

```
SQL> connect system
Enter password:
Connected.
```

```
SQL> create table employee(emp_id number(10),f_name varchar(10),l_name
varchar(10),job_type varchar(10),salary number(10),dept varchar(10),commission
number(10),manager_id number(10));
```

Table created.

```
SQL> desc employee;
```

Name	Null?	Type
EMP_ID		NUMBER(10)
F_NAME		VARCHAR2(10)
L_NAME		VARCHAR2(10)
JOB_TYPE		VARCHAR2(10)
SALARY		NUMBER(10)
DEPT		VARCHAR2(10)
COMMISSION		NUMBER(10)
MANAGER_ID		NUMBER(10)

```
SQL> alter table employee add primary key(emp_id);
```

Table altered.

```
SQL> desc employee;
```

Name	Null?	Type
EMP_ID	NOT NULL	NUMBER(10)
F_NAME		VARCHAR2(10)
L_NAME		VARCHAR2(10)
JOB_TYPE		VARCHAR2(10)
SALARY		NUMBER(10)
DEPT		VARCHAR2(10)
COMMISSION		NUMBER(10)
MANAGER_ID		NUMBER(10)

```
SQL> alter table employee modify(f_name not null,salary not null);
```

Table altered.

```
SQL> desc employee;
```

Name	Null?	Type
EMP_ID	NOT NULL	NUMBER(10)
F_NAME	NOT NULL	VARCHAR2(10)
L_NAME		VARCHAR2(10)
JOB_TYPE		VARCHAR2(10)
SALARY	NOT NULL	NUMBER(10)
DEPT		VARCHAR2(10)
COMMISSION		NUMBER(10)
MANAGER_ID		NUMBER(10)

```
SQL> alter table employee add date_of_joining date;
```

Table altered.

```
SQL> desc employee;
```

Name	Null?	Type
EMP_ID	NOT NULL	NUMBER(10)
F_NAME	NOT NULL	VARCHAR2(10)
L_NAME		VARCHAR2(10)
JOB_TYPE		VARCHAR2(10)

SALARY	NOT NULL	NUMBER(10)
DEPT		VARCHAR2(10)
COMMISSION		NUMBER(10)
MANAGER_ID		NUMBER(10)
DATE_OF_JOINING		DATE

```
SQL> create table department(d_name varchar(15) primary key,d_loc varchar(10),hod_id
number(10));
```

Table created.

```
SQL> desc department;
```

Name	Null?	Type
-----	-----	-----
D_NAME	NOT NULL	VARCHAR2(15)
D_LOC		VARCHAR2(10)
HOD_ID		NUMBER(10)

```
SQL> create table location(loc_id number(10),city varchar(10),contact_no
number(10));
```

Table created.

```
SQL> desc location;
```

Name	Null?	Type
-----	-----	-----
LOC_ID		NUMBER(10)
CITY		VARCHAR2(10)
CONTACT_NO		NUMBER(10)

```
SQL> alter table location modify city varchar(5);
```

Table altered.

```
SQL> desc location;
```

Name	Null?	Type
-----	-----	-----
LOC_ID		NUMBER(10)
CITY		VARCHAR2(5)
CONTACT_NO		NUMBER(10)

```
SQL> alter table location drop column contact_no;
```

Table altered.

```
SQL> desc location;
```

Name	Null?	Type
-----	-----	-----
LOC_ID		NUMBER(10)
CITY		VARCHAR2(5)

```
SQL> alter table department rename column d_name to dept;
```

Table altered.

```
SQL> alter table employee add foreign key(dept) references department(dept);
```

Table altered.

```
SQL> alter table location rename column city to address;
```

Table altered.

```
SQL> desc location;
```

Name	Null?	Type
LOC_ID		NUMBER(10)
ADDRESS		VARCHAR2(5)

SQL> rename location to loc;

Table renamed.

SQL> desc loc;

Name	Null?	Type
LOC_ID		NUMBER(10)
ADDRESS		VARCHAR2(5)

SQL> alter table loc modify address varchar(10);

Table altered.

SQL> insert into loc values('&loc_id','&address');

Enter value for loc_id: 1

Enter value for address: Kolkata

old 1: insert into loc values('&loc_id','&address')

new 1: insert into loc values('1','Kolkata')

1 row created.

SQL> /

Enter value for loc_id: 2

Enter value for address: Mumbai

old 1: insert into loc values('&loc_id','&address')

new 1: insert into loc values('2','Mumbai')

1 row created.

SQL> truncate table loc;

Table truncated.

SQL> drop table loc;

Table dropped.

SQL> insert into department values('&dept','&d_loc','&hod_id');

Enter value for dept: Sales

Enter value for d_loc: Kol

Enter value for hod_id: 4

old 1: insert into department values('&dept','&d_loc','&hod_id')

new 1: insert into department values('Sales','Kol','4')

1 row created.

SQL> /

Enter value for dept: Accounts

Enter value for d_loc: Delhi

Enter value for hod_id: 6

old 1: insert into department values('&dept','&d_loc','&hod_id')

new 1: insert into department values('Accounts','Delhi','6')

1 row created.

SQL> /

Enter value for dept: Production

Enter value for d_loc: Kol

Enter value for hod_id: 1

old 1: insert into department values('&dept','&d_loc','&hod_id')

new 1: insert into department values('Production','Kol','1')

1 row created.

```
SQL> /
Enter value for dept: Marketing
Enter value for d_loc: Kol
Enter value for hod_id: 2
old 1: insert into department values('&dept','&d_loc','&hod_id')
new 1: insert into department values('Marketing','Kol','2')
```

1 row created.

```
SQL> /
Enter value for dept: R & D
Enter value for d_loc: Marketing
Enter value for hod_id: 8
old 1: insert into department values('&dept','&d_loc','&hod_id')
new 1: insert into department values('R & D','Marketing','8')
```

1 row created.

```
SQL> insert into employee
values('&emp_id','&f_name','&l_name','&job_type','&salary','&dept','&commission','&manager_id','&date_of_joining');
Enter value for emp_id: 1
Enter value for f_name: Arun
Enter value for l_name: Khan
Enter value for job_type: Manager
Enter value for salary: 90000
Enter value for dept: Production
Enter value for commission:
Enter value for manager_id:
Enter value for date_of_joining: 04-Jan-1998
old 1: insert into employee
values('&emp_id','&f_name','&l_name','&job_type','&salary','&dept','&commission','&manager_id','&date_of_joining')
new 1: insert into employee
values('1','Arun','Khan','Manager','90000','Production','','','04-Jan-1998')
```

1 row created.

```
SQL> /
Enter value for emp_id: 2
Enter value for f_name: Barun
Enter value for l_name: Kumar
Enter value for job_type: manager
Enter value for salary: 80000
Enter value for dept: arketing
Enter value for commission:
Enter value for manager_id:
Enter value for date_of_joining: 09-Feb-1998
old 1: insert into employee
values('&emp_id','&f_name','&l_name','&job_type','&salary','&dept','&commission','&manager_id','&date_of_joining')
new 1: insert into employee
values('2','Barun','Kumar','manager','80000','arketing','','','09-Feb-1998')
```

1 row created.

```
SQL> /
Enter value for emp_id: 3
Enter value for f_name: Chitra
Enter value for l_name: Kapoor
Enter value for job_type: Engineer
Enter value for salary: 60000
Enter value for dept: Production
Enter value for commission:
```

```
Enter value for manager_id: 1
Enter value for date_of_joining: 08-Jan-1998
old 1: insert into employee
values('&emp_id','&f_name','&l_name','&job_type','&salary','&dept','&commission','&manager_id','&date_of_joining')
new 1: insert into employee
values('3','Chitra','Kapoor','Engineer','60000','Production','','1','08-Jan-1998')
```

1 row created.

```
SQL> /
Enter value for emp_id: 4
Enter value for f_name: Dheeraj
Enter value for l_name: Mishra
Enter value for job_type: Manager
Enter value for salary: 75000
Enter value for dept: Sales
Enter value for commission:
Enter value for manager_id: 4
Enter value for date_of_joining: 27-Dec-2001
old 1: insert into employee
values('&emp_id','&f_name','&l_name','&job_type','&salary','&dept','&commission','&manager_id','&date_of_joining')
new 1: insert into employee
values('4','Dheeraj','Mishra','Manager','75000','Sales','','4','27-Dec-2001')
```

1 row created.

```
SQL> /
Enter value for emp_id: 5
Enter value for f_name: Emma
Enter value for l_name: Dutt
Enter value for job_type: Engineer
Enter value for salary: 55000
Enter value for dept: Production
Enter value for commission:
Enter value for manager_id: 1
Enter value for date_of_joining: 20-Mar-2002
old 1: insert into employee
values('&emp_id','&f_name','&l_name','&job_type','&salary','&dept','&commission','&manager_id','&date_of_joining')
new 1: insert into employee
values('5','Emma','Dutt','Engineer','55000','Production','','1','20-Mar-2002')
```

1 row created.

```
SQL> /
Enter value for emp_id: 6
Enter value for f_name: Floki
Enter value for l_name: Dutt
Enter value for job_type: Accountant
Enter value for salary: 70000
Enter value for dept: Accounts
Enter value for commission:
Enter value for manager_id:
Enter value for date_of_joining: 16-Jul-2000
old 1: insert into employee
values('&emp_id','&f_name','&l_name','&job_type','&salary','&dept','&commission','&manager_id','&date_of_joining')
new 1: insert into employee
values('6','Floki','Dutt','Accountant','70000','Accounts','','','16-Jul-2000')
```

1 row created.

```
SQL> /
Enter value for emp_id: 7
Enter value for f_name: Dheeraj
```

```

Enter value for l_name: Kumar
Enter value for job_type: Clerk
Enter value for salary: 40000
Enter value for dept: Accounts
Enter value for commission:
Enter value for manager_id: 6
Enter value for date_of_joining: 01-Jul-2016
old 1: insert into employee
values('&emp_id','&f_name','&l_name','&job_type','&salary','&dept','&commission','&manager_id','&date_of_joining')
new 1: insert into employee
values('7','Dheeraj','Kumar','Clerk','40000','Accounts','','6','01-Jul-2016')

```

1 row created.

```

SQL> /
Enter value for emp_id: 8
Enter value for f_name: Saul
Enter value for l_name: Good
Enter value for job_type: Engineer
Enter value for salary: 60000
Enter value for dept: R & D
Enter value for commission:
Enter value for manager_id:
Enter value for date_of_joining: 06-Sep-2014
old 1: insert into employee
values('&emp_id','&f_name','&l_name','&job_type','&salary','&dept','&commission','&manager_id','&date_of_joining')
new 1: insert into employee values('8','Saul','Good','Engineer','60000','R & D','','','06-Sep-2014')

```

1 row created.

```

SQL> /
Enter value for emp_id: 9
Enter value for f_name: Mou
Enter value for l_name: Bhat
Enter value for job_type: Clerk
Enter value for salary: 30000
Enter value for dept: Sales
Enter value for commission:
Enter value for manager_id: 4
Enter value for date_of_joining: 08-Mar-2018
old 1: insert into employee
values('&emp_id','&f_name','&l_name','&job_type','&salary','&dept','&commission','&manager_id','&date_of_joining')
new 1: insert into employee
values('9','Mou','Bhat','Clerk','30000','Sales','','4','08-Mar-2018')

```

1 row created.

```

SQL> /
Enter value for emp_id: 10
Enter value for f_name: Sunny
Enter value for l_name: Deol
Enter value for job_type: Salesman
Enter value for salary: 20000
Enter value for dept: arketing
Enter value for commission: 10000
Enter value for manager_id: 2
Enter value for date_of_joining: 31-Mar-2001
old 1: insert into employee
values('&emp_id','&f_name','&l_name','&job_type','&salary','&dept','&commission','&manager_id','&date_of_joining')
new 1: insert into employee
values('10','Sunny','Deol','Salesman','20000','arketing','10000','2','31-Mar-2001')

```

1 row created.

SQL> /

```
Enter value for emp_id: 11
Enter value for f_name: Bobby
Enter value for l_name: Deol
Enter value for job_type: Engineer
Enter value for salary: 35000
Enter value for dept: R & D
Enter value for commission:
Enter value for manager_id: 8
Enter value for date_of_joining: 17-Oct-2017
old 1: insert into employee
values('&emp_id','&f_name','&l_name','&job_type','&salary','&dept','&commission','&manager_id','&date_of_joining')
new 1: insert into employee values('11','Bobby','Deol','Engineer','35000','R & D','','8','17-Oct-2017')
```

1 row created.

SQL> /

```
Enter value for emp_id: 12
Enter value for f_name: Aamir
Enter value for l_name: Khan
Enter value for job_type: Salesman
Enter value for salary: 15000
Enter value for dept: arketing
Enter value for commission: 5000
Enter value for manager_id: 2
Enter value for date_of_joining: 11-Jan-2013
old 1: insert into employee
values('&emp_id','&f_name','&l_name','&job_type','&salary','&dept','&commission','&manager_id','&date_of_joining')
new 1: insert into employee
values('12','Aamir','Khan','Salesman','15000','arketing','5000','2','11-Jan-2013')
```

1 row created.

SQL> select * from department;

DEPT	D_LOC	HOD_ID
Sales	Kol	4
Accounts	Delhi	6
Production	Kol	1
arketing	Kol	2
R & D	Marketing	8

SQL> select dept,d_loc from department;

DEPT	D_LOC
Sales	Kol
Accounts	Delhi
Production	Kol
arketing	Kol
R & D	Marketing

SQL> select f_name,l_name,salary,salary+1000 from employee;

F_NAME	L_NAME	SALARY	SALARY+1000
Arun	Khan	90000	91000
Barun	Kumar	80000	81000
Chitra	Kapoor	60000	61000
Dheeraj	Mishra	75000	76000

Emma	Dutt	55000	56000
Floki	Dutt	70000	71000
Dheeraj	Kumar	40000	41000
Saul	Good	60000	61000
Mou	Bhat	30000	31000
Sunny	Deol	20000	21000
Bobby	Deol	35000	36000

F_NAME	L_NAME	SALARY	SALARY+1000
-----	-----	-----	-----
Aamir	Khan	15000	16000

12 rows selected.

SQL> select salary+1000,salary+12*100 from employee;

SALARY+1000	SALARY+12*100
-----	-----
91000	91200
81000	81200
61000	61200
76000	76200
56000	56200
71000	71200
41000	41200
61000	61200
31000	31200
21000	21200
36000	36200

SALARY+1000	SALARY+12*100
-----	-----
16000	16200

12 rows selected.

SQL> select f_name as name and salary as annsal from employee;
select f_name as name and salary as annsal from employee
*

ERROR at line 1:
ORA-00923: FROM keyword not found where expected

SQL> select f_name as name,salary as annsal from employee;

NAME	ANNSAL
-----	-----
Arun	90000
Barun	80000
Chitra	60000
Dheeraj	75000
Emma	55000
Floki	70000
Dheeraj	40000
Saul	60000
Mou	30000
Sunny	20000
Bobby	35000

NAME	ANNSAL
-----	-----
Aamir	15000

12 rows selected.

SQL> select l_name as "LasT",salary+100 as "NewSal" from employee;

LasT	NewSal
-----	-----
Khan	90100
Kumar	80100
Kapoor	60100
Mishra	75100
Dutt	55100
Dutt	70100
Kumar	40100
Good	60100
Bhat	30100
Deol	20100
Deol	35100

LasT	NewSal
-----	-----
Khan	15100

12 rows selected.

```
SQL> select emp_id,f_name,l_name,job_type from employee where salary=(select
max(salary) from employee);
```

EMP_ID	F_NAME	L_NAME	JOB_TYPE
-----	-----	-----	-----
1	Arun	Khan	Manager

```
SQL> select emp_id,f_name,l_name,job_type from employee where salary=(select
min(salary) from employee);
```

EMP_ID	F_NAME	L_NAME	JOB_TYPE
-----	-----	-----	-----
12	Aamir	Khan	Salesman

```
SQL> select avg(salary) from employee;
```

AVG(SALARY)

52500

```
SQL> create table person(driverid varchar2(20) primary key,name varchar2(20),address
varchar2(20));
```

Table created.

```
SQL> desc person;
```

Name	Null?	Type
-----	-----	-----
DRIVERID	NOT NULL	VARCHAR2(20)
NAME		VARCHAR2(20)
ADDRESS		VARCHAR2(20)

```
SQL> create table car(regno varchar2(20) primary key,model varchar2(20),year
number);
```

Table created.

```
SQL> desc car;
```

Name	Null?	Type
-----	-----	-----
REGNO	NOT NULL	VARCHAR2(20)
MODEL		VARCHAR2(20)
YEAR		NUMBER

```
SQL> create table accident(reportno number primary key,accdate date,location
varchar2(20));
```

Table created.

SQL> desc accident;

Name	Null?	Type
REPORTNO	NOT NULL	NUMBER
ACCDATE		DATE
LOCATION		VARCHAR2(20)

SQL> create table owns(driverid varchar2(20) references person(driverid), regno
varchar2(20) references car(regno));

Table created.

SQL> desc owns;

Name	Null?	Type
DRIVERID		VARCHAR2(20)
REGNO		VARCHAR2(20)

SQL> create table participated(driverid varchar2(20) references person(driverid),
regno varchar2(20) references car(regno), reportno number references
accident(reportno), dmg_amt number(10,2));

Table created.

SQL> desc participated;

Name	Null?	Type
DIRVERID		VARCHAR2(20)
REGNO		VARCHAR2(20)
REPORTNO		NUMBER
DMG_AMT		NUMBER(10,2)

SQL> insert into person values('&driverid','&name','&address');
Enter value for driverid: 1
Enter value for name: Ram
Enter value for address: Kolkata
old 1: insert into person values('&driverid','&name','&address')
new 1: insert into person values('1','Ram','Kolkata')

1 row created.

SQL> /
Enter value for driverid: 103
Enter value for name: Shyam
Enter value for address: Newtown
old 1: insert into person values('&driverid','&name','&address')
new 1: insert into person values('103','Shyam','Newtown')

1 row created.

SQL> /
Enter value for driverid: 104
Enter value for name: Sagnik
Enter value for address: Rajarhat
old 1: insert into person values('&driverid','&name','&address')
new 1: insert into person values('104','Sagnik','Rajarhat')

1 row created.

SQL> /
Enter value for driverid: 105
Enter value for name: Sam
Enter value for address: Howrah
old 1: insert into person values('&driverid','&name','&address')

```
new 1: insert into person values('105','Sam','Howrah')
```

```
1 row created.
```

```
SQL> /
```

```
Enter value for driverid: 102
```

```
Enter value for name: Evan
```

```
Enter value for address: Bandel
```

```
old 1: insert into person values('&driverid','&name','&address')
```

```
new 1: insert into person values('102','Evan','Bandel')
```

```
1 row created.
```

```
SQL> insert into car values('&regno','&model','&year');
```

```
Enter value for regno: 111
```

```
Enter value for model: Hundai
```

```
Enter value for year: 1998
```

```
old 1: insert into car values('&regno','&model','&year')
```

```
new 1: insert into car values('111','Hundai','1998')
```

```
1 row created.
```

```
SQL> /
```

```
Enter value for regno: 222
```

```
Enter value for model: BMW
```

```
Enter value for year: 2001
```

```
old 1: insert into car values('&regno','&model','&year')
```

```
new 1: insert into car values('222','BMW','2001')
```

```
1 row created.
```

```
SQL> /
```

```
Enter value for regno: 333
```

```
Enter value for model: Innova
```

```
Enter value for year: 2003
```

```
old 1: insert into car values('&regno','&model','&year')
```

```
new 1: insert into car values('333','Innova','2003')
```

```
1 row created.
```

```
SQL> /
```

```
Enter value for regno: 444
```

```
Enter value for model: Maruti
```

```
Enter value for year: 2005
```

```
old 1: insert into car values('&regno','&model','&year')
```

```
new 1: insert into car values('444','Maruti','2005')
```

```
1 row created.
```

```
SQL> /
```

```
Enter value for regno: 555
```

```
Enter value for model: Suzuki
```

```
Enter value for year: 2000
```

```
old 1: insert into car values('&regno','&model','&year')
```

```
new 1: insert into car values('555','Suzuki','2000')
```

```
1 row created.
```

```
SQL> insert into owns values('&driverid','&regno');
```

```
Enter value for driverid: 1
```

```
Enter value for regno: 111
```

```
old 1: insert into owns values('&driverid','&regno')
```

```
new 1: insert into owns values('1','111')
```

```
1 row created.
```

```
SQL> /
```

```
Enter value for driverid: 102
Enter value for regno: 222
old 1: insert into owns values('&driverid','&regno')
new 1: insert into owns values('102','222')
```

1 row created.

```
SQL> /
Enter value for driverid: 103
Enter value for regno: 333
old 1: insert into owns values('&driverid','&regno')
new 1: insert into owns values('103','333')
```

1 row created.

```
SQL> /
Enter value for driverid: 104
Enter value for regno: 444
old 1: insert into owns values('&driverid','&regno')
new 1: insert into owns values('104','444')
```

1 row created.

```
SQL> /
Enter value for driverid: 105
Enter value for regno: 555
old 1: insert into owns values('&driverid','&regno')
new 1: insert into owns values('105','555')
```

1 row created.

```
SQL> insert into accident values('&reportno','&accddate','&location');
Enter value for reportno: 201
Enter value for accdate: 11-Jan-2006
Enter value for location: Kolkata
old 1: insert into accident values('&reportno','&accddate','&location')
new 1: insert into accident values('201','11-Jan-2006','Kolkata')
```

1 row created.

```
SQL> /
Enter value for reportno: 202
Enter value for accdate: 12-Jan-2007
Enter value for location: Delhi
old 1: insert into accident values('&reportno','&accddate','&location')
new 1: insert into accident values('202','12-Jan-2007','Delhi')
```

1 row created.

```
SQL> /
Enter value for reportno: 203
Enter value for accdate: 12-Jan-2008
Enter value for location: Newtown
old 1: insert into accident values('&reportno','&accddate','&location')
new 1: insert into accident values('203','12-Jan-2008','Newtown')
```

1 row created.

```
SQL> /
Enter value for reportno: 204
Enter value for accdate: 13-Jan-2006
Enter value for location: Rajarhat
old 1: insert into accident values('&reportno','&accddate','&location')
new 1: insert into accident values('204','13-Jan-2006','Rajarhat')
```

1 row created.

```

SQL> /
Enter value for reportno: 205
Enter value for accdate: 13-Jan-2009
Enter value for location: Bandel
old 1: insert into accident values('&reportno','&accdate','&location')
new 1: insert into accident values('205','13-Jan-2009','Bandel')

1 row created.

SQL> insert into participated values('&driverid','&regno','&reportno','&dmg_amt');
Enter value for driverid: 1
Enter value for regno: 111
Enter value for reportno: 201
Enter value for dmg_amt: 10000
old 1: insert into participated
values('&driverid','&regno','&reportno','&dmg_amt')
new 1: insert into participated values('1','111','201','10000')

1 row created.

SQL> /
Enter value for driverid: 102
Enter value for regno: 222
Enter value for reportno: 202
Enter value for dmg_amt: 20000
old 1: insert into participated
values('&driverid','&regno','&reportno','&dmg_amt')
new 1: insert into participated values('102','222','202','20000')

1 row created.

SQL> /
Enter value for driverid: 103
Enter value for regno: 333
Enter value for reportno: 203
Enter value for dmg_amt: 30000
old 1: insert into participated
values('&driverid','&regno','&reportno','&dmg_amt')
new 1: insert into participated values('103','333','203','30000')

1 row created.

SQL> /
Enter value for driverid: 104
Enter value for regno: 444
Enter value for reportno: 204
Enter value for dmg_amt: 40000
old 1: insert into participated
values('&driverid','&regno','&reportno','&dmg_amt')
new 1: insert into participated values('104','444','204','40000')

1 row created.

SQL> /
Enter value for driverid: 105
Enter value for regno: 555
Enter value for reportno: 205
Enter value for dmg_amt: 50000
old 1: insert into participated
values('&driverid','&regno','&reportno','&dmg_amt')
new 1: insert into participated values('105','555','205','50000')

1 row created.

SQL> update participated set dmg_amt=dmg_amt +2500 where 12<=regno<=25000;
update participated set dmg_amt=dmg_amt +2500 where 12<=regno<=25000
*
```

ERROR at line 1:
ORA-00933: SQL command not properly ended

SQL> update participated set dmg_amt=dmg_amt +2500 where regno<=25000 and regno>=12;
5 rows updated.

SQL> select * from participated;

DIRVERID	REGNO	REPORTNO	DMG_AMT
1	111	201	12500
102	222	202	22500
103	333	203	32500
104	444	204	42500
105	555	205	52500

SQL> insert into accident values('&reportno','&accddate','&location');
Enter value for reportno: 206
Enter value for accdate: 22-Jan-2001
Enter value for location: Kerala
old 1: insert into accident values('&reportno','&accddate','&location')
new 1: insert into accident values('206','22-Jan-2001','Kerala')

1 row created.

SQL> select count(*) from accident where accdate between '01-Jan-2006' and '31-Dec-2006';

COUNT(*)
2

SQL> select count(*) as Suzuki_Accident from car,accident,participated where
car.regno= participated.regno and accident.reportno=participated.reportno and
car.model='Suzuki';

SUZUKI_ACCIDENT
1

create table employee(EID varchar2(10) primary key, EName varchar2(20), DOJ Date, Salary number(10,2), DID Varchar2(10), DName varchar2(20));

create table department(DID varchar2(10) primary key, DName varchar2(20), Mgr varchar2(20));

desc employee;

Name	Null?	Type
EID	NOT NULL	VARCHAR2(10)
ENAME		VARCHAR2(20)
DOJ		DATE
SALARY		NUMBER(10,2)
DID		VARCHAR2(10)
DNAME		VARCHAR2(20)

desc department;

Name	Null?	Type
DID	NOT NULL	VARCHAR2(10)
DNAME		VARCHAR2(20)
MGR		VARCHAR2(20)

insert into employee values('E203', 'Abhijit', '19-oct-2019', 56000, 'D120', 'Sales');

insert into employee values('E049', 'Sumit', '08-jan-2019', 65000, 'D420', 'Marketing');

insert into employee values('E713', 'Priyam', '01-nov-2016', 86000, 'D420', 'Marketing');

insert into employee values('E629', 'Ranjita', '29-oct-2018', 49000, 'D120', 'Sales');

insert into employee values('E191', 'Abhijit', '18-jul-2017', 60000, 'D846', 'IT');

insert into employee values('E432', 'Zoya', '18-jul-2017', 30000, 'D245', 'Entry');

insert into employee values('E542', 'Jasmine', '16-feb-2017', 308000, 'D247', 'BackOfc');

select * from employee;

EID	ENAME	DOJ	SALARY	DID	DNAME
E203	Abhijit	19-OCT-19	56000	D120	Sales
E049	Sumit	08-JAN-19	65000	D420	Marketing
E713	Priyam	01-NOV-16	86000	D420	Marketing
E629	Ranjita	29-OCT-18	49000	D120	Sales
E191	Abhijit	18-JUL-17	60000	D846	IT
E432	Zoya	18-JUL-17	30000	D245	Entry
E542	Jasmine	16-FEB-17	308000	D247	BackOfc

insert into department values('D120', 'Sales', 'Shanaya');

insert into department values('D420', 'Marketing', 'Akash');

insert into department values('D846', 'IT', 'Inaya');

insert into department values('D245', 'Entry', 'Shaurya');

insert into department values('D194', 'Management', 'Akash');

select * from department;

DID	DNAME	MGR
D120	Sales	Shanaya
D420	Marketing	Akash
D846	IT	Inaya
D245	Entry	Shaurya
D194	Management	Akash

select * from employee cross join department;

//cross/cartesian join

EID	ENAME	DOJ	SALARY	DID	DNAME	DID	DNAME	MGR
E203	Abhijit	19-OCT-19	56000	D120	Sales	D120	Sales	Shanaya
E049	Sumit	08-JAN-19	65000	D420	Marketing	D120	Sales	Shanaya
E713	Priyam	01-NOV-16	86000	D420	Marketing	D120	Sales	Shanaya
E629	Ranjita	29-OCT-18	49000	D120	Sales	D120	Sales	Shanaya
E191	Abhijit	18-JUL-17	60000	D846	IT	D120	Sales	Shanaya
E432	Zoya	18-JUL-17	30000	D245	Entry	D120	Sales	Shanaya
E542	Jasmine	16-FEB-17	308000	D247	BackOfc	D120	Sales	Shanaya
E203	Abhijit	19-OCT-19	56000	D120	Sales	D420	Marketing	Akash
E049	Sumit	08-JAN-19	65000	D420	Marketing	D420	Marketing	Akash
E713	Priyam	01-NOV-16	86000	D420	Marketing	D420	Marketing	Akash
E629	Ranjita	29-OCT-18	49000	D120	Sales	D420	Marketing	Akash
E191	Abhijit	18-JUL-17	60000	D846	IT	D420	Marketing	Akash
E432	Zoya	18-JUL-17	30000	D245	Entry	D420	Marketing	Akash
E542	Jasmine	16-FEB-17	308000	D247	BackOfc	D420	Marketing	Akash
E203	Abhijit	19-OCT-19	56000	D120	Sales	D846	IT	Inaya
E049	Sumit	08-JAN-19	65000	D420	Marketing	D846	IT	Inaya
E713	Priyam	01-NOV-16	86000	D420	Marketing	D846	IT	Inaya
E629	Ranjita	29-OCT-18	49000	D120	Sales	D846	IT	Inaya
E191	Abhijit	18-JUL-17	60000	D846	IT	D846	IT	Inaya
E432	Zoya	18-JUL-17	30000	D245	Entry	D846	IT	Inaya
E542	Jasmine	16-FEB-17	308000	D247	BackOfc	D846	IT	Inaya
E203	Abhijit	19-OCT-19	56000	D120	Sales	D245	Entry	Shaurya
E049	Sumit	08-JAN-19	65000	D420	Marketing	D245	Entry	Shaurya
E713	Priyam	01-NOV-16	86000	D420	Marketing	D245	Entry	Shaurya

EID	ENAME	DOJ	SALARY	DID	DNAME	DID	DNAME	MGR
E629	Ranjita	29-OCT-18	49000	D120	Sales	D245	Entry	Shaurya
E191	Abhijit	18-JUL-17	60000	D846	IT	D245	Entry	Shaurya
E432	Zoya	18-JUL-17	30000	D245	Entry	D245	Entry	Shaurya
E542	Jasmine	16-FEB-17	308000	D247	BackOfc	D245	Entry	Shaurya
E203	Abhijit	19-OCT-19	56000	D120	Sales	D194	Management	Akash
E049	Sumit	08-JAN-19	65000	D420	Marketing	D194	Management	Akash
E713	Priyam	01-NOV-16	86000	D420	Marketing	D194	Management	Akash
E629	Ranjita	29-OCT-18	49000	D120	Sales	D194	Management	Akash
E191	Abhijit	18-JUL-17	60000	D846	IT	D194	Management	Akash
E432	Zoya	18-JUL-17	30000	D245	Entry	D194	Management	Akash
E542	Jasmine	16-FEB-17	308000	D247	BackOfc	D194	Management	Akash

35 rows selected.

select * from employee E cross join department D where D.did=E.did;

//cross/cartesian join with a where clause

EID	ENAME	DOJ	SALARY	DID	DNAME	DID	DNAME	MGR
E203	Abhijit	19-OCT-19	56000	D120	Sales	D120	Sales	Shanaya
E049	Sumit	08-JAN-19	65000	D420	Marketing	D420	Marketing	Akash
E713	Priyam	01-NOV-16	86000	D420	Marketing	D420	Marketing	Akash
E629	Ranjita	29-OCT-18	49000	D120	Sales	D120	Sales	Shanaya
E191	Abhijit	18-JUL-17	60000	D846	IT	D846	IT	Inaya
E432	Zoya	18-JUL-17	30000	D245	Entry	D245	Entry	Shaurya

select * from employee E, employee D where D.did=E.did;

//self join

EID	ENAME	DOJ	SALARY	DID	DNAME	EID	ENAME	DOJ	SALARY	DID	DNAME
E629	Ranjita	29-OCT-18	49000	D120	Sales	E203	Abhijit	19-OCT-19	56000	D120	Sales
E203	Abhijit	19-OCT-19	56000	D120	Sales	E203	Abhijit	19-OCT-19	56000	D120	Sales
E713	Priyam	01-NOV-16	86000	D420	Marketing	E049	Sumit	08-JAN-19	65000	D420	Marketing
E049	Sumit	08-JAN-19	65000	D420	Marketing	E049	Sumit	08-JAN-19	65000	D420	Marketing
E713	Priyam	01-NOV-16	86000	D420	Marketing	E713	Priyam	01-NOV-16	86000	D420	Marketing
E049	Sumit	08-JAN-19	65000	D420	Marketing	E713	Priyam	01-NOV-16	86000	D420	Marketing
E629	Ranjita	29-OCT-18	49000	D120	Sales	E629	Ranjita	29-OCT-18	49000	D120	Sales
E203	Abhijit	19-OCT-19	56000	D120	Sales	E629	Ranjita	29-OCT-18	49000	D120	Sales
E191	Abhijit	18-JUL-17	60000	D846	IT	E191	Abhijit	18-JUL-17	60000	D846	IT
E432	Zoya	18-JUL-17	30000	D245	Entry	E432	Zoya	18-JUL-17	30000	D245	Entry
E542	Jasmine	16-FEB-17	308000	D247	BackOfc	E542	Jasmine	16-FEB-17	308000	D247	BackOfc

11 rows selected.

select * from employee E inner join department D on D.did=E.did;

//inner join

EID	ENAME	DOJ	SALARY	DID	DNAME	DID	DNAME	MGR
E203	Abhijit	19-OCT-19	56000	D120	Sales	D120	Sales	Shanaya
E049	Sumit	08-JAN-19	65000	D420	Marketing	D420	Marketing	Akash
E713	Priyam	01-NOV-16	86000	D420	Marketing	D420	Marketing	Akash
E629	Ranjita	29-OCT-18	49000	D120	Sales	D120	Sales	Shanaya
E191	Abhijit	18-JUL-17	60000	D846	IT	D846	IT	Inaya
E432	Zoya	18-JUL-17	30000	D245	Entry	D245	Entry	Shaurya

select * from employee E full outer join department D on D.did=E.did;
 //(full) outer join

EID	ENAME	DOJ	SALARY	DID	DNAME	DID	DNAME	MGR
E629	Ranjita	29-OCT-18	49000	D120	Sales	D120	Sales	Shanaya
E203	Abhijit	19-OCT-19	56000	D120	Sales	D120	Sales	Shanaya
E713	Priyam	01-NOV-16	86000	D420	Marketing	D420	Marketing	Akash
E049	Sumit	08-JAN-19	65000	D420	Marketing	D420	Marketing	Akash
E191	Abhijit	18-JUL-17	60000	D846	IT	D846	IT	Inaya
E432	Zoya	18-JUL-17	30000	D245	Entry	D245	Entry	Shaurya
E542	Jasmine	16-FEB-17	308000	D247	BackOfc			
						D194	Management	Akash

8 rows selected.

select * from employee E left join department D on D.did=E.did;
 //left join

EID	ENAME	DOJ	SALARY	DID	DNAME	DID	DNAME	MGR
E629	Ranjita	29-OCT-18	49000	D120	Sales	D120	Sales	Shanaya
E203	Abhijit	19-OCT-19	56000	D120	Sales	D120	Sales	Shanaya
E713	Priyam	01-NOV-16	86000	D420	Marketing	D420	Marketing	Akash
E049	Sumit	08-JAN-19	65000	D420	Marketing	D420	Marketing	Akash
E191	Abhijit	18-JUL-17	60000	D846	IT	D846	IT	Inaya
E432	Zoya	18-JUL-17	30000	D245	Entry	D245	Entry	Shaurya
E542	Jasmine	16-FEB-17	308000	D247	BackOfc			

7 rows selected.

select * from employee E right join department D on D.did=E.did;

//right join

EID	ENAME	DOJ	SALARY	DID	DNAME	DID	DNAME	MGR
E203	Abhijit	19-OCT-19	56000	D120	Sales	D120	Sales	Shanaya
E049	Sumit	08-JAN-19	65000	D420	Marketing	D420	Marketing	Akash
E713	Priyam	01-NOV-16	86000	D420	Marketing	D420	Marketing	Akash
E629	Ranjita	29-OCT-18	49000	D120	Sales	D120	Sales	Shanaya
E191	Abhijit	18-JUL-17	60000	D846	IT	D846	IT	Inaya
E432	Zoya	18-JUL-17	30000	D245	Entry	D245	Entry	Shaurya
						D194	Management	Akash

select * from employee E full join department D on D.did=E.did;

//full join

EID	ENAME	DOJ	SALARY	DID	DNAME	DID	DNAME	MGR
E629	Ranjita	29-OCT-18	49000	D120	Sales	D120	Sales	Shanaya
E203	Abhijit	19-OCT-19	56000	D120	Sales	D120	Sales	Shanaya
E713	Priyam	01-NOV-16	86000	D420	Marketing	D420	Marketing	Akash
E049	Sumit	08-JAN-19	65000	D420	Marketing	D420	Marketing	Akash
E191	Abhijit	18-JUL-17	60000	D846	IT	D846	IT	Inaya
E432	Zoya	18-JUL-17	30000	D245	Entry	D245	Entry	Shaurya
E542	Jasmine	16-FEB-17	308000	D247	BackOfc			
						D194	Management	Akash

8 rows selected.

select * from employee natural join department;

//natural join

DID	DNAME	EID	ENAME	DOJ	SALARY	MGR
D120	Sales	E203	Abhijit	19-OCT-19	56000	Shanaya
D420	Marketing	E049	Sumit	08-JAN-19	65000	Akash
D420	Marketing	E713	Priyam	01-NOV-16	86000	Akash
D120	Sales	E629	Ranjita	29-OCT-18	49000	Shanaya
D846	IT	E191	Abhijit	18-JUL-17	60000	Inaya
D245	Entry	E432	Zoya	18-JUL-17	30000	Shaurya

ASSIGNMENT 1

Rule- connect sys/oracle as sysdba

1. Write a query to create a table employee with empno, ename, designation and salary.
2. Write a query to display the column name and data type of the table employee
3. Write a query to create a table from an existing table with all the fields.
4. Write a query to create table from an existing table with selected fields.
5. Write a query to create a new table from an existing table without any record.
6. Write a query to Alter the column empno number(4) to empno number(6).
7. Write a query to Alter the table employee with multiple columns (empno, ename).
8. Write a query to add a new column in employee table.
9. Write a query to add multiple columns in employee table.
10. Write a query to drop a column from an existing table employee.
11. Write a query to drop multiple columns from the employee table.
12. Write a query to rename table employee to emp

```
use assignment_1;
create table employee(empno binary(10),ename varchar(10),designation varchar(10),salary
binary(20));
desc employee;
```

```
select * from employee;
create table emp as select * from employee where 1>1;
```

```
select * from emp;
create table emp1 as select * from emp where 1>1;
```

```
create table emp2 as select * from employee where 1=0;
```

```
alter table employee modify empno binary(6);
```

```
alter table employee modify ename varchar(20);
alter table employee modify empno binary(12);
```

```
alter table employee add (doj date);
```

```
alter table employee add (dept varchar(10),age binary(10));
```

```
alter table employee drop column age;
```

```
alter table employee drop column doj;
alter table employee drop column dept;
```

```
alter table emp1 rename to employee1;
```

1. Create a table employee with attributes emp_id, f_name, l_name, job_type, salary, dept, commission, manager_id.
2. Make emp_id as the primary key of employee table.
3. Make f_name and salary NOT NULL type.
4. Add a column date_of_joining in the employee table.
5. Create a table department with attribute d_name, d_loc and HOD_id where d_name is primary key.
6. Create a table location with attributes loc_id, city and contact_no.
7. Enhance the size of the 'city' attribute by 5, in the location table.
8. Delete the contact_no attribute from the location table.
9. Make the department attribute of the employee table its foreign key referencing the department table.
10. Rename the city attribute to 'address' in the location table.
11. Rename the location table name to 'loc'.
12. Insert the following rows in 'loc' table

loc_id	address
1	Kolkata
2	Mumbai

13. Truncate the table 'loc'.
14. Drop the table 'loc'.
15. Insert the following rows in the department table:

d_name	d_loc	HOD_id
Sales	Kol	4
Accounts	Delhi	6
Production	Kol	1
Marketing	Kol	2
R & D	Marketing	8

16. Insert the following rows in the employee table:

Emp id	E_Name	L_name	Job_type	Salary	Comission	Name_idDOJ
1	Arun	Khan	Manager	90000	500	5 04-Jan-1999

17. Show the values of departmental table.
18. Select the department names and their locations.
19. Show the employees f_name, l_name, salary and the salary after 1000rs. Bonus.
20. Show the employees annual salary with a 1000rs. Yearly bonus and the annual salary with a 100rs. Monthly bonus.
21. Show f_name as NAME and annual salary as ANNSAL from the employee table.
22. Show the l_name as LasT AND 100rs. Incremented salary as NewSal.
23. Show the emp_id, f_name, l_name, job_type of the employee getting highest salary.
24. Show the emp_id, f_name, l_name, job_type of the employee getting minimum salary.
25. Show the average salary of employees in the employee table.
26. Consider the Insurance database given below. The primary keys are underlined and the data types are specified:
 PERSON (driver-id: string, name: string, address: string)
 CAR (Regno:string,model:string,year:int)
 ACCIDENT (report-number:int,date:date,location:string)
 OWNS (driver-id:string,regno:string)
 PARTICIPATED (driver-id:string,regno:string,report-number:int,damage-amount:int)
- i. Create the above tables by properly specifying the primary keys and the foreign keys
- ii. Enter atleast five tuples for each relation
- iii. Demonstrate how you a. Update the damage amount for the car with a specific regno in accident with report number 12 to 25000 b. Add a new accident to the database
- iv. Find the total number of people who owned cars that were involved in accidents in 2006.
- v. Find the number of accidents in which cars belonging to a specific model were involved.

Consider the following employee table and execute the queries based on it

Emp id	E_Name	L_name	Job_type	Salary	Comission	Name_idDOJ
1	Arun	Khan	Manager	90000	500	5 04-Jan-1999
.
.
.
.

1. Show f_name, L_name and job_type from employees.
2. Show employee details in the following fashion:

Employee details

Arun is a manager

3. Show the monthly salary details in the following fashion

Monthly Salary Details

Arun's monthly salary is Rs. 90000

Consider the Department table to answer the queries

d_name	d_loc	HOD_id
Sales	Kol	4
Accounts	Delhi	6
Production	Kol	1
Marketing	Kol	2
R & D	Marketing	8

4. Show the different department names from department table
5. Show the employee names who works in 'Sales'
6. Show the employee names who gets salary of more than 50000 per month
7. Show the details of the employee whose manager id is not 1
8. Show the employee details whose salary ranges between 40000 and 70000
9. Show the details of the employees who works under the manager having id 1, 6 and 8
10. Select the f_name and salary of those employees whose last name starts with 'K'
11. Select the f_name and salary of those employees whose last name starts with 'K' and ends with 'R'
12. Show the details of those employees where 3rd letter of L_name is 'o'
13. Select the details of those employees who works as an engineer with monthly salary more than 50000
14. Select the employees whose department is 'Production' or monthly salary is more than 60000 per month.
15. Find the minimum salary, maximum salary, total salary, average salary of the

employees who work in 'Sales' department

16. Find the employee L_name that is first and f_name that is last if they are arranged in an order
17. Find the number of employees working in each department
18. Find the number of departments from employee table
19. Find the average commission of the employees.
20. Find the average salaries of the employees department wise
21. Find the sum of salary of different job_type according to different departments
22. Find the department name and average salaries of those departments whose average salary is greater than 40000
23. Find the department name and maximum salaries of those departments whose maximum salary is greater than 55000
24. Display the job_type and total monthly salary for each job_type where total payroll is exceeding 100000
25. Display the name of the department having maximum average salary

1. Show the use of upper and lower function.
2. Show the use of concat, instr and length function
3. Show the use of the following functions on numeric values:
 - a. Sqrt()
 - b. Power()
 - c. Ceil()
 - d. Substr()
 - e. Max()
 - f. min()

- g. Round()
- h. avg()
- i. count()
- j. Exp()
- k. mod()
4. Solve the following queries
 - a. Find the ceiling and floor value of 14.887.
 - b. Find out the round-off 17.49989.
 - c. Calculate 87
5. Show the current date
6. Find the total experience of the employees in weeks who works in Sales department
7. Display the use of the following functions on date
 - a. Months_between
 - b. Add_months
 - c. Next_day
 - d. Last_day
 - e. Round
 - f. Trunc
 - g. To_char
8. Show the employee details with a revised salary. The salary is incremented in the following way:
 - a. 10% for sales department
 - b. 20% for marketing department
 - c. No increment for others
9. Determine the tax for each employee in production department based on the monthly salary. The tax rate are as per the following data:

Monthly Salary	Range Rate
0 – 19,999	0%
20,000 – 39,999	9%
40,000 – 59,999	20%
60,000 – 79,999	30%
80,000 or more	45%

10. Find the Cartesian product between Employee and Department table.
11. Show the employee names and the respective department location.
12. Give an example of the following joins considering employee and department tables.
 - k. Natural join
 - l. Inner join
 - m. Left outer join
 - n. Right outer join
 - o. Full outer join

13. Write a query to find the addresses(location_id,street_address,city,state_province,country_name) of all departments
 HINT : Use NATURAL JOIN.
 Sample Table : locations

location_id	street_address	postal_code	city	state_province	country_id
-------------	----------------	-------------	------	----------------	------------

worker_ref_id bonus_date bonus_amount

worker_ref_id worker_title affected_form

1. Write An SQL Query To Fetch "FIRST_NAME" From Worker Table In Upper Case alias as WORKER_FIRSTNAME.
2. Write An SQL Query To Print The First Three Characters Of FIRST_NAME From Worker Table.
3. Write An SQL Query To Find The Position Of The Alphabet ('A') In The First Name Column 'Amitabh' From Worker Table.
4. Write An SQL Query To Print The FIRST_NAME And LAST_NAME From Worker Table Into A Single Column COMPLETE_NAME. A Space Char Should Separate Them.
5. Write An SQL Query To Print All Worker Details From The Worker Table Order By FIRST_NAME Ascending And DEPARTMENT Descending.
6. Write An SQL Query To Print Details Of The Workers Whose FIRST_NAME Contains 'A'.
7. Write An SQL Query To Print Details Of The Workers Whose FIRST_NAME Ends With 'A'.
8. Write An SQL Query To Print Details Of The Workers Whose SALARY Lies Between 100000 And 500000.
9. Write An SQL Query To Fetch The Count Of Employees Working In The Department 'Admin'.
10. Write An SQL Query To Fetch The No. Of Workers For Each Department In The Descending Order.
11. Write An SQL Query To Print Details Of The Workers Who Are Also Managers.
12. Write An SQL Query To Show Only Odd Rows From A Table
13. Write An SQL Query To Show Records From One Table That Another Table Does Not Have.
14. Write An SQL Query To Show The Top N (Say 10) Records Of A Table.
15. Write An SQL Query To Fetch The List Of Employees With The Same Salary.

16. Write An SQL Query To Show All Departments Along With The Number Of People Working There.
17. Write An SQL Query To Print The Name Of Employees Having The Highest Salary In Each Department.
18. Write An SQL Query To Fetch Departments Along With The Total Salaries Paid For Each Of Them.
19. Consider the following relations for an order processing database application in a company.
CUSTOMER (Cust #: int, Cname: string, City: string)
ORDER (Order #: int, Odate: date, Cust #: int, Ord-Amt: int) ORDER-ITEM (Order #: int, Item #: int, qty: int)
ITEM (Item #: int, Unit Price: int)
SHIPMENT (Order #: int, Warehouse #: int, Ship-Date: date) WAREHOUSE (Warehouse #: int, City: string)
- i) Create the above tables by properly specifying the primary keys and the foreign keys.
- ii) Enter at least five tuples for each relation.
- iii) Produce a listing: CUSTNAME, NO_OF_ORDERS, AVG_ORDER_AMT, where the middle column is the total number of orders by the customer and the last column is the average order amount for that customer
- iv) List the Order# for the orders that were shipped from all the warehouses that the company has in a specific city
- v) Demonstrate how you delete Item# 10 from the ITEM table and make that field null in the ORDER- ITEM table.
20. Create a table Emp(e_no, e_name, e_phone, e_addr, e_salary) to store records of 10 employees:
 - i) Alter the data type of e_no from number to varchar
 - ii) Alter table by setting e_no as primary key
 - iii) Alter table by adding a column e_pin
 - iv) Update the phone number of an employee in the table
21. Create a table Dept(dept_no, dept_name, e_no, dept_loc, dept_hod) to store records of 10 departments:
 - i) Create the reference between Emp and Dept table with e_no attribute.
 - ii) Assign dept_no as primary key.
 - iii) Update the dept_hod for one department.
 - iv) Delete one department.
22. Solve the following queries
 - i) Write a query to find the employee name and dept_hod whose dept_hod is SAY, "John".
 - ii) Write a query to find the average salary of the employee of CSE department.

ASSIGNMENT 8

1. The following tables are maintained by a book dealer

AUTHOR(author-id: int, name: string, city: string, country: string)

PUBLISHER(publisher-id: int, name: string, city: string, country: string)

CATALOG(book-id: int, title : string, author-id: int, publisher-id: int, category: int, year: int, price: int)

CATEGORY(category-id: int, description: string)

ORDER-DETAILS(order-no: int, book-id: int, quantity: int)

i) Create above tables by properly specifying the primary keys and the foreign keys.

ii) Enter atleast five tuples for each relation.

iii) Give the details of the authors who have 2 or more books in the catalog and the price of the books is

greater than the average price of the books in the catalog and the year of publication is after 2010.

iv) Find the author of the book which has maximum sales.

v) Demonstrate how to increase price of books published by specific publisher by 10%

2. Consider the following database for BANK.

BRANCH(branch-name: string, branch-city: string, assets: real)

ACCOUNT(accno: int, branch-name: string, balance: real)

DEPOSITOR(customer-name: string, accno: int)

CUSTOMER(customer-name: string, customer-street: string, customer-city: string)

LOAN(loan-no: int, branch-name: string, amount: real)

BORROWER(customer-name: string, loan-no: int)

i) Create the above tables by properly specifying the primary keys and foreign keys.

ii) Enter atleast five tuples for each relation.

iii) Find all the customers who have atleast two accounts at the main branch.

iv) Find all customer who have an account at all the branches located in a specific city.

v) Demonstrate how to delete all account tuples at every branch located in specific city.

3. Consider the following database for ORDER PROCESSING.

CUSTOMER(cust-no: int, cname: string, city: string)

ORDER(orderno: int, odate: date, ord-amt: real)

ORDER_ITEM(orderno: int, itemno: int, qty: int)

ITEM(itemno: int, unitprice: real)

SHIPMENT(orderno: int, warehouseno: int, ship-date: date)

WAREHOUSE(warehouseno: int, city: string)

i) Create the above tables by properly specifying the primary keys and the foreign keys

ii) Enter atleast five tuples for each relation.

iii) List the order number and ship date for all orders shipped from particular warehouse

iv) Produce a listing: customer name, no of orders, average order amount

v) List the orders that were not shipped within 30 days of ordering

University of Engineering & Management