

Prof. Dr. Ermir Rogova

# BAZAT E TË DHËNAVE

Pjesa 4

---

# ER DHE UML NË CIKLIN E JETËS

- ✘ Në këtë pjesë do të shikojmë se si diagramet ER dhe UML aplikohen në ciklin e jetës së databazës
- ✘ Kjo përfshin hapat prej analizës së kerkesave deri te integrimi i pamjeve
- ✘ Qasja në dizajnimin logjik që do të shtjellohet formon nje kornizë punuese për transformimin e konstrukteve konceptuale në tabela.

# ANALIZA E KËRKESAVE

---

- ✘ Është hapi i parë në ciklin e jetës së databazës dhe zakonisht kërkon punë intensive
- ✘ Dizajneri duhet të intervistojë përdoruesët dhe të përcaktojë saktësisht se:
  - + për çka do të përdoret databaza dhe
  - + çka do të përmbajë ajo

# OBJEKTIVAT E ANALIZËS SË KËRKESAVE

- ✘ Të paraqesë kërkesat për të dhëna në atë ambient në formë të elementeve themelore të të dhënave (data elements)
- ✘ Të përshkruaj informacionin mbi elementët e të dhënave dhe marrëdhëniet në mes tyre
- ✘ Të përcaktojë llojet e transaksioneve që planifikohet të ekzekutohen në databazë dhe interaksionin në mes të tyre dhe elementeve të të dhënave

# OBJEKTIVAT E ANALIZËS SË KËRKESAVE (2)

- ✘ Të definojë kufizimet që duhen të imponohen në databazën rezultuese si:
  - + Integritetit
  - + Përformancës
  - + Sigurisë
  - + Administrativë
- ✘ Të specifikojë kufizimet dizajnuese apo implementuese si:
  - + Teknologjitë specifike
  - + Hardverin dhe softverin
  - + Gjuhët programuese
  - + Politikat
  - + Standardet
  - + Interfejsët e jashtëm

# OBJEKTIVAT E ANALIZËS SË KËRKESAVE (3)

- ✘ Të dokumentojë rrënjësisht të gjitha pikat e lartëpërmendura në dokument të detalizuar të kërkesave
- ✘ Elementet e të dhënave (data elements) duhet të definohen në një fjalor të të dhënave i cili zakonisht jepet si pjesë integrale e sistemit të menagjimit të databazës

# MODELIMI KONCEPTUAL

---

- ✘ Modelimi konceptual realizohet duke kryer hapat e mëposhtëm:
  - + Klasifikimi i entiteteve dhe attributeve (klasët dhe atributet -> UML)
  - + Identifikimi i hierarkive gjeneralizuese
  - + Definimi i marrëdhënieve (asocijimeve ->UML)

# KLASIFIKIMI I ENTITETEVE DHE ATRIBUTEVE

- ✘ Si vendoset se a është një data element entitet, atribut apo marrëdhënie?
  - + P.sh. Vendndodhja e projekteve është nëpër qytete. A është qyteti entitet apo atribut?
- ✘ Udhëzimet për klasifikimin e entiteteve dhe attributeve janë:
  - + Entitetet duhet të përmbajnë informata përshkruese
  - + Atributet shumëvlerëshe duhet të klasifikohen si entitete
  - + Atributet duhen të bashkangjiten entiteteve të cilat i përshkruajnë në mënyrën më direkte



# PËRMBAJTJA E ENTITETEVE

- ✘ Nëse ekziston informacion përshkrues për data elementin, atëherë ai duhet të klasifikohet si entitet
- ✘ Nëse elementit i nevojitet vetem një identifikues dhe nuk ka marrëdhënie, atëhere duhet të klasifikohet si atribut.
- ✘ Qyteti:
  - + Entitet, (popullata, sipërfaqja, etj)
  - + Atribut (identifikues i vendndodhjes se projektit)
  - + Si entitet nëse identiteti duhet të kufizohet

# ATRIBUTET SHUMËVLERËSHE

- ✘ Kur një atribut mund të ketë shumë vlera, p.sh. Nr. Tel apo adresat e e-mailit të studentit
- ✘ Këto attribute duhet të riklasifikohen si entitete dhe të lidhen me entiteti prej të cilit rrjedhin
- ✘ Me këtë arrihet thjeshtësimi i dizajnit dhe implementimit

# BASHKANGJITJA E ATRIBUTEVE

- ✘ Atributet ti bashkangjiten entiteteve te cilat i përshkruajnë sa me direkt
  - + P.sh. Salla\_e\_mesimit duhet ti bashkangjitet entitetit Objekti në vend të entitetit Studenti
- ✘ Procedura e identifikimit te entiteteve dhe attributeve është iterative (përsëritëse)
- ✘ Gjatë modelimit shpesh elementet ndyshojnë vendet prej attribute në entitete dhe anasjelltas, varësisht se a i shkelin apo jo udhëzimet e dhëna

# IDENTIFIKIMI I GJENERALIZIMEVE

- ✘ P.sh. Nëse identifikohen 5 entitete si:
  - + Employee, with identifier empno and descriptors
  - + empname, address, and date-of-birth.
  - + Manager, with identifier empno and descriptors
  - + empname and jobtitle.
  - + Engineer, with identifier empno and descriptors
  - + empname, highest-degree, and jobtitle.
  - + Technician, with identifier empno, and descriptors
  - + empname and specialty.
  - + Secretary, with identifier empno, and descriptors
  - + empname and best-skill.

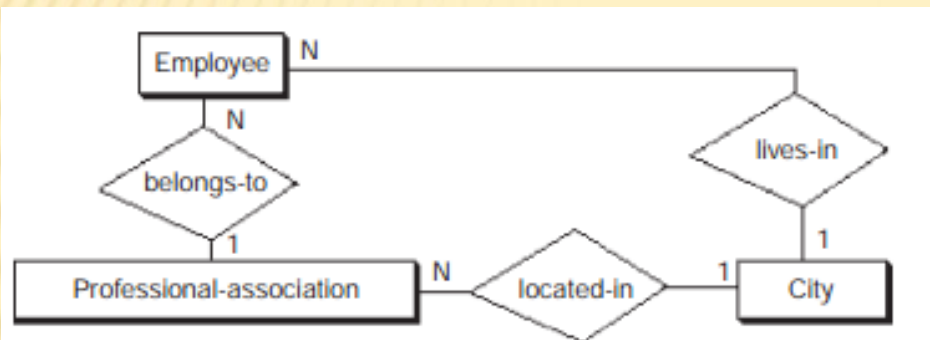
# IDENTIFIKIMI I GJENERALIZIMEVE (2)

- ✘ Me analizim përcaktojmë se entiteti Employee mund të krijohet si gjeneralizim për Manager, Engineer, Technician dhe Secretary
- ✘ Vendosim empono dhe përshkruesit e përgjithshëm si, empname, address, DOB në supertip, kurse empno dhe përshkruesit specifik në secilin nëntip (subtip)
- ✘ Nëse më vonë vendosim të eliminojmë entitetin employee, atëherë identifikuesi dhe përshkruesit e përgjithshëm rishpërndahen

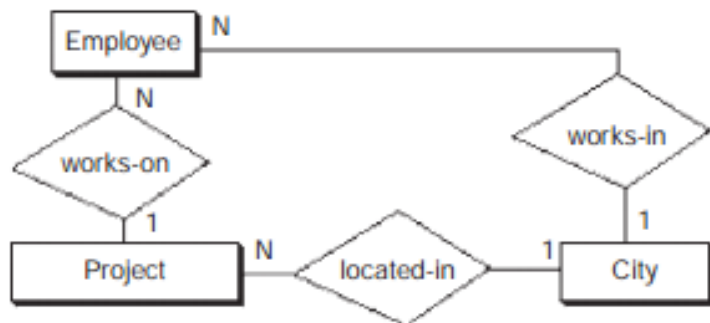
# DEFINIMI I MARRËDHËNIEVE

- ✘ Marrëdhëniet janë data elementet që paraqesin lidhje (asocijim) në mes entiteteve
- ✘ Gramatikisht identifikohen me folje (punon-ne, menagjon, blen, ligjëron, ndëgjon, etj)
- ✘ Për secilën marrëdhënie duhet të specifikohen:
  - + Shkalla (binare, trinare, etj)
  - + Lidhja (1-1, 1-M, N-M)
  - + Ekzistenca (opcionale apo e detyrueshme)
  - + Çfardo atributi që është asociuar me marrëdhënien e jo entitetet

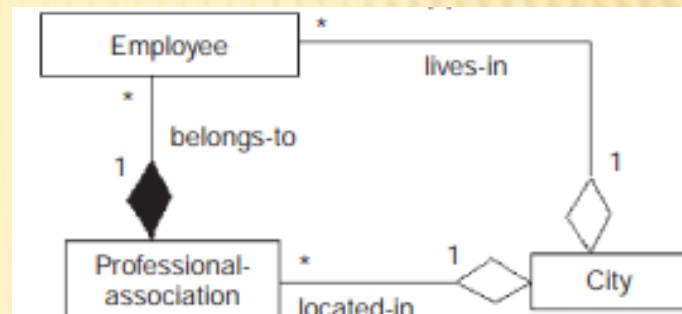
# MARRËDHËNIET E TEPËRTA (REDUNDANTE)



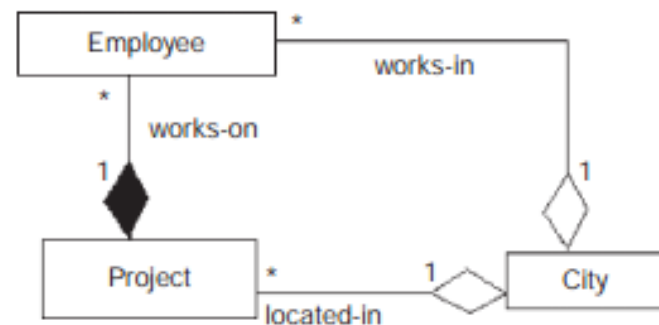
(a)



(b)

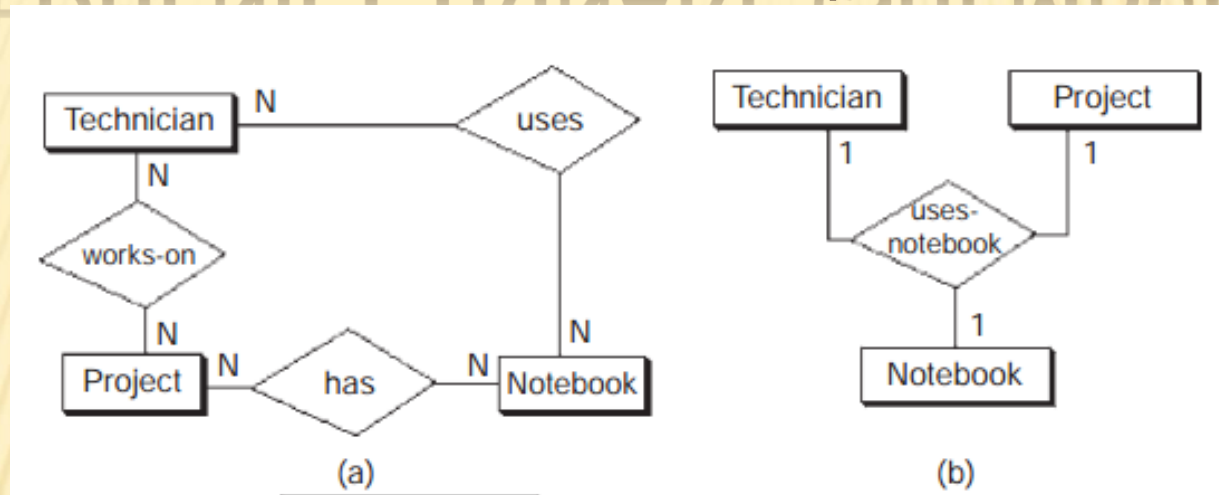


(c)



(d)

# MARRËDHËNIET TRINARE (SHEMBULL)



- ✘ Nëse secili teknik mund të punojë në secilin prej disa projekteve duke përdorur të njëjtën fletore shënimesh në secilin projekt, atëherë mund të definohen tri m-m marrëdhënie.
- ✘ Nëse secili teknik është i kufizuar të përdorë saktësisht 1 fletore për secilin projekt dhe ajo fletore i takon vetem një tekniku, atëherë duhet të definohet një marrëdhëni trinare 1-1-1



---

**PYETJE ???**