Agenda

```
- DQL(Select)
- Computed Columns
- Distinct
- LIMIT
- Order By
- WHERE clause
- Relational Operators
- IN,BETWEEN Operator
- DML- UPDATE,DELETE
- DDL- DROP,TRUNCATE
- DUAL
```

DQL (Data Query Language)

```
- Select
```

```sql
USE classwork_db;

SHOW TABLES;

-- Display all the data from our tables inside classwork_db
SELECT * FROM bonus;
SELECT * FROM books;
SELECT * FROM dept;
SELECT * FROM dummy;
SELECT * FROM emp;
SELECT * FROM salgrade;

--Display empno,ename and sal of the employees
SELECT empno,ename,sal FROM emp;

SELECT sal,ename,empno FROM emp;
```

Adding/Copying Data from one table to another

```sql
--Insert into dummy table all the empno from emp table
INSERT INTO dummy SELECT empno FROM emp;
```

Computed Columns

```
-- display all the emps with their name,sal and DA(sal * 0.5)
SELECT * FROM emp;
SELECT ename,sal FROM emp;
SELECT ename,sal,sal*0.5 FROM emp;

-- display all the emps with their name,sal and DA(sal * 0.5) and Total salary
(sal + DA)
SELECT ename,sal,sal*0.5 FROM emp;

SELECT ename,sal,sal*0.5, sal+(sal*0.5) FROM emp;

SELECT ename,sal,sal*0.5 AS DA, sal+(sal*0.5) FROM emp;
SELECT ename,sal,sal*0.5 AS DA, sal+(sal*0.5) AS TS FROM emp;

SELECT ename,sal,sal*0.5 DA, sal+(sal*0.5) TS FROM emp;

SELECT ename,sal,sal*0.5 `DA`, sal+(sal*0.5) `Total Salary` FROM emp;

-- display employee no,name and their dept names
(10=ACCOUNTING,20=RESEARCH,30=SALES)
SELECT empno,ename,deptno FROM emp;

SELECT empno,ename,deptno,CASE
WHEN deptno=10 THEN 'ACCOUNTING'
WHEN deptno=20 THEN 'RESEARCH'
WHEN deptno=30 THEN 'SALES'
END
FROM emp;

SELECT empno,ename,deptno,CASE
WHEN deptno=10 THEN 'ACCOUNTING'
WHEN deptno=20 THEN 'RESEARCH'
WHEN deptno=30 THEN 'SALES'
END AS dname
FROM emp;
```

Distinct

```
--display unique job profiles from emp
SELECT job FROM emp;
SELECT DISTINCT job FROM emp;


--display unique deptno from emp
SELECT DISTINCT deptno FROM emp;

--display unique jobs of all dept from emp
SELECT job,deptno FROM emp;
```

```
SELECT DISTINCT job,deptno FROM emp;
SELECT DISTINCT deptno,job FROM emp;
```

Limit

```
-- display only 5 rows from emp table.
SELECT * FROM emp LIMIT 5;
SELECT empno,ename,sal FROM emp LIMIT 5;

--skip first 3 emps and then dispaly 5 emps
SELECT * FROM emp LIMIT 3,5;
```

Order By

```
-- display emp and their sal arranged in ascending order
SELECT ename,sal FROM emp;
SELECT ename,sal FROM emp ORDER BY sal;

-- display emp and their sal arranged in descending order
SELECT ename,sal FROM emp ORDER BY sal DESC;

-- display all the deptno and job from emp in ascending order
SELECT deptno,job FROM emp;
SELECT deptno,job FROM emp ORDER BY deptno;
SELECT deptno,job FROM emp ORDER BY deptno,job;

-- display all the deptno in ascending order and job in descending order from emp
SELECT deptno,job FROM emp ORDER BY deptno ,job DESC;

-- display all the deptno,job,ename from emp in ascending order
-- TODO
```

Practice Examples

```
-- display  top 3 emps based on salary
SELECT ename,job,sal FROM emp;
SELECT ename,job,sal FROM emp ORDER BY sal DESC
SELECT ename,job,sal FROM emp ORDER BY sal DESC LIMIT 3;

-- dispaly emp whose name is last in alphabetical order
SELECT ename FROM emp ORDER BY ename DESC;
SELECT ename FROM emp ORDER BY ename DESC LIMIT 1;

-- sort the emps based on comm and observe the output
SELECT * FROM emp ORDER BY comm;
```

```sql
-- Display emp with lowsest salary
SELECT ename,sal FROM emp;
SELECT ename,sal FROM emp ORDER BY sal;
SELECT ename,sal FROM emp ORDER BY sal LIMIT 1;

-- Display emp with 3rd lowsest salary
SELECT ename,sal FROM emp;
SELECT ename,sal FROM emp ORDER BY sal;
SELECT ename,sal FROM emp ORDER BY sal LIMIT 2,1;

-- Display emp with 2nd highest salary
SELECT ename,sal FROM emp;
SELECT ename,sal FROM emp ORDER BY sal DESC;
SELECT ename,sal FROM emp ORDER BY sal DESC LIMIT 1,1;

-- Display all the emps in asc order based on DA
SELECT ename,sal,sal*0.5 FROM emp;
SELECT ename,sal,sal*0.5 FROM emp ORDER BY sal*0.5;
SELECT ename,sal,sal*0.5 DA FROM emp ORDER BY sal*0.5;
SELECT ename,sal,sal*0.5 DA FROM emp ORDER BY DA;
SELECT ename,sal,sal*0.5 DA FROM emp ORDER BY 3;
```

Where Clause

```sql
-- dispaly all emps of dept 20
SELECT *  FROM emp WHERE deptno=20;

-- display all the emps with sal more than 2000
SELECT * FROM emp WHERE sal > 2000;

-- dispaly all emps with job as ANALYST
SELECT * FROM emp WHERE job="ANALYST";

--display all emps who are not in dept 30;
SELECT *  FROM emp WHERE deptno!=30;
SELECT *  FROM emp WHERE deptno<>30;

--display all emps who are not SALESMAN;
SELECT * FROM emp WHERE job != "SALESMAN";
SELECT * FROM emp WHERE job <> "SALESMAN";
SELECT * FROM emp WHERE NOT job = "SALESMAN";

-- display all emps who are analyst and manager
SELECT ename,job FROM emp;
SELECT ename,job FROM emp WHERE job="ANALYST";
SELECT ename,job FROM emp WHERE job="MANAGER";
SELECT ename,job FROM emp WHERE job="ANALYST" OR job="MANAGER";
SELECT ename,job FROM emp WHERE job IN("ANALYST","MANAGER");

-- display emp having sal in range of 2000 to 3500
```

```sql
SELECT * FROM emp WHERE sal>2000 AND sal<3500;

-- display emp having sal in range of 2000 to 3000
SELECT * FROM emp WHERE sal>2000 AND sal<3000;
SELECT * FROM emp WHERE sal>=2000 AND sal<=3000;
SELECT * FROM emp WHERE sal BETWEEN 2000 AND 3000;

-- display all emps hired in 1982;
SELECT * FROM emp WHERE hire>="1982-01-01" AND hire<="1982-12-31";
SELECT * FROM emp WHERE hire BETWEEN "1982-01-01" AND "1982-12-31";

-- dispaly all emp having comm as null.
SELECT * FROM emp WHERE comm=NULL; --will not work
SELECT * FROM emp WHERE comm IS NULL;
SELECT * FROM emp WHERE comm <=> NULL;

-- dispaly all emp having comm as not null.
SELECT * FROM emp WHERE comm IS NOT NULL;

INSERT INTO emp(ename) VALUES("B"),("K"),("J");
--Display all emps whose name start with B to J
SELECT * FROM emp WHERE ename BETWEEN 'B' AND 'J';

SELECT * FROM emp WHERE ename>= 'B' AND ename<'K';

SELECT * FROM emp WHERE ename BETWEEN 'B' AND 'K' and ename!="K";
```

Like

```
1. % -> Any no of occurance
2. _ -> Single no of occurance
```

```sql
-- display all the emps starting with letter M
SELECT * FROM emp WHERE ename>"M" and ename<"N";
SELECT * FROM emp WHERE ename LIKE "M%";

-- display all the emps ending with letter H
SELECT * FROM emp WHERE ename LIKE "%H";

-- display all emps having letter U in their ename
SELECT * FROM emp WHERE ename LIKE "%U%";

-- display all emps having letter A twice in ename
SELECT * FROM emp WHERE ename LIKE "%A%A%";

-- Display all emps whose name start with S to Z
SELECT * FROM emp WHERE ename BETWEEN 'S' AND 'Z';
SELECT * FROM emp WHERE ename BETWEEN 'S' AND 'Z' OR ename LIKE "Z%";
```

```
-- display emp consisting name of only 4 letters
SELECT * FROM emp WHERE ename LIKE "____";

-- display emp who is having letter R at 3rd position
SELECT * FROM emp WHERE ename LIKE"__R%";

-- display emp consisting name of only 4 letters and having letter R at 3rd
position
SELECT * FROM emp WHERE ename LIKE "__R_";
```

Practice Examples

```
-- display emp with highest salary betwen the range of 1000 to 2000;
SELECT * FROM emp WHERE sal BETWEEN 1000 AND 2000;
SELECT * FROM emp WHERE sal BETWEEN 1000 AND 2000 ORDER BY sal DESC;
SELECT * FROM emp WHERE sal BETWEEN 1000 AND 2000 ORDER BY sal DESC LIMIT 1;

-- display clerk with min salary
SELECT * FROM emp WHERE job = "CLERK";
SELECT * FROM emp WHERE job = "CLERK" ORDER BY sal;
SELECT * FROM emp WHERE job = "CLERK" ORDER BY sal LIMIT 1;

--display 5th lowest sal from dept 20 and 30;
SELECT sal FROM emp WHERE deptno IN(20,30);
SELECT sal FROM emp WHERE deptno IN(20,30) ORDER BY sal;
SELECT DISTINCT sal FROM emp WHERE deptno IN(20,30) ORDER BY sal;
SELECT DISTINCT sal FROM emp WHERE deptno IN(20,30) ORDER BY sal LIMIT 4,1;
```

DML (DATA MANIPULATION LANGUAGE)

UPDATE

```
-- update the empno of emp with ename as B
UPDATE emp SET empno=7935 WHERE ename = "B";

-- provide sal of 1500 to emp J
UPDATE emp SET sal=1500 WHERE ename="J";

-- update the salary of all clerks by 200
SELECT * FROM emp WHERE job="CLERK";
UPDATE emp SET sal = sal+200 WHERE job="CLERK";

UPDATE emp SET sal = 3000 WHERE mgr=7698;
SELECT DISTINCT sal FROM emp WHERE deptno IN(20,30) ORDER BY sal LIMIT 4,1;
```

Delete

```
-- delete the emp named as k;
DELETE FROM emp WHERE ename = "K";

-- What will Happen??

DELETE FROM emp WHERE job="CLERK";-- Will delete all emps having job as clerk

DELETE FROM emp WHERE comm IS NULL;

DELETE FROM emp;
-- data can be recovered if transactions are started

TRUNCATE dummy;
-- data cannot be recovered

DROP TABLE emp;
DROP TABLE dummy;
-- Delete the entire table structure
```

Dual

```
- Imagianary/Temporary Table
```