

ADATBÁZIS ALAPÚ RENDSZEREK

Az Oracle
Database

SQL*Plus

SQL

ORACLE AZONOSÍTÓK REGISZTRÁLÁSA

- SQL Developer elindítása
- Használt adatbázisszerver (csak belső hálózatról érhető el):
 - Cím: `orania.inf.u-szeged.hu`
 - Port: 1521
 - SID: `kabinet`
- Regisztrálás folyamata:
 1. Kapcsolódjunk a fenti szerverhez `reg` felhasználónévvel (jelszó: `reg`)
 2. Futtassuk a köv. parancsot: `exec register.register('h-s azonosító', 'cser');`
 3. COMMIT kiadása, kijelentkezés a `reg` fiókból.
- A gyakorlatvezető jóváhagyása után be lehet lépni (a jelszó is a `h-s azonosító`); változtassuk meg a jelszót: `ALTER USER hxxxxxx IDENTIFIED BY ujjelszo;`

ORACLE

■ Felhasználók:

- SYS: rendszergazda
- SCOTT: demonstrációs adatbázis, táblái: EMP (dolgozó), DEPT (osztály)
- "közönséges" felhasználók

■ Adatszótár:

- metaadatokat tartalmazó, csak olvasható táblák
- táblanév-prefixek:
 - ALL: az adott felhasználó által elérhető összes objektum;
 - USER: a felhasználó sémájában lévő objektumok;

ORACLE

- **Felhasználók listája:** `SELECT * FROM all_users;`
- **A felhasználó tábláinak nevei:** `SELECT table_name FROM user_tables;`
- **DUAL segédtábla: egy bejegyzést tartalmazó tesztábla**
 - **A tábla kiírása:** `SELECT * FROM DUAL;`
 - **Teszt:** `SELECT 'a' || 'b' FROM DUAL;`
 - **Teszt 2:** `SELECT sysdate FROM DUAL;`

SQL*PLUS

- kezelőfelület:
 - háromféle parancsot adhatunk ki
 - *SQL parancsok* (adatbázis műveletek)
 - *PL/SQL blokkok* (adatbázis műveletek)
 - *SQL*Plus parancsok* (opciók beállítása, az előző két parancstípus szerkesztése és tárolása, lekérdezések formázása)
- SQL parancsbuffer: az utoljára begépelte utasítást tartalmazza (az SQL*Plus parancsok nem kerülnek bele)
 - Enter: sortörés, de nem a parancs végrehajtása
 - SQL-parancs lezárása: pontosvessző (;) vagy törtvonal (\): ezek nem kerülnek a bufferbe
 - üres sor: ekkor nem hajtódik azonnal végre a parancs, de a bufferbe bekerül.
- SQL*Plus parancsok után nem kell ;

SQL*PLUS PARANCSOK

- `CLEAR SCREEN` : képernyő törlése
- `CLEAR BUFFER` : buffer törlése
- `EXIT` : kilépés az SQL*Plusból
- `HELP parancsnev` : egy SQL*Plus parancsról kérhetünk segítséget
- `HELP INDEX` : felsorolja a parancsokat
- `SHOW ALL` : SQL*Plus paraméterek listája
- `SHOW parameter` : adott paraméter értéke
- `SET parameter ertek` : adott paraméter beállítása
- `DESCRIBE tablanev` : adott tábla szerkezetének kiírása

SQL*PLUS PARANCSOK (PÉLDA)

- `SHOW LINESIZE` – sorméret
- `SHOW PAGESIZE` – lapméret
- `SHOW NEWPAGE` – két lap közötti üres sorok száma
- `SET LINESIZE 300`
- `DESCRIBE dual`
- `DESCRIBE emp`

MANIPULÁLÓ, FUTTATÓ PARANCSOK

- **Listázzuk ki a DEMO felhasználóhoz tartozó táblákat!**
- `SAVE filename` : **buffer tartalmának mentése**
- `SAVE filename REPLACE` : **létező file felülírása**
- `EDIT filename` : **file megnyitása szerkesztésre**
- `GET filename` : **file betöltése a bufferbe**
- `SAVE else` - **létrehozza az else.sql file-t a buffer tartalmával**
- `EDIT else` - **megnyitjuk szerkesztésre a file-t**

- **Javítsuk ki a DEMO nevet saját azonosítónkra, majd mentjük a file-t.**

NYELVI ELEMELK

■ Megjegyzések:

- REMARK szoveg (PL/SQL blokkban nem használható)
- -- szoveg (tetszőleges nem SQL*Plus utasítás után)
- /* */ (többsoros megjegyzés)

■ Fontosabb adattípusok:

- NUMBER (hossz, tizedes)
- INTEGER - NUMBER-rel egyenértékű
- CHAR (n) - n fix hosszú (alpból n=1) karaktersorozat
- VARCHAR2 (n) - változó, de max. n hosszúságú karaktersorozat
- LONG -változó hosszúságú karaktersorozat
- BINARY_FLOAT
- BINARY_DOUBLE
- DATE - dátum
- TIMESTAMP - idő

■ Változók:

- rendszerváltozók (SQL*Plus paraméterek)
- felhasználói vagy helyettesítő változók (input)
- hozzárendelt változók (output PL/SQL blokkból)

VÁLTOZÓKEZELÉS

- `DEFINE` : definiált felhasználói változók listázása
- `DEFINE változo = ertek` : változó definiálás
- `UNDEFINE változo` : változó törlése
- `ACCEPT változo PROMPT 'szoveg'` : változó értékének bekérése a szoveg megjelenítésével

- **Definiáljunk egy változót:** `DEFINE nevem = 'X Y'` (alapból mindenkeppen string lesz, az `ACCEPT` paranccsal explicite is meg lehet adni a típust)
- `ACCEPT nev PROMPT 'mi a nev:'`
- `ACCEPT változo NUMBER PROMPT`

VÁLTOZÓKEZELÉS (&)

Bekérés & paranccsal (a változó nem definiálódik):

- `SELECT owner,table_name FROM all_tables WHERE owner LIKE &name;`
- **Adja meg a(z) name értékét: ' DEMO '**
- **régi 1:** `select owner, table_name from all_tables where owner like &name`
- **új 1:** `select owner, table_name from all_tables where owner like ' DEMO '`

VÁLTOZÓKEZELÉS (&&)

- **Bekérés && paranccsal (a változó definiálódik):**
- `SELECT owner, table_name FROM all_tables WHERE owner LIKE &&name;`
- **Adja meg a(z) name értékét: ' DEMO '**

- **Paraméter átadás a START paranccsal:**
- **Ha a MYFILE tartalma az alábbi:**
`SELECT * FROM emp WHERE job=' &1' AND sal=' &2' ;`
akkor a START MYFILE param1 param2 parancsot használjuk

HOZZÁRENDELTELT VÁLTOZÓ

- Deklaráljunk egy hozzárendelt változót, majd kérdezzük le a deklarált változókat:
 - `VARIABLE XY CHAR`
 - `VARIABLE`
- **Hozzárendelt változó értékének kiíratása:** `PRINT`
`valtozonev`

1. FELADAT

Hozzuk létre egy könyvek nevű SQL file-t, amely futtatás után létrehoz egy táblát könyv néven és benne két adatot, majd kiírja azokat.

- 1.** Adjunk jogot a szomszédunknak a táblánk lekérdezésére.
- 2.** Kérdezzük le a szomszéd tábláját.
- 3.** Vonjuk meg a jogot.

2. FELADAT

Írjunk egy rövid SQL*Plus szkriptet az alábbi adatok megjelenítésére!

1. Azon táblák neve, tulajdonosa, sorainak száma, amelyek egy bekért számnál több sort tartalmaznak.
 2. A táblák száma felhasználónként.
 3. A korábban bekért paraméternél több táblával rendelkező felhasználók listája.
- **Szorgalmi:** a 2. lekérdezést úgy adjuk meg, hogy a 0 táblával rendelkező felhasználók is szerepeljenek benne (ők nincsenek benne all_tables-ben!)

IRODALOM

[1] Balázs Péter: SQL*Plus ismertető.

<http://www.inf.u-szeged.hu/~pbalazs/teaching/SQLPlus.pdf>