Készítette: Nagy Dávid 2018. 01. 08

Adatbázis

1. CREATE TABLE gyumolcsok(

id INT(11),

megnevezes VARCHAR(50),

leiras TEXT

);

2. Új rekord(ok) beszúrása az adattáblába.

```
INSERT INTO gyumolcsok(megnevezes, leiras) VALUES('alma', 'Rövid leírás');
```

Lehetőség van egyszerre több rekord beszúrására is

```
INSERT INTO gyumolcsok(megnevezes, leiras) VALUES
('alma', 'Rövid leírás'),
('barack', 'Rövid leírás'),
('dinnye', 'Rövid leírás');
```

Lekérdezés :A lekérdezéseket mindig a SELECT kulcsszóval kezdjük.
A SELECT után kell megadni azokat a mezőket amiket eredményként visszakapunk.

Minden nev és leiras mező értékének lekérdezése a gyumolcsok táblából

```
SELECT nev, leiras FROM gyumolcsok
```

Minden mező lekérdezése a gyumolcsok táblából

```
SELECT * FROM gyumolcsok
```

Logikai operátorok:

Operátor	Leírás
=	Egyenlőség
<> vagy !=	Nem egyenlő
>	Nagyobb mint
<	Kisebb mint
<=	Kisebb vagy egyenlő
NOT	Feltétel ellenkezője

A megjelenő mezőnév módosítása

```
            megnevezes AS 'Gyümölcs neve';
```

Distinc:

Az egymástól különböző elemek lekérdezésére szolgál.

Lekérdezi a különböző nevű gyümölcsöket.

```
            SELECET DISTINCT nev FROM gyumolcsok
```

FROM:

A lekérdezések része a forrás adattábla(k) meghatározása, melyet használtunk az előző példákban is.

GROUP BY:

A megadott feltétel(mező értéke) szerint azonos rekordok csoportosítása, melyet jellemzően aggregáló függvényekkel kiegészítve használunk.

Ebben a lekérdezésben minden rekord külön sorként kerül listázásra és mindenhol 1 lesz a második mező értéke.

```
            SELECET gyumolcsnev, COUNT(*) FROM kiszallitasok;
```

HAVING:

A HAVING segítségével a már csoportosított(GROUP BY) eredményhalmazra tudunk extra feltételeket megadni.

```
            SELECET gyumolcsnev, COUNT(*) FROM kiszallitasok GROUP BY gyumolcsnev
HAVING COUNT(*) > 5;
```

LIMIT:

A visszatérési rekordok számának limitálása.

Az első 5 sort adja vissza eredményként.

```
            SELECT * FROM gyumolcsok LIMIT 5
```

ORDER BY:

Rendezési feltétel meghatározása. Megadhatjuk melyik mezőre történjön a rendezés és milyen irányban.

Növekvő sorrendben történő rendezés

```
            SELECET * FROM gyumolcsok ORDER BY nev ASC
```

Csökkenő sorrendben történő rendezés

```
SELECET * FROM gyumolcsok ORDER BY nev DESC
```

WHERE:

A lekérdezés eredményének szűkítésére szolgál. A mezőnevekre megadott logikai műveletek segítségével szűkíti az eredmény halmazt.

Lekérdezzük az összes gyümölcsöt ahol ki van töltve a leírás.

```
SELECT * FROM gyumolcsok WHERE leiras <> '';
```

Aritmetikai operátorok:

AND:

```
SELECT column1, column2, ...
FROM table_name
WHERE condition1 AND condition2 AND condition3 ...;
```

AVG():

Átlagszámítás, visszaadja az átlag értéket.

SELECT AVG(osszeg) FROM megrendelesek;

BETWEEN:

Az operátor segítségével olyan feltételeket fogalmazhatunk meg, amikor egy mező értéke két érték közötti tartományba esik.
Az értékek lehetnek szám, szöveg és dátum típusúak.

```
fizetes BETWEEN 100000 AND 20000;
```

A NOT operátorral tetszőlegesen kiegészíthető, ha a feltétel ellenkezőjét szeretnénk.

```
fizetes NOT BETWEEN 100000 AND 20000;
```

COUNT():

Számlálás, visszaadja a sorok számát.

SELECT COUNT(*) FROM megrendelesek;

IN:

Az IN operátor segítségével lehetséges egy mező vizsgálata egyszerre több értékkel.

```
megnevezes IN ('alma', 'körte', 'barack');
```

Készítette: Nagy Dávid 2018. 01. 08

Ez megegyezik az alábbi feltételekkel.

```
            megnevezes = 'alma' OR megnevezes = 'körte' OR megnevezes = 'barack';
```

IS NULL:

```
SELECT column_names
FROM table_name
WHERE column_name IS NULL;
```

IS NOT NULL:

```
SELECT column_names
FROM table_name
WHERE column_name IS NOT NULL;
```

LIKE:

Keresés az adott mező értékben a meghatározott minta alapján. Wildcardként a % jel használható.

Minden almával kezdődő megnevezés

```
            megnevezes LIKE 'alma%';
```

Minden almával végződő megnevezés

```
            megnevezes LIKE '%alma';
```

Minden almát tartalmazó megnevezés

```
            megnevezes LIKE '%alma%';
```

MAX():

isszaadja a legnagyobb értéket, jelen esetben a legnagyobb értékű megrendelés.

SELECT MAX(osszeg) FROM megrendelesek;

MIN():

Visszaadja a legkisebb értéket, jelen esetben a legkisebb értékű megrendelés.

SELECT MIN(osszeg) FROM megrendelesek;

NOT:

```
SELECT column1, column2, ...
FROM table_name
WHERE NOT condition;
```

OR:

Készítette: Nagy Dávid 2018. 01. 08

```
SELECT column1, column2, ...
FROM table_name
WHERE condition1 OR condition2 OR condition3 ...;
```

SUM():

Mező értékeinek összegzése, jelen esetben az össze megrendelés összértéke..

SELECT SUM(osszeg) FROM megrendelesek;