

Adatbázisrendszerek. Adatbázis, adatbázisrendszer, adatbázis-kezelő rendszer (DBMS) fogalma és jellemzői. Egyed, tulajdonság és kapcsolat fogalma és tulajdonságai. Relációs, objektum-relációs és NoSQL adatbázisok jellemzése. A funkcionális függés fogalma. Koncepcionális adatbázis-tervezés, az ER modell és leképezése relációs modellre. Az SQL elemei: DDL, DML, DCL, egyszerű lekérdezések és táblák összekapcsolása.

Adatbázis:

- Egymással logikailag összefüggő, egymáshoz kapcsolódó belső jelentéssel bíró adatok összessége, kollektívja, együttese. Speciális célra tervezett, felépített és közzétett adatok együttese.
- Az adatmodell, valamint az egyed-előfordulások, tulajdonság-előfordulások és kapcsolat-előfordulások együttese.

Adatbázisrendszer:

- Adatbázisrendszernek nevezzük az Adatbázis-kezelő rendszert és a benne szereplő adatokat együttesen.

Adatbázis-kezelő rendszer (DBMS):

- Olyan szoftvercsomag/rendszer, amely számítógépes adatbázisok létrehozását és karbantartását támogatja.
- Feladatai:
  - Adatbázisok létrehozása
  - Adatbázisok tartalmának definiálása
  - Adatok tárolása
  - Adatok lekérdezése
  - Adatok védelme
  - Adatok titkosítása
  - Hozzáférési jogok kezelése
  - Fizikai adatszerkezet szervezése

Az adatbázis megközelítés fő jellemzői:

- Az adatbázisrendszer önleíró természete:
  - Egy DBMS katalógus egy önálló adatbázis leírását tárolja (adatszerkezetek, adattípusok, megszorítások).
  - A leírás metaadatokból áll.
  - Ez teszi lehetővé, hogy a DBMS szoftver különböző adatbázis alkalmazásokkal működjön együtt.
- A programok és az adatok elszigetelése
  - Ezt program-adat függetlenségnek nevezzük.
  - Lehetővé teszi az adatszerkezetek és a tárolás módjának megváltoztatását anélkül, hogy a DBMS-t eltérő programot meg kellene változtatni.

- Adat absztrakció:
  - Egy adatmodellt használunk arra, hogy a tárolási részleteket elrejtjük és csak az adatbázis koncepcionális képét jelenítjük meg a felhasználóknak.
  - A programok az adatmodellre hivatkoznak az adattárolási részletekkel szemben.
- Az adatok többféle nézetének támogatása:
  - Minden felhasználó különböző képet láthat az adatbázisról, ami csak a számára érdekes adatokat írja le.
- Adatok megosztása és többfelhasználós tranzakció feldolgozás:
  - Megengedi konkurens felhasználóknak az adatkinyerést és frissítést ugyanazon adatbázisban.
  - A konkurencia ellenőrzés a DBMS-en belül garantálja, hogy minden tranzakció helyesen hajtodik végre vagy szakad meg.
  - A helyreállító alrendszerek biztosítják, hogy minden végrehajtott tranzakció állandó bejegyzésre kerül az adatbázisban.
  - A közvetlen tranzakció feldolgozás (OLTP – Online Transaction Processing) az adatbázis alkalmazások fő része. Ez teszi lehetővé a konkurens tranzakciók százainak másodpercek alatti végrehajtását.

Egyed:

- Egyed: A valós világnak az az eleme (tárgy, jelenség, elképzelés, személy, fogalom), amely a modellezés tárgyát képi.
- Egyed típus: Az azonos tulajdonságokkal rendelkező egyedek absztrakciója.

Tulajdonság:

- Tulajdonság: Az egyednek a modellezés szempontjából lényeges jellemzője.
- Tulajdonságtípus: Az azonos szerepű tulajdonságok absztrakciója.
- Osztályozása:
  - A tulajdonság-előfordulás szerkezete (összetettsége szerint)
    - egyszerű (atomi)
    - összetett
  - A tulajdonság-előfordulás hány értéket vehet föl egyszerre
    - egyértékű
    - halmazértékű (többértékű)
  - A tulajdonság-előfordulás minden esetben megjelenik-e a háttértáron (a fizikai adatbázisban)
    - tárolt
    - származtatott
  - A null érték mint tulajdonság-előfordulás
    - nem alkalmazható, nem értelmezett
    - ismeretlen
      - létezik, de hiányzik
      - nem tudjuk, hogy létezik-e

#### Kapcsolat:

- Kapcsolattípus: Két vagy több egyedtípus közötti jól meghatározott viszony.
- Kapcsolat: Két vagy több egyedtípus egyedi közötti jól meghatározott viszony.
- Osztályzása:
  - A kapcsolat foka (hány egyedtípus vesz részt a kapcsolatban):
    - bináris (másodfokú)
    - ternális (harmadfokú)
    - ...
  - A (bináris) kapcsolat számossága (legfeljebb hány kapcsolat-előfordulásban vehet részt egy egyed-előfordulás):
    - 1:1
    - 1:N
    - M:N
  - A (bináris) kapcsolat szorossága (a kapcsolatban részt vevő egyedtípusok minden egyedének részt kell-e vennie legalább egy kapcsolat-előfordulásban):
    - kötelező
    - félig kötelező
    - opcionális

#### Relációs adatbázis:

- Részei:
  - Reláció: Értékek egy táblázata, amely sorok egy halmazából áll.
  - Sor: Minden egyes sor adatelemei a modellezett kisvilág egy egyed-előfordulásáról vagy egy kapcsolat előfordulásról tartalmaznak tényeket (információt).
  - Oszlopok: Minden egyes oszlop egy oszlop fejléccel (címmel) rendelkezik, amely az illető oszlopban lévő adatok jelentéséről ad információt.
  - Reláció kulcsa: Minden sor rendelkezik egy olyan adatelem értékkel (vagy azok egy halmazával), amely egyértelműen azonosítja a sort a táblázatban. Ezt kulcsnak nevezzük.

#### Objektum-relációs adatbázis:

- Relációs adatbázis kibővítése. Belül minden relációsan működik. Erre egy ráépülő réteggént alakítják ki az objektum-orientált felületet.
- Főbb bővítések:
  - Kollektívtípusok: struktúra, halmaz, multihalmaz, lista, struktúrák halmaza
  - Metódusok: műveletek a definiált objektumtípusokhoz
  - Hivatkozások: a kapcsolatok kialakítására
- Objektumok:
  - Sorobjektum: A tábla egy sorát tekintjük objektumnak
  - Oszlopobjektum: a relációséma egy attribútumát tekintjük objektumnak
  - Az objektumazonosító tartalmilag azonos sorokat is megkülönböztet
- Táblák:
  - Relációs tábla: a relációs modellből
  - Objektumtábla: sorobjektumok halmaza

## NoSQL adatbázis:

- A relációs (SQL) adatbázisoktól eltérően: gyorsan változó, struktúratlan adatok nagy tömegének kezelésére szolgál.
- Típusai:
  - Kulcs-érték: A kulcs-érték típus kivonattáblában tárolja a kulcs-érték párokat. A kulcs-érték típus akkor a leghasznosabb, ha egy kulcs ismert, de a hozzá társított érték ismeretlen.
  - Dokumentum: A dokumentum-adatbázisok a kulcs-érték típusú adatbázis elvét terjesztik ki azzal, hogy teljes dokumentumokat szerveznek gyűjteménynek nevezett csoportokba. Támogatják a beágyazott kulcs-érték párokat, és adott dokumentum bármely attribútuma alapján végzett lekérdezést lehetővé tesznek.
  - Oszlopalapú: Az oszlopalapú, széles oszlopú vagy oszlopcsalád-alapú adatbázisok hatékonyan tárolják az adatokat és kérdezik le a ritka adatok sorait. Előnyök az adatbázis megadott oszlopain végzett lekérdezések esetén.
  - Gráf: A gráf típusú adatbázisok csomópontokból és élekből felépülő modellt használnak az egymással összefüggő adatok – például egy közösségi hálózat tagjai közötti kapcsolatok – leképezésére, és az összetett kapcsolatok egyszerűbb tárolását és követését teszik lehetővé.

## Funkcionális függés:

- Az  $R$  két attribútumhalmaza  $X$  és  $Y$  között  $X \rightarrow Y$ -nal jelölt funkcionális függés előír egy megszorítást azokra a lehetséges rekordokra, amelyek egy  $R$  fölötti  $r$  relációt alkothatnak. A megszorítás az, hogy bármely két  $r$ -beli  $t_1$  és  $t_2$  rekord esetén, amelyekre  $t_1[X] = t_2[X]$  teljesül, teljesülnie kell  $t_1[Y] = t_2[Y]$ -nak is.
- Más szavakkal: egy  $R$  relációsémában  $X$  akkor és csak akkor határozza meg funkcionálisan  $Y$ -t, ha valahányszor  $r(R)$  két rekordja megegyezik az  $X$  értékeken, szükségszerűen megegyezik az  $Y$  értékeken is.

## Koncepcionális adatmodell (séma):






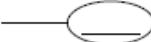
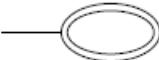
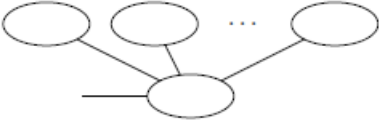
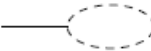
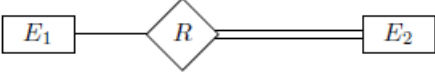
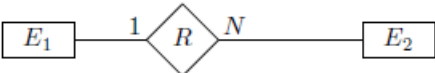
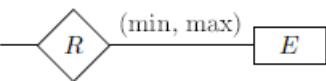
- Véges számú tulajdonságtípussal megadott véges számú egyed típus és a közöttük fennálló véges számú kapcsolattípus összessége.

## Koncepcionális adatbázis-tervezés:

- Mi a feladat és mely területen? Mi a kisvilág?
- Milyen igényeink, követelményeink, elvárásaink vannak? Mit tudjon a rendszer?
- A szükséges és lehetséges adatok számbavétele. A kívánt funkciók végig gondolása.
- Az elemzés és tervezés folyamán csoportmunka más területek szakértőivel, iteráció.
- Koncepcionális adatmodell elkészítése: az egyed-, tulajdonság- és kapcsolattípusok meghatározása.
- Ezen az adatmodellen (absztrakt) műveletek / tranzakciók definiálása.
- A logikai tervezés a DBMS leíró nyelvén.
- Logikai séma (konkrét) kialakítása.
- A logikai séma lefordítása (ez sokszor automatikus) belső (fizikai) sémává.

ER modell:

- Gyenge egyedtípus: Azok az egyedtípusok, melyek nem rendelkeznek saját kulcs attribútumokkal.
- Erős egyedtípus: Azok az egyedtípusok melyek rendelkeznek saját kulcsattribútummal.
- Részleges kulcs: A gyenge egyedtípusoknak részleges kulcsuk van, mely azon attribútumok halmaza, amelyek egyértelműen azonosítják azokat a gyenge egyedeket, amelyek ugyanazon tulajdonos egyed(ek)hez kapcsolódnak.

<u>Szimbólum</u>	<u>Jelentés</u>
	egyedtípus
	gyenge egyedtípus
	kapcsolattípus
	azonosító kapcsolattípus
	attribútum
	kulcsattribútum
	többértékű attribútum
	összetett attribútum
	származtatott attribútum
	az $E_2$ egyedtípus totális résztvevője a $R$ kapcsolatnak
	az $E_1$ és $E_2$ egyedtípusok $1 : N$ számosságú $R$ kapcsolata
	az $E$ egyedtípus $R$ -beli részvételére vonatkozó strukturális megszorítás (min, max)

ER séma leképezése:

1. Erős egyedtípusok leképezése
2. Gyenge egyedtípusok leképezése
3. Bináris 1:1 számosságú kapcsolattípusok leképezése
  - a. külső kulcs használata
  - b. összevonás
  - c. kereszthivatkozás v. kapcsoló reláció használata
4. Bináris 1:N számosságú kapcsolattípusok leképezése
5. Bináris M:N számosságú kapcsolattípusok leképezése
6. Többértékű attribútumok leképezése
7. N-edfokú kapcsolattípusok leképezése

Structured Query Language (SQL):

- Strukturált lekérdező nyelv egy komplex adatbázis-lekérdező nyelv, mellyel megvalósíthatjuk a lekérdezéseket és különböző adatbázis-kezelő műveleteket.

Data Definition Language (DDL):

- Create: pl.: tábla létrehozása
- Alter: pl.: tábla módosítása
- Drop: pl.: tábla törlése
- Truncate: pl.: tábla kiürítése (csak a tartalma törlődik)

Data Manipulation Language (DML):

- Insert: pl.: táblába új sor beszúrása
- Update: pl.: tábla feltételben megadott sorainak módosítása
- Delete: pl.: tábla feltételben megadott sorainak törlése

Data Control Language (DCL):

- Tranzakciókezelés:
  - Commit: helyi módosítások többi felhasználó számára láthatóvá tétele
  - Rollback: helyi módosítások törlése (utolsó commitig, vagy megadott savepointig)
  - Savepoint: helyi módosítások mentése (helyi módosítás marad, de rollbackel visszatörölhető a mentésig)
- Jogosultságkezelés:
  - Grant: jogosultság adása
  - Revoke: jogosultság elvétele

Query Language (QL):

- Select: Adatbázisból adat lekérdezése

Lekérdezések részei:

- SELECT: lekérdezendő attribútumok megadása
- FROM: lekérdezendő táblák megadása
- WHERE: lekérdezés feltételeinek megadása
- GROUP BY: csoportképzés megadott attribútumok alapján
- HAVING: csoportképzésre vonatkozó feltételek megadása
- ORDER BY: rendezés megadása

Tábla kapcsolások módjai:

- Belső kapcsolat:
  - INNER JOIN: A két megadott tábla kapcsoláshoz használt attribútumai alapján a közös sorok kerülnek az eredménybe
  - NATURAL JOIN: A két megadott táblából automatikusan választ ki egy-egy attribútumot, mely alapján kapcsolja a táblákat, az eredménybe a közös sorok kerülnek, az rossz attribútum választásakor hibás lehet.
- Külső kapcsolat:
  - LEFT OUTER JOIN: A bal oldalon megadott táblából azok a sorok is az eredménybe kerülnek, amelyekhez nincs a jobb oldalon megadott táblában megfelelő érték, null értékek kerülnek az eredményben ezek helyére.
  - RIGHT OUTER JOIN: A jobb oldalon megadott táblából azok a sorok is az eredménybe kerülnek, amelyekhez nincs a bal oldalon megadott táblában megfelelő érték, null értékek kerülnek az eredményben ezek helyére.
  - FULL OUTER JOIN: Mindkét megadott táblából az eredménybe kerülnek azok a sorok is, amelyeknek nincs a másik táblában megfelelő sora.