



Adatbázisrendszerek

9. gyakorlat

BETWEEN

- A BETWEEN operátorral vizsgálható, hogy egy érték egy adott intervallumon belül van-e.
 - Az `attribútum >= érték1 AND attribútum <= érték2` kiváltására használható.
- Az értékek lehetnek szám, szöveges vagy dátum típusúak is.
- A BETWEEN operátor inkluzív, azaz az intervallum kezdő és végértéke is beletartozik az intervallumba.

Példa

- Kérdezzük le a szerzők vezetéknévét „Asimov”-tól „Burnie”-ig!
Rendezzük az eredményt ABC sorrendbe!
 - `SELECT VEZETEKNEV FROM SZERZO
WHERE VEZETEKNEV BETWEEN 'Asimov' AND 'Burnie'
ORDER BY VEZETEKNEV;`

Beágyazott lekérdezések

- Az SQL lehetővé teszi SELECT utasítások egymásba ágyazását, akár többszörösen is.
 - Oracle SQL esetében ez maximum 255 szintig lehetséges.
- A beágyazott lekérdezéseknek („alszelekteknek”) két fő típusa különböztethető meg az alapján, hogy a „külső” és a „belső” lekérdezés milyen viszonyban áll egymással:
 - Korrelálatlan (egyszerű) alszelekt
 - Korrelált (kapcsolt) alszelekt

Beágyazott lekérdezések

Korrelálatlan (egyszerű) alszelekt

- Önmagában is kiértékelhető, azaz nem függ a külső SELECT utasítástól.
- Kiértékelés:
 - Először az alszelekt értékelődik ki, és az az által visszaadott értékek kerülnek át a külső SELECT utasításhoz.
- Csak egyszer fut le.

Beágyazott lekérdezések

Korrelálatlan (egyszerű) alszelekt

- Példa: Kérdezzük le azon könyvek címét és oldalszámát, amelyek hosszabbak az átlagosnál!
 - `SELECT CIM, OLDALSZAM FROM KONYV
WHERE OLDALSZAM >= (SELECT AVG(OLDALSZAM) FROM
KONYV) ;`

Beágyazott lekérdezések

Korrelálatlan (egyszerű) alszelekt

- Példa: Kérdezzük le azon könyvek címét, amelyek szerzője 1950-ben vagy az után született! Minden könyv címe csak egyszer szerepeljen!

```
▫ SELECT CIM FROM (SELECT SZERZO_AZON FROM SZERZO
WHERE SZULETESI_DATUM >= TO_DATE('1950-01-
01', 'YYYY-MM-DD'))
INNER JOIN KONYVSZERZO USING (SZERZO_AZON)
INNER JOIN KONYV USING (KONYV_AZON)
GROUP BY KONYV_AZON, CIM;
```

Beágyazott lekérdezések

Korrelált (kapcsolt) alszelekt

- Önmagában nem értékelhető ki, mert hivatkozik egy külső SELECT utasításban szereplő attribútumra/kifejezésre.
- Kiértékelés:
 - A kiértékelés a külső SELECT-nél kezdődik, majd az átadja a hivatkozott értéket a belső SELECT-nek, amely előállítja az értékekhez tartozó eredményt.
 - Ezután a belső SELECT által előállított értékkel folytatódik a külső SELECT kiértékelése.
- Újra lefut minden átadott értéknél.

Beágyazott lekérdezések

Korrelált (kapcsolt) alszelekt

- Fontos: Ha a külső és a belső SELECT ugyan azt a táblát használja, akkor vagy a külső vagy a belső lekérdezésben annak alias nevet kell adni, és annak attribútumait minősített nevekkel kell hivatkozni.

Beágyazott lekérdezések

Korrelált (kapcsolt) alszelekt

- Példa: Kérdezzük le, hogy mennyi az egyes szerzők összhonoráriumuma! A szerzők teljes neve jelenjen meg és az érték legyen kerekítve!

```
▫ SELECT VEZETEKNEV || ' ' || KERESZTNEV,  
       (SELECT ROUND(SUM(HONORARIUM)) FROM KONYVSZERZO  
WHERE KONYVSZERZO.SZERZO_AZON =  
      SZERZO.SZERZO_AZON) FROM SZERZO;
```

Beágyazott lekérdezések

Alszelektek kezelése

- Az alszelekteknek mindig zárójelek közt kell szerepelniük.
- Minden alszelekt eredménye kezelhető táblaként, azaz bárhol szerepelhet, ahol táblakifejezés szerepelhet.
- Ha az alszelekt feltételben szerepel, akkor annak megfelelően kell megírni és kezelni, hogy hány sort és oszlopot adhat az adott helyen vissza.

ROWNUM

- A ROWNUM egy pseudo-oszlop, azaz fizikailag nincs tárolva, de ugyanúgy kezelhető, mint a többi attribútum.
- Megadja, hogy hányadikként olvasta ki a DBMS a táblából vagy az eredményhalmazból az adott sort.
 - A számozás 1-től indul és egyesével növekszik.
- Olyankor használható, amikor egy táblának vagy halmaznak a legelső n sorára van szükség.
- Használata rendezett adatok esetében gyakoribb.
 - Például: 5 legdrágább könyv, 3 legfiatalabb tag stb.

ROWNUM

- Példa: Kérdezzük le az 5 legdrágább könyv címét és árát!
 - `SELECT * FROM (SELECT CIM, AR FROM KONYV WHERE AR IS NOT NULL ORDER BY AR DESC) WHERE ROWNUM <= 5;`
- Példa: Kérdezzük le a 3 legfiatalabb tag teljes nevét és születési dátumát!
 - `SELECT * FROM (SELECT VEZETEKNEV || ' ' || KERESZTNEV, SZULETESI_DATUM FROM TAG WHERE SZULETESI_DATUM IS NOT NULL ORDER BY SZULETESI_DATUM DESC) WHERE ROWNUM <= 3;`