

Μοντέλο Οντοτήτων – Σχέσεων (Entity-Relationship Model)

- Εξειδίκευση (Specialization)
- Μια οντότητα μπορεί να περιλαμβάνει υπο-ομάδες οντοτήτων οι οποίες διακρίνονται από άλλες οντότητες στην ίδια ομάδα καθώς χαρακτηρίζονται από γνωρίσματα τα οποία δεν χαρακτηρίζουν όλες τις οντότητες σε αυτό το σύνολο
- Η διαδικασία προσδιορισμού υπο-ομάδων μέσα σε σύνολα οντοτήτων ονομάζεται εξειδίκευση
- Η εξειδίκευση δημιουργεί ιεραρχίες εξειδίκευσης (specialization or IsA hierarchies) με χρήση της σχέσης «είναι (υπο-ομάδα)» (IsA)
- Μια σχέση IsA επίσης ορίζει μια σχέση υπερκλάσης – υποκλάσης (superclass – subclass)

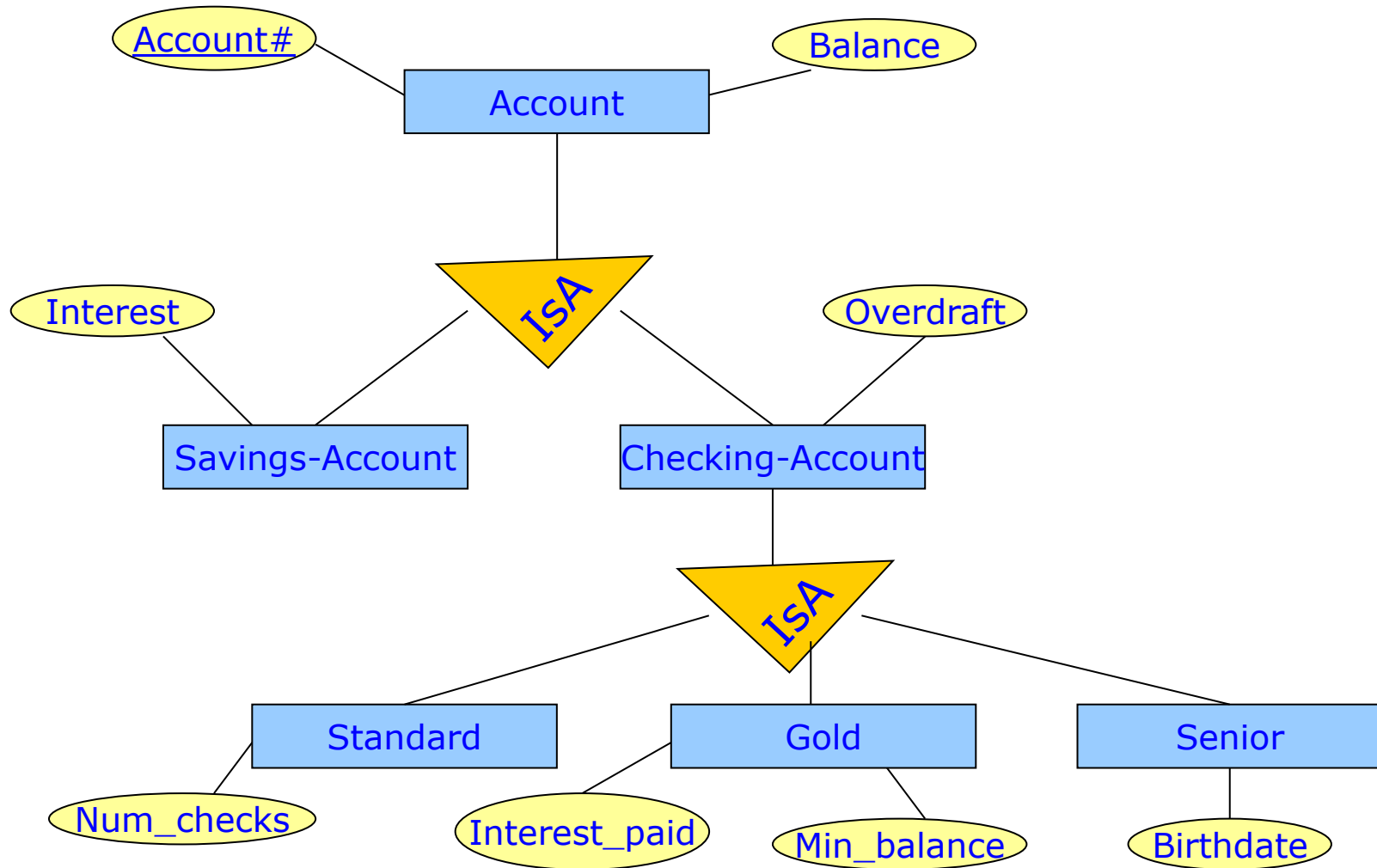
Μοντέλο Οντοτήτων – Σχέσεων (Entity-Relationship Model)

- Εξειδίκευση (Specialization)
- **Παράδειγμα:** Η οντότητα **account** με γνωρίσματα **account-number** και **balance** μπορεί να εξειδικευθεί σε:
 - **savings-account**
 - **checking-account**
- Κάθε είδος λογαριασμού περιγράφεται από ένα σύνολο γνωρισμάτων τα οποία περιλαμβάνουν όλα τα γνωρίσματα της οντότητας **account**. Επιπλέον, μπορεί να έχει ιδιαίτερα γνωρίσματα.
 - η οντότητα **savings-account** έχει το γνώρισμα **interest-rate**
 - η οντότητα **checking-account** έχει το γνώρισμα **overdraft-amount**

Μοντέλο Οντοτήτων – Σχέσεων (Entity-Relationship Model)

- Εξειδίκευση (Specialization)
- Μια οντότητα μπορεί να εξειδικεύεται σύμφωνα με περισσότερα από ένα γνωρίσματα
 - η οντότητα **account** μπορεί να εξειδικευθεί σε σχέση με τους κατόχους ενός λογαριασμού σε **commercial-account** και **personal-account**
- Όταν υπάρχουν περισσότερες από μια εξειδικεύσεις για μια οντότητα, ένα στιγμιότυπο μπορεί να ανήκει και στις δύο εξειδικεύσεις.
 - Π.χ. ένας λογαριασμός μπορεί να είναι **personal-account** και **savings-account** συγχρόνως
- Η εξειδίκευση μπορεί να εφαρμοστεί επαναληπτικά
 - Π.χ. η οντότητα **checking-account** μπορεί να εξειδικευθεί σε **standard, gold, senior**

Μοντέλο Οντοτήτων – Σχέσεων (Entity-Relationship Model)



Μοντέλο Οντοτήτων – Σχέσεων (Entity-Relationship Model)

- Γενίκευση
- Η εξειδίκευση οντοτήτων σε υπο-ομάδες αντιστοιχεί σε μια **top-down** διαδικασία σχεδιασμού ενός εννοιολογικού μοντέλου
- Ο σχεδιασμός μπορεί να γίνει και **bottom-up**. Σε αυτή την περίπτωση, οντότητες χρησιμοποιούνται για να **συνθέσουν** άλλες οντότητες σε υψηλότερα επίπεδα. Η σύνθεση γίνεται βάσει των κοινών γνωρισμάτων των οντοτήτων.
- Η διαδικασία αυτή λέγεται **γενίκευση** (generalization) και αναπαριστά μια σχέση υποσυνόλου μεταξύ των συντιθέμενων οντοτήτων και της νέας οντότητας που δημιουργείται.
- Η γενίκευση είναι η **δυϊκή** σχέση της εξειδίκευσης.

Μοντέλο Οντοτήτων – Σχέσεων (Entity-Relationship Model)

- Κληρονομικότητα Γνωρισμάτων (Attribute Inheritance)
- Όταν οντότητες οργανώνονται σε ιεραρχίες εξειδίκευσης / γενίκευσης, τα γνωρίσματα των οντοτήτων που βρίσκονται στα υψηλότερα επίπεδα **κληρονομούνται** από τις οντότητες που βρίσκονται σε χαμηλότερα επίπεδα.
 - Π.χ. οι οντότητες **savings-account** και **checking-account** κληρονομούν όλα τα γνωρίσματα της οντότητας **account**
- Επίσης κληρονομείται η συμμετοχή σε σχέσεις με τους ίδιους περιορισμούς.
- Οι σχέσεις γενίκευσης / εξειδίκευσης υπόκεινται σε περιορισμούς που αφορούν τα στιγμιότυπα των οντοτήτων που συμμετέχουν σε αυτές.

Μοντέλο Οντοτήτων – Σχέσεων (Entity-Relationship Model)

- Περιορισμοί
- Η σχέση μέλους ενός στιγμιότυπου μιας οντότητας μπορεί να είναι:
 - **υπό συνθήκη (condition-defined)**: ελέγχεται μία συνθήκη προκειμένου να προσδιοριστεί αν ένα στιγμιότυπο ανήκει σε μια οντότητα
 - Π.χ. Υποθέστε ότι η οντότητα **account** έχει ένα γνώρισμα **account-type**. Τα στιγμιότυπα τα οποία ικανοποιούν τη συνθήκη **account-type=savings-account** ανήκουν στην οντότητα **savings-account**, ενώ αυτά που ικανοποιούν τη συνθήκη **account-type=checking-account** ανήκουν στην οντότητα **checking-account**
 - **ορισμένη από το χρήστη (user-defined)**: στιγμιότυπα ορίζονται ως μέλη συνόλων οντοτήτων από το χρήστη

Μοντέλο Οντοτήτων – Σχέσεων (Entity-Relationship Model)

- Περιορισμοί
- **Αποκλειστικότητα (Disjointness)**: ένα στιγμιότυπο δεν μπορεί να ανήκει σε περισσότερες από μία οντότητες στο ίδιο επίπεδο μιας ιεραρχίας IsA
 - π.χ. ένας λογαριασμός θα είναι **savings-account** ή **checking-account** αλλά όχι και τα δύο
- **Επικάλυψη (overlapping)**: το ίδιο στιγμιότυπο μπορεί να ανήκει σε περισσότερες από μία οντότητες σε μια ιεραρχία
- **Πληρότητα (completeness)**: καθορίζει αν ένα στιγμιότυπο μιας οντότητας **πρέπει** να ανήκει σε τουλάχιστο μία οντότητα σε χαμηλότερο επίπεδο

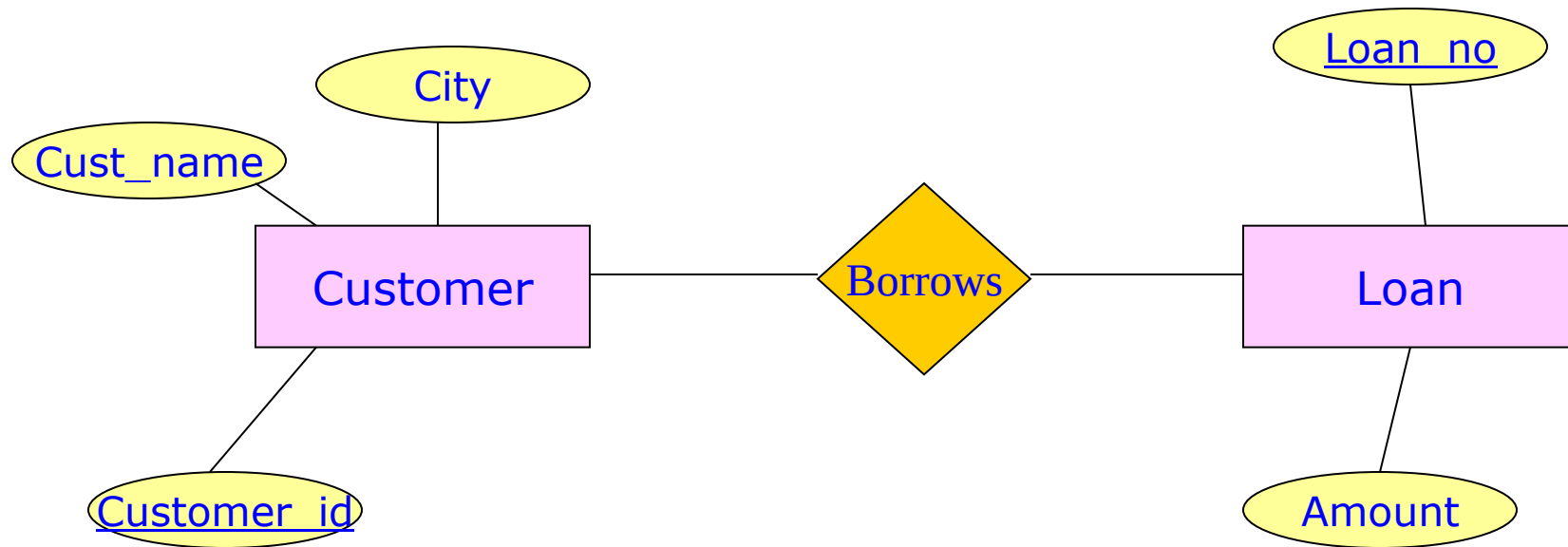
Μοντέλο Οντοτήτων – Σχέσεων (Entity-Relationship Model)

- Περιορισμοί
- Περιορισμοί πληρότητας μπορεί να είναι:
 - **ολικοί (total)** : κάθε στιγμιότυπο πρέπει να ανήκει σε μια οντότητα σε χαμηλότερο επίπεδο
 - **μερικοί (partial)**: κάποια στιγμιότυπα μπορούν να μην ανήκουν σε κάποια από τις οντότητες σε χαμηλότερα επίπεδα
- Αν ισχύει ο ολικός περιορισμός πληρότητας, τότε όποτε εισάγεται ένα στιγμιότυπο ως μέλος μιας οντότητας, πρέπει να εισαχθεί και ως μέλος μιας οντότητας σε χαμηλότερο επίπεδο
- Η διαγραφή ενός στιγμιότυπου από μια οντότητα πρέπει να συνοδεύεται από τη διαγραφή του από όλες τις οντότητες χαμηλότερου επιπέδου στις οποίες ανήκει.

Μοντέλο Οντοτήτων – Σχέσεων (Entity-Relationship Model)

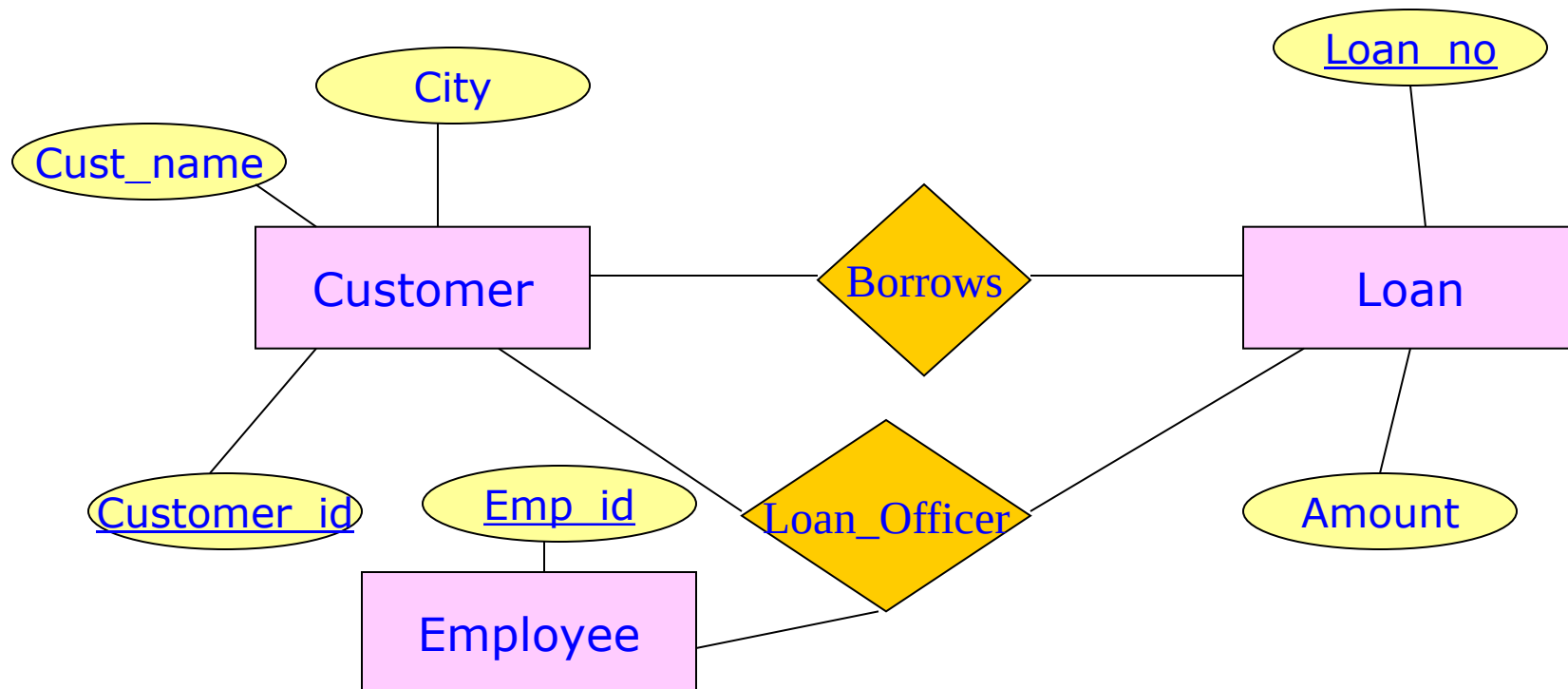
Συνάθροιση (Aggregation)

- 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 – Σχέσεων (Entity-Relationship Model)

- 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.



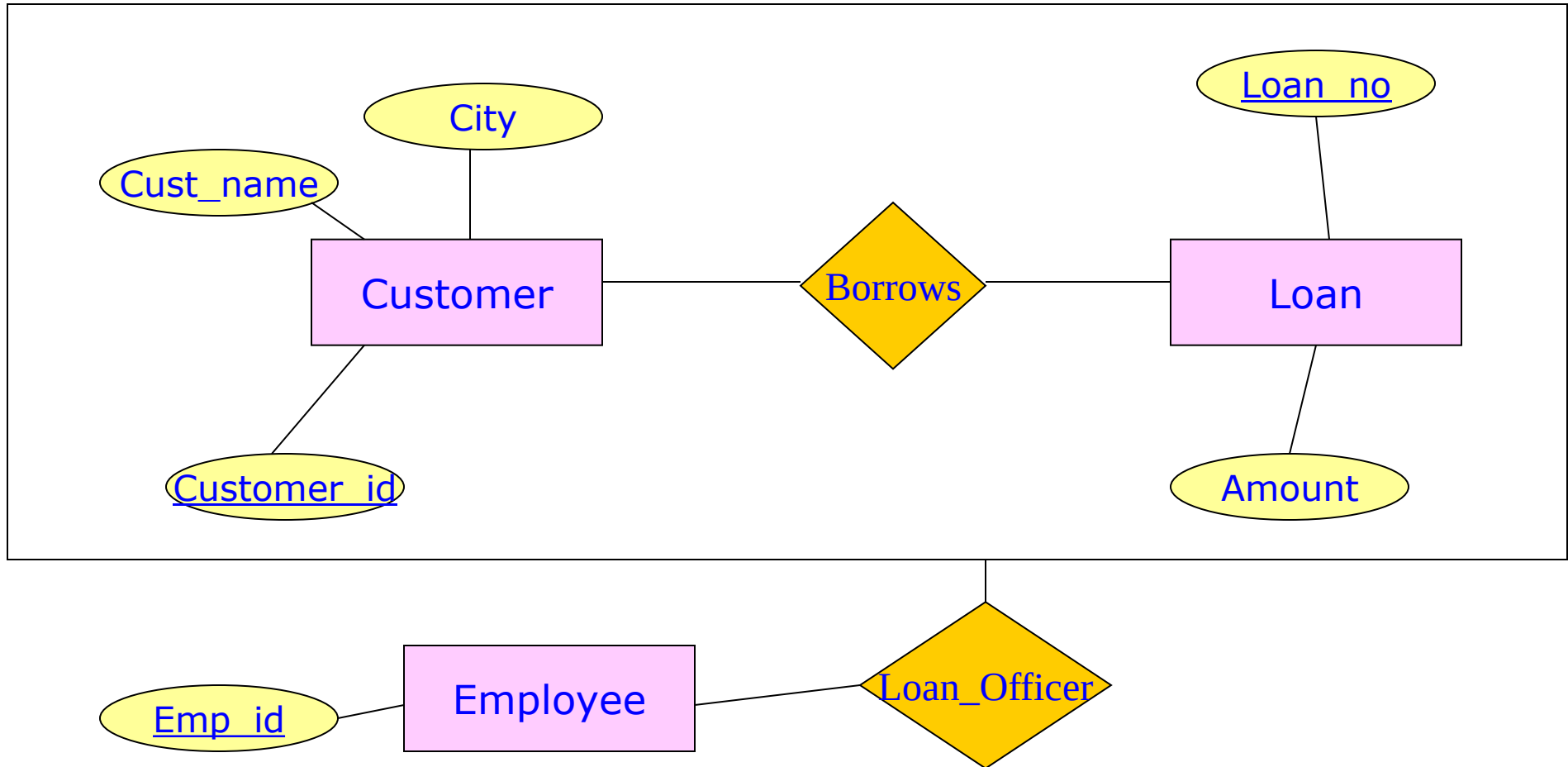
Μοντέλο Οντοτήτων – Σχέσεων (Entity-Relationship Model)

- 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 – Σχέσεων (Entity-Relationship Model)

- 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 – Σχέσεων (Entity-Relationship Model)



Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

- 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, βασισμένη στη σχεσιακή άλγεβρα

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

- **Παράδειγμα:** CAP database
 - ΒΔ που χρησιμοποιείται από εταιρία χονδρικής πώλησης προϊόντων ώστε να κρατά αρχείο των **πελατών (customers)**, των **προϊόντων (products)** που πουλάει και των **πρακτόρων (agents)** που κάνουν **παραγγελίες (orders)** προϊόντων εκ μέρους των πελατών.
 - Οι πελάτες είναι εταιρίες λιανικής που παραγγέλνουν μεγάλες ποσότητες προϊόντων από την εταιρία χονδρικής.
 - Οι πελάτες, τα προϊόντα, οι πράκτορες και οι παραγγελίες χαρακτηρίζονται από μοναδικά αναγνωριστικά (identifiers).

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

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

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

- Το σύνολο των σχημάτων όλων των σχέσεων ονομάζεται **σχήμα της βάσης δεδομένων (database schema)**.
- Το σύνολο όλων των πλειάδων των σχέσεων λέγεται **περιεχόμενο** της βάσης δεδομένων (**database content, extension**). Για μια σχέση **R**, το περιεχόμενο συμβολίζεται ως **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.
- Το πεδίο τιμών ενός γνωρίσματος **A** συμβολίζεται ως **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 (Relational Data Model)

- Παράδειγμα: CAP Database

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

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

- 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.
- Κανόνας 1 (Πρώτη Κανονική Μορφή)
 - Μια σχέση είναι σε **Πρώτη Κανονική Μορφή (1NF)** αν δεν έχει πλειότιμα γνωρίσματα

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

- Κανόνας 1 (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	Superintendent	David M. Jr.
e003	Jones, Franklin	Agent	Andrew K. Mark W. Lois M.

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

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

eid	ename	position	dep1	dep2	dep3
e001	Smith, John	Agent	Michael J.	Susan R.	
e002	Andrews, David	Superintendent	David M. Jr.		
e003	Jones, Franklin	Agent	Andrew K.	Mark W.	Lois M.

Μη αποδοτική λύση: σπατάλη χώρου, δυσκολία στις ερωτήσεις

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

- Κανόνας 1 (Πρώτη Κανονική Μορφή)
 - 2η επιλογή: να δημιουργηθεί μια νέα σχέση για τους υπαλλήλους και τα εξαρτούμενα 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 η ελάχιστη δυνατή.

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

- Υπάρχουν συστήματα (extended RDBMS) τα οποία επιτρέπουν τον ορισμό σχέσεων που δεν είναι σε 1NF.
- **Κανόνας 2: 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 3η πλειάδα της σχέσης?» είναι μη-έγκυρες
 - οι ερωτήσεις πρέπει να βασίζονται στις τιμές των γνωρισμάτων
 - πολλά εμπορικά συστήματα παραβιάζουν τον κανόνα αυτό και παρέχουν ένα εσωτερικό αναγνωριστικό πλειάδων (tuple id). Πρόσβαση βάσει του tuple id ενδέχεται να έχει πλεονεκτήματα από την άποψη της απόδοσης του συστήματος.

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

- Κανόνας 3: 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 (Relational Data Model)

- Κλειδιά και υπερ-κλειδιά (keys and superkeys)
 - ο 3ος 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 διακρίνομε οποιεσδήποτε δύο πλειάδες μιας σχέσης
 - 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 (Relational Data Model)

- Κλειδιά και υπερ-κλειδιά (keys and superkeys)

- Η επιλογή των κλειδιών και υπερ-κλειδιών είναι θέμα της σχεδίασης μιας βάσης δεδομένων 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	Boston	203000	0.50
p03	razor	Paris	150500	1.00
p04	pen	Paris	125000	1.00
p05	pencil	Dallas	221400	1.00

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

- 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 $\{A_{i_1}, A_{i_2}, \dots, A_{i_k}\} \subseteq \{A_1, A_2, \dots, A_n\}$, τότε ο περιορισμός (restriction) $t[A_{i_1}, A_{i_2}, \dots, A_{i_k}]$ της t στα γνωρίσματα $\{A_{i_1}, A_{i_2}, \dots, A_{i_k}\}$ είναι η πλειάδα τιμών της t στα γνωρίσματα $A_{i_1}, A_{i_2}, \dots, A_{i_k}$
- Παράδειγμα: Αν $t=(p01, \text{comb}, \text{Dallas}, 111400, 0.50)$ τότε
 $t[\text{pid}, \text{city}, \text{price}]=(p01, \text{Dallas}, 0.50)$
- Ορισμός Ένα κλειδί μιας σχέσης R με σχήμα $\text{head}(R) = \{A_1, A_2, \dots, A_n\}$ είναι ένα σύνολο γνωρισμάτων $K = \{A_{i_1}, A_{i_2}, \dots, A_{i_k}\}$ για το οποίο:
 - αν u, v είναι διακεκριμένες πλειάδες της R , τότε $u[K] \neq v[K]$
 - δεν υπάρχει $H \subset K$ με την παραπάνω ιδιότητα

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

- Ένα σύνολο γνωρισμάτων το οποίο πληροί 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
- **Απόδειξη:** Έστω ότι $\text{head}(R) = \{A_1, A_2, \dots, A_n\}$ και έστω S_1 το σύνολο αυτό. Από τον σχεσιακό κανόνα 3, το S_1 είναι υπερ-κλειδί. Άρα για κάποιες πλειάδες u, v με $u \neq v$, πρέπει να υπάρχει γνώρισμα $A_i \in S_1$ τέτοιο ώστε $u[A_i] \neq v[A_i]$

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

- Απόδειξη (συνέχεια):

- Περίπτωση 1: S_1 είναι κλειδί
- Περίπτωση 2: Υπάρχει $S_2 \subset S_1$ τέτοιο ώστε S_2 είναι υπερκλειδί. Αν το S_2 δεν είναι κλειδί, τότε πρέπει να υπάρχει $S_3 \subset S_2$ τέτοιο ώστε S_3 είναι υπερκλειδί. Συνεχίζοντας με αυτό τον τρόπο συμπεραίνουμε ότι πρέπει να υπάρχει ένα σύνολο S_k το οποίο είναι υποσύνολο όλων των προηγούμενων συνόλων και το οποίο δεν έχει γνήσιο υποσύνολο το οποίο να είναι υπερκλειδί.

Αν $\#S_i$ δηλώνει τον αριθμό των γνωρισμάτων στο σύνολο S_i , τότε $\#S_1=n$, $\#S_i > \#S_{i+1}$ και $\#S_k>0$.

Άρα, το S_k είναι κλειδί.

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

- 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	Boston	203000	0.50
p03	razor	Paris	150500	1.00
p04	pen	Paris	125000	1.00
p05	pencil	Dallas	221400	1.00

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

- 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.
- **Σημασιολογία κενών τιμών:**
 - υπαρκτή αλλά άγνωστη τιμή, ή
 - η εισαγωγή μιας τιμής δεν έχει νόημα για κάποια πλειάδα

Σχεσιακό Μοντέλο Δεδομένων (Relational Data Model)

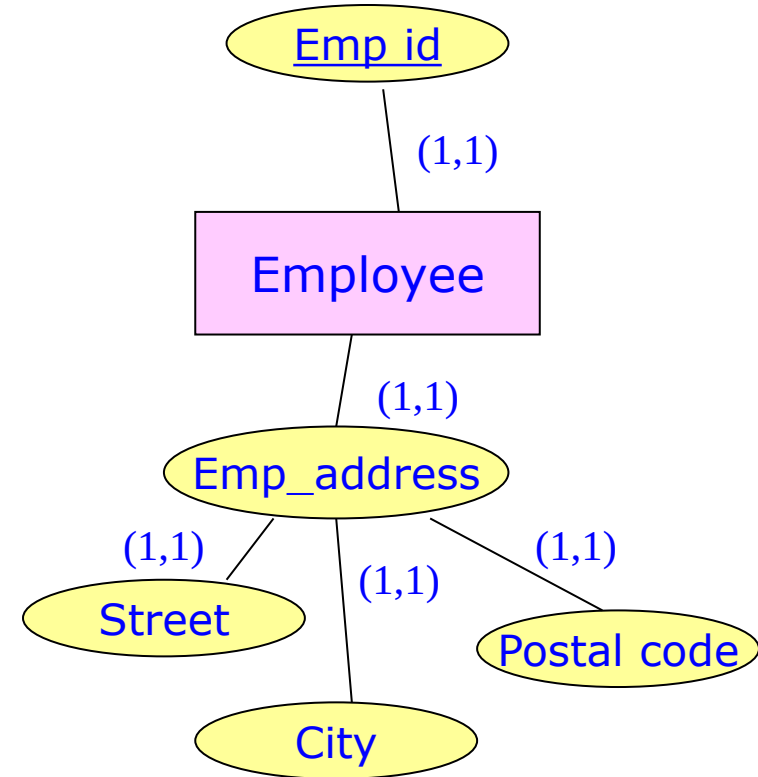
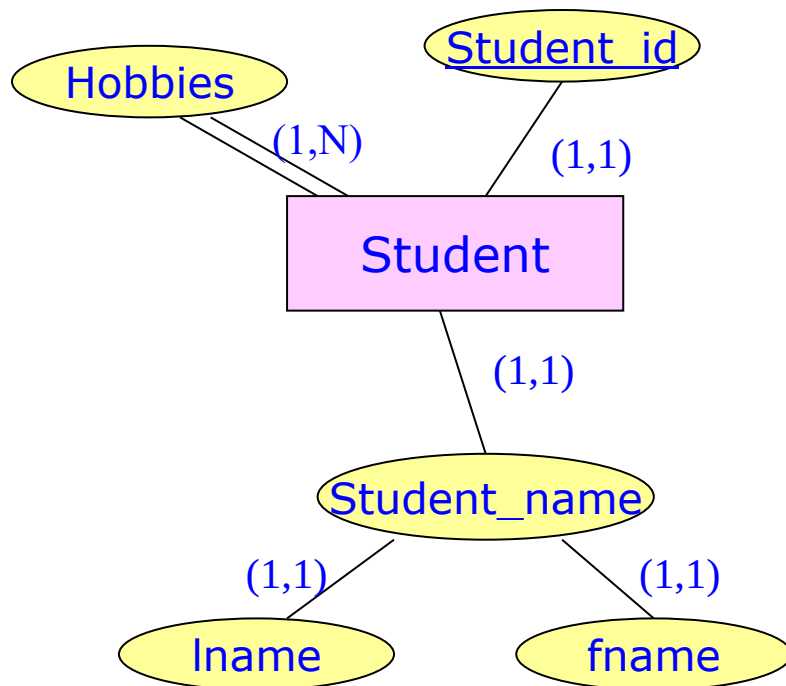
- Κενές Τιμές
- Οι κενές τιμές γνωρισμάτων σχέσεων είναι διαφορετικές από τις αντίστοιχες «κενές τιμές» των τύπων δεδομένων (π.χ. null string)
- **Προσοχή:** ορισμένα γνωρίσματα δεν επιτρέπεται να δέχονται κενές τιμές
- **Κανόνας 4: Ακεραιότητα Οντοτήτων (Entity Integrity)**
 - γνωρίσματα τα οποία ανήκουν στο πρωτεύον κλειδί μιας σχέσης R δε 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 (απλά και σύνθετα) γίνονται γνωρίσματα της αντίστοιχης σχέσης.
 - Τα αναγνωριστικά γνωρίσματα της οντότητας σχηματίζουν ένα υποψήφιο κλειδί της σχέσης
 - Τα στιγμιότυπα της οντότητας γίνονται πλειάδες της σχέσης

Μοντέλο Οντοτήτων – Σχέσεων → Σχεσιακό Μοντέλο

• Παράδειγμα:



Student

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

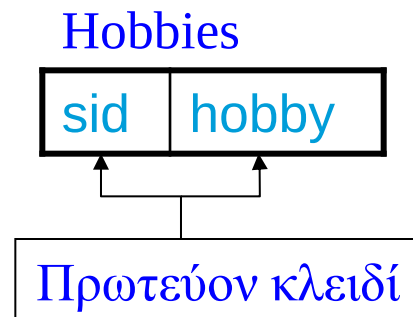
Υποψήφια
Κλειδιά

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:



Μοντέλο Οντοτήτων – Σχέσεων → Σχεσιακό Μοντέλο

3. Μια N-N 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 .

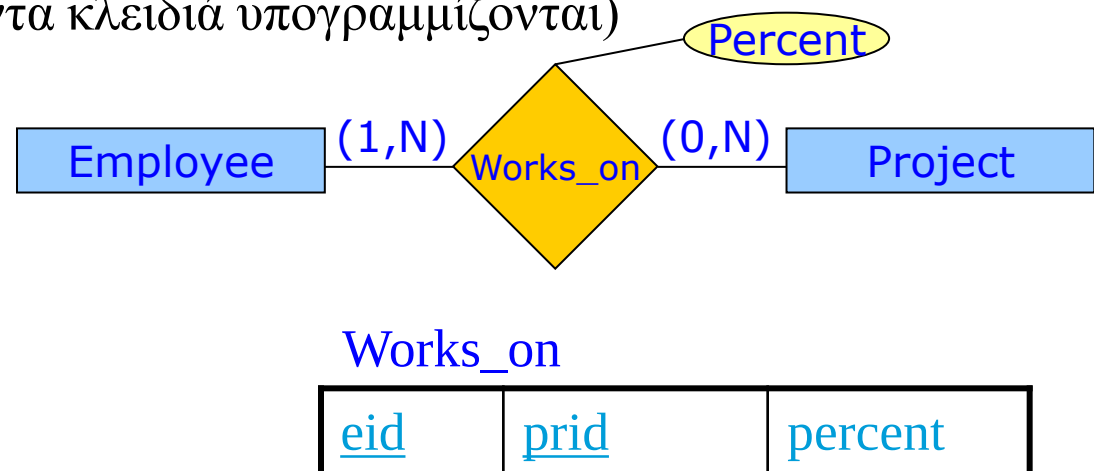
Παράδειγμα: (τα πρωτεύοντα κλειδιά υπογραμμίζονται)

Employee

<u>eid</u>	street	city	postal_code
------------	--------	------	-------------

Projects

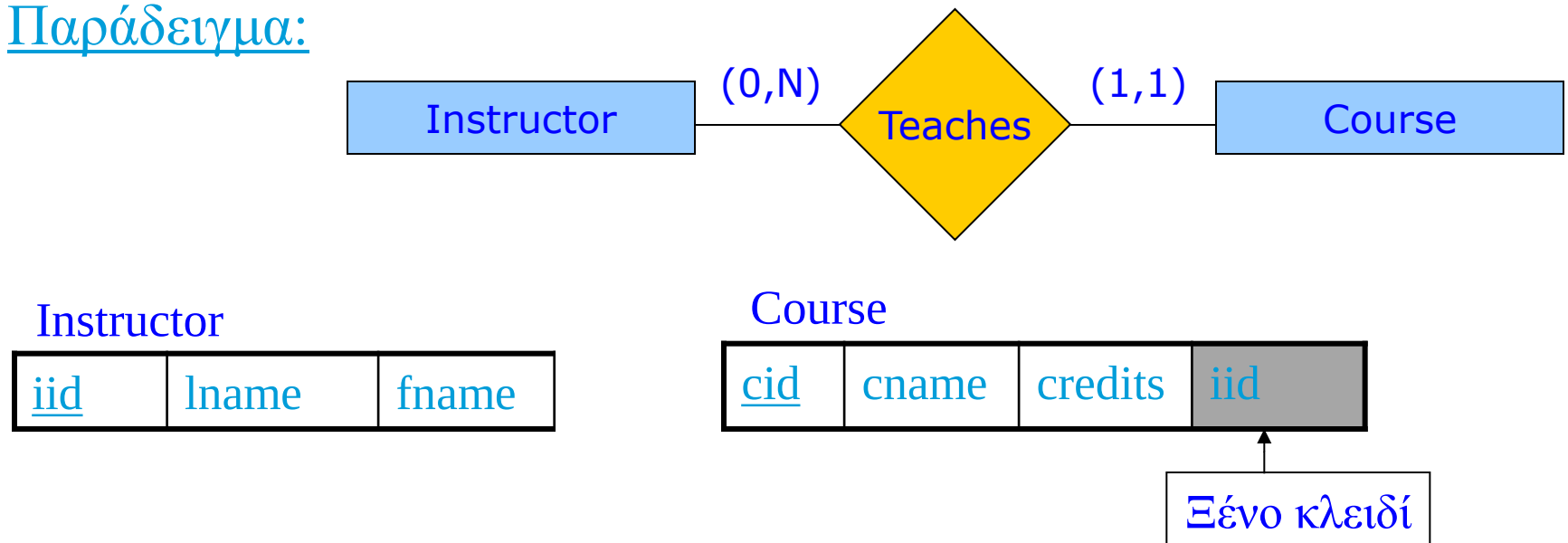
<u>prid</u>	pr_name	due_date
-------------	---------	----------



Μοντέλο Οντοτήτων – Σχέσεων → Σχεσιακό Μοντέλο

4. Για μια N-1 σχέση r μεταξύ οντοτήτων E και F δε 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 δε δέχεται κενές τιμές.

Παράδειγμα:



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 προσθέτομε 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.

Παράδειγμα:



Employee

<u>eid</u>	street	city	postal_code	prid
------------	--------	------	-------------	------

Projects

<u>prid</u>	pr_name	due_date
-------------	---------	----------

ή

Employee

<u>eid</u>	street	city	postal_code
------------	--------	------	-------------

Projects

<u>prid</u>	pr_name	due_date	supid
-------------	---------	----------	-------

Μοντέλο Οντοτήτων – Σχέσεων → Σχεσιακό Μοντέλο

6. Στην περίπτωση σχέσεων εξειδίκευσης (IsA), υπάρχουν δύο περιπτώσεις:
- Αν κάθε στιγμιότυπο της πατρικής οντότητας είναι **υποχρεωτικά** και στιγμιότυπο **μιας μόνο** από τις εξειδικευμένες οντότητες, τότε δημιουργούμε μία σχέση μόνο για κάθε εξειδικευμένη οντότητα και όχι για την πατρική. Οι σχέσεις αυτές περιέχουν όλα τα γνωρίσματα της πατρικής.

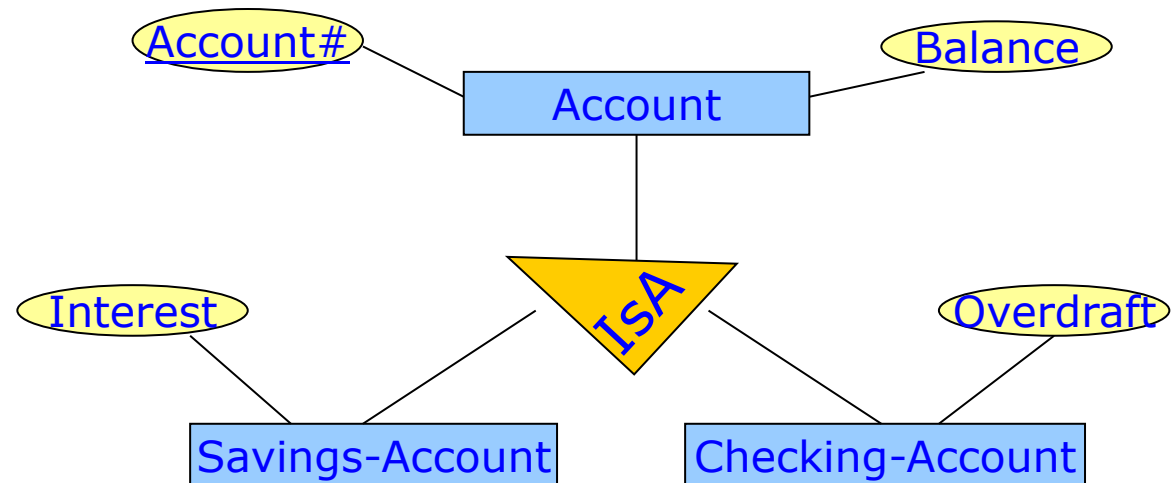
Παράδειγμα:

Savings-Account

<u>acc_no</u>	balance	interest
---------------	---------	----------

Checking-Account

<u>acc_no</u>	balance	overdraft
---------------	---------	-----------



Μοντέλο Οντοτήτων – Σχέσεων → Σχεσιακό Μοντέλο

- Στην αντίθετη περίπτωση, όταν δηλαδή υπάρχουν στιγμιότυπα της πατρικής οντότητας που μπορεί να μην αντιστοιχούν σε στιγμιότυπο κάποιας από τις εξειδικευμένες οντότητες (ή αντιστοιχούν σε περισσότερες από μία) δημιουργούμε μια σχέση για την πατρική οντότητα, και μια για κάθε εξειδικευμένη, προσθέτοντας σ' αυτές το πρωτεύον κλειδί της πατρικής.

Παράδειγμα:

Person

<u>ssn</u>	name
------------	------

Student

<u>ssn</u>	sid
------------	-----

Instructor

<u>ssn</u>	dept
------------	------

