Ex.No.9	SET OPERATIONS AND AGGEREGATE FUNCTIONS
30.04.2025	

AIM:

To perform various set operations, aggregate functions, group by and having clause on the relational database.

CREATE TABLE

```
SQL> create table student(id varchar2(10),name varchar2(10),city varchar2(10));

Table created.

SQL> create table std_info(id varchar2(10),name varchar2(10),city varchar2(10));

Table created.

SQL> create table emp4(name varchar2(10),dept varchar2(10),address varchar2(10),salary number(10));

Table created.
```

INSERTING VALUES

```
SQL> insert into student values('it01','kabesh','namakkal');

1 row created.

SQL> insert into student values('it02','kamalesh','erode');

1 row created.

SQL> insert into student values('it03','karthik','chennai');

1 row created.

SQL> insert into student values('it04','jegan','salem');

1 row created.

SQL> insert into student values('it05','sanjay','madurai');

1 row created.

SQL> insert into std_info values('ece01','kavin','trichy');

1 row created.

SQL> insert into std_info values('ece02','babu','mumbai');

1 row created.
```

```
SQL> insert into std_info values('ece03','shangav','ooty');

1 row created.

SQL> insert into std_info values('ece04','iyyappan','coimbatore');

1 row created.

SQL> insert into emp4 values('kumar','it','erode',60000);

1 row created.

SQL> insert into emp4 values('vijay','eee','chennai',50000);

1 row created.

SQL> insert into emp4 values('ajith','ece','coimbatore',45000);

1 row created.

SQL> insert into emp4 values('kamal','cse','mumbai',78000);

1 row created.

SQL> insert into emp4 values('siva','ft','tirupur',30000);

1 row created.

SQL> commit;

Commit complete.
```

UNION KEYWORD

```
SQL> select id,name from student union select id,name from std_info;

ID         NAME ----
------ ---------- ece01
kavin ece02      babu
ece03      shangav
ece04      iyyappan
it01       kabesh
it02       kamalesh
it03       karthik
it04       jegan
it05       sanjay

9 rows selected.
```

UNION ALL KEYWORD

```
SQL> select id,name,city from student union all select id,name,city from std_info;
```

ID NAME CITY ---
------- ---------- ---------- it01
kabesh namakkal it02
kamalesh erode it03
karthik chennai
it04 jegan salem

it05 sanjay madurai
ece01 kavin trichy
ece02 babu mumbai
ece03 shangav ooty
ece04 iyyappan coimbatore

9 rows selected.

INTERSECT KEYWORD

SQL> select * from student intersect select * from std_info;

no rows selected

MINUS KEYWORD

SQL> select * from std_info minus select * from student;

ID NAME CITY ------
---- ---------- ---------- ece01
kavin trichy ece02 babu
mumbai ece03 shangav
ooty ece04 iyyappan
coimbatore

AGGREGATE FUNCTIONS

SQL> select max(salary) from emp4;

MAX(SALARY)

78000

SQL> select min(salary) from emp4;

MIN(SALARY)

30000

SQL> select avg(salary) from emp4;

AVG(SALARY)

52600

SQL> select sum(salary) from emp4;

```
SUM(SALARY)
-------------------
263000

SQL> select count(salary) from emp4;

COUNT(SALARY)
--------------------
5
```

SECOND MAXIMUM SALARY

```
SQL> select max(salary) from emp4 where salary not in (select max(salary) from emp4);

MAX(SALARY)
-------------------
60000
```

SECOND MINIMUM SALARY

```
SQL> select min(salary) from emp4 where salary not in (select min(salary) from emp4);

MIN(SALARY)
--------------------
    45000
```

AGGREGATE FUNCTIONS WITH GROUPBY AND HAVING:

GROUP BY

```
SQL> select dept,sum(salary) from emp4 group by dept;

DEPT       SUM(SALARY)
----------  -------------------
IT             60000
EEE            50000
CSE            78000
ECE            45000
FT             30000

SQL> select dept,sum(salary) from emp4 group by dept having avg(salary)>43000;

DEPT       SUM(SALARY)
----------  --------------------
IT             60000
EEE            50000
CSE            78000
ECE            4500
```

DISTINCT

```
SQL> select distinct dept from emp4;

DEPT
----------
IT
EEE
```

CSE
ECE
FT

TO FIND THE EMPLOYEES WHO EARN SALARY HIGHER THAN THE AVG SALARY OF THEIR CITY

SQL>select name from employee_info e where salary > (select avg(salary) from employee_info where address = e.address);

NAME

KARTHIK

TO FIND THE NAME OF THE PERSONS WHO HAVE HIGHER SALARY THAN THE AVERAGE SALARY OF THEIR DEPARTMENT

SQL>select name from employee_info e where salary > (select avg(salary) from employee_info where dept = e.dept);

NAME

KAVIN
KARTHI

CONTENTS	MARKS ALLOTED	MARKS OBTAINED
Aim,Algorithm,SQL,PL/SQL	30	
Execution and Result	20	
Viva	10	
Total	60	

RESULT

Thus various set operations, aggregate functions, group by and having clause operations are executed.