

Tartalomjegyzék

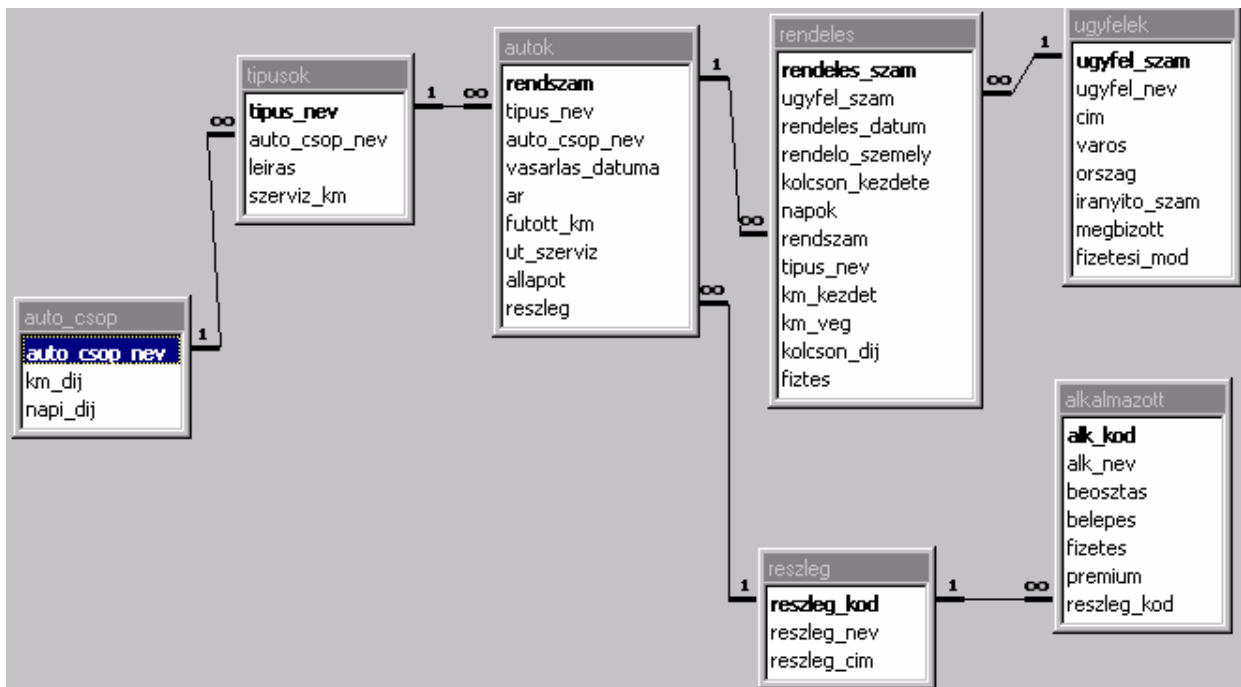
Tartalomjegyzék	1
Az SQL nyelv	1
Az SQL DDL alapjai	2
Adatbázis parancsok	2
Táblaparancsok	2
A táblázat létrehozása	2
A táblázat módosítása	3
A tábla törlése	3
Indextábla létrehozása	3
Az SQL DML alapjai	4
A SELECT ... FROM alparancs	4
A WHERE alparancs	4
A GROUP BY alparancs	5
Fontosabb (csoport)függvények	5
Az ORDER BY alparancs	6
Virtuális tábla létrehozása	6
Sorok bevitele a táblába	6
A tábla rekordjainak módosítása	6
A tábla rekordjainak törlése	6
Az SQL DCL nyelv alapjai	7
Hozzáférési jogok adományozása	7
A táblákra vonatkozó jogosultság adományozása	7
Jogosultság adományozása az adatbázison végzett műveletekre	7
Jogosultságok visszavonása	8
Táblákra vonatkozó jogosultság visszavonása	8
Adatbázis-jogosultságok visszavonása	8

Az SQL nyelv

Az SQL (Structured Query Language) 1986-ban az Amerikai Szabványügyi Hivatal (ANSI) által szabványosított **strukturált lekérdezőnyelv**, mely relációs adatmodell alapján felépülő adatbázisok kezelésére képes. Az SQL három résznyelvre bontható:

- SQL DDL, azaz az SQL adatdefiníciós nyelv;
- SQL DML, azaz az SQL adatmanipulációs nyelv;
- SQL DCL, azaz az SQL adatvezérlő nyelv.

A továbbiakban a példákat mindig az alábbi adatbázisra vonatkoztatjuk:



Az SQL DDL alapjai

Adatbázis parancsok

Az adatbázis-parancsokat az alábbi táblázatban foglalhatjuk össze:

A parancs alakja:	A parancs funkciója:
CREATE DATABASE <i>adatbázisnév</i> ;	Adatbázis létrehozása
SHOW DATABASE ;	Információ egy adatbázisról
START DATABASE <i>adatbázisnév</i> ;	Adatbázis megnyitása
STOP DATABASE ;	Adatbázis bezárása
DROP DATABASE <i>adatbázisnév</i> ;	Adatbázis törlése

Táblaparancsok

A táblázat létrehozása

A táblázat definiálását az SQL-ben a CREATE TABLE parancs végzi. Általános alakja

```
CREATE TABLE táblanév ( oszlopnév adattípus ( méret )
[ , oszlopnév adattípus ( méret ) ] ) ;
```

Az oszlopneveknél az adattípus mellé beírható a NULL illetve a NOT NULL opció. Ez utóbbi azt jelenti, hogy a definiált táblában az adott oszlop kitöltése kötelező. A gyakoribb SQL adattípusokat a következő táblázat tünteti fel:

Adattípus neve:	Jelentése:
CHAR (n)	N hosszúságú string
INTEGER	Egész szám -9 999 999 999 és 99 999 999 999 között
SMALLINT	Egész szám -99 999 és 999 999 között
DECIMAL (x , y)	Előjellel együtt x számjegyű, fixpontos decimális szám y tizedesjeggyel, ahol x 1 és 19, y 0 és 18 közötti értéket vehet fel
FLOAT (x , y)	Előjellel együtt x számjegyű, lebegőpontos decimális szám y tizedesjeggyel, ahol x 1 és 20, y 0 és 18 közötti értéket vehet fel
DATE	Dátum típus
LOGICAL	Logikai típus, értéke .T. és .F. lehet

A különböző adatbázis-kezelő programok SQL motorjai általában megengednek más adattípusokat is. Az Access '97 SQL pl. megengedi a BYTE, CURRENCY típusok használatát. Lehetséges eltérés a szabvány táblakészítő utasítástól is. Az Access '97 SQL például az elsődleges kulcs kijelölésére más utasítást használ:

```
CREATE TABLE auto_csop (auto_csop_nev CHAR(6)NOT NULL PRIMARY
KEY,km_dij INTEGER,napi_dij INTEGER);
```

```
CREATE TABLE auto_csop (auto_csop_nev CHAR(6)NOT NULL,km_dij
INTEGER,napi_dij INTEGER CONSTRAINT auto_csop_nev PRIMARY
KEY);
```

A táblázat módosítása

A táblázat módosítására az ALTER TABLE parancs szolgál. Általános alakja:

```
ALTER TABLE táblanév
ADD/MODIFY (oszlopnév adattípus [ , oszlopnév adattípus ] );
```

Az Access'97 SQL itt is egy kicsit más:

```
ALTER TABLE auto_csop ADD megjegyzes CHAR(20);
ALTER TABLE auto_csop ADD COLUMN megjegyzes CHAR(20);
```

A tábla törlése

A tábla törlésére a katalógusból a DROP TABLE parancs szolgál. Alakja:

```
DROP TABLE táblanév;
```

Indextábla létrehozása

Az *indexállomány* egy adott táblából kiemelt néhány rendezett oszlopból áll. Az SQL-ben rendezett oszlopokból álló indextáblát a következő alakú parancs hozza létre:

```
CREATE [UNIQUE] INDEX indextábla-név ON táblanév
(oszlopnév [[ASC/DESC]] [, oszlopnév[ASC/DESC]] ... );
```

A parancs hatása: Az ON után adott tábla felsorolt oszlopait rendezi (növekedően ASC esetén és csökkenően DESC esetén) és belőlük egy az INDEX szó után megadott nevű táblát készít. Az UNIQUE azt jelenti, hogy az oszlop értékei egyediek, s ha ez esetben ismétlődő értékek is vannak az oszlopban, a rendszer hibát jelez.

Az SQL DML alapjai

Az SQL lekérdezőparancsa a SELECT parancs összetett parancs, több részparancsból áll.

A **SELECT ... FROM** alparancs

A SELECT ... FROM alparancs a projekció megvalósítására alkalmas. Általános alakja:

```
SELECT [DISTINCT] oszlopnév-lista FROM [táblanév];
```

A SELECT parancs hatására egy ideiglenes **eredménytábla** keletkezik. Amennyiben a DISTINCT szerepel, úgy az azonos értékek közül csak egyet választ ki az adott oszlopba. Az oszlopnév-lista elemei nemcsak oszlopnevek, hanem oszlopokból álló kifejezések, aggregáló (egyesítő) függvények, s más kifejezések is lehetnek.

A **WHERE** alparancs

A WHERE alparancs a szelekciót valósítja meg. Általános alakja:

```
WHERE feltétel
```

A parancs hatására az eredménytáblában a SELECT után felsorolt oszlopokba azoknak a soroknak az értékei kerülnek, amelyekre teljesül a feltétel. A feltételre vonatkozó szabályokat az alábbi táblázattal foglalhatjuk össze:

A feltétel típusa:	A feltétel alakja:	Az operátor:
Egyszerű összehasonlítás	<i>oszlopnév operátor kifejezés</i>	=; != ;< ;> ;<> ;^= ; >=; <=
Összehasonlítás egy halmaz elemeivel	<i>oszlopnév operátor halmazdef.</i>	BETWEEN szám1 AND szám2; IN (lista) LIKE karakterminta
Összehasonlítás NULL értékkel	<i>oszlopnév IS NULL</i>	
Összetett kereséséi feltétel	<i>feltétel1 operátor feltétel2</i>	AND; OR; NOT

Néhány példa:

```
SELECT tipus_nev,ar FROM autok WHERE ar>1350000;  
SELECT * FROM alkalmazott WHERE alk_nev LIKE "H*";  
SELECT * FROM autok WHERE futott_km BETWEEN 120000 AND 200000;
```

A GROUP BY alparancs

A GROUP BY alparancs alakja:

GROUP BY *oszlopnév* [,*oszlopnév*] [HAVING *feltétel*]

A parancs hatására a megadott oszlop azonos értékei szerint csoportosítja a rekordokat. Amennyiben szerepel a HAVING, az azt jelenti, hogy a GROUP BY által kialakított eredménytáblából kiválasztja azon sorokat, amelyek eleget tesznek a HAVING utáni feltételnek.

Fontosabb (csoport)függvények

INITCAP(kifejezés)	Nagy kezdőbetű, karakteres kifejezésekre vonatkozik
LOWER(kifejezés)	Kisbetűs, karakteres kifejezésekre vonatkozik
UPPER(kifejezés)	Nagybetűs, karakteres kifejezésekre vonatkozik
AVG([DISTINCT ALL] kifejezés)	Átlag, numerikus vonatkozik, a Null értéket figyelmen kívül hagyja.
COUNT([DISTINCT ALL] { <i>*</i> kifejezés})	Számláló, numerikus dátum és karakteres kifejezésre is vonatkozik, a Null értéket figyelmen kívül hagyja.
MAX([DISTINCT ALL] kifejezés)	Maximum, numerikus dátum és karakteres kifejezésre is vonatkozik.
MIN([DISTINCT ALL] kifejezés)	Minimum, numerikus dátum és karakteres kifejezésre is vonatkozik.
SUM([DISTINCT ALL] kifejezés)	Összeg, numerikus kifejezésre vonatkozik.
ABS(kifejezés)	Abszolútérték, numerikus kifejezésre vonatkozik.
SQRT(kifejezés)	Négyzetgyök, numerikus kifejezésre vonatkozik.

Néhány példa:

```
SELECT reszleg_kod, COUNT(*) FROM alkalmazott  
GROUP BY reszleg_kod;  
  
SELECT reszleg_kod,MIN(fizetes),  
MAX(fizetes),SUM(fizetes),AVG(fizetes)  
FROM alkalmazott GROUP BY reszleg_kod;
```

Az ORDER BY alparancs

Az ORDER BY alparancs az eredménytábla rendezésére szolgál. Általános alakja:

```
ORDER BY oszlopnév [ASC/DESC][,oszlopnév [ASC/DESC]];
```

A parancs a megadott oszlop (vagy oszlopok) szerint rendezi az eredménytáblát, ASC megadása esetén növekedő sorrendben (ez az alapértelmezés is), DESC esetén csökkenő sorrendben.

Példa:

```
SELECT * FROM alkalmazott  
WHERE beosztas="Elado" OR beosztas="Szerelo"  
ORDER BY fizetes DESC;
```

Virtuális tábla létrehozása

A virtuális tábla létrehozására a CREATE VIEW parancs szolgál. Általános alakja:

```
CREATE VIEW táblanév [oszlopnévlista] AS SELECT [WITH CHECK OPTION];
```

A parancs hatására a SELECT parancs által generált eredménytáblának a VIEW után írt táblanevet adja az SQL és ennek a táblának az oszlopai az oszlopnévlistában felsorolt oszlopok lesznek. Amennyiben nem adunk meg oszlopnévlistát, akkor az eredménytábla nevei lesznek a VIEW tábla oszlopnevei.

Sorok bevitele a táblába

A parancs alakja:

```
INSERT INTO táblanév [(oszlopnév-lista)] VALUES (értéklista)/szelekciós utasítás;
```

Példa:

```
INSERT INTO auto_csop (auto_csop_nev,km_dij,napi_dij)  
VALUES ("Luxus",6000,8000);
```

A tábla rekordjainak módosítása

A parancs alakja:

```
UPDATE táblanév SET oszlopnév=kifejezés [oszlopnév=kifejezés] [WHERE logikai kifejezés];
```

Példa:

```
UPDATE auto_csop SET napi_dij=8500  
WHERE auto_csop_nev="Luxus";
```

A tábla rekordjainak törlése

A parancs alakja:

```
DELETE FROM táblanév [WHERE feltétel];
```

Az SQL DCL nyelv alapjai

Hozzáférési jogok adományozása

A táblákra vonatkozó jogosultság adományozása

A parancs formája:

```
GRANT ALL [PRIVILEGES]/ jogosultságlista ON [TABLE] táblalista  
TO PUBLIC/felhasználólista [WITH GRANT OPTION];
```

A parancs minden jogot (ALL PRIVILEGES) vagy a jogosultságlistában szereplő műveletekre való jogot adja a táblalistában szereplő táblákra mindenkinek (PUBLIC esetén) vagy a felhasználólistában szereplő személyeknek. Amennyiben a WITH GRANT OPTION szerepel, akkor az e jogokat kapók át is adhatják ezeket a jogokat másoknak. A jogosultságlista elemeit a következő táblázatban foglalhatjuk össze:

A jogosultság neve:	A jogosultság jelentése:
ALTER	Jogosultság a tábla módosítására
DELETE	Jogosultság a tábla törlésére
INDEX	Jogosultság indextábla létrehozására
INSERT	Jogosultság új sor felvételére a táblázatba
SELECT	Jogosultság lekérdezésre
UPDATE	Jogosultság a tábla módosítására

Jogosultság adományozása az adatbázison végzett műveletekre

A parancs formája:

```
GRANT adatbázisjog TO PUBLIC/felhasználólista
```

A parancs jogosultságot ad az adatbázisra vonatkozóan vagy mindenkinek (PUBLIC) vagy adott felhasználóknak a felhasználólista szerint.

Az adatbázisjogokat a következő táblázatban foglalhatjuk össze:

A jog neve:	A jog jelentése:
CONNECT	<ul style="list-style-type: none"> - Hozzáférés a teljes adatbázishoz - Jog arra, hogy SELECT, INSERT, DELETE, UPDATE műveleteket végezzon más felhasználók tábláin, ha ilyen jogosultságot kapott a táblákra vonatkozó GRANT-tal - Jog nézettáblák és szinonim táblák létrehozására.
RESOURCE	<ul style="list-style-type: none"> - Minden CONNECT jogosultság - Jogosultság táblák és inextáblák létrehozására, jogosultságok adományozása ezekre a táblákra
DBA	-Teljes adatbázis-adminisztrátori jogkör

Jogosultságok visszavonása

Táblákra vonatkozó jogosultság visszavonása

A parancs formája:

```
REVOKE ALL [PRIVILEGES]/ jogosultságlista ON [TABLE] táblalista
TO PUBLIC/felhasználólista ;
```

A parancs hatása: Az összes jogosultságot (ALL PRIVILEGES vagy csak a jogosultságlistában felsoroltakat a megadott táblákra vonatkozóan mindenkitől (PUBLIC) vagy csak a listában szereplő felhasználóktól visszavonja.

Adatbázis-jogosultságok visszavonása

A parancs formája:

```
REVOKE adatbázisjog FROM PUBLIC/felhasználólista ;
```

A parancs adatbázisjogokat mindenkitől (PUBLIC) vagy a listában szereplőktől visszavonja;