

ASSIGNMENT NO-23 DEMONSTRATE TO INSERT,UPDATE AND DELETE RECORDS IN TABLE

```
SQL> create table emp62
```

```
2 (eid int,  
3 ename char(20),  
4 eadd char(30),  
5 emob long);
```

Table created.

```
SQL> insert into emp62
```

```
2 values(&eid,'&ename','&eadd','&emob');
```

Enter value for eid: 1

Enter value for ename: ram

Enter value for eadd: shirpur

Enter value for emob: 12345

```
old 2: values(&eid,'&ename','&eadd','&emob')
```

```
new 2: values(1,'ram','shirpur','12345')
```

1 row created.

.....

```
SQL> update emp62  
2 set eadd='imrd'  
3 where eid=3;
```

1 row updated.

```
SQL> delete from emp62  
2 where eid=5;
```

1 row deleted.

```
SQL> create table student  
2 (rollno int primary key,  
3 sname char(30),  
4 sgender char(7) check(sgender='male' or sgender='female'),  
5 scity char(10));
```

Table created.

```
SQL> create table exam  
2 (rollno int references student(rollno),  
3 Firstsubname char(20) not null,  
4 secondsubname char(20) not null,  
5 thirdsubname char(20) not null,  
6 forthsubname char(20) not null);
```

Table created.

```
SQL> create table fees  
2 (rollno int references student(rollno),  
3 totalfees int,  
4 paidfees int,  
5 duefees int);
```

Table created.

```
SQL> insert into student  
2 values(&rollno,'&sname','&sgender','&scity');  
Enter value for rollno: 101  
Enter value for sname: Umesh  
Enter value for sgender: male  
Enter value for scity: shirpur
```

```
old  2: values(&rollno,'&sname','&sgender','&scity')
new  2: values(101,'Umesh','male','shirpur')
```

1 row created.

```
SQL> /
Enter value for rollno: 102
Enter value for sname: aniket
Enter value for sgender: male
Enter value for scity: karvand
old  2: values(&rollno,'&sname','&sgender','&scity')
new  2: values(102,'aniket','male','karvand')
```

1 row created.

```
SQL> /
Enter value for rollno: 103
Enter value for sname: neel
Enter value for sgender: male
Enter value for scity: surat
old  2: values(&rollno,'&sname','&sgender','&scity')
new  2: values(103,'neel','male','surat')
```

1 row created.

```
SQL> /
Enter value for rollno: 104
Enter value for sname: geeta
Enter value for sgender: female
Enter value for scity: boradi
old  2: values(&rollno,'&sname','&sgender','&scity')
```

```
new 2: values(104,'geeta','female','boradi')
```

1 row created.

```
SQL> /
```

Enter value for rollno: 105

Enter value for sname: kajal

Enter value for sgender: female

Enter value for scity: holi

```
old 2: values(&rollno,'&sname','&sgender','&scity')
```

```
new 2: values(105 , 'kajal','female','holi')
```

1 row created.

```
SQL> /
```

Enter value for rollno: 106

Enter value for sname: nalini

Enter value for sgender: female

Enter value for scity: shirpur

```
old 2: values(&rollno,'&sname' , '&sgender','&scity')
```

```
new 2: values(106,'nalini','female','shirpur')
```

1 row created.

```
SQL> insert into fees
```

```
2 values(&rollno,&totalfees,&paidfees,&duefees);
```

Enter value for rollno: 101

Enter value for totalfees: 50000

Enter value for paidfees: 25000

Enter value for duefees: 25000

```
old 2: values(&rollno,&totalfees,&paidfees,&duefees)
```

```
new 2: values(101,50000,25000,25000)
```

1 row created.

```
SQL> /
```

```
Enter value for rollno: 102
```

```
Enter value for totalfees: 50000
```

```
Enter value for paidfees: 30000
```

```
Enter value for duefees: 20000
```

```
old 2: values(&rollno,&totalfees,&paidfees,&duefees)
```

```
new 2: values(102,50000,30000,20000)
```

1 row created.

```
SQL> /
```

```
Enter value for rollno: 103
```

```
Enter value for totalfees: 50000
```

```
Enter value for paidfees: 24000
```

```
Enter value for duefees: 26000
```

```
old 2: values(&rollno,&totalfees,&paidfees,&duefees)
```

```
new 2: values(103,50000,24000,26000)
```

1 row created.

```
SQL> /
```

```
Enter value for rollno: 104
```

```
Enter value for totalfees: 50000
```

```
Enter value for paidfees: 26000
```

```
Enter value for duefees: 24000
```

```
old 2: values(&rollno,&totalfees,&paidfees,&duefees)
```

```
new 2: values(104,50000,26000,24000)
```

1 row created.

```
SQL> insert into exam  
  2 values(&rollno,'&Firstsubname','&secondsubname','&thirdsubname','&forthsubname');  
Enter value for rollno: 101  
Enter value for firstsubname: ISA  
Enter value for secondsubname: DS  
Enter value for thirdsubname: c#.net  
Enter value for forthsubname: RDBMS  
old  2: values(&rollno,'&Firstsubname','&secondsubname','&thirdsubname','&forthsubname')  
new  2: values(101,'ISA','DS','c#.net','RDBMS')
```

1 row created.

```
SQL> /  
Enter value for rollno: 102  
Enter value for firstsubname: ISA  
Enter value for secondsubname: ACCOUNT  
Enter value for thirdsubname: DA  
Enter value for forthsubname: DBA  
old  2: values(&rollno,'&Firstsubname','&secondsubname','&thirdsubname','&forthsubname')  
new  2: values(102,'ISA','ACCOUNT','DA','DBA')
```

1 row created.

```
SQL> /  
Enter value for rollno: 103  
Enter value for firstsubname: IBM  
Enter value for secondsubname: ACCOUNT  
Enter value for thirdsubname: STOCK
```

Enter value for forthsubname: MOB

old 2: values(&rollno,'&Firstsubname','&secondsubname','&thirdsubname','&forthsubname')
new 2: values(103,'IBM','ACCOUNT','STOCK','MOB')

1 row created.

SQL> /

Enter value for rollno: 104

Enter value for firstsubname: MA

Enter value for secondsubname: SCS

Enter value for thirdsubname: BEAUTY

Enter value for forthsubname: MEAKUP

old 2: values(&rollno,'&Firstsubname','&secondsubname','&thirdsubname','&forthsubname')
new 2: values(104,'MA','SCS','BEAUTY','MEAKUP')

1 row created.

SQL> /

Enter value for rollno: 105

Enter value for firstsubname: DRAMA

Enter value for secondsubname: PHTOGRAPHY

Enter value for thirdsubname: SHOOT

Enter value for forthsubname: FOOD

old 2: values(&rollno,'&Firstsubname','&secondsubname','&thirdsubname','&forthsubname')
new 2: values(105,'DRAMA','PHTOGRAPHY','SHOOT','FOOD')

1 row created.

SQL> /

Enter value for rollno: 106

Enter value for firstsubname: RS

Enter value for secondsubname: CA

Enter value for thirdsubname: MA

Enter value for forthsubname: BS

old 2: values(&rollno,'&Firstsubname','&secondsubname','&thirdsubname','&forthsubname')

new 2: values(106,'RS','CA','MA','BS')

1 row created.

JOIN QUERY :-

SQL> select * from fees inner join student on fees.rollno=student.rollno;

ROLLNO	TOTALFEES	PAIDFEES	DUEFEES	ROLLNO
-----	-----	-----	-----	-----

SNAME	SGENDER	SCITY
-----	-----	-----

101	50000	25000	25000	101
-----	-------	-------	-------	-----

Umesh	male	shirpur
-------	------	---------

102	50000	30000	20000	102
-----	-------	-------	-------	-----

aniket	male	karvand
--------	------	---------

103	50000	24000	26000	103
-----	-------	-------	-------	-----

neel	male	surat
------	------	-------

ROLLNO	TOTALFEES	PAIDFEES	DUEFEES	ROLLNO
-----	-----	-----	-----	-----

SNAME	SGENDER	SCITY
-----	-----	-----

104	50000	26000	24000	104
-----	-------	-------	-------	-----

geeta	female	boradi
-------	--------	--------

```
SQL> select * from fees left join student on fees.rollno=student.rollno;
```

ROLLNO	TOTALFEES	PAIDFEES	DUEFEES	ROLLNO
SNAME	SGENDER	SCITY		
101	50000	25000	25000	101
Umesh	male	shirpur		
102	50000	30000	20000	102
aniket	male	karvand		
103	50000	24000	26000	103
neel	male	surat		

ROLLNO	TOTALFEES	PAIDFEES	DUEFEES	ROLLNO
SNAME	SGENDER	SCITY		
104	50000	26000	24000	104
geeta	female	boradi		

```
SQL> select * from fees right join student on fees.rollno=student.rollno;
```

ROLLNO	TOTALFEES	PAIDFEES	DUEFEES	ROLLNO
SNAME	SGENDER	SCITY		

```
-----  
      101    50000    25000    25000    101  
Umesh           male   shirpur
```

```
      102    50000    30000    20000    102  
aniket          male   karvand
```

```
      103    50000    24000    26000    103  
neel            male   surat
```

```
ROLLNO TOTALFEES PAIDFEES DUEFEES ROLLNO
```

```
-----  
SNAME           SGENDER SCITY
```

```
      104    50000    26000    24000    104  
geeta          female  boradi
```

```
      106  
nalini          female shirpur
```

```
      105  
kajal          female holi
```

6 rows selected.

```
SQL> select * from fees full join student on fees.rollno=student.rollno;
```

```
ROLLNO TOTALFEES PAIDFEES DUEFEES ROLLNO
```

SNAME	SGENDER	SCITY		
101	50000	25000	25000	101
Umesh	male	shirpur		
102	50000	30000	20000	102
aniket	male	karvand		
103	50000	24000	26000	103
neel	male	surat		

ROLLNO	TOTALFEES	PAIDFEES	DUEFEES	ROLLNO
104	50000	26000	24000	104
geeta	female	boradi		
105				
kajal	female	holi		
106				
nalini	female	shirpur		

6 rows selected.

NEASTED QUREY:-

Q). Display First Subject Name , Second Subject Name,Third Subject Name of student whos due fees is ≥ 25000 .

```
SQL> select firstsubname,secondsubname,thirdsubname from exam  
2 where rollno in(select rollno from student  
3           where rollno in (select rollno from fees  
4           where duefees<=25000));
```

FIRSTSUBNAME	SECONDSUBNAME	THIRDSUBNAME
--------------	---------------	--------------

ISA	DS	c#.net
IBM	ACCOUNT	STOCK

Assignment No – 19 Demonstrate use of Group By and Having clause

1) SQL> create table customer

```
2 (
3   customerid int,
4   customername char(20),
5   country char(20));
```

2) SQL> insert into customer

```
2 values(&customerid,'&customername','&country');
```

Enter value for customerid: 1

Enter value for customername: aniket

Enter value for country: india

```
old 2: values(&customerid,'&customername','&country')
```

```
new 2: values(1,'aniket','india')
```

1 row created.

3) SQL> select count(customerid),country from customer group by country;

COUNT(CUSTOMERID) COUNTRY

1 india

1 america

4) SQL> select count(customerid),country from customer

```
2 group by country
```

```
3 having count(customerid)>5;
```

64 Pathak Varad Anant .

Assignment – aggregate function..

SQL> select * from sal64;

EMPID	BASIC	HPA	DA
101	21000	21000	23000
102	22000	23000	24000
103	22000	23000	24000
104	21000	23000	24000
105	26000	27000	25000

SQL> select max(basic) from sal64;

MAX(BASIC)

26000

SQL> select min(basic) from sal64;

MIN(BASIC)

21000

```
SQL> select count(basic) from sal64;
```

```
COUNT(BASIC)
```

```
-----
```

```
5
```

```
SQL> select count (da)from sal64
```

```
2 where da<= 25000;
```

```
COUNT(DA)
```

```
-----
```

```
5
```

```
SQL> select avg (basic)from sal64;
```

```
AVG(BASIC)
```

```
-----
```

```
22400
```

```
SQL> select sum (basic)from sal64;
```

```
SUM(BASIC)
```

```
-----
```

```
112000
```

```
SQL> select max(basic)from sal64;
```

MAX(BASIC)

26000

```
SQL> select max(da)from sal64;
```

MAX(DA)

25000

```
SQL> select max(hpa)from sal64;
```

MAX(HPA)

27000

```
SQL> create table student  
2 (rollno int primary key,  
3 sname char(30),  
4 sgender char(7) check(sgender='male' or sgender='female'),  
5 scity char(10));
```

Table created.

```
SQL> create table fees  
2 (rollno int references student(rollno),  
3 totalfees int,  
4 paidfees int,  
5 duefees int);
```

Table created.

```
SQL> insert into student  
2 values(&rollno,'&sname','&sgender','&scity');  
Enter value for rollno: 101  
Enter value for sname: umesh  
Enter value for sgender: male  
Enter value for scity: shirpur  
old  2: values(&rollno,'&sname','&sgender','&scity')  
new  2: values(101,'umesh','male','shirpur')
```

1 row created.

```
SQL> /  
Enter value for rollno: 102  
Enter value for sname: aniket
```

Enter value for sgender: male

Enter value for scity: shirpur

old 2: values(&rollno,'&sname','&sgender','&scity')

new 2: values(102,'aniket','male','shirpur')

1 row created.

SQL> /

Enter value for rollno: 103

Enter value for sname: kajal

Enter value for sgender: female

Enter value for scity: shirpur

old 2: values(&rollno,'&sname','&sgender','&scity')

new 2: values(103 , 'kajal','female','shirpur')

1 row created.

SQL> /

Enter value for rollno: 104

Enter value for sname: arpita

Enter value for sgender: female

Enter value for scity: shirpur

old 2: values(&rollno,'&sname','&sgender','&scity')

new 2: values(104,'arpita','female','shirpur')

1 row created.

SQL> /

Enter value for rollno: 105

Enter value for sname: vipul

Enter value for sgender: male

Enter value for scity: dhule

```
old 2: values(&rollno,'&sname','&sgender','&scity')  
new 2: values(105,'vipul','male','dhule')
```

1 row created.

SQL> insert into fees

```
2 values(&rollno,&totalfees,&paidfees,&duefees);
```

Enter value for rollno: 101

Enter value for totalfees: 50000

Enter value for paidfees: 30000

Enter value for duefees: 20000

```
old 2: values(&rollno,&totalfees,&paidfees,&duefees)
```

```
new 2: values(101,50000,30000,20000)
```

1 row created.

SQL> /

Enter value for rollno: 102

Enter value for totalfees: 50000

Enter value for paidfees: 40000

Enter value for duefees: 10000

```
old 2: values(&rollno,&totalfees,&paidfees,&duefees)
```

```
new 2: values(102,50000,40000,10000)
```

1 row created.

SQL> /

Enter value for rollno: 103

```
Enter value for totalfees: 50000
Enter value for paidfees: 20000
Enter value for duefees: 30000
old 2: values(&rollno,&totalfees,&paidfees,&duefees)
new 2: values(103,50000,20000,30000)
```

1 row created.

```
SQL> /
Enter value for rollno: 104
Enter value for totalfees: 50000
Enter value for paidfees: 25000
Enter value for duefees: 25000
old 2: values(&rollno,&totalfees,&paidfees,&duefees)
new 2: values(104,50000,25000,25000)
```

1 row created.

```
SQL> /
Enter value for rollno: 105
Enter value for totalfees: 50000
Enter value for paidfees: 27000
Enter value for duefees: 23000
old 2: values(&rollno,&totalfees,&paidfees,&duefees)
new 2: values(105,50000,27000,23000)
```

1 row created.

In

```
SQL> select * from fees
```

```
2 where rollno in (select rollno from student  
3           where sname='umesh');
```

ROLLNO TOTALFEES PAIDFEES DUEFEES

```
-----  
101   50000   30000   20000
```

Or

```
SQL> select * from fees  
2 where duefees=30000 or paidfees=30000;
```

ROLLNO TOTALFEES PAIDFEES DUEFEES

```
-----  
101   50000   30000   20000  
103   50000   20000   30000
```

And

```
SQL> select * from fees  
2 where duefees=30000 and paidfees=20000;
```

ROLLNO TOTALFEES PAIDFEES DUEFEES

```
-----  
103   50000   20000   30000
```

Between

```
SQL> select * from fees  
2 where rollno between 101 and 103;
```

ROLLNO TOTALFEES PAIDFEES DUEFEES

```
-----  
101   50000   30000   20000
```

```
102 50000 40000 10000  
103 50000 20000 30000
```

Not

```
SQL> select * from fees  
2 where rollno in(select rollno from student  
3           where sname not like '%h%');
```

ROLLNO	TOTALFEES	PAIDFEES	DUFEES
102	50000	40000	10000
103	50000	20000	30000
104	50000	25000	25000
105	50000	27000	23000

Exists

```
SQL> select * from student  
2 where exists (select * from fees  
3           where totalfees=50000);
```

ROLLNO	SNAME	SGENDER	SCITY
101	umesh	male	shirpur
102	aniket	male	shirpur
103	kajal	female	shirpur
104	arpita	female	shirpur
105	vipul	male	dhule

Like

```
SQL> select * from fees  
2 where rollno in(select rollno from student  
3           where sname like '%h%');
```

ROLLNO	TOTALFEES	PAIDFEES	DUFEES
101	50000	30000	20000

```
SQL> create table student  
2 (sroll int primary key,  
3 sanme varchar(30),  
4 city varchar(30),  
5 gender char(7) check(gender='male' or gender='female'),  
6 class varchar(10),  
7 subject char(20));
```

Table created.

```
SQL> insert into student  
2 values(&sroll,'&sanme','&city','&gender','&class','&subject');  
Enter value for sroll: 101  
Enter value for sanme: Rahul  
Enter value for city: shirpur  
Enter value for gender: male  
Enter value for class: sybba  
Enter value for subject: account  
old  2: values(&sroll,'&sanme','&city','&gender','&class','&subject')  
new  2: values(101,'Rahul','shirpur','male','sybba','account')
```

1 row created.

```
SQL> /  
Enter value for sroll: 102  
Enter value for sanme: sumit
```

Enter value for city: Pachora

Enter value for gender: male

Enter value for class: syimca

Enter value for subject: DA

old 2: values(&scroll,'&sanme','&city','&gender','&class','&subject')

new 2: values(102,'sumit','Pachora','male','syimca','DA')

1 row created.

SQL> /

Enter value for scroll: 103

Enter value for sanme: manish

Enter value for city: rajkot

Enter value for gender: male

Enter value for class: sybms

Enter value for subject: fcl

old 2: values(&scroll,'&sanme','&city','&gender','&class','&subject')

new 2: values(103,'manish','rajkot','male','sybms','fcl')

1 row created.

SQL> /

Enter value for scroll: 104

Enter value for sanme: Saloni

Enter value for city: Nandurbar

Enter value for gender: female

Enter value for class: sybms

Enter value for subject: RRR

```
old 2: values(&sroll,'&sanme','&city','&gender','&class','&subject')
```

```
new 2: values(104,'Saloni','Nandurbar','female','sybms','RRR')
```

1 row created.

```
SQL> /
```

Enter value for sroll: 105

Enter value for sanme: Mohini

Enter value for city: Dhule

Enter value for gender: female

Enter value for class: sybca

Enter value for subject: RDBMS

```
old 2: values(&sroll,'&sanme','&city','&gender','&class','&subject')
```

```
new 2: values(105,'Mohini','Dhule','female','sybca','RDBMS')
```

1 row created.

```
SQL> update student
```

```
2 set class='ABCD';
```

5 rows updated.

```
SQL> update student
```

```
2 set subject='software engg';
```

5 rows updated.

Assignment 2:

Create table cricket_team

```
SQL> create table Cricket_team  
2 ( team_id int,  
3   t_name char(20),  
4   Captain char(20),  
5   Rank int  
6 );
```

Table created.

```
SQL> insert into Cricket_team  
2 values(&team_id,'&t_name','&Captain',&Rank);  
Enter value for team_id: 1  
Enter value for t_name: india  
Enter value for captain: virat kohli  
Enter value for rank: 1  
old 2: values(&team_id,'&t_name','&Captain',&Rank)  
new 2: values(1,'india','virat kohli',1)
```

1 row created.

```
SQL> select * from Cricket_team;
```

TEAM_ID	T_NAME	CAPTAIN	RANK
1	india	virat kohli	1
2	australia	aron finch	2
3	england	eoin morgan	3
4	new zealand	kane williamson	4
5	sri lanka	dimuth karunaratne	5

1. Find name of all captains having 'm' as 1st character:

```
SQL> select captain from cricket_team  
2 where captain like 'm%';
```

no rows selected

2. Find name of all captains having 'a' as 2nd character:

```
SQL> select captain from cricket_team  
2 where captain like '_a%';
```

CAPTAIN

kane williamson

3. Find all team whose rank between 1 and 3

```
SQL> select t_name , rank from cricket_team  
2 where rank >= 1 and rank <= 3;
```

T_NAME	RANK
india	1
australia	2
england	3

4. Demonstrate truncate and drop command.

```
SQL> truncate table cricket_team;
```

Table truncated.

```
SQL> select * from cricket_team;
```

no rows selected

```
SQL> drop table cricket_team;
```

Table dropped.

Assignment : 07

Q. Create Table Patient.

```
SQL> create table patient
```

```
2 (pid int,  
3 pname char(20),  
4 disease char(20),  
5 admitdate date);
```

Table created.

```
SQL> insert into patient
```

```
2 values(&pid,'&pname','&disease','&admitdate');
```

```
SQL> /
```

Enter value for pid: 10

Enter value for pname: ram

Enter value for disease: corona

Enter value for admitdate: 12/nov/2020

```
old 2: values(&pid,'&pname','&disease','&admitdate')
```

```
new 2: values(10,'ram','corona','12/nov/2020')
```

1 row created.

```
SQL> '
```

SP2-0042: unknown command "" - rest of line ignored.

SQL> /

Enter value for pid: 11

Enter value for pname: sham

Enter value for disease: typhoid

Enter value for admitdate: 1/jan/2018

old 2: values(&pid,'&pname','&disease','&admitdate')

new 2: values(11,'sham','typhoid','1/jan/2018')

1 row created.

1. Display Patient whose Admitted From 01/01/2018 to 01/02/2018.

SQL> select *from patient

2 where admitdate between '1/jan/2018' and '01/feb/2018';

PID	PNAME	DISEASE	ADMITDATE
11	sham	typhoid	01-JAN-18
13	john	fever	01-FEB-18

2. Display Patient with Disease ‘Maleria’ or ‘Typhoid’.

```
SQL> select pname from patient  
2 where (disease ='malaria') or (disease='typhoid');
```

PNAME

sham

jitesh

3. Change Patientname of Second Patient as ‘Amruta’.

```
SQL> update patient  
2 set pname='Amruta' where pname='sham';
```

1 row updated.

```
SQL> select *from patient;
```

PID	PNAME	DISEASE	ADMITDATE
10	ram	corona	12-NOV-20
11	Amruta	typhoid	01-JAN-18
12	jitesh	maleria	02-MAR-18
13	john	fever	01-FEB-18

14 nick

HIV

04-OCT-20

11. Create table and insert 10 records and perform ALTER operation.

Table creation

```
SQL> create table stud999
```

```
(sroll int,  
 sname char(20),  
 sadd char(20),  
 sgender char(6)  
);
```

Table created.

Inserting 10 records

```
SQL> insert into stud999  
values(&sroll,'&sname', '&sadd', '&sgender');  
  
Enter value for sroll: 1  
  
Enter value for sname: hitesh  
  
Enter value for sadd: shirpur  
  
Enter value for sgender: male  
  
old  2: values(&sroll,'&sname', '&sadd', '&sgender')  
new  2: values(1,'hitesh', 'shirpur', 'male')
```

1 row created.

SQL> /

Enter value for sroll: 2

Enter value for sname: jayesh

Enter value for sadd: dhule

Enter value for sgender: male

old 2: values(&sroll,'&sname', '&sadd', '&sgender')

new 2: values(2,'jayesh', 'dhule', 'male')

1 row created.

.....
.....

Showing inserted records

SQL> select * from stud999;

SROLL	SNAME	SADD	SGENDE
1	hitesh	shirpur	male
2	jayesh	dhule	male
3	bhatu	shirpur	male
4	bhupesh	mumbai	male
5	neel	jalgaon	male
6	gaurav	pune	male
7	priya	mumbai	female

8 supriya	pune	female
9 sahil	dhule	male
10 aniket	dhule	male

10 rows selected.

Alter operation

```
SQL> alter table stud999
```

```
    add smob int;
```

Table altered.

```
SQL> select * from stud999;
```

SROLL	SNAME	SADD	SGENDE	SMOB
1	hitesh	shirpur	male	
2	jayesh	dhule	male	
3	bhatu	shirpur	male	
4	bhupesh	mumbai	male	
5	neel	jalgaon	male	
6	gaurav	pune	male	
7	priya	mumbai	female	
8	supriya	pune	female	
9	sahil	dhule	male	

10 aniket dhule male

10 rows selected.

```
SQL> create synonym ep2 for emp1;
```

Synonym created.

```
SQL> select* from ep2;
```

EID	ENAME	EADD.	ESALARY
1	anjali	shirpur.	20000
2	saili	kharde.	25000
3	vijay	dhule.	21000
5.	divya	jalgoan.	30000

1) SQL> create table product_master
-> (pno int primary key,
-> pname char(20), > price int); Query OK, 0 rows
affected (0.057 sec)

2) SQL> create table order_detail
-> (orderno int ,
-> pno int references product_master (pno),
-> qty char (20),
-> rate int);
Query OK, 0 rows affected (0.075 sec)

SQL> insert into product_master
➔ Values(01,'dantkanti',100);
Query OK, 1 row affected (0.020 sec)

SQL> insert into product_master values(02,'pears',500);
Query OK, 1 row affected (0.002 sec)

SQL> insert into product_master values(03,'patanjali',700);
Query OK, 1 row affected (0.003 sec)

3) SQL> select pname,price from product_master where price=(select max(price) from product_master);

--	--

pname	price
-------	-------

--	--

patanjali	700
-----------	-----

--	--

1 row in set (0.003 sec)

10) Demonstrate Date Function in RDBMS

SQL> select current_date from dual;

CURRENT_D

04-APR-22

SQL> select current_timestamp from dual;

CURRENT_TIMESTAMP

04-APR-22 09.08.43.042000 AM +05:30

SQL> select extract(month from to_date('20/jan/2022'))from dual;

EXTRACT(MONTHFROMTO_DATE('20/JAN/2022'))

1

SQL> select extract(year from to_date('20/jan/2022'))from dual;

EXTRACT(YEARFROMTO_DATE('20/JAN/2022'))

2022

SQL> select extract(day from to_date('20/jan/2022'))from dual;

```
EXTRACT(DAYFROMTO_DATE('20/JAN/2022'))
```

20

```
SQL> select months_between(date'2022-04-03',date'2022-01-03')month_diff from dual;
```

```
MONTH_DIFF
```

3

Assignment no.22

SQL> select*from emp11;

EMPID	EMPNAME	EMPADD
101	lalita	nashik
102	sandhya	pune
103	nikita	aamdhde
104	divya	shirpur
105	vaishnavi	shirpur

SQL> create view v4 as select empname from emp11;

View created.

SQL> select*from v4;

EMPNAME
lalita
sandhya
nikita
divya
vaishnavi

SQL>

Assignment 25: Demonstrate number functions

```
SQL> create table sal62
```

```
2 (eid int,  
3 ename char(20),  
4 da int,  
5 basic int,  
6 ta int,  
7 hra int);
```

Table created.

```
SQL> insert into sal62
```

```
2 values(&eid,'&ename',&da,&basic,&ta,&hra);
```

Enter value for eid: 101

Enter value for ename: aish

Enter value for da: 2000.3

Enter value for basic: 3000

Enter value for ta: 234.43

Enter value for hra: 2345.9

```
old 2: values(&eid,'&ename',&da,&basic,&ta,&hra)
```

```
new 2: values(101,'aish',2000.3,3000,234.43,2345.9)
```

1 row created.

```
SQL> select abs(da)from sal62;
```

ABS(DA)

2000

```
SQL> select power(da,3)from sal62;
```

POWER(DA,3)

8000000000

```
SQL> select round(basic)from sal62;
```

ROUND(BASIC)

3000

```
SQL> select sqrt(ta)from sal62;
```

SQRT(TA)

15.2970585

```
SQL> select mod(da,2)from sal62;
```

MOD(DA,2)

0

SQL> select sign(da)from sal62;

SIGN(DA)

1

Assignment No. 6

Create table vehical

```
SQL> create table vehical
```

```
2 (vehicalno varchar(20),  
3 vehicalname char(20),  
4 type char(20),  
5 color char(10),  
6 cost int  
7 );
```

Table created.

Insert record's

```
SQL> insert into vehical
```

```
2 values('&vehicalno','&vehicalname','&type','&color','&cost);
```

Enter value for vehicalno: AS8919

Enter value for vehicalname: Discover

Enter value for type: petrol

Enter value for color: black

Enter value for cost: 90000

```
old 2: values('&vehicalno','&vehicalname','&type','&color','&cost)
```

```
new 2: values('AS8919','Discover','petrol','black',90000)
```

1 row created.

- Display all vehical in black color

SQL> select* from vehical

2 where color='black';

VEHICALNO	VEHICALNAME	TYPE	COLOR
-----------	-------------	------	-------

COST

AS8919	Discover	petrol	black
--------	----------	--------	-------

90000

BS9024	honda	petrol	black
--------	-------	--------	-------

95000

AS2156	hero	diesel	black
--------	------	--------	-------

80000

VEHICALNO	VEHICALNAME	TYPE	COLOR
-----------	-------------	------	-------

COST

DS3232	yamaha	petrol	black
--------	--------	--------	-------

- Display vehical whose cost is highest

```
SQL> select vehicalname,cost
```

```
2 from vehical
```

```
3 where cost=
```

```
4 (
```

```
5 select max(cost)
```

```
6 from vehical
```

```
7 );
```

VEHICALNAME	COST
-------------	------

```
-----
```

yamaha	99000
--------	-------

- Display vehical in the series of 'AS'

```
SQL> select *from vehical
```

```
2 where vehicalno like 'AS%';
```

VEHICALNO	VEHICALNAME	TYPE	COLOR
-----------	-------------	------	-------

```
-----
```

COST

```
-----
```

AS8919	Discover	petrol	black
--------	----------	--------	-------

```
90000
```

AS2156 hero diesel black
80000

AS2583 TVS petrol
98000

VEHICALNO VEHICALNAME TYPE COLOR

COST

AS5289 Bajaj petrol white

```
SQL> create table films6  
2 (filmno int primary key,filmname char (10) unique ,filmdir char(10),budget int);
```

Table created.

```
SQL> insert into films6 values(1,'War','sanket',2000000);
```

1 row created.

```
SQL> insert into films6 values(2,'simmba','sanket',2000000);
```

1 row created.

```
SQL> insert into films6 values(3,'singham','sanket',2000000);
```

1 row created.

```
SQL> insert into films6 values(4,'shershaah','sanket',2000000);
```

1 row created.

```
SQL> insert into films6 values(5,'baaghi','sanket',2000000);
```

1 row created.

```
SQL> select * from films6;
```

FILMNO	FILMNAME	FILMDIR	BUDGET
1	War	sanket	2000000
2	simmba	sanket	2000000
3	singham	sanket	2000000
4	shershaah	sanket	2000000
5	baaghi	sanket	2000000

```
SQL> select filmname from films6;
```

FILMNAME
War
simmba
singham
shershaah
baaghi

```
SQL> drop table films6;
```

Table dropped.

```
SQL>
```

Assignment no-12

Demonstrate String Functions in RDBMS.

SQL> select * from stud107;

SROLL SNAME SADD

101	anjali	kurkhedi
102	harshada	shirpur
103	vishakha	kharde
104	namrata	kharde
105	himani	shirpur

SQL> select upper(sname) from stud107;

UPPER(SNAME)

ANJALI

HARSHADA

VISHAKHA

NAMRATA

HIMANI

SQL> select lower(sname) from stud107;

LOWER(SNAME)

Anjali

Harshada

Vishakha

Namrata

Himani

SQL> select initcap(sname) from stud107;

INITCAP(SNAME)

Anjali

Harshada

Vishakha

Namrata

Himani

SQL> select sname,length(sname) as sname_length from stud107;

SNAME	SNAME_LENGTH
Anjali	20
Harshada	20
Vishakha	20
Namrata	20
Himani	20

SQL> select replace(sname,'himani','hema') from stud107;

REPLACE(SNAME,'HIMANI','HEMA')

Anjali

Harshada

Vishakha

Namrata

Hema

SQL> select substr(sname,1,4)sroll from stud107;

SROL

Anja

Hars

Vish

Namr

Hima

SQL> select sname ||'has roll_no'||sroll from stud107;

SNAME||'HASROLL_NO'||SROLL

Anjali has roll_no101

Harshada has roll_no102

Vishakha has roll_no103

Namrata has roll_no104

Himani has roll_no105

ASSIGNMENT NO-13

CREATE TABLE: EMP(ENO,NAME,SALARY,DEPT)

SQL> create table emp71

```
2  (eno char(10),  
3   ename char(20),  
4   esalary int,  
5   edept int);
```

Table created.

SQL> insert into emp71

```
2  values('&eno','&ename',&esalary,&edept);
```

Enter value for eno: E101

Enter value for ename: pranjal

Enter value for esalary: 4000

Enter value for edept: 1

old 2: values('&eno','&ename',&esalary,&edept)

new 2: values('E101','pranjal',4000,1)

1 row created.

1.Display all records having eno that start with 'E'.

SQL> select*from emp71

```
2  where eno like'E%';
```

ENO	ENAME	ESALARY	EDEPT
E101	pranjali	4000	1
E102	yogesh	5000	2
E105	kalyani	5500	5

2..Display records having dept must between 1 and 3.

SQL> select*from emp71

2 where edept between 1 and 3;

ENO	ENAME	ESALARY	EDEPT
E101	pranjali	4000	1
E102	yogesh	5000	2
D103	kuldip	5500	3

3.Display all records having salary greater than 5000.

SQL> select*from emp71

2 where esalary>5000;

ENO	ENAME	ESALARY	EDEPT
D103	kuldip	5500	3

D104 divya 6000 4

E105 kalyani 5500 5

ASS.17 CREATE TABLE COURSE

SQL> create table course

```
2 (cid int,  
3 cname char(20),  
4 cduration int,  
5 cfees int,  
6 csub char(10)  
7 );
```

Table created.

SQL> desc course;

Name	Null?	Type
CID		NUMBER(38)
CNAME		CHAR(20)
CDURATION		NUMBER(38)
CFEES		NUMBER(38)
CSUB		CHAR(10)

SQL> insert into course values(&id,'&cname','&cduration','&fees','&csub');

Enter value for id: 401

Enter value for cname: bca

Enter value for cduration: 10

Enter value for fees: 29000

Enter value for csub: RDBMS

old 1: insert into course values(&id,'&cname','&cduration','&fees','&csub')

new 1: insert into course values(401,'bca','10','29000','RDBMS')...

ASS.17 CREATE TABLE COURSE

1 row created

SQL> select*from course;

CID	CNAME	CDURATION	CFEES	CSUB
-----	-------	-----------	-------	------

SQL> select*from course;

CID	CNAME	CDURATION	CFEES	CSUB
-----	-------	-----------	-------	------

401	bca	10	29000	RDBMS
402	BBA	11	17000	account
403	MMs	12	20000	java
404	bms	2	16000	tally

SQL> update course

2 set cfees=2000

3 where cname='MMs';

1 row updated.

SQL> select*from course;

CID	CNAME	CDURATION	CFEES	CSUB
-----	-------	-----------	-------	------

401	bca	10	29000	RDBMS
-----	-----	----	-------	-------

ASS.17 CREATE TABLE COURSE

402 BBA	11	17000	account
403 MMs	12	2000	java
404 bms	2	16000	tally

SQL> delete from course

```
2 where cname='BBA';
```

1 row deleted.

SQL> select*from course;

CID	CNAME	CDURATION	CFEES	CSUB
401	bca	10	29000	RDBMS
403	MMs	12	2000	java
404	bms	2	16000	tally

SQL> Truncate table course;

1) Create table: Film(filmno,filmname,director,budget) with constraint and insert 5 records.

Filmno-primary key

Filmname-unique and solve following queries.

1.Display all records

2.display name of all films

3.Destroy the table.

2) Create table: cricket_team(teamid, t_name,captain,rank) n insert 5 records and solve following queries.

1.find name of all captain having 'M' as first character

2.find name of all captain having 'A' as second letter

3.find all team whose rank between 1 & 3

4.Demonstrate truncate & drop command

3) Create table: product_master(pno, pname,price) & order_detail(ono,pno,qty,rate)

Solve following queries.

1.pno in "product_master" is primary key

2.pno in "order_detail" references pno in "product_master"

3.find the product whose cost is highest

4) Create table: student(sroll,sname,city,gender,class,subject) insert 5 records & solve following queries.

1.roll no. is primary key

2.gender can be male or female

3.change all classes with 'ADCA' & subject with 'software engg'

5) Create table: dept(dno,dnmae,loc,noofemp) & insert 5 records in table. Create view on dept table having fields dno,dname

Create table: salesman with following column & constraints.

Sid-primary key

Name-not null

Address-unique

State_Salary=should be greater than 10000

1.Display all records

2. Destroy the table

3.Demonstrate view

6) Createtable: vehical(vehicalno,vehicalname,type,color,cost) & solve following queries.

- 1.Display all vehical in black color
- 2.Dipslay vehical whose cost is highest
- 3.Display vehical in the series of 'AS'

7) Create table: patient(pid,pname,disease,admitdate) & solve queries.

- 1.Display patient whose admitted from 01/01/2018 to 01/02/2018
- 2.Display patient with disease 'Maleria' or 'Typhoid'
3. Change patientname of second patient as 'Amruta'

8) Create table: emp(empno,deptno,mgr,sal,hiredate,job)

Dept(deptno,dname,loc)

- 1.Give demonstration of group by clause
2. Display highest paid employee working as clerk
3. Count no. of employee in each department

9) Demonstrate aggregate function in RDBMS.

10) Demonstrate Date function in RDBMS.

11) Create table and insert 10 records and perform ALTER operation.

12) Demonstrate String function in RDBMS.

13) Create table: emp (eno,name,salary,dept)

- 1.Display all records having eno that start with 'E'
2. Display records having dept must between 1 & 3
3. Display records having salary greater than 5000

14) Create table: book & solve following queries.

Accno-int primary key

Bname –varchar(20) not null

Author-varchar(20)

Subject-char(10) not null

- 1.select accno not in 1000 & 2000
- 2.List all author for JAVA & SQL
- 3.Add one more field "noofpages" to table
- 4.Delete the table

15) Create table: menu(dishno,dishname,price)

- 1.insert 5 records
- 2.Display all dishes present with their price
- 3.change the price of the dish icecream to 200
- 4.alter table & ad one field category for veg/non-veg
- 5.Display dishname and price having maximum price

16) Demonstrate the Synonyms in RDBMS.

17) Create table: course(coursed,cname,duration ,fees)

Insert 5 records & solve following queries.

- 1.Display all course information
- 2.Update fees of course MMS to 20000
- 3.Delete course BBA
- 4.Truncate the table

18) Demonstrate the use of operators: IN,OR,AND,BETWEEN,NOT,LIKE,EXISTS

19) Demonstrate use of GROUP by and HAVING clause

20) Demonstrate JOINS & NESTED queries.

21) Create table with various constraints , insert records and perform ALTER,DELETE & UPDATE operation.

22) Demonstrate VIEW.

23) Demonstrate to INSERT, UPDATE, and DELETE Records in Table.

24) Create table with various constraints as PRIMARY KEY, FOREIGN KEY, and CHECK & NOT NULL Constraints

Assignment No (5):

1) Create table :dept Insert 5 records ,create view on dept table having fields dno,dname

```
SQL> create table dept79  
2 ( dno int,  
3 dname char(20),  
4 dloc char(20),  
5 nofemp int);
```

B)

```
insert into dept79  
2 values(&dno,'&dname','&dloc',&nofemp);
```

c)

```
.create view d1 as  
2 select dno from dept79;  
  
. create view t1 as select dname from dept79;
```

2) Create table

A) display all records

```
select * from salesman79;
```

SID	SNAME	SADD	STATE_SAL
-----	-----	-----	-----

1	laksh	naka	11000
2	tushar	karvand	150000

b)destroy table:

```
drop table salesman79;
```

21) Create table with various constraints , insert records and perform ALTER,DELETE & UPDATE operation.

SQL> create table emp88

```
2 (empid int primary key,  
3 empname char(20) not null,  
4 empmno int not null,  
5 empadd char (20));
```

Table created.

SQL> insert into emp88 values(111,'sneha',9850798076,'pune');

1 row created.

SQL> insert into emp88 values(112,'karan',9656789890,'nasik');

1 row created.

SQL> insert into emp88 values(145,'ram',9656789890,'mumbai');

1 row created.

SQL> insert into emp88 values(144,'tom',8010457890,'shirpur')

1 row created.

SQL> insert into emp88 values(888,'priya',8056547870,'dhule');

1 row created.

SQL> select * from emp88;

EMPID	EMPNAME	EMPMNO	EMPADD
111	sneha	9850798076	pune
112	karan	9656789890	nasik
145	ram	9656789890	mumbai
144	tom	8010457890	shirpu
888	priya	8056547870	dhule

SQL> alter table emp88

2 add (Gender char(20));

Table altered.

SQL> select * from emp88;

EMPID	EMPNAME	EMPMNO	EMPADD
			GENDER
111	sneha	9850798076	pune
112	karan	9656789890	nasik
145	ram	9656789890	mumbai
EMPID	EMPNAME	EMPMNO	EMPADD
			GENDER
144	tom	8010457890	shirpur
888	priya	8056547870	dhule

```
SQL> update emp88 set Gender= 'female'where empid =111;
```

1 row updated.

```
SQL> select * from emp88;
```

EMPID	EMPNAME	EMPMNO	EMPADD
-----	-----	-----	-----
GENDER			
-----	-----	-----	-----
111	sneha	9850798076	pune
female			
112	karan	9656789890	nasik
145	ram	9656789890	mumbai
-----	-----	-----	-----
EMPID	EMPNAME	EMPMNO	EMPADD
-----	-----	-----	-----
GENDER			
-----	-----	-----	-----
144	tom	8010457890	shirpur
888	priya	8056547870	dhule

```
SQL> delete from emp88 where Gender = 'female';
```

```
1 row deleted.
```

```
SQL> select * from emp88;
```

EMPID	EMPNAME	EMPMNO	EMPADD
GENDER			
112	karan	9656789890	nasik
145	ram	9656789890	mumbai
144	tom	8010457890	shirpur
EMPID	EMPNAME	EMPMNO	EMPADD
GENDER			
888	priya	8056547870	dhule

```
SQL>
```

Query = 15

Create table : menu (dishno,dishname,price)

1. insert 5 records.
2. display all dishes present with their price.
3. change the price of the dish icecream to 200.
4. alter table & add one field category for veg/non-ved.
5. display dishname and price having maximum price.

SQL> create table menu

```
2. ( dishno int ,  
3. dishname char(20),  
4. price int );
```

SQL> insert into menu values(1,'pavbhaji',200);

SQL> insert into menu values(2,'icecream',150);

SQL> insert into menu values(3,'panipuri',50);

SQL> insert into menu values(4,'chikken tikka',250);

SQL> insert into menu values(5,'biryani',300);

||

SQL> select dishname,price from menu;

SQL> update menu

2 set price=200

3 where dishname='icecream';

SQL> alter table menu

2 add veg char(10);

SQL> select dishname,price from menu

2 where price=(select max(price) from menu);