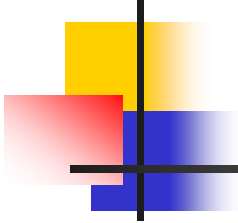
eid	street	city	postal_code
-----	--------	------	-------------

Projects

prid	pr_name	due_date
------	---------	----------

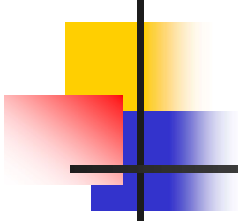
Works_on

eid	prid	percent
-----	------	---------



Μοντέλο Οντοτήτων-Σχέσεων → Σχεσιακό Μοντέλο

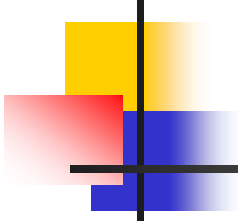
4. Για μια 1-N σχέση r μεταξύ οντοτήτων F και E δεν δημιουργούμε νέα σχέση για την αναπαράστασή της. Αν $\max\text{-card}(F,R) = 1$, τότε η σχέση της F πρέπει να περιέχει γνωρίσματα που αντιστοιχούν στο πρωτεύον κλειδί της E (ξένο κλειδί). Αν η F έχει υποχρεωτική συμμετοχή στην r , τότε το ξένο κλειδί δεν δέχεται κενές τιμές.



Μοντέλο Οντοτήτων-Σχέσεων → Σχεσιακό Μοντέλο

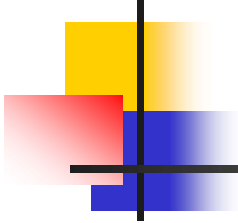
- Παράδειγμα:

Η σχέση teach μεταξύ instructors και courses είναι 1-N. Αν `i_id` και `c_id` είναι τα πρωτεύοντα κλειδιά των σχέσεων `instructors` και `courses` αντίστοιχα, τότε η σχέση `courses` πρέπει να περιέχει το `i_id` ως ξένο κλειδί, το οποίο δεν μπορεί να δέχεται κενές τιμές.



Μοντέλο Οντοτήτων-Σχέσεων → Σχεσιακό Μοντέλο

5. Αν οι οντότητες E και F συμμετέχουν σε μια 1-1 σχέση r , τότε αν η συμμετοχή των οντοτήτων είναι προαιρετική, δημιουργούμε σχέσεις για τις E και F και προσθέτουμε στη μία από αυτές ένα γνώρισμα για το πρωτεύον κλειδί της άλλης. Αν η συμμετοχή τους είναι υποχρεωτική, τότε οι δύο σχέσεις μπορούν να συνδυαστούν σε μία.

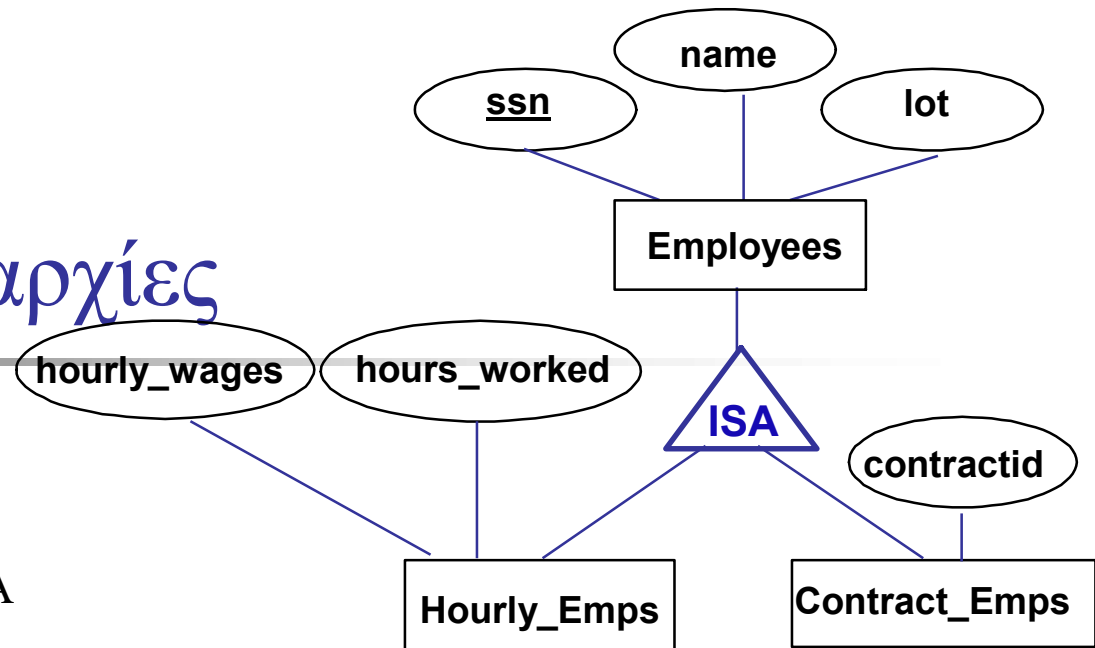


Μοντέλο Οντοτήτων-Σχέσεων → Σχεσιακό Μοντέλο

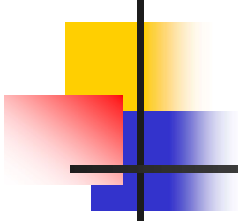
- Παράδειγμα:

Η σχέση supervise μεταξύ employees και projects είναι 1-1 με προαιρετική συμμετοχή και από τις δύο πλευρές. Στη σχέση employees προστίθεται το γνώρισμα rid (πρωτεύον κλειδί της σχέσης projects) ως ξένο κλειδί. Ισοδύναμα, το eid (πρωτεύον κλειδί της σχέσης employees) μπορεί να προστεθεί στη σχέση projects ως ξένο κλειδί.

ISA (is a) Ιεραρχίες

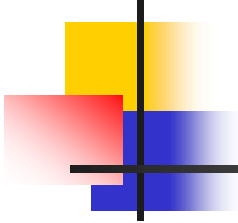


- Όπως στη C++, ή άλλες PLs, τα γνωρίσματα κληρονομούνται.
- Εάν δηλώσουμε ότι A ISA B, κάθε οντότητα A θεωρείται ότι είναι μια οντότητα B.
- Overlap constraints: Can Joe be an Hourly_Emps as well as a Contract_Emps entity? (default: disallowed; A overlaps B)
- Covering constraints: Does every Employees entity also have to be an Hourly_Emps or a Contract_Emps entity? (default: no; A AND B COVER C)
- Γιατί να χρησιμοποιούμε ISA;
 - Για να προσθέσουμε περιγραφικά γνωρίσματα σε μια υποκλάση.
 - Για να αναγνωρίσουμε οντότητες που συμμετέχουν σε σχέσεις.



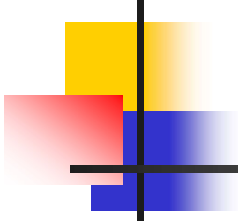
Μετατρέποντας ISA Ιεραρχίες σε Σχέσεις

- Γενική προσέγγιση:
 - 3 σχέσεις: Employees, Hourly_Emps and Contract_Emps.
 - Hourly_Emps: Every employee is recorded in Employees. For hourly emps, extra info recorded in Hourly_Emps (hourly_wages, hours_worked, ssn); must delete Hourly_Emps tuple if referenced Employees tuple is deleted).
 - Queries involving all employees easy, those involving just Hourly_Emps require a join to get some attributes.
- Εναλλακτικά: Just Hourly_Emps and Contract_Emps.
 - Hourly_Emps: ssn, name, lot, hourly_wages, hours_worked.
 - Each employee must be in one of these two subclasses.



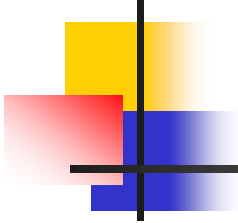
Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Αποτελούν μηχανισμό για τον έλεγχο της συνέπειας των δεδομένων.
- Χρησιμοποιούνται για να εξασφαλιστεί ότι μια βάση δεδομένων δεν θα βρεθεί ποτέ σε ασυνεπή κατάσταση.
- Περιορισμοί ακεραιότητας στο μοντέλο Οντοτήτων – Σχέσεων:
 - Ορισμός κλειδιών: η δήλωση ενός πρωτεύοντος (ή υποψήφιου κλειδιού) για ένα σύνολο οντοτήτων περιορίζει τις αποδεκτές εισαγωγές και ενημερώσεις σε αυτές που δεν δημιουργούν οντότητες με ίδια τιμή στο κλειδί.



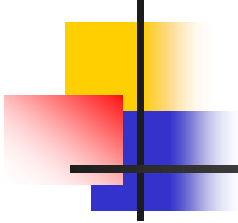
Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Περιορισμοί ακεραιότητας στο μοντέλο Οντοτήτων – Σχέσεων:
 - Περιορισμοί πληθικότητας: περιορίζουν το σύνολο των αποδεκτών σχέσεων μεταξύ συνόλων οντοτήτων
- Εν'γένει, οι περιορισμοί ακεραιότητας μπορούν να είναι αυθαίρετα κατηγορήματα που αναφέρονται σε μια βάση δεδομένων.



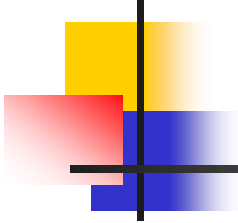
Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Περιοριζόμαστε σε τύπους περιορισμών οι οποίοι είναι εύκολο να ελεγχθούν μετά από κάθε αλλαγή που συμβαίνει στη βάση δεδομένων.
- Τύποι περιορισμών ακεραιότητας:
 1. Περιορισμοί Πεδίων Τιμών (Domain Constraints)
 - Στοιχειώδης τύπος περιορισμών ακεραιότητας
 - Χρησιμοποιείται για να ελεγχθεί αν τιμές γνωρισμάτων ανήκουν στα επιθυμητά πεδία τιμών



Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Ορισμός περιορισμών πεδίων τιμών στην SQL:
 - Ορισμός πεδίου τιμών: e.g., `create domain person-name char(20);` ορίζει το `person-name` σαν ένα πεδίο τιμών αποτελούμενο από strings μήκους 20. Το `person-name` μπορεί ακόλουθα να χρησιμοποιηθεί σαν τύπος γνωρισμάτων.
 - Η κενή τιμή (`null`) ανήκει σε κάθε πεδίο τιμών.



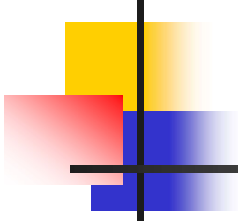
Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Παραδείγματα:

```
create domain hourly-wage numeric(5,2)
constraint wage-value-test
check(value >= 4.00);
```

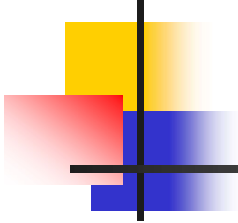
```
create domain account-number char(10)
constraint account-number-null-test
check(value not null);
```

```
create domain account-type char(10)
constraint account-type-test check(value in
("Checking", "Savings"));
```



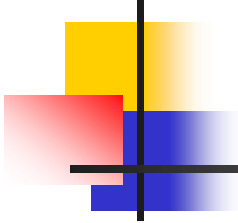
Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Περιορισμοί Αναφοράς (Referential Integrity Constraints)
 - Χρησιμοποιούνται για να διασφαλιστεί ότι τιμές γνωρισμάτων που εμφανίζονται σε μια σχέση εμφανίζονται και σε άλλες σχέσεις (όταν αυτές έχουν κοινά γνωρίσματα)
 - Εκκρεμείς Πλειάδες (dangling tuples): πλειάδες σε σχέσεις στις οποίες μπορεί να εφαρμοστεί join οι οποίες όμως δεν συμμετέχουν στο αποτέλεσμα του join. Δηλαδή, αν για την πλειάδα $tr \in r$, δεν υπάρχει πλειάδα $ts \in s$ έτσι ώστε $tr[r \cap s] = ts[r \cap s]$, τότε η πλειάδα tr λέγεται εκκρεμής.



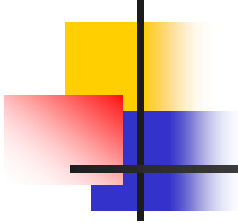
Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Παράδειγμα: έστω η σχέση account με σχήμα (account-number, branch-name, balance), η σχέση branch με σχήμα (branch-name, branch-city, assets). Έστω πλειάδα t στη σχέση account με t[branch name]="Atlantis". Αν στη σχέση branch δεν υπάρχει πλειάδα με αυτό το branch-name, τότε η πλειάδα t είναι εκκρεμής.



Περιορισμοί Ακεραιότητας (Integrity Constraints)

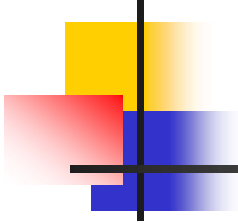
- Θέλουμε να έχουμε περιορισμούς οι οποίοι αποκλείουν τέτοιες καταστάσεις, ιδιαίτερα όταν το γνώρισμα στο οποίο εμφανίζονται «ανύπαρκτες» τιμές είναι ξένο κλειδί (foreign key).
- Ξένα κλειδιά: έστω $r1$ και $r2$ σχέσεις με κλειδιά $K1$ και $K2$ αντίστοιχα. Ένα υποσύνολο α του $K2$ είναι ξένο κλειδί αναφερόμενο στο $K1$, αν απαιτείται για κάθε πλειάδα $t2$ στην $r2$ να υπάρχει πλειάδα $t1$ στην $r1$ έτσι ώστε $t1[K1] = t2[\alpha]$.



Περιορισμοί Ακεραιότητας (Integrity Constraints)

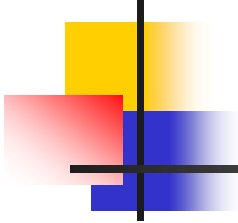
- Περιορισμοί αναφοράς και μεταβολή της ΒΔ.
 - Μεταβολές στην ΒΔ μπορούν να προκαλέσουν την παραβίαση περιορισμών αναφοράς.
 - Για κάθε είδος μεταβολής, υπάρχει μια συνθήκη η οποία πρέπει να ελεγχθεί για να διασφαλιστεί ότι δεν παραβιάζεται κανένας περιορισμός.
 - Έστω ότι ο περιορισμός αναφοράς περιγράφεται από την ακόλουθη συνθήκη:

$$\pi_a(r_2) \subseteq \pi_K(r_1)$$



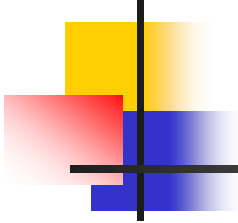
Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Εισαγωγή: αν η πλειάδα $t2$ εισαχθεί στη σχέση $r2$, πρέπει να ελεγχθεί ότι υπάρχει πλειάδα $t1$ στην $r1$ έτσι ώστε $t1[K] = t2[\alpha]$. Άρα η συνθήκη που πρέπει να ελεγχθεί είναι: $t2[\alpha] \in \pi_K(r1)$
- Διαγραφή: αν η πλειάδα $t1$ διαγραφεί από την $r1$ πρέπει να βρεθεί το σύνολο των πλειάδων της $r2$ που αναφέρονται στην $t1$: $\sigma_{\alpha = t1[K]}(r2)$. Αν αυτό το σύνολο δεν είναι κενό, τότε είτε η διαγραφή δεν θα εκτελεστεί, είτε οι πλειάδες που αναφέρονται στην $t1$ πρέπει να διαγραφούν επίσης.



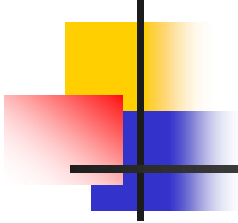
Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Διαγραφή: Η δεύτερη επιλογή μπορεί να οδηγήσει σε συνακόλουθες (cascaded) διαγραφές καθώς πλειάδες άλλων σχέσεων μπορεί να αναφέρονται σε πλειάδες που αναφέρονται στην $t1$ κ.ο.κ.
- Ενημέρωση: υπάρχουν δύο περιπτώσεις
 - Ενημέρωση στην $r2$: αν μια πλειάδα $t2$ της $r2$ ενημερώνεται και η ενημέρωση μεταβάλλει την τιμή του ξένου κλειδιού a , τότε αν $t2'$ είναι η νέα πλειάδα, πρέπει να ελεγχθεί αν $t2'[a] \in \pi_K(r1)$



Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Ενημέρωση: υπάρχουν δύο περιπτώσεις
 - Ενημέρωση στην r1: αν μια πλειάδα t1 της r1 ενημερώνεται και η ενημέρωση μεταβάλλει την τιμή του πρωτεύοντος κλειδιού K, τότε πρέπει να βρεθεί το σύνολο $\sigma_{a=t1[K]}(r2)$. Αν το σύνολο αυτό είναι μη-κενό, τότε είτε η ενημέρωση δεν εκτελείται είτε η ενημέρωση γίνεται με συνακόλουθες ενημερώσεις (όπως στην περίπτωση της διαγραφής).

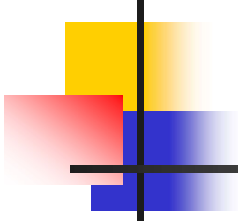


Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Ορισμός περιορισμών αναφοράς στην SQL:

```
create table customer (customer-name char(20)
not null, customer-street char(30), customer-
city char(30),
primary key (customer-name));
```

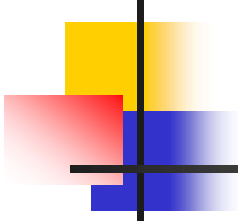
```
create table branch (branch-name char(15) not
null, branch-city char(30), assets integer,
primary key (branch-name),
check (assets>=0));
```



Περιορισμοί Ακεραιότητας (Integrity Constraints)

```
create table account (account-number char(10)
not null, branch-name char(15), balance integer,
primary key (account-number),
foreign key (branch-name) references branch,
check(balance >=0));
```

```
create table depositor (customer-name char(20)
not null, account-number char(10) not null,
primary key (customer-name, account-number),
foreign key (customer-name) references customer,
foreign key (account-number) references
account);
```

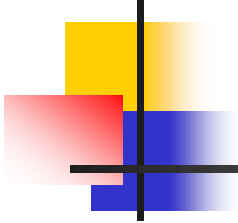


Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Καθορίζεται επίσης το πώς αντιμετωπίζεται μια παραβίαση περιορισμού:

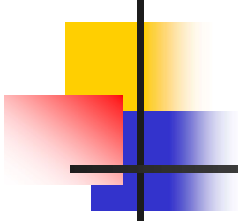
```
create table account (account-number char(10)
not null, branch-name char(15), balance integer,
primary key (account-number,
foreign key (branch-name) references branch,
on delete cascade,
on update cascade,
check(balance >=0));
```

- Διαγραφή μιας πλειάδας από τη σχέση branch ακολουθείται από τη διαγραφή των πλειάδων της σχέσης account που την αναφέρουν. Παρόμοια, στην περίπτωση της ενημέρωσης, ενημερώνονται τα αντίστοιχα γνωρίσματα της σχέσης account.



Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Άλλες ενέργειες για την αντιμετώπιση της παραβίασης περιορισμών περιλαμβάνουν την εισαγωγή κενής τιμής στο πεδίο το οποίο αναφέρεται σε πεδίο πλειάδας άλλης σχέσης η οποία διαγράφεται ή ενημερώνεται.
- Αν η παραβίαση ενός περιορισμού δεν μπορεί να διορθωθεί με διαγραφές / ενημερώσεις, τότε η πράξη που προκαλεί την παραβίαση διακόπτεται και αναιρούνται οι πράξεις που εκτελέστηκαν.



Περιορισμοί Ακεραιότητας (Integrity Constraints)

- Χειρισμός κενών τιμών:
 - Όλα τα γνωρίσματα του πρωτεύοντος κλειδιού θεωρούνται not null
 - Γνωρίσματα ξένων κλειδιών μπορούν να δέχονται κενές τιμές. Οι κενές τιμές δεν παραβιάζουν τους περιορισμούς αναφοράς.