1.

**SELECT** \*

**FROM** Customers;

2.

**INSERT INTO** Customers (CustomerName, ContactName)

**VALUES** ('Gabes Company', 'Gabesz');

**or VALUES** ('', ''); 🡪 *to avoid null value*

3.

**DELETE** **FROM** Customers

**WHERE** CustomerID = 91;

4.

**UPDATE** Customers

**SET** ContactName = 'Gabes Company', Adress = 'Fő út', City = 'Budapest', PostalCode = '1139', Country = 'Sehol'

**WHERE** CustomerID = 91;

5.

**SELECT** \*

**FROM** Customers

**WHERE** City = ’Budapest’ **OR/AND** ContactName = ’Gabesz Moreno’;

6.

**SELECT** ContactName, PostalCode, City

**FROM** Customers

**WHERE** City = ’Mexico D. F.’

**AND** ContactName **LIKE** ' %Antonio% ';

7.

**SELECT** **DISTINCT** City

**FROM** Customers;

8.

**SELECT** **COUNT** (**DISTINCT** Country)

**FROM** Customer;

9.

**SELECT** **DISTINCT** Country

**FROM** Customers

**WHERE** Country **NOT** NULL

**ORDER** **BY** Country **DESC**;

10.

**SELECT** \*

**FROM** Customers

**WHERE** Country **IS** null;

11.

**SELECT** **DISTINCT** Country

**FROM** Customers

**WHERE** Country **NOT** null

**ORDER** **BY** Country

**DESC** **LIMIT** 3; 🡪 *pl: max 3 város neve csökkenő sorrendben*

12.

**SELECT** \* **FROM** Employees **ORDER** **BY** BirthDate **LIMIT** 1;

**SELECT** **MIN**(BirthDate) **FROM** Employees;

**SELECT** **MAX**(BirthDate) **FROM** Employees;

**SELECT** \* **FROM** Employees **WHERE** Notes **LIKE** '\_du%';

**SELECT** \* **FROM** Customers **WHERE** Country **IN** ('Germany','France','UK');

13.

**SELECT** \*

**FROM** Employees

**WHERE** BirthDate

**BETWEEN** '1961-01-01' **AND** '1971-12-31');

**SELECT** LastName **AS** ’Vezeteknev’

**FROM** Employees

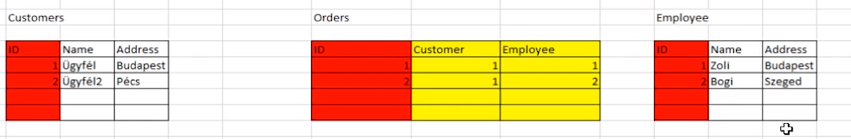
**WHERE** BirthDate **NOT** **BETWEEN** '1961-01-01' **AND** '1971-12-31');

14. **JOINS (manual mode)**

**SELECT** \*

**FROM** Orders, Customers, Employees

**WHERE** Customers.CustomerID = Orders.CustomerID **AND** Orders.EmployeeID = Employees.EmployeeID;



15. **INNER JOIN**

**SELECT** \*

**FROM** Orders

**INNER** **JOIN** Customers **ON** Orders.CustomerID = Customers.CustomerID;

**INNER** **JOIN** Employees **ON** Orders.EmployeeID = Employees.EmployeeID;

***Inner Join:*** *what data elements are in both table…*

16.

**SELECT** Customers.CustomerName, Orders.OrderID

**FROM** Customers

**LEFT** **JOIN** Orders **ON** Customers.CustomerID = Orders.CustomerID

**ORDER** **BY** Customers.CustomerName;

**SELECT** Orders.OrderID, Employees.LastName, Employees.FirstName

**FROM** Orders

**RIGHT** **JOIN** Employees **ON** Orders.EmployeeID = Employees.EmployeeID

**ORDER** **BY** Orders.OrderID;

17.

**SELECT** Customers.CustomerName, Orders.OrderID

**FROM** Customers

**FULL** **OUTER** **JOIN** Orders **ON** Customers.CustomerID=Orders.CustomerID

**ORDER** **BY** Customers.CustomerName;

18. sub\_qurey

**SELECT** \*

**FROM** Customers

**GROUP** **BY** **HAVING**;

**SELECT** SupplierName

**FROM** Suppliers

**WHERE** **EXISTS** (**SELECT** ProductName **FROM** Products **WHERE** Products.SupplierID = Suppliers.supplierID **AND** Price < 20);

19.

**SELECT** City **FROM** Customers

**UNION**

**SELECT** City **FROM** Suppliers

**ORDER BY** City;

SELECT \* FROM Bank WHERE lakcim REGEXP Petőfi | Loczy;

SELECT \* FROM Bank WHERE lakcim REGEXP Petőfi Sándor utca [1234];

SELECT UPPER CONCAT Vezeteknev, ’ ’ , Keresztnev Nevek FROM Bank LIM29; (nevek lesz az oszlop neve)

UPPER: nagybetűvel iratja ki

LOWER: kisbetűvel iratja ki

LENGHT: hány karakter a kiiradnó adat pl.: SELECT LENGHT Vezeteknev;

szövegkezelő függvények:

SUBSTR; RPAD; LPAD;

**Matematikai függvények:**

COUNT – hány db van, pl.: hány olyan város van, amelyiknek az irányitószáma 8200? Hány olyan sor van, amelyikben ez szerepel?

ABS – abszolút érték

SQRT – gyökvonás

POW – hatványra emelés

ROUND – kerekítés

MAX, MIN - pl.: legnagyobb egyenleg kilistázása

TRUNCATE – tizedesjegy csonkolás

CEIL – felfelé kerekítés a csonkolás után – double-ből csinál integert

FLOOR – lefelé kerekít csonkolás után

SIGN - +1/-1 érték visszakapása

SELECT COUNT Varos

FROM Bank

WHERE Iranyitószam = 8200;

SELECT ABS (-1\*Egyenleg)

FROM Bank;

SELECT Egyenleg SQRT Egyenleg

SQRT FROM Bank;

SELECT Egyenleg POW Egyenleg

POW FROM Bank;

SELECT Egyenleg SQRT Egyenleg

ROUND\_SQRT FROM Bank;

SELECT Vezeteknev, Keresztnev, Szamaszam MAX Egyenleg

FROM Bank;

SELECT Vezeteknev, Keresztnev, Szamaszam MIN Egyenleg

FROM Bank;

SELECT Egyenleg SQRT Egyenleg SQRT, TRUNCATE(SQRT(Egyenleg), 2)

TRUNCATE\_SQRT, CEIL(SQRT(Egyenleg)) CEIL,

FLOOR FROM Bank LIMIT 29;

**Dátumkezelő függvények:**

SELECT CURDATE();

SELECT CURTIME();

Egyben:

SELECT CONCAT(CURDATE(), ’ ’, CURTIME()) TIME;

SELECT YEAR(CURDATE()) YEAR;

SELECT CURDATE() MOST, ADDDATE (CURDATE(), 10) 10 nappal később;

SELECT YEAR(Gyartas\_Datuma) Év, Hónap, Nap

FROM Auto;

**GROUP BY:** hány ügyfelem van pl.: adott városból? (Where konfliktus – helyette: HAVING)

SELECT Varos VÁROSOK,

COUNT(\*) Ügyfelek\_szama

FROM Bank

GROUP BY Varos

HAVING COUNT(\*)>2

ORDER BY COUNT(\*) DESC;

SELECT Tipus, COUNT(\*)

FROM Auto

GROUP BY Tipus

HAVING Tipus

LIKE ’%e%’;

UNION, UNION ALL,

MATCH AGAINST – keresés bizonyos oszlopban,m de előtte hozzá kell adni mint full text-et

Azok a lakcímek amikben szerepel az h petőfi pl

ALTER TABLE Bank ADD FULLTEXT(Lakcím);

SELECT \*

FROM Bank

WHERE MATCH(Lakcim) AGAINST(’Petőfi’);

**Oszlop hozzáfűzése:**

ALTER TABLE Users

DROP COLUMN email;

Nézet táblázat: