$\nearrow$ S Q L $\nwarrow$

Structured   Query   Language   eS Quewe eL
Sequel

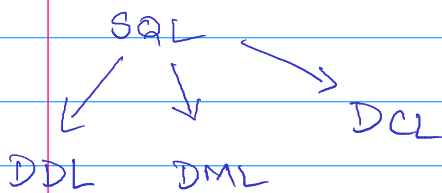| IBM	NCR	Microsoft	Sun MicroSystems	Oracle
↓	↓	↓	↓	↓
DB2	Teradata		Sun H/w	Oracle
		MS Access	Sun Solaris	
		MS SQL Server		

Java

Star Office
↳ Open Office

SQL → DCL
↓ ↓
DDL DML

MySQL
MariaDB Percona

DDL → Data Description Language

DML → Data Manipulation Language

DCL → Data Control Language

Data types

Numeric → Integers → TINYINT   1 Byte   [-128-127]
               → SMALLINT   [0-256]
                        2 Bytes $\begin{bmatrix} -32k & -32k \\ 0 & -65k \end{bmatrix}$
→ Decimals → INT   4 Byte   ±4B
           → LONG   8 Byte   ±2⁶³

FLOAT   DOUBLE   **INT (5)**

FLOAT (5,3)

String → Fixed Length → CHAR   **CHAR(8)**
     → Variable Length → VARCHAR   **VARCHAR (10)**
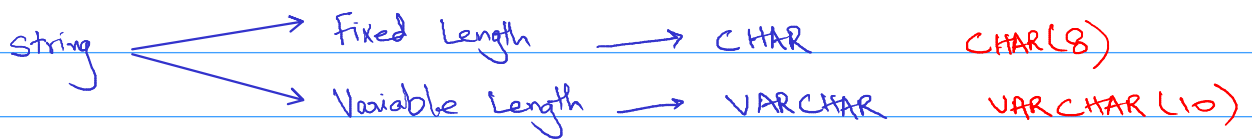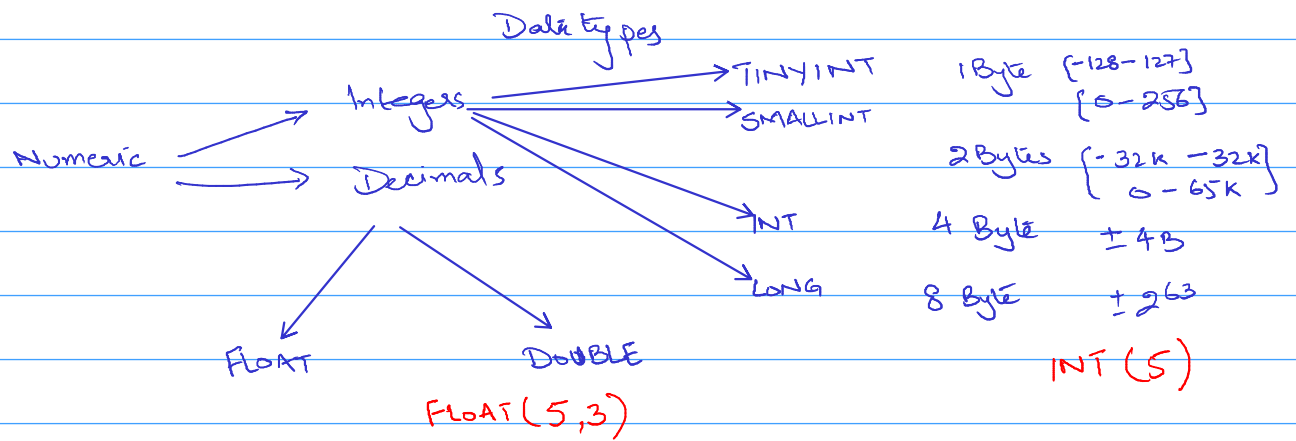
BOOL

DATE   **YYYY - MM - DD**

TIME   **HH : mm : ss**

DATETIME

Binary → BLOB → TINYBLOB   256 Bytes
                    → SMALLBLOB   65 KB
   Binary large Objects → BLOB   4 MB
                      → LARGEBLOB   2GB
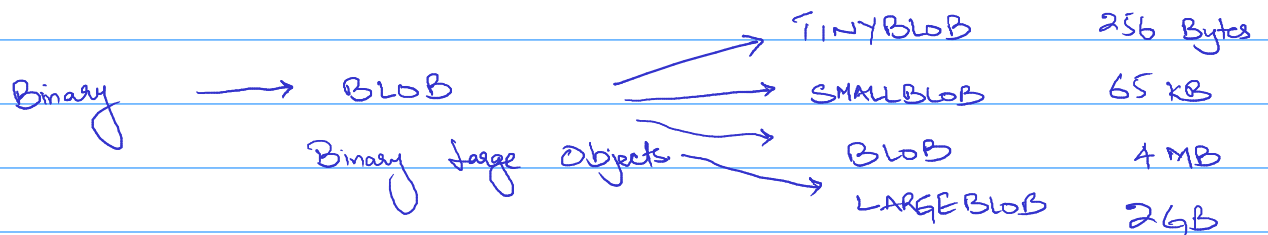
```sql
-- this is a comment
-- which dbs are available
show databases;

-- create a new db
Create database D17_Material;

-- all further commands should go to this db
Use d17_material;

-- remove database from system
drop database d17_material;

-- which tables are present
Show tables;

-- lets make a table
CREATE TABLE Student
(
        sID INT(5),
    sName VARCHAR(20),
    gpa FLOAT(3,2),
    sex CHAR(1)
);

-- whats is the structure of the table
DESCRIBE student;

-- discard table and data
DROP table student;

CREATE TABLE Student
(
        sid INT(5) NOT NULL UNIQUE,
    fName varchar(10),
    lName varchar(10) NOT NULL,
    dob DATE,
    phone char(11),
    gpa FLOAT(3,2) NOT NULL,
    Sex CHAR(1) DEFAULT 'M',
    PRIMARY KEY(SID)
);
```

```sql
-- add a new record in the table
INSERT into Student VALUES (1, 'Ali', 'Ahmad', NULL,
                    '12345678901', 3.33, 'M');


Select * from Student;


INSERT into Student VALUES (2, 'Ali', 'Again', '2000-07-15',
                    '12345678901', 3.33, NULL);


-- adding data in specific columns only
INSERT INTO Student (fName, lName, Sid, dob, gpa)
                    VALUES ('Saleem', 'Elahi', 3, '1999-01-23', 2.8);


INSERT INTO Student (fName, lName, Sex, Sid, dob, gpa)
                    VALUES  ('Fatima', 'Azam', 'F', 4, '1999-01-23', 3.8),
                    ('Ayesha', 'Waseem', 'F', 5, '1998-02-23', 2.5),
                    ('Farooq', 'Saleem', 'M', 6, '2000-01-23', 2.2),
                    ('Noraiz', 'Rasheed', 'M', 7, '2001-09-23', 2.0);


Select * from Student where GPA < 3.0 AND Sex = 'M';


delete from Student where sex = 'M';


Update Student set gpa = 3.1, phone = '98765432101' where sid = 7;
```

```sql
Select * from Customers

Select * from Customers where Country = 'France'

Select * from Customers where Country = 'France' or Country = 'Germany'

Select * from Products whjere price < 22

Select * from Products where price > 15 and price < 22

Select * from Products where supplierid <= 10

Select * from Products where supplierid <= 10 and price > 15 and price < 22

select * from Products where (price > 15 and price < 22)
                    or (supplierid <= 10 and supplierid > 5)

Select * from Customers where country != 'France'

Select * from Customers where country <> 'Germany'

Select * from Orders where orderdate = '1996-07-08'

Select * from Customers where Country > 'Spain'

Select CustomerName, City, Country from Customers;

Select CustomerName, City, Country from Customers where Country = 'Germany';

Select CustomerName from Customers where country = 'Germany';

Select distinct Country, City from Customers;

Select * from products where price BETWEEN 10 and 15;

Select * from products where price NOT BETWEEN 10 AND 15;

Select * from products where price IN (10, 12, 20, 22, 25, 25.5)

Select * from products where price NOT IN (10, 12, 20, 22, 25, 25.5)
```

< >

$(A \cup B)' = A' \cap B'$

$(A \cap B)' = A' \cup B'$

Wild Cards

— matches exactly 1 character

% matches any 0 or more length string

CustomerName starts with L
```
Select * from Customers where ContactName LIKE 'L%';
```

end with n
```
Select * from Customers where ContactName LIKE '%n';
```

2nd name start with L
```
Select * from Customers where ContactName LIKE '% L%';
```

name has an O anywhere
```
Select * from Customers where ContactName LIKE '%o%';
```

2nd last letter is O
```
Select * from Customers where ContactName LIKE '%o_';
```

name
```
Select * from Customers where ContactName LIKE '%n%n%';
```

```
Select * from Customers where ContactName NOT LIKE '%o%';
```

```sql
SELECT CustomerId, ContactName as President, City, Country FROM Customers
where ContactName like 'A%';
```

```sql
Select * from Customers where ContactName IS NULL;
                                          IS NOT NULL
```

		SSN
A	Ids of the Employees whose name start with J	---------
		123456789
	Select SSN from Employee where fname like 'J%';	987654321
		453453453
		888665555

		ESSN
B	Ids of the Employees who have a male dependent	----------
		333445555
	Select ESSN from Dependent where sex = 'M';	987654321
		123456789

Find Ids of Employees whose name start with J or they
have a male dependent

Select SSN from Employee where fname like 'J%'

UNION

Select ESSN from Dependent where sex = 'M';

Union Compatibility
1. # of columns in result of both queries must be same
2. Domain/datatype of corrosponding pairs of columsn must be same

Select SSN from Employee where fname like 'J%'

INTERSECT

Select ESSN from Dependent where sex = 'M';

Select SSN from Employee where fname like 'J%'

EXCEPT

Select ESSN from Dependent where sex = 'M';

Intersection in MYSQL

Select SSN from Employee where fname like 'J%' AND SSN IN(

Select ESSN from Dependent where sex = 'M');

Except in MySQL

Select SSN from Employee where fname like 'J%' AND SSN NOT IN(

Select ESSN from Dependent where sex = 'M')

Selection from Multiple Tables

Cross Product

Find dependents of female employees

E.fname, E.lname, E.ssn

Select ⎰ , D.* from Employees E , Dependent-D
where E.sex = 'F' and E.ssn = D.ESSN

FEMALE_EMPS

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
Alicia	J	Zelaya	999887777	1968-07-19	3321Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291Berry, Bellaire, TX	F	43000	888665555	4
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

EMP_DEPENDENTS

Fname	Lname	Ssn	Essn	Dependent_name	Sex	Bdate	...
Alicia	Zelaya	999887777	333445555	Alice	F	1986-04-05	...
Alicia	Zelaya	999887777	333445555	Theodore	M	1983-10-25	...
Alicia	Zelaya	999887777	333445555	Joy	F	1958-05-03	...
Alicia	Zelaya	999887777	987654321	Abner	M	1942-02-28	...
Alicia	Zelaya	999887777	123456789	Michael	M	1988-01-04	...
Alicia	Zelaya	999887777	123456789	Alice	F	1988-12-30	...
Alicia	Zelaya	999887777	123456789	Elizabeth	F	1967-05-05	...
Jennifer	Wallace	987654321	333445555	Alice	F	1986-04-05	...
Jennifer	Wallace	987654321	333445555	Theodore	M	1983-10-25	...
Jennifer	Wallace	987654321	333445555	Joy	F	1958-05-03	...
Jennifer	Wallace	987654321	987654321	Abner	M	1942-02-28	...
Jennifer	Wallace	987654321	123456789	Michael	M	1988-01-04	...
Jennifer	Wallace	987654321	123456789	Alice	F	1988-12-30	...
Jennifer	Wallace	987654321	123456789	Elizabeth	F	1967-05-05	...
Joyce	English	453453453	333445555	Alice	F	1986-04-05	...
Joyce	English	453453453	333445555	Theodore	M	1983-10-25	...
Joyce	English	453453453	333445555	Joy	F	1958-05-03	...
Joyce	English	453453453	987654321	Abner	M	1942-02-28	...
Joyce	English	453453453	123456789	Michael	M	1988-01-04	...
Joyce	English	453453453	123456789	Alice	F	1988-12-30	...
Joyce	English	453453453	123456789	Elizabeth	F	1967-05-05	...

24/5/23

If you use MultiTable Select in SQL
in any Assignment, Quiz, Final Exam, lab
after Today you will be awarded 0 (Zero)
marks

JOIN

Select E.fname, E.lname, E.SSN, D.* from
Employee E JOIN Dependent D
on E.SSN = D.ESSN
where E.sex = 'F';

1. Show the details of the departments located in Houston,
   Dept Join Dept Locs on D.Dnumber = DL.Dnumber

2. Show the names of the depts and the names & salaries
   of male employees working in those depts.
   Dept Join Emp on D.Dnumber = E.Dno

3. Show the name of the employees along with name of their
   supervisors

4. Show the names of the dependents whose parents work on
   Some project located in Houston or Stafford
   Proj → Works On → Employee → Dependent

5. Show the names of the projects managed by the dept
   whose manager is James Borg.
   Emp → Dept →(manager) Proj

6. Show the names of female employees along with the name
   of the projects where they spent more than 10 hrs.
   Emp → Works On → Project

7. Show the names of the departments along with the name
   of their managers.

Select D.dname, E.fname, E.lname from
Department D JOIN Employee E
on D.mgr_ssn = E.SSN;

Select E.* from Employee E JOIN Dependent D
on E.SSN = D.ESSN
where D.sex = 'M' and E.sex = 'M';

Select * from Employee where SSN IN
(Select ESSN from Dependent where sex = 'M')
and sex = 'M'

Show details of Employees who have a dependent of same
gender as the Employee

Select E.* from Employee E JOIN Dependent D
on E SSN = D.ESSN
where E sex = D.sex ;


```
For Each Employee E                      For Each Emp E
{                                          if E.sex = 'f'
{ if E.sex = 'M'                           for Each Dep D
        For Each Dependent D                 if E.SSN = D.ESSN
          if E.ssn = D.Essn                    if D.Sex = f
          {                                      print E
            if D.Sex = E sex
            {
              print E
            }
          }
        }
      }
}
```
.

Select * from Employee where sex = 'M' and
SSN IN (Select ESSN from Dependent where sex: 'M')

UNION

Select * from Employee where sex = 'F' and
SSN IN (Select ESSN from Dependent where sex: 'F')

Co Related Query _____

Select E.* from Employee E
where E.SSN IN (Select D.ESSN from Dependent D
                 where D.Sex = E.Sex
                 and D.ESSN = ESSN)

Select E.* from Employee E
where E.SSN NOT EXISTS (Select D.* from Dependent D
                 where D.Sex = E.Sex
                 and D.ESSN = ESSN)

Show the details of the employees and their male dependents

Inner

Select E.*, D.* from Employee E JOIN Dependent D

on E.SSN = D.ESSN

where D.Sex = 'M';

Show the details of all ⌐male⌐ employees and ~~only~~ their ~~male~~ dependent

outer

Select E.*, D.* from Employee E LEFT JOIN Dependent D
on E.SSN = D.ESSN
where E.Sex = 'M';

~~Left~~ [Outer] Join → Display all rows from left table and
only matching rows from right.

Right [Outer] Join → Display all rows from right and
only matchi rows from left.

Full [Outer] Join → Display all rows from left & right table
rows are matched where possible else
columns are NULL.

```
int count = 0
    Students[] allStudents;
        for ( i=0 ; i< allStudents.size() ; i++)
        {
                Student s = allStudents[i];
                if (s. isGlasses == true)
                {
                        print(s);        count ++
                }
        }
    }
        print (count);
```

Aggregate Functions

1. Count
2. Sum
3. Min
4. Max
5. Avg

How many female Employees

```sql
SELECT count(ssn) FROM Employeewhere Sex = 'F';
```

What is total salary paid to female employees

```sql
SELECT sum(Salary) FROM Employee where Sex = 'F';
```

What is total salary paid to dept managers

```sql
Select SUM(Salary) FROM Employee where SSN In
(Select mgr_ssn from Department);
```

Avg salary of dept manager

```sql
Select AVG(Salary) FROM Employee where SSN In
(Select mgr_ssn from Department);
```

Min salary for dept manager

```sql
Select MIN(Salary) FROM Employee where SSN In
(Select mgr_ssn from Department);
```

Select AVG(Salary), MIN(Salary), MAX(Salary), Count(SSN) FROM Employee where SSN In
(Select mgr_ssn from Department);

Avg salary of dno 5

Select   Avg(Salary)   from   Employee   where   dno = 5;

Group by

Select Avg(salary), Count(ssn), dno from Employee where Sex = 'M'
group by dno;

Join   Grouped data with rest of table

Select Count(S.SSN), S.Super_ssn, E.Fname, E.Lname, E.SSn
from Employee S LEFT JOIN EMPLOYEE E ON S.Super_SSN = E.SSN
group by S.super_ssn;

Multiple   groupings

Select Count(SSN), AVG(Salary), dno, sex from Employee
group by dno, sex;

Conditions   &   Aggregate   Values

1. Condition contain Aggregate value
   Find   details of the employees who earn more than
   the highest paid female employee

Select * from Employee
where Salary > (Select MIN(Salary) from Employee where Sex = 'F');
                                      max

2. Condition on aggregate function
   Show dept no where avg salary of male employee is
   more than 35k
                35

                                                → condition on data

Select avg(salary), dno from Employee where Sex = 'M' group by dno
having avg(salary) > 35000;

                    ↳ condition on aggregate functions