DBMS SLIPS – ASSIGNMENT 2

NAME : SOHAIL SHAIKH

CLASS : FY BBA(CA)

INVIGILATOR : VAISHALI KARKARE

COLLEGE : SMT. SUDHATAI MANDKKE COLLEGE.

SUBMISSION DATE : 18/10/2023

Slip no1:

Consider the following entities and their relationships.

Create a RDB in 3 NF with appropriate data types and Constraints. [15 Marks]

Emp(eno ,ename ,designation ,salary, Date_Of_Joining)

Dept(dno,dname ,loc)

The relationship between Dept & Emp is one-to-many. Constraints: - Primary Key, ename should not be NULL, salary must be greater than 0.

SQL> create table emp(eno number primary key,ename varchar(20),designation varchar(20),salary number,date_of_joining varchar(20));

SQL> desc emp;

SQL> insert into emp(eno,ename,designation,salary,date_of_joining)

2 values(1,'Mr. Advait','Assistant',54000,'23/03/2002');

SQL> insert into emp(eno,ename,designation,salary,date_of_joining)

2 values(2,'Mr. Roy','ceo',50000,'15/06/2019');

SQL> insert into emp(eno,ename,designation,salary,date_of_joining)

2 values(3,'Mr. Abhay','manager',60000,'10/06/2013');

SQL> insert into emp(eno,ename,designation,salary,date_of_joining)

2 values(4,'Mr. Raghav','manager',420000,'01/03/2003');

1 row created.

SQL> select * from emp;

SQL> create table dept(dno number primary key,dname varchar(20),loc varchar(10),eno references emp);

SQL> desc dept

SQL> insert into dept(dno,dname,loc,eno)

2 values(101,'computer','pune',1);

SQL> insert into dept(dno,dname,loc,eno)

2 values(102,'computer science','mumbai',2);

SQL> insert into dept(dno,dname,loc,eno)

2 values(103,'Quqlity','mumbai',3);

SQL>

SQL> insert into dept(dno,dname,loc,eno)

2 values(104,'Account','mumbai',4);

SQL> select * from dept;

Q.3 Consider the above tables and Execute the following queries:

1. Add column phone_No into Emp table with data type int.

SQL> alter table emp

2 add phone_no int;

SQL> desc emp;

2. Delete the details of Employee whose designation is 'Manager'.

SQL> Delete from emp

2 where designation='manager';

Q4. Consider the above database and execute the following queries: [25 Marks]

1. **Display the count of employees department wise.**

   SQL> select count(emp.eno),dname from emp,dept

   2 where emp.eno=dept.eno

   3 group by dname;

2. **Display the name of employee who is 'Manager' of "Account Department".**

   SQL> select ename from emp,dept

   2 where emp.eno=dept.eno

3 and designation='manager' and dname='Account';

3. **Display the name of department whose location is “Pune” and “Mr. Advait” is working in it**

SQL> select dname from emp,dept

2 where emp.eno=dept.eno

3 and loc='pune' and ename='Mr. Advait';

4. **Display the names of employees whose salary is greater than 50000 and department is “Quality”.**

SQL> select ename from emp,dept

2 where emp.eno=dept.eno

3 and salary>50000 and dname='Quqlity';

5. **Update Dateofjoining of employee to ‘15/06/2019’ whose department is ‘computer science’ and name is “Mr. Roy’.**

update emp set date_of_joining='15/06/2019'

where ename='Mr.Roy' and dno in(select dno from dept where dname='computer science');

slip no:2--Q3. Consider the following entities and their relationships. Create a

RDB in 3 NF with appropriate data types and Constraints. [15 Marks]

Sales_order (ordNo, ordDate)

Client (clientNo, ClientName, addr)

The relationship between Client & Sales_order is one-to-many.

Constraints: - Primary Key, ordDate should not be NULL

SQL> create table client(cno varchar(10) primary key,cname varchar(20),addr varchar(20));

SQL> desc client

SQL> insert into client values('CN001','Abhay','Pune');

SQL> insert into client values('CN002','Patil','Pune');

SQL> insert into client values('CN003','Mr.Roy','Pimpri');

SQL> insert into client values('CN004','Raj','Mumbai');

SQL> select * from client;

SQL> create table sales_order(ordno int primary key,ordDate varchar(23) not null,

cno varchar(10) references client on delete cascade);

SQL> desc sales_order;

SQL> insert into sales_order values(1,'23/06/2015','CN001');

SQL> insert into sales_order values(2,'09/03/2019','CN002');

SQL> insert into sales_order values(3,'09/08/2009','CN004');

SQL> insert into sales_order values(4,'09/08/2019','CN002');

SQL> select * from sales_order;

Q.3Consider the above tables and execute the following queries:

1. Add column amount into Sales_order table with data type int.

SQL> alter table sales_order

2 add amount int;

Table altered.

SQL> desc sales_order;

2. Delete the details of the clients whose names start with 'A' character.

SQL> delete from client

2 where cname like'A%';

1 row deleted.

SQL> select * from client;

Q4. Consider the above tables and execute the following queries: [25 Marks]

1. **Delete sales order details of client whose name is "Patil" and order date is "09/08/2019".**

SQL> delete from sales_order

2 where ordDate='09/08/2019'

3 and cno in(select cno from client where cname='Patil');

1 row deleted.

SQL> select * from sales_order;

2)Change order date of client_No ‘CN001’ ‘18/03/2019’.

SQL> update sales_order

2 set ordDate='18/03/2019'

3 where cno='CN001';

0 rows updated.

3) Delete all sales_record having order date is before ‘10 /02/2018’.

SQL> delete from sales_order

2 where ordDate<'20/10/2019';

2 rows deleted.

4)Display date wise sales_order given by clients.

SQL> select ordDate,ordno,amount,cno from sales_order

2 order by ordDate;

no rows selected

5) Update the address of client to “Pimpri” whose name is ‘Mr. Roy’

SQL> update client

2 set addr='pimpri'

3 where cname='Mr.Roy';

1 row updated.

Slip no-3:-Q3. Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints. [15 Marks]

Hospital (hno ,hname , city, Est_year, addr)

Doctor (dno , dname , addr, Speciality)

The relationship between Hospital and Doctor is one - to – Many Constraints: - Primary Key, Est_year should be greater than 1990.

SQL> create table hospital(hno int primary key,hname varchar(20),city varchar(20),est_year numeric(4) check(est_year>1990),addr varchar(20));

Table created.

SQL> desc hospital;

SQL> insert into hospital values(101,'balaji','pune',1993,'kharadi road');

1 row created.

SQL> insert into hospital values(103,'vedant','mumbai',1993,'dharavi');

1 row created.

SQL> insert into hospital values(104,'ruby','pimpri',1993,'kharadi road');

1 row created.

SQL> insert into hospital values(105,'birla','chinchwad',1993,'tyr');

1 row created.

SQL> insert into hospital values(106,'qw','pune',1993,'kalptaru');

1 row created.

SQL> select * from hospital;

SQL> create table doctor(dno int primary key,dname varchar(20),addr1 varchar(20),speciality varchar(20),hno int references hospital on delete cascade);

Table created.

SQL> desc doctor;

SQL> insert into doctor values(1,'dr.joshi','pune','skin',104);

1 row created.

SQL> insert into doctor values(2,'dr.mane','nashik','surgeon',103);

1 row created.

SQL> insert into doctor values(3,'dr.patil','pune','gynecologist',101);

1 row created.

SQL> insert into doctor values(4,'dr.Raghav','pune','skin',105);

1 row created.

SQL> insert into doctor values(5,'dr.Abhay','mumbai','internist',104);

1 row created.

SQL> insert into doctor values(6,'dr.joshi','pune','surgeon',106);

1 row created.

SQL> insert into doctor values(7,'dr.Riya','pune','skin',103);

1 row created.

SQL> insert into doctor values(8,'dr.Gawade','pune','head',104);

1 row created.

SQL> select * from doctor;

8 rows selected.

Q.3Consider the above tables and execute the following queries:

1. Delete addr column from Hospital table.

SQL> alter table hospital

2 drop column addr1;

2. Display doctor name, Hospital name and specialty of doctors from “Pune City” .

SQL> select dname,hname,speciality from doctor,hospital

2 where doctor.hno=hospital.hno

3 and city='pune';

Q4. Consider the above tables and execute the following queries: [25 Marks]

1. **Display the names of the hospitals which are located at “Pimpri” city.**

SQL> select hname from hospital,doctor

2 where doctor.hno=hospital.hno

3 and city='pimpri';

2. **Display the names of doctors who are working in “Birla” Hospital and city name is “Chinchwad”**

SQL> select dname from doctor,hospital

2 where doctor.hno=hospital.hno

3 and hname='birla' and city='chinchwad';

3. **Display the specialty of the doctors who are working in “Ruby” hospital.**

```
SQL> select speciality from hospital,doctor
  2  where doctor.hno=hospital.hno
  3  and hname='ruby';
```

4. **Give the count of doctor’s hospital wise which are located at “Pimple Gurav”.**

```
SQL> select hname,count(dno) from doctor,hospital
  2  where doctor.hno=hospital.hno
  3  and addr='kharadi road'
  4  group by hname;
```

5. **Update an address of Doctor to “Pimpri” whose hospital is “Ruby clinic”**

```
SQL> update doctor set addr1='pimpri'
  2  where hno in(select hno from hospital where hname='ruby');

3 rows updated.
```

Slip no-4:Q3. Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints. [15 Marks]

Patient (PCode, Name, Addr, Disease)

Bed (Bed_No, RoomNo, loc)

Relationship: - There is one-one relationship between patient and bed.
Constraints: - Primary key, RoomNo must be greater than Bed_No, Addr should not be null.

SQL> create table patient(pcode int primary key,name varchar(20) not null,addr varchar(20),disease varchar(10));

Table created.

SQL> desc patient;

SQL> insert into patient values(11,'Raghav','pimple gurav','listeria');

1 row created.

SQL> insert into patient values(12,'Abhay','pune','norovirus');

1 row created.

SQL> insert into patient values(13,'Mr.Roy','mumbai','cholera');

1 row created.

SQL> insert into patient values(14,'Sachin','pimple gurav','dengue');

1 row created.

SQL> insert into patient values(15,'Priya','nashik','listeria');

1 row created.

SQL> select * from patient;

SQL> create table bed(bno int primary key,rno int not null,loc varchar(10) not null,pcode int references patient on delete cascade);

Table created.

SQL> desc bed;

SQL> insert into bed values(1,105,'pune',11);

1 row created.

SQL> insert into bed values(2,102,'2nd floor',12);

1 row created.

SQL> insert into bed values(3,103,'4th floor',13);

1 row created.

SQL> insert into bed values(4,104,'1st floor',11);

1 row created.

SQL> insert into bed values(5,105,'3rd floor',14);

1 row created.

SQL> insert into bed values(6,106,'2nd floor',15);

1 row created.

SQL> select * from bed;

6 rows selected.

Q.3Consider the above tables and execute the following queries:

1. Display the details of patients who are from “Pimple Gurav”

SQL> select * from patient

2 where addr='pimple gurav';

2. **Delete the details of patient whose Bed_No is 1 and RoomNo is 105.**

SQL> select * from patient,bed

2 where patient.pcode=bed.pcode

3 and bno=1 and rno=105;

Q4. Consider the above tables and execute the following queries: [25 Marks]

1. **Display the count of patient room wise**.

SQL> select count(patient.pcode) from patient,bed

2 where patient.pcode=bed.pcode

3 group by rno;

2. **Display the names of patients who are admitted in room no 101.**

SQL> select name from patient,bed

2 where patient.pcode=bed.pcode

3 and rno=102;

3. **Display the disease of patient whose bed_No is 1**

```
SQL> select disease from patient,bed
  2  where patient.pcode=bed.pcode
  3  and bno=1;
```

4. **Display the room_no and bed_no of patient whose name is “Mr Roy”**

```
SQL> select rno,bno from patient,bed
  2  where patient.pcode=bed.pcode
  3  and name='Mr.Roy';
```

5. **Give the details of Patient who is admitted on 2nd flr in roomno 102.**

```
SQL> select * from patient,bed
  2  where patient.pcode=bed.pcode
  3  and loc='2nd floor' and rno=102;
```

Slip no-5:Q3. Consider the following entities and their relationships.

Create a RDB in 3 NF with appropriate data types and Constraints. [15 Marks]

Customer (cust_no, cust_name, address, city)

Loan (loan_no, loan_amt)

The relationship between Customer and Loan is Many to Many Constraint:

Primary key, loan_amt should be > 0.

Connected.

SQL> create table customer(cno int primary key,cname varchar(20) not null,addr varchar(20),city varchar(10));

Table created.

SQL> desc customer

SQL> insert into customer values(101,'Dhiraj','kharadi','pune');

1 row created.

SQL> insert into customer values(102,'Patil','kalptaru','pimpri');

1 row created.

SQL> insert into customer values(103,'Abhay','west','pimpri');

1 row created.

SQL> insert into customer values(104,'Raghav','rt','nashik');

1 row created.

SQL> insert into customer values(105,'Dhanu','bvh','pune');

1 row created.

SQL> select * from customer;

SQL> create table loan(lno int primary key,lamt int check(lamt>0),cno int references customer on delete cascade);

Table created.

SQL>

SQL> insert into loan values(1,120000,101);

1 row created.

SQL> insert into loan values(2,100000,102);

1 row created.

SQL> insert into loan values(3,30000,103);

1 row created.

SQL> insert into loan values(4,120,104);

1 row created.

SQL> insert into loan values(5,1000000,105);

1 row created.

SQL> select * from loan;

Q.3Consider the above tables and execute the following queries:

1. Add Phone_No column in customer table with data type int.

SQL> alter table customer

2 add phone_no int;

Table altered.

SQL> desc customer

2)Delete the details of customer whose loan_amt<1000.

Delete cno,cname,addr,city, from customer

Where customer.cno=loan.cno

And lamt<1000;

Q4. Consider the above tables and execute the following queries: [25 Marks]

1. **Find details of all customers whose loan_amt is greater than 10 lack.**

SQL> select * from customer,loan

2 where customer.cno=loan.cno

3 and lamt>1000000;

no rows selected

2. **List all customers whose name starts with 'D' character.**

SQL> select * from customer

2 where cname like 'D%';

3. List the names of customer in descending order who has taken a loan from Pimpri city.

SQL> select * from customer

2 where city='pimpri'

3 order by cname desc;

4.Display customer details having maximum loan amount

SQL> select max(lamt) from customer,loan

2 where customer.cno=loan.cno;

5.Update the address of customer whose name is “Mr. Patil” and loan_amt is greater than 100000.

update customer set addr='pune'

where cname='patil' and lno in(select lno from laon where lamt>100000);

Q3. Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints. [15 Marks]

Project (pno, pname, start_date, budget, status) Department (dno, dname, HOD, loc)

The relationship between Project and Department is Many to One. Constraint: Primary key. Project Status Constraints:

C – Completed,

P - Progressive,

I – Incomplete

SQL> create table project(pno int primary key,pname varchar(20),sdate date,budget int,status varchar(20) check(status in('c','i','p')));

Table created.

SQL> desc project;

SQL> insert into project values(1,'abc','09/mar/20',2300000,'c');

1 row created.

SQL> insert into project values(2,'xyz','01/apr/18',200000,'i');

1 row created.

SQL> insert into project values(3,'st','23/mar/27',1200000,'p');

1 row created.

SQL> insert into project values(4,'vb','12/feb/20',600000,'c');

1 row created.

SQL> insert into project values(5,'qrt','16/jan/23',3400000,'p');

1 row created.

SQL> select * from project;

SQL> create table department(dno int primary key,dname varchar(20),hod varchar(20),loc varchar(20),pno int references project on delete cascade);

Table created.

SQL> desc department

SQL> insert into department values(101,'computer','desai','pune',1);

1 row created.

SQL> insert into department values(102,'commerce','mane','pune',2);

1 row created.

SQL> insert into department values(103,'computer','kadam','pune',3);

1 row created.

SQL> insert into department values(104,'engineering','sam','pune',4);

1 row created.

SQL> select * from department;

Consider the above tables and execute the following queries:

1. Drop loc column from department table.

alter table department

drop column loc;

2. Display the details of project whose start_date is before one month and status is “Progressive”

SQL> select * from project

2 where sdate>'12/feb/20' and status='p';

Q4. Consider the above tables and execute the following queries: [25 Marks]

1. **Display the names of project and department who are worked on projects whose status is ‘Completed’**

SQL>

```
SQL> Select pname,dname,hod,loc from department,project
  2  where department.pno= project.pno
  3  and project.status='c';
```

2. **Display total budget of each department.**

```
SQL> Select sum(budget),dname from department,project
  2  where department.pno=project.pno
  3  group by dname;
```

3. **Display incomplete project of each department.**

```
SQL> select pname,status ,count(department.dno) from department,project
  2  where department.pno=project.pno
  3  and project.status='i'
  4  group by status,pname;
```

4. **Display all project working under 'Mr.Desai'.**

```
SQL> Select pname from department,project
  2  where department.pno=project.pno
  3  and hod= 'desai';
```

PNAME

Abc

5.Display department wise HOD.

SQL> select dname,hod from department,project

2 where department.pno=project.pno

3 order by dname;

slip no_7:Q3. Consider the following entities and their relationships.

Create a RDB in 3 NF with appropriate data types and Constraints. [15 Marks]

Room (roomno, desc, rate)

Guest (gno, gname, no_of_days)

The relationship between Room and Guest is One to One. Constraint:

Primary key, no of days should be > 0.

SQL> create table room(rno int primary key,des varchar(20),rate number);

Table created.

SQL> desc room;

SQL> insert into room values(101,'A/C',1500);

1 row created.

SQL> insert into room values(102,'Non A/C',750);

1 row created.

SQL> insert into room values(103,'A/C',2000);

1 row created.

SQL> insert into room values(104,'Non A/C',1200);

1 row created.

SQL> select * from room;

SQL> create table guest(gno int primary key,gname varchar(20),nod number check (nod>0));

Table created.

SQL> desc guest;

SQL> insert into guest values(101,'Mr.Bharat',3);

1 row created.

SQL> insert into guest values(102,'Mr.Nilesh',4);

1 row created.

SQL> insert into guest values(103,'Mr.Advait',7);

1 row created.

SQL> insert into guest values(104,'Miss.Sapana',2);

1 row created.

SQL> select * from guest;

Consider the above tables and execute the following queries:

1. **Update the rate of room to 5000 whose type is "AC"**

```
SQL> update room set rate=5000
  2  where des='A/C';

2 rows updated.

SQL> select * from room;
```

2. **Display the name of guest who is staying 2 days in roomno 101**

```
select gname from room,guest
where room.rno=guest.rno
and nod=2 and rno=101;
```

Q4. Consider the above tables and execute the following queries: [25 Marks]

1. **Display room details according to its rates in ascending order**

```
SQL> select des,rate from room
  2  order by des asc;
```

2. **Display the roomno in which "Mr. Advait" is staying for 7 days**

select rno from room,guest

where room.rno=guest.rno

and gname='Mr.Advait' and nod=7;

3. **Find no. of AC rooms.**

SQL> select count(rno) from room

2 where des='A/C';

4. **Find names of guest with maximum room charges.**

select gname from room,guest

where guest.rno=room.rno

and rate=(select max(rate) from room);

5. **Display guest wise halt days.**

Select gname,nod from guest

Order by gname;

SQL> Select gname,nod from guest

2 Order by gname;

Slip_no 8:Q3. Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints. [15 Marks]

Book (Book_no, title, author, price, year_published) Customer (cid, cname, addr)

Relation between Book and Customer is Many to Many with quantity as descriptive attribute. Constraint: Primary key, price should be >0;

SQL> create table book(bno int primary key,title varchar(10),author varchar(20),

price int check(price>0),yp number);

Table created.

SQL> desc book;

SQL> insert into book values(101,'dreams','mr.Raj',150,2017);

1 row created.

SQL> insert into book values(102,'life','mr.Raghav',100,2019);

1 row created.

SQL> insert into book values(103,'rt story','mr.Gadhave',190,2011);

1 row created.

SQL> insert into book values(104,'Dad','dr.Sam',200,2001);

1 row created.

SQL> insert into book values(105,'Struggle','mr.Raj',250,2017);

1 row created.

SQL> insert into book values(106,'Joker','Mr. Talore',230,2011);

1 row created.

SQL> select * from book;

6 rows selected.

SQL> create table customer(cid int primary key,cname varchar(20),addr varchar(20),bno int references book);

Table created.

SQL> desc customer;

SQL> insert into customer values(1,'Abhay','pune',101);

1 row created.

SQL> insert into customer values(2,'Sam','Mumbai',102);

1 row created.

SQL> insert into customer values(3,'Raghav','pimpri',103);

1 row created.

SQL> insert into customer values(4,'Abhay','mumbai',104);

1 row created.

SQL> insert into customer values(5,'Ganesh','Nashik',105);

1 row created.

SQL> select * from customer;

SQL> create table customerbook(bcid int primary key,bno int references book,

cid int references customer);

Table created.

SQL> desc customerbook;

SQL> insert into customerbook values(11,101,1);

1 row created.

SQL> insert into customerbook values(12,102,2);

1 row created.

SQL> insert into customerbook values(13,101,3);

1 row created.

SQL> insert into customerbook values(14,103,1);

1 row created.

SQL> insert into customerbook values(15,106,4);

1 row created.

SQL> select * from customerbook;

Consider the above tables and execute the following queries:

1.Display the name of book whose author is “Mr. Gadhave”.

```
SQL> select title from book
  2  where author='mr.Gadhave';
```

2.Add column EMailId into customer table.

```
SQL> alter table customer
  2  add emailID varchar2(20);
```

Table altered.

SQL> desc customer;

Q4. Consider the above tables and execute the following queries: [25 Marks]

1. **Display customer details from 'Mumbai'.**

```
SQL> select * from customer
  2  where addr='mumbai';
```

2. Display author wise details of book.

```
SQL> select author,title from book
  2  order by author;
```

```
6      rows selected.
```

3)Display customer name that has purchased more than 3 books.

```
SQL> select count(book.bno),cname from customer,book,customerbook
  2  where customer.cid=customerbook.cid
  3  and book.bno=customerbook.bno and book.bno>3
  4  group by cname;
```

3. Display book names having price between 100 and 200 and published year is 2019.

```
SQL> select book.title from book,customer,customerbook
  2  where customer.cid=customerbook.cid
  3  and book.bno=customerbook.bno
  4  and yp=2019 and price between 100 and 200;

TITLE
----------
```

life

5. Update the title of book to “DBMS” whose author is “Mr. Talore”.

SQL> update book set title='DBMS'

2 where author='Mr. Talore';

1 row updated.

SQL> select * from book;

6 rows selected.

Slip_no:9 Q3. Consider the following entities and their relationships. Create a

RDB in 3 NF with appropriate data types and Constraints. [15 Marks]

Property (pno, desc, area, rate)

Owner (owner_name, addr, phno) The relationship between owner and Property is One to Many. Constraint: Primary key, rate should be > 0

SQL> create table property(pno int primary key,des varchar(20) not null,area varchar(20) not null,rate int check(rate>0));

Table created.

SQL> desc property;

SQL> insert into property values(101,'vegr','nashik',1030000);

1 row created.

SQL> insert into property values(102,'tr','Pune',100000);

1 row created.

SQL> insert into property values(103,'vbh','pune',1030000);

1 row created.

SQL> insert into property values(104,'vsdr','mumbai',20000);

1 row created.

SQL> insert into property values(105,'hjjr','nashik',10000);

1 row created.

SQL> select * from property;

SQL> create table owner(name varchar(20),addr varchar(20),phno int,pno int references property);

Table created.

SQL> desc owner;

SQL> insert into owner values('Mr.Mane','Mumbai',1762386534,101);

1 row created.

SQL> insert into owner values('Mr.Patil','Mumbai',1762386534,102);

1 row created.

SQL> insert into owner values('Mr.Joshi','Pune',6892386534,103);

1 row created.

SQL> insert into owner values('Mr.Bhagat','Pune',6876783865,101);

1 row created.

SQL> insert into owner values('Mr.Abhay','Pune',6753386534,104);

1 row created.

SQL> select * from owner;

Consider the above tables and execute the following queries:

1. Display area of property whose rate is less than 100000

SQL> select area from property

2 where rate>100000;

2. Give the details of owner whose property is at “Pune”

SQL> select * from owner

2 where addr='Pune';

Q4. Consider the above tables and execute the following queries: [25 Marks]

1. Display area wise property details.

SQL> select area,des from property

2 order by area;

2. **Display property owned by 'Mr.Patil' having minimum rate.**

```
SQL> select min(rate) from property,owner
  2  where property.pno=owner.pno
  3  and name='Mr.Patil';
```

3. **Delete all properties from “pune” owned by “Mr. Joshi”.**

```
SQL> delete from owner
  2  where addr='Pune' and name='Mr.Joshi';

1 row deleted.

SQL> select * from owner;
```

4. Update the phone Number of “Mr. Joshi” to 9922112233 who is having property

at “Uruli Kanchan”

```
SQL> update owner set phno=9922112233
  2  where addr='Urali Kanchan';

1 row updated.
```

SQL> select * from owner;

5.Delete column address from Owner table.

Alter table owner

drop column addr;

slip_no-10:Q3. Consider the following entities and their relationships. Create a

RDB in 3 NF with appropriate data types and Constraints. [15 Marks]

Employee (emp_no, name, skill, payrate)

Position (posting_no, skill)

The relationship between Employee and Position is Many to Many with day and

shift as descriptive attribute. Constraint: Primary key, payrate should be > 0.

Connected.

SQL> create table employee(eno int primary key,name varchar(20),skill varchar(20) not null,payrate int check(payrate>0));

Table created.

SQL> desc employee;

SQL> insert into employee values(1,'Rghav','manager',23000);

1 row created.

SQL> insert into employee values(2,'Mane','waiter',23000);

1 row created.

SQL> insert into employee values(3,'Priya','ceo',23000);

1 row created.

SQL> insert into employee values(4,'Abhay','chef',23000);

1 row created.

SQL> select * from employee;

SQL> create table position(pno int primary key,skill varchar(20),eno int references employee);

Table created.

SQL> desc position;

SQL> insert into position values(201,'mg',1);

1 row created.

SQL> insert into position values(203,'ceo',2);

1 row created.

SQL> insert into position values(202,'wt',3);

1 row created.

SQL> insert into position values(205,'wdf',4);

1 row created.

SQL> insert into position values(204,'whd',2);

1 row created.

SQL> select * from position;

SQL> create table ep(epno int primary key,eno int references employee,pno int references position);

Table created.

SQL> desc ep;

SQL> insert into ep values(11,1,201);

1 row created.

SQL> insert into ep values(12,2,202);

1 row created.

SQL> insert into ep values(13,2,203);

1 row created.

SQL> insert into ep values(14,3,202);

1 row created.

SQL> insert into ep values(15,1,204);

1 row created.

SQL> select * from ep;

Consider the above tables and execute the following queries:

1. Display skill of employees name wise.

SQL> select name,skill from employee

2 order by name;

2)Update the posting of employee to 220 whose skill is “Manager”.

SQL> update position set pno=220

2 where skill='mg';

1 row updated.

SQL> select * from position;

6 rows selected.

Q4. Consider the above tables and execute the following queries: [25 Marks]

1. Find the names and rate of pay of all employees who has allocated a duty.

SQL> select name,payrate from employee;

2. Give employee number who is working at posting_no. 201, but don’t have the

skill of waiter

SQL> select employee.name,employee.skill from employee,position,ep

2 where employee.eno=ep.eno

3 and position.pno=ep.pno

4 and position.pno=201 and employee.skill not in('waiter');

no rows selected

3)Display a list of names of employees who have skill of chef and who has

assigned a duty.

select name from employee,position,ep

where employee.eno=ep.eno

and position.pno=ep.pno

and employee.skill='chef';

4. **Display shift wise employee details.**

SQL> select name,employee.skill from employee,position,ep

2 where employee.eno=ep.eno

3 and position.pno=ep.pno

4 group by employee.skill,name;

no rows selected

5. **Update payrate of employees to 20000 whose skill is waiter.**

SQL> update employee set payrate=20000

2 where skill='waiter';

1 row updated.

SQL> select * from employee;

Slip_no:11:Q3. Consider the following entities and their relationships. Create a

RDB in 3 NF with appropriate data types and Constraints. [15 Marks]

Bill (billno, day, tableno, total)

Menu (dish_no, dish_desc, price)

The relationship between Bill and Menu is Many to Many with quantity as descriptive attribute.

Constraint: Primary key, price should be > 0.

SQL> create table bill(bno int primary key not null,day varchar(10),tbno int,

total int);

Table created.

SQL> desc bill;

SQL> insert into bill values(301,'monday',109,1120);

1 row created.

SQL> insert into bill values(302,'sunday',123,9120);

1 row created.

SQL> insert into bill values(303,'tuesday',122,4200);

1 row created.

SQL> insert into bill values(304,'monday',176,2210);

1 row created.

SQL> select * from bill;

SQL> create table menu(dno int primary key not null,ddes varchar(10), price int check(price>0),bno int references bill);

Table created.

SQL> desc menu;

SQL> insert into menu values(101,'veg',200,301);

1 row created.

SQL> insert into menu values(102,'non-veg',300,303);

1 row created.

SQL> insert into menu values(103,'non-veg',400,301);

1 row created.

SQL> insert into menu values(104,'veg',250,301);

1 row created.

SQL> insert into menu values(105,'non-veg',800,302);

1 row created.

SQL> insert into menu values(106,'veg',600,304);

1 row created.

SQL> select * from menu;

6 rows selected.

SQL> create table bm(bmno int primary key,ddate varchar(10),bno int references bill,mno int references menu);

Table created.

SQL> desc bm;

SQL> insert into bm values(1,'12/02/10',301,102);

1 row created.

SQL> insert into bm values(2,'09/07/19',303,104);

1 row created.

SQL> insert into bm values(3,'02/06/11',302,101);

1 row created.

SQL> insert into bm values(4,'12/02/09',304,102);

1 row created.

SQL> select * from bm;

Consider the above tables and execute the following queries:

1. **Display the tableno whose dish_desc is “Veg”.**

SQL> select tno from menu,bill,bm

```
  2  where bill.bno=bm.bno
  3  and menu.mno=bm.mno
  4  and dis='veg';
```

2. **Display the special menu of Monday.**

```
SQL> select dis from bill,menu,bm
  2  where bill.bno=bm.bno
  3  and menu.mno=bm.mno
  4  and day='monday';
```

Q4. Consider the above tables and execute the following queries: [25 Marks]

1. **Display receipt which includes bill_no with Dish description, price, quantity and total amount of each menu.**

```
SQL> select sum(bill.total),menu.dis,menu.price,bm.qunt from bill,menu,bm
  2  where bill.bno=bm.bno
  3  and menu.mno=bm.mno
  4  group by menu.dis,menu.price,bm.qunt;
```

2)Find total amount collected by hotel on date 09/07/2019.

```
SQL> select sum(total) from bill,menu,bm
  2  where bill.bno=bm.bno
```

```
3  and menu.mno=bm.mno
4  and ddate='09/07/19';
```

3)Count number of menus of billno 301

```
SQL> select count(dis) from bill,menu,bm
  2  where bill.bno=bm.bno
  3  and menu.mno=bm.mno
  4  and bill.bno=301;
```

4)Display menu details having price between 100 and 500.

```
SQL> select dis,price from menu
  2  where price between 100 and 500;
```

5. Display the tableno and day whose bill amount is zero.

```
SQL> select tno,day from bill
  2  where total=0;

no rows selected
```

slip-no:12 Q3 Consider the following entities and their relationships. Create a

RDB in 3 NF with appropriate data types and Constraints. [15 Marks]

Movies (M_name, release_year, budget)

Actor (A_name, role, charges, A_address)

Producer (producer_id, name, P_address)

Relationship:- Each actor has acted in one or more movie. Each producer has produced many movies but each movie can be produced by more than one producers.

Each movie has one or more actors acting in it, in different roles.

Constraint: Primary key, release_year > 2000, A_address and P_address should not be same.

Consider the above tables and execute the following queries:

1. List the names of movies with the highest budget.

2. Display the details of producer who have produced more than one movie in a year.

Q4. Consider the above tables and execute the following queries: [25 Marks]

1. List the names of movies with the second highest budget 2. List the names of actors who have acted in the maximum number of movies.

3. List the names of movies, produced by more than one producer.

4. List the names of actors who are given with the maximum charges for their movie.

5. List the names of actors who have acted in at least one movie, in which 'Akshay' has acted.

Q3. Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints. [15 Marks]

Driver (driver_id, driver_name, address)

Car (license_no, model, year)

Relation between Driver and Car is Many to Many with date and time as descriptive attribute.

Constraint: Primary key, driver_name should not be null

SQL> create table driver(did int primary key,dname varchar(10),addr varchar(10));

Table created.

SQL> desc driver;

SQL> insert into driver values(101,'Raghav','pune');

1 row created.

SQL> insert into driver values(102,'ram','mumbai');

1 row created.

SQL> insert into driver values(103,'Abhay','pune');

1 row created.

SQL> insert into driver values(104,'Ganesh','Nanded');

1 row created.

SQL> insert into driver values(105,'Ritik','Nashik');

1 row created.

SQL> select * from driver;

SQL> create table car(lno varchar(10) primary key,model varchar(10),year number,did int references driver);

Table created.

SQL> desc car;

SQL> insert into car values('DPU123','w12b',1987,101);

1 row created.

SQL> insert into car values('DPU781','SUV300',2019,103);

1 row created.

SQL> insert into car values('DPU231','swif',2001,105);

1 row created.

SQL> insert into car values('DPU018','ty12',1999,102);

1 row created.

SQL> insert into car values('DPU810','nh79',2001,104);

1 row created.

SQL> select * from car;

SQL> create table dc(dco int primary key,did int references driver,lno varchar(10) references car);

Table created.

SQL> desc dc;

SQL> insert into dc values(301,101,'DPU123');

1 row created.

SQL> insert into dc values(302,102,'DPU781');

1 row created.

SQL> insert into dc values(303,103,'DPU123');

1 row created.

SQL> insert into dc values(304,101,'DPU018');

1 row created.

SQL> insert into dc values(305,105,'DPU810');

1 row created.

SQL> select * from dc;

Consider the above tables and execute the following queries:

1. Display the name of driver whose license no is “DPU123”.

```
SQL> select dname from driver,car,dc
  2  where driver.did=dc.did
  3  and car.lno=dc.lno
  4  and car.lno='DPU123';
```

2. Delete the details of car whose model is “swift”.

```
SQL> delete from car
  2  where model='swif';

1 row deleted.

SQL> select * from car;
```

Q4. Consider the above tables and execute the following queries: [25 Marks]

1. **Display details of all persons who are driving ‘Alto’ car**

```
SQL> select dname from driver,car,dc
  2  where driver.did=dc.did
  3  and car.lno=dc.lno
```

```
  4  and model='Alto';
```

2.Update model of car to “SUV300” whose manufactured year is 2019.

```
SQL> update car set model='SUV300'
  2  where year=2019;

2 rows updated.

SQL> select * from car;

6      rows selected.
```

3.Display car details manufactured before year 2000.

4.In which day ‘Mr. Ram’ drives maximum number of cars.

```
SQL> select count(car.model),dname from driver,car,dc
  2  where driver.did=dc.did
  3  and car.lno=dc.lno
  4  and dname='ram'
  5  group by dname;
```

5.Display total number of drivers who drives car in each year.

```
SQL> select count(driver.did),year,dname from driver,car,dc
  2  where driver.did=dc.did
  3  and car.lno=dc.lno
  4  group by year,dname;


6 rows selected.
```

2013 Pattern Slips

Slip 1: Consider the following Entities and Relationships

Customer (cust_no, cust_name, address, city)

Loan (loan_no, loan_amt)

Relation between Customer and Loan is **Many to Many**

Constraint: Primary key, loan_amt should be > 0.

```
Create table customer
(custnonumber(4) primary key,
cnamevarchar(20),dd
address varchar(20),
city varchar(20)
);

Insert into customer values(101,'bajirao','Maharastra chowk','Pimpri');
Insert into customer values(102,'Seeta','Bhosari','Pimpri');
Insert into customer values(103,'Geeta','Navi Peth','Nashik');
Insert into customer values(104,'Sachin','Sagavi','Nashik');

Create table loan
(loannonumber(4) primary key,
loanamtnumber(10) constraint loanchk check(loanamt>0)
```

);

Insert into loan values(201,'10000012');

Insert into loan values(202,'32467236');

Insert into loan values(203,'348118447');

Insert into loan values(204,'342349223');

Insert into loan values(205,'5697689122');

Create table cl

(custnonumber(4) references customer(custno),

loannonumber(4) references loan(loanno));

Insert into cl values (101,201);

Insert into cl values (101,202);

Insert into cl values (102,203);

Insert into cl values (104,205);

Insert into cl values (103,201);

- Find details of all customers whose loan is greater than 10 lakhs.

Select distinct cname,address,city,loanamt

```
From customer,loan,cl
Where customer.custno=cl.custno and
    loan.loanno=cl.loanno and
    loanamt> 1000000;
```

- List all customers whose name starts with 'ba'.

```
Select distinct cname
From customer,loan,cl
Where cname like 'ba%' and
customer.custno=cl.custno and
    loan.loanno=cl.loanno ;
```

- Display details of customer maximum loan amount.

```
Select distinct customer.custno,cname,address,city,loanamt
From customer,loan,cl
Where customer.custno=cl.custno and
    loan.loanno=cl.loanno and
        loanamt=(select max(loanamt) from loan);
```

- Calculate total of all loan amount

-

```
Select sum(loanamt)
from customer, loan, cl
where  customer.custno=cl.custno and
   loan.loanno=cl.loanno;
```

- List names of all customers in descending order who has taken a loan in Nasik city.me

```
Select distinct cname,city
From customer,loan,cl
Where customer.custno=cl.custno and
   loan.loanno=cl.loanno and
      city='Nashik' order by cnamedesc;
```

Slip 2 Consider the following Entities and Relationships

Department (dept_no, dept_name, location)

Employee (emp_no, emp_name, address, salary, designation)

Relation between Department and Employee is **One to Many**

Constraint: Primary key, salary should be > 0

Create table dept

```
(dnonumber(4) primary key,
Dnamevarchar(20),
location varchar(20));

insert into dept values(101,'computer','pune');
insert into dept values(102,'finance','pimpri');
insert into dept values(103,'computer','nashik');
insert into dept values(104,'warehouse','pune');
insert into dept values(105,'account','bhosali');

create table emp11
(enonumber(4) primary key,
enamevarchar(20),
address varchar(20),
salary number(6) constraint salch11 check (salary>0),
designation varchar(20),
dnonumber(4) references dept(dno));

insert into emp11 values(201,'seema','Pimpri','10253','Manager','101');
insert into emp11 values(202,'alok','pimpri','356854','HR','102');
insert into emp11 values(203,'anil','nashik','45854','IT','103');
```

insert into emp11 values(204,'rahul','pune','56253','HR','104');

insert into emp11 values(205,'atual','bhosali','1253','Supervicer','105');

insert into emp11 values(206,'geeta','Pimpri','10223','Manager','101');

write queries for following.

- Find total salary of all computer department employees.

```
Select sum(salary)as total_salary
From emp, dept
Where emp.dno=dept.dno and
      dname='computer';
```

- Find the name of department whose salary is above 10000.

```
Select distinct dname,salary
Fromemp, dept
wheredept.dno=emp.dno and
            salary>10000;
```

- Count the number of employees in each department.

```
Select dname,count(*) as count
from emp,dept
```

wheredept.dno=emp.dnogroup by dname;

- Display the maximum salary of each department.

select max(salary),dname from emp,dept

where dept.dno=emp.dno group by dname;

- Display department wise employee list.

select distinctdname, ename

from emp, dept

wheredept.dno=emp.dnoorder by dname;

Slip 3 Consider the following Entities and Relationships [30 Marks]

Project (pno, pname, start_date, budget, status)

Department (dno, dname, HOD)

Relation between Project and Department is **Many to One**

Constraint: Primary key.

Project Status Constraints: C – completed,

P-Progressive, I-Incomplete

create table dep1

```
(dnonumber(4)primary key,
dnamevarchar(20),
HOD varchar(10)
);

insert into dep1 values (101,'computer','Mr.Desai');
insert into dep1 values(102,'math','sanjay');
insert into dep1 values(103,'computer','rohit');
insert into dep1 values(104,'stat','pawar');
insert into dep1 values(105,'electronics','rakesh');

create table pro1
(pnonumber(4)primary key,
pnamevarchar(20),
sdate date,
budget number(5),
status varchar(25),
dnonumber(4) references dep1(dno));

insert into pro1 values(1,'mathematics',’1 mar 2013’,’500000’,'C',102);
insert into pro1 values(2,'cost',’23 feb 2014’,’3437’,'I',104);
```

insert into pro1 values(3,'mathematics',’3 apr 2011’,’2345096’,'P',102);

insert into pro1 values(4,'eng',’11 Nov 2013’,’23431’,’C’,105);

insert into pro1 values(5,’programming',’14 oct 2011’,’23453’, 'C',101);

Create a Database in 3NF & write queries for following.

- List the project name and department details worked in projects that are ‘Complete’.

```
select pname,dname,HOD,status from dep1,pro1
where dep1.dno=pro1.dno and
status=’C’;
```

- Display total budget of each department.

```
Select dname, sum(budget) from dep1,pro1
where dep1.dno=pro1.dno group by dname;
```

- Display incomplete project of each department

```
Select dname, status from dep1,pro1
Where status=’I’and
```

dep1.dno=pro1.dno group by dname,status;

- Find the names of departments that have budget greater than 50000

Select dname, budget from dep1,pro1

Where budget> 50000 and

dep1.dno=pro1.dno;

- Display all project working under 'Mr.Desai'.

Select pname from dep1,pro1

Where HOD='Mr.Desai' and

dep1.dno=pro1.dno;

Slip 4

Consider the following Entities and Relationships [30 Marks]

Room (roomno, desc, rate)

Guest (gno, gname, no_of_days)

Relation between Room and Guest is **One to One**.

Constraint: Primary key, no of days should be > 0.

```
create table room
(rnonumber(4)primary key,
roomtypevarchar(20),
rate varchar(10));

insert into room values(1,'ac',500);
insert into room values(2,'nonac',1500);
insert into room values(3,'ac',300);
insert into room values(4,'ac',800);
insert into room values(5,'nonac',600);

create table guest
(gnonumber(4),
gnamevarchar(10),
no_of_daysvarchar(22) constraint no_day check(no_of_days>0));

insert into guest values(1,'akshay',5);
insert into guest values(2,'sanjay',3);
insert into guest values(3,'raje',1);
insert into guest values(4,'rohit',5);
insert into guest values(5,'mane',4);
```

Create a Database in 3NF & write queries for following.

- Display room details according to its rates in ascending order.

select * from room order by rateasc;

- Find the names of guest who has allocated room for more than 3 days

select distinctgname from guest

where no_of_days>3;

.

- Find no. of AC rooms.

select distinct rno,roomtype from room,guest

where guest.gno=room.rno and

roomtype='ac';

- Display total amount for NON-AC rooms.

select roomtype, sum(rate) from room

where roomtype='nonac' group by roomtype;

- Find names of guest with maximum room charges.

select gnamefromguest,room

where guest.gno=room.rno and

rate=(select max(rate) from room);

Slip 5Consider the following Entities and Relationships [30 Marks]

Book (Book_no, title, author, price, year_published)

Customer (cid, cname, addr)

Relation between Book and Customer is **Many to Many** with quantity as

descriptive attribute.

Constraint: Primary key, price should be >0.

create table book

(bnonumber(4) primary key,

title varchar(20),

author varchar(20),

price number(5) constraint prchk check(price>0),

```
pu_yearnumber(4)

);


Insert into book values(101,'C programming', 'Yashwant Kantekar','524','1995');

Insert into book values(102,'Database','Korth','124','2013');

Insert into book values(103,'Finance Accounting','Sachin','322','2014');

Insert into book values(104,'OB','Alok','824','2015');

Insert into book values(105,'E Commerce','Khan','254','2012');


Create table cust

(cidnumber(4) primary key,

cnamevarchar(20),

addrvarchar(20)

);


Insert into cust values(201,'alok','pune');

Insert into cust values(202,'atual','pimpri');

Insert into cust values(203,'neetin','pune');

Insert into cust values(204,'seema','hadapsar');

Insert into cust values(205,'geeta','bhosari');
```

```
Create table bc1
(bnonumber(4) references book(bno),
cidnumber(4) references cust(cid),
quantity number(3)
);

Insert into bc1values(101,201,3);
Insert into bc1values(102,202,2);
Insert into bc1values(103,203,4);
Insert into bc1values(104,204,1);
Insert into bc1values(105,205,30);
```

Create a Database in 3NF & write queries for following.

- Display customer details from 'Pune'.

```
select * from cust
where addr ='pune';
```

- Display author wise details of book.

```
Select author,bno,title,price,pu_year
```

from book order by author;

- Display all customers who have purchased the books published in the year 2013.

```
Select cname,pu_year
From cust,book,bc1
wherecust.cid=bc1.cid and
book.bno=bc1.bno and
    pu_year='2013';
```

- Display customer name that has purchased more than 3 books.

```
Select distinct cname,title, quantity fromcust,book,bc1
Where cust.cid=bc1.cid and
book.bno=bc1.bno and
quantity>3;
```

- Display book names having price between 100 and 200 and published in the year 2013.

```
select title,price
frombook
where   price between 100 and 200 and
```

pu_year='2013';

Slip 6

Property (pno, desc, area, rate)

Owner (owner_name, addr, phno)

Relation between owner and Property is **One to Many**.

Constraint: Primary key, rate should be > 0

Create a Database in 3NF & write queries for following.

```
create table property1
  (pnonumber(4) primary key,
desvarchar(20),
areavarchar(20),
rate number(10) constraint rtchk1 check(rate>0)
);

insert into property1 values(1,'landline','chinchwad','10000');
insert into property1 values (2,'House','chinchwad','10450');
insert into property1 values (3,'landline','pune','10540');
insert into property1 values (4,'Agriculturalland','hdapsar','14500');
insert into property1 values (5,'House','pune','10626');
```

```
create table owner1
  (ownernamevarchar(20),
addressvarchar(20),
phnonumber(10),
pnonumber(4) REFERENCES property1(pno));

insert into owner1 values('patil','chinchwad',1234567,1);
insert into owner1 values('seema','chinchwad',45342333,2);
insert into owner1 values('seeta','pune',12334237,3);
insert into owner1 values('suhas','hadapsar',123213567,4);
insert into owner1 values('patil','pune',1234567,5);
insert into owner1 values('seeta','pune',12334237,2);
```

- Display area wise property details

```
Select distinct area,ownername,des,rate from  property, owner
Where property.pno=owner.pno order by area;
```

- Display property owned by 'Mr.Patil' having minimum rate.

```
Select distinct desfrom property1, owner1
Where property1.pno=owner1.pnoand
```

rate=(select min(rate) from property1, owner1

Where property1.pno=owner1.pno and ownername='patil');

- Display all properties with owner name that having highest rate of properties located in Chinchwad area.

Select distinct ownername,des from property1, owner1

Where property1.pno=owner1.pno and

rate=(select max(rate) from property1 where area='chinchwad');

- Display owner name having maximum no. of properties

Select distinct ownername, count(*) from property1, owner1

Where property1.pno=owner1.pno group by ownername

Having max(owner1.pno)= (select count (*) from property1);

Slip 7

Employee (emp_no, name, skill, payrate, workdate)

Position (posting_no, skill)

Relation between Employee and Position is **Many to Many** with day and shift as descriptive attribute.

Constraint: Primary key, payrate should be > 0

```
create table employee11
 (enonumber(4) primary key,
namevarchar(15),
skillvarchar(15),
payratenumber(10) constraint paychk check (payrate>0),
workdate date
);

Insert into employee11 values(1,'seema','manager',25423,'1 Mar 2010');
Insert into employee11 values(2,'sheeta','supervisor',56478,'2 Feb 2012');
Insert into employee11 values(3,'geeta','cleark',65423, '13 Dec 1999');
Insert into employee11 values(4,'suhas','chef',55423, '21 Nov 2013');
Insert into employee11 values(5,'sachin','waiter',27423, '11 Oct 2012');

create table position
(pnonumber(4) primary key,
```

```
skillvarchar(15)
);

Insert into position values(201,'manager');
Insert into positionvalues(202,'supervisor');
Insert into position values(203,'cleark');
Insert into positionvalues(204,'chef');
Insert into positionvalues(205,'waiter');


Create table ep
(enonumber(4) references employee11(eno),
pnonumber(4) references position(pno),
dayvarchar(10),
shiftvarchar(10)
);

Insert into ep values (1,201,'Monday','allocate');
Insert into ep values (2,202,'Tuesday','notallocat');
Insert into ep values (3,203,'Friday','allocate');
Insert into epvalues (4,204,'Monday','allocate');
```

Insert into ep values (5,205,'Friday','notallocat');

Create a Database in 3NF& write queries for following.

- Find the names and rate of pay all employees who allocated a duty.

```
Select distinct name,payrate from position, employee11,ep
Where employee11.eno=ep.eno and
position.pno=ep.pno and
shift='allocate';
```

- Give employee number who are working at posting_no. 201, but don't have the skills of waiter.

```
Select ep.eno from position, employee11,ep
Where employee11.eno=ep.eno and
position.pno=ep.pno and
ep.pno='201' and
position.skill<>'waiter';
```

- Display a list of names of employees who have skill of chef and who has assigned a duty.

```
Select name from position, employee11,ep
Where employee11.eno=ep.eno and
position.pno=ep.pno and
```

position.skill='chef' and

shift='allocate';

- Display emp_no and dates for all employees who are working on Tuesday and at least one other day.

Select ep.eno, workdate from position, employee11,ep

Where employee11.eno=ep.eno and

position.pno=ep.pno and

day in ('Tuesday','Friday');

- Display shiftwise employee details.

Select shift,ep.eno, name,position.skill,workdate from position, employee11,ep

Where employee11.eno=ep.eno and

position.pno=ep.pno order by shift;

Slip 8

Bill (billno, day, tableno, total)

Menu (dish_no, dish_desc, price)

Relation between Bill and Menu is **Many to Many** with quantity as descriptive attribute.

Constraint: Primary key, price should be > 0.

```
create table bill
(bnonumber(4) primary key,
day date,
tablenonumber(4),
total number(7)
);

Insert into bill values(301,'8 Jan 2013',1,1000);
Insert into bill values(302,'1 Dec 2013',2,1020);
Insert into bill values(303,'23 Feb 2014',3,3424);
Insert into bill values(304,'8 Jan 2013',2,2422);
Insert into bill values(305,'30 Nov 2013',6,1000);

create table menu
(dnonumber(4) primary key,
ddescvarchar(20),
price number(5) constraint pchk check(price>0)
);
```

```
Insert into menu values(211,'Pav Bhaji',600);

Insert into menu values(212,'Panner Mahkanwala',634);

Insert into menu values(213,'Roti',20);

Insert into menu values(214,'Kaju Masala',345);


create table bm1

(bnonumber(4) references bill(bno),

dnonumber(4) references menu(dno),

quantity number(10));


    Insert into bm1values(301,211,3);

    Insert into bm1values(302,212,2);

    Insert into bm1values(302,213,4);

    Insert into bm1values(303,214,1);

    Insert into bm1values(303,213,5);

    Insert into bm1values(304,212,1);

    Insert into bm1values(304,213,4);
```

Create a Database in 3NF & write queries for following.

- Display receipt which includes bill_no with Dish description, price, quantity and total amount of each menu.

Select bm1.bno,ddesc,price,total, quantity from bill,menu,bm1

Where bm1.bno=bill.bno and

bm1.dno=menu.dno order by ddesc;

- Find total amount collected by hotel on date 08/01/2013

Select sum (price) from bill,menu,bm1

Where bm1.bno=bill.bno and

bm1.dno=menu.dno and

day='8 Jan 2013';

- Count number of menus of billno 301.

Select count(bm1.dno) from bill,menu,bm1

Where bm1.bno=bill.bno and

bm1.dno=menu.dno and

bm1.bno='301';

- Display menu details having price between 100 and 500.

Select bm1.bno,ddesc,price,total from bill,menu,bm1

Where bm1.bno=bill.bno and

bm1.dno=menu.dno and

price between 100 and 500;

- Display total number of bills collected from each table on 01/12/2013.

Select count (bm1.bno), tableno from bill,menu,bm1

Where bm1.bno=bill.bno and

bm1.dno=menu.dno and

day='1 Dec 2013' group by tableno;

Slip 9

Musician (mno, mname, addr, phno)

Album (title, copy_right_date, format)

Relation between Musicians and Album is **One to Many**.

Constraint: Primary key.

```
create table musician
  (mnonumber(4) primary key,
namevarchar(20),
addrvarchar(20),
phnonumber(10)
);
```

```
insert into musician values (1,'A RRehman','Pune',123456);
insert into musician values (2,'Arijit Singh','Bombay',2323413);
insert into musician values (3,'Nihira Joshi','Nashik',344553);
insert into musician values (4,'Anu Malik','Pune',12324231);

create table album
  (titlevarchar(20),
codate date,
formatvarchar(20),
mnonumber(4) REFERENCES musician(mno));

        insert into album values('track1','2 March 2010','audio',1);
        insert into album values('track2','30 Nov 2012','video',2);
        insert into album values('track2','22 Jan 1999','video',3);
        insert into album values('track3','22 Jan 1999','audio',4);
        insert into album values('track5','4 Jun 1889','video',3);
```

Create a Database in 3NF & write queries for following.

- Display all albums composed by 'A R Rehman'.

Select title from musician, album

Where musician.mno=album.mno and

name='A RRehman';

- Display musician details who have composed Audio album.

Select distinct name,addr,phno,title,codate,format from musician, album

Where musician.mno=album.mno and

Format='audio';

- Find all musicians who have composed maximum albums.

select name, format from musician, album

Where musician.mno=album.mno and title=(select max(title)from album);

Select name,count(*) from musician,album

Where musician.mno=album.mno group by name

Having max(album.mno)= (select count (*) from album);

- Display musician wise album details.

Select name,title from musician, album

Where musician.mno=album.mno order by name;

Slip 10

Consider the following Entities and Relationships [30 Marks]

Sailor (sid, sname, age)

Boats (bid, bname, color)

Relation between Sailer and Boats is **Many to Many** with day as descriptive attribute.

Constraint: Primary key, age should be > 0.

Create table sailor

(sidnumber(4) primary key,

snamevarchar(10),

age number(5) constraint ack check(age>0)

);

Insert into sailor values(101,'ram',34);

Insert into sailor values(102,'alok',44);

Insert into sailor values(103,'surya',42);

Insert into sailor values(104,'priti',37);

Insert into sailor values(105,'atual',41);

```
Create table boats
(bid number(4) primary key,
bnamevarchar(10),
color varchar(10)
);
Insert into boats values(201,'Hindustan','blue');
Insert into boats values(202,'ABC','red');
Insert into boats values(203,'XYZ','black');
Insert into boats values(204,'Joshi','blue');
Insert into boats values(205,'PQR','green');

Create table sb
( sid number(4) references sailor(sid),
  bid number(4) references boats(bid),
  day varchar(10)
);
Insert into sb values(101,201,'Sunday');
Insert into sb values(102,202,'Friday');
Insert into sb values(103,203,'Tuesdy');
Insert into sb values(104,204,'Monday');
```

Insert into sb values(105,205,'Friday');

Insert into sb values(105,202,'Sunday');

Create a Database in 3NF & write queries for following.

- Display details of all boats sailed by sailor 'Ram'.

  Select sb.bid,bname,color from sb,boats,sailor

  Where sb.bid=boats.bid and

  sb.sid =sailor.sid and

  sname='ram';

- Display Sailor names working on blue boat.

Select sname from sb,boats,sailor

Where sb.bid=boats.bid and

sb.sid =sailor.sid and

color='blue';

- Count number of boats sailed by each sailor.

Select sname,count(*) from sb,boats,sailor

Where sb.bid=boats.bid and

sb.sid =sailor.sid group by sname;

- Find the name of sailor who sailed the boat on both Tuesday & Friday.

Select sname,day from sb,boats,sailor

Where sb.bid=boats.bid and

sb.sid =sailor.sid and

day in ('Tuesdy','Friday');

- Display details of the boats which is sailed maximum times on Sundays.

Select distinct bname, count(*) from sb,boats,sailor

Where sb.bid=boats.bid and

sb.sid =sailor.sidand day='Sunday' group by bname

having max(boats.bid)=(select count(*)from sp);

Slip 11:-

Supplier (sid, sname, addr)

Parts (pid, pname, pdesc)

Relation between Supplier and Parts is **Many to Many** with cost as descriptive attribute.

Constraint: Primary key, cost should be > 0.

create table supplier1

(snonumber(4) primary key,

```
snamevarchar(20),
addrvarchar(20)
);

insert into supplier1values(201,'Mr Pawar','Mumbai');
insert into supplier1values(202,'Mr Suhas','Pune');
insert into supplier1values(203,'Mr Sachin','Nashik');
insert into supplier1values(204,'Miss Sonali','Mumbai');
insert into supplier1values(205,'Miss Geeta','Pune');

create table parts1
(pnonumber(4) primary key,
pnamevarchar(20),
desvarchar(20)
);

insert into parts1 values(301,'wheel','rounded');
insert into parts1 values(302,'pencil','natraj');
insert into parts1 values(303,'tire','plated');
insert into parts1values(304,'break line','steel');
insert into parts1 values(305,'skrew','rounded');
```

```
create table sp1
(snonumber(4) references supplier1(sno),
pnonumber(4) references parts1(pno),
price number(5) constraints pck1 check(price>0)
);

insert into sp1values(201,301,500);
insert into sp1 values(202,302,40);
insert into sp1values(203,303,1200);
insert into sp1values(204,304,300);
insert into sp1values(205,305,20);
```

Create a Database in 3NF & write queries for following.

- Display Supplier details from 'Mumbai' city.

  ```
  Select * from supplier1
  Where addr='Mumbai';
  ```

- Update cost by 25 % for all parts supplied by supplier 'Mr. Pawar'.

  ```
  update sp1Set price=price+(0.5)
  ```

Where sp1.sno=(select sp1.sno from supplier1, sp1

Where supplier1.sno=sp1.sno and

sname='MrPawar');

- Display all parts supplied by each supplier.

Select pname,sname from supplier1,parts1,sp1

Where supplier1.sno=sp1.sno and

parts.pno=sp1.pno group by pname,sname;

- Display details of parts which are supplied at maximum price by each supplier.

Select distinct sname,max(price) from supplier1,parts1,sp1

Where supplier1.sno=sp1.sno group by sname;

- Display all suppliers who supply part 'wheel' and also display its cost.

Select sname,price from supplier1,parts1,sp1

Where supplier1.sno=sp1.sno and

parts1.pno=sp1.pno and

pname='wheel';

Slip 12:-

Medical_store (mno, mname, city, phno)

Drug (dno, dname, type, company, price)

Relation between Medical_store and Drug as **Many to Many** with quantity as descriptive attribute.

Constraint: Primary key, price should be > 0.

Create a Database in 3NF & write queries for following.

```
create table medst
(mnonumber(4) primary key,
mnamevarchar(20),
cityvarchar(20),
phnonumber(10)
);

Insert into medst values(101,'Sai medical','Pune',1234567);
Insert into medst values(201,'Priti medical','Pimpri',234567);
Insert into medst values(301,'Om medical','Hadpsar',3456789);
Insert into medst values(401,'Surbhi medical','Pimpri',4567890);
Insert into medst values(501,'Sai medical','Nashik',5678912);

create table drug
```

```
(dnonumber(4) primary key,
dnamevarchar(20),
typevarchar(20),
companyvarchar(10),
price number(5) constraint pcheck1 check(price>0)
);

Insert into drug values(211,'Omini','Acidity','SunPharma',40);
Insert into drug values(311,'Crocin','sardi','ABC',32);
Insert into drug values(411,'Crocin','Flue','XYZ',35);
Insert into drug values(511,'Calcium','Pain','ABC',120);
Insert into drug values(611,'Action 500','Sardi','SunPharma',40);

Create table md1
(mnonumber(4) references medst(mno),
dnonumber(4) references drug(dno),
quantity number(4)
);

Insert into md1values(101,211,1);
```

Insert into md1values(201,311,2);

Insert into md1values(301,411,1);

Insert into md1values(401,511,5);

Insert into md1values(501,611,3);

Insert into md1values(101,311,4);

Insert into md1 values(501,311,2);

- Update price of drug by 5 % of 'ABC' Company.

Update drug set price=price+(0.05)

Where drug.dno=(select md1.dno from medst,drug,md1

Where medst.mno=md1.mno and

drug.dno=md1.dno and company =’XYZ’);

- Display names of all medical store where ‘Crocin’ is available.

SelectDISTINCT company from medst,drug,md1

Where medst.mno=md1.mno and

drug.dno=md1.dno and

dname=’Crocin’;

- Count total number of drug of ‘SunPharma’ company in ‘Sai medical’ store.

Select count(md1.dno) from medst,drug,md1

Where medst.mno=md1.mno and

drug.dno=md1.dno and

mname=’Sai medical’ and

company =’SunPharma’;

- Delete all drugs supplied by ‘SunPharma‘

  Delete from(select * from drug inner join md1 on drug.dno=md1.dno andcompany =’SunPharma’);

  Delete from drug where company =’SunPharma’;

delete from drug

Where dno=(select md1.dno from medst,drug,md1

Where medst.mno=md1.mno and

drug.dno=md1.dno and

mname=’Om medical’ and company =’XYZ’);

- Display the details of medical store having maximum quantity of Crocin.

```
select mname, count(*) from medst,drug,md1
Where medst.mno=md1.mno and
drug.dno=md1.dno group by mname
having max(md1.mno)=(select count(*) from medst,drug
                     where dname='Crocin');
```

Slip 13:-

Account (ano, branchname, balance)

Customer (cust_no, cust_name, street, city)

Relation between Account and Customer is **Many to Many**.

Constraint: Primary key, balance should be > 500.

```
Create table account
(anonumber(4) primary key,
bnamevarchar(20),
balnumber(7) constraint bchk check(bal>0)
);
```

```
Insert into account values(101,'Chinchwad','1098453');

Insert into account values(102,'Pune','239843');

Insert into account values(103,'Pimpri','342333');

Insert into account values(104,'Chinchwad','1098453');

Insert into account values(105,'Mumbai','3458453');

Insert into account values(106,'Pune','34');


Create table cus

(cnonumber(4) primary key,

cnamevarchar(20),

streetvarchar(10),

cityvarchar(10)

);

Insert into cusvalues(201,'alok','XYZ road','Mumbai');

Insert into cusvalues(202,'ram','PQR road','Pune');

Insert into cusvalues(203,'geeta','Ring road','Pimpri');

Insert into cusvalues(204,'raju','NPM road','Mumbai');

Insert into cusvalues(205,'sham','ASD road','Chimchwad');


Create table ac

(accnonumber(4) references account(accno),
```

```
cnonumber(4) references cus(cno)
);
Insert into ac values(101,201);
Insert into ac values(104,204);
Insert into ac values(102,202);
Insert into ac values(103,203);
Insert into ac values(105,205);
Insert into ac values(101,202);
Insert into ac values(101,204);
Insert into ac values(106,204);
```

Create a Database in 3NF & write queries for following.

- Display customer details with balance between 100000 and 200000.

```
Select ac.cno,cname,street,city,bal from account,cus,ac
Where ac.cno=cus.cno and
account.ano=ac.ano and
bal between 1000000 and 2000000;
```

- Display customers having more than two accounts in Chinchwad branch.

Select cname,count(account.ano) from account,cus,ac

Where bname='Chinchwad' and

ac.cno=cus.cno and

account.ano=ac.ano group by cname,bname

having count(account.ano)>=2;

- Delete account whose balance is below the balance <500.

  Delete from

  (select * from account inner join ac on account.accno=ac.accno where balance<500);

- Select names of all Customers whose street name include the substring "road" and whose city is 'Mumbai'.

Select cname,street from cus

Where city='Mumbai' and street like '%road%';

Find number of depositor for each branch.

Select cname,count(*) from account,cus,ac

Where ac.cno=cus.cno and

account.ano=ac.ano group by cname;

Slip 14

Consider the following Entities and Relationships [30 Marks]

Branch (bname ,bcity ,assets)

Loan (loan_no, amount)

Relation between Branch and Loan is **One to Many**.

Constraint: Primary key, amount and assets should be > 0.

Branch (bname ,bcity ,assets)

Loan (loan_no, amount)

```
Create table branch
(bnonumber(4) primary key,
bnamevarchar(10),
bcityvarchar(10),
asetsnumber(10) constraints aschk check(asets>0)
);
Insert into branch values(101,'DYP','Mumbai',20000);
Insert into branch values(201,'BOI','Pune',23430);
Insert into branch values(301,'BOB','Nashik',34200);
Insert into branch values(401,'SBI','Mumbai',435210);
Insert into branch values(501,'BOM','Pune',453420);

Create table loan1
```

```
(lnonumber(4) primary key,
amount number(10) constraints achk check(amount>0),
bnonumber(4) references branch(bno)
);
Insert into loan1 values(301,2341,101);
Insert into loan1 values(302,3341,201);
Insert into loan1 values(303,2312,301);
Insert into loan1 values(304,2123,401);
Insert into loan1 values(305,900,501);
Insert into loan1 values(306,500,501);
```

Create a Database in 3NF & write queries for following.

- Display total loan amount given by DYP branch.

  ```
  Select amount,bname from branch, loan1
  Where branch.bno=loan1.bno and
  bname='DYP';
  ```

- Find total number of loans given by each branch.

```
Select bname,count(*) from branch, loan1
Where branch.bno=loan1.bno group by bname;
```

- Find the name of branch that have maximum assets located in Mumbai.

Select bname,bcity from branch, loan1

Where branch.bno=loan1.bno and

asets=(select max(asets) from branch where

bcity='Mumbai');

- Display loan details in descending order of their amount.

select loan1.bno,bname,bcity,asets,amount from branch,loan1

where branch.bno=loan1.bno order by amountdesc;

- Display all branches located in Mumbai, Pune and Nasik.

select bname,bcity from branch,loan1

where branch.bno=loan1.bno and

bcity in ('Mumbai','Pune','Nashik');

Slip15

Consider the following Entities and

Relationships

Employee (eno, ename, deptname, salary)

Project (pno, name, budget)

Relation between Employee and Project is Many to Many.

Constraint: Primary key, salary should be > 0.

Solution:-

Create a Database in 3NF & write queries for following.

+List the name of employee and department having salary >

50000.

SQL>select ename ,dname from Employee where salary >50000;

+List names of all employees who works with ‘Ramesh’ on same

project.

SQL> select ename ,pname from Employee,project ,ep where

employee.eno=ep.eno and project.pno=ep.pno and

ename=’Ramesh’;;

-Find the names of employees who are working on project having

budget greater than 30000.

SQL>select ename ,pname,budget from Employee,project ,ep

where employee.eno=ep.eno and project.pno=ep.pno and

budget>300000;

List name of department that have at least two projects under

them.

SQL>select dname,pname from Employee,project ,ep where

employee.eno=ep.eno and project.pno=ep.pno group by

dname,pname having count(project.pname)>2;

-Updatebudget of a project done by employees of Computer

Department by 15%.

SQL>

Slip16

Consider the following Entities and

Relationships

Branch (bno, bname, bcity, assets)

Account (acc_no ,balance)

Relation between Branch and Account is One

to Many.

Constraint: Primary key, balance and assets should be > 0.

Solution:-

Create a Database in 3NF & write queries for following.

***Find the maximum account balance of each branch.**

SQL>Select max(balance),bname from branches,accounts where branches.bno=accounts.bno group by bname;

***Find branches where average account balance is more than 30000.**

SQL>Select avg(balance) from branches,accounts where branches.bno=accounts.bno and balance >30000;

***Find names of all branches that have assets value greater than that of each branch in ‘pune’.**

SQL>Select max(assets),bname from branches,accounts where branches.bno=accounts.bno and bcity=’pune’ group by bname;

***Decrease 3% balance on account whose balance is greater than 100000.**

SQL> Update accounts set balance=balance-balance*0.03 where balance >3000;

***Display details of branchwhose city**

starts from ‘A’.

SQL>select * from branches where bcity like

Slip17

Consider the following Entities and

Relationships

Donor (donor_no, donor_name, city)

Blood_Donation(bid,blood_group,quantity,date_of_collection)

Relation between Donor and Blood_Donation is One to Many.

Constraint: Primary key, blood_group should not be null.

Solution:-

Create a Database in 3NF & write queries for following.

***Display total blood quantity collected on 25th December 2013.**

SQL>Select sum(quantity)from Blood_Donation where date_of_collection='25-12-2013';

***Display total blood donated by each donor.**

SQL>select donor_name,sum(quantity) from Blood_Donation,Donor where Blood_Donation.donor_no=Donor.donor_no group by donor_name;

***Display Donor details having blood group 'A+tve'.**

SQL>Select donor_name,city from Donor, Blood_Donation where Donor .donor_no= Blood_Donation .donor _no and blood_group= 'A+VE';

***Display the donor who has donated blood more than two times.**

SQL>Select donor_name from Donor, Blood_Donation where Donor.donor_no=Blood_Donation.donor_no and

Blood_Donation. donor_no>=2;

SQL>Select donor_name from Donor, Blood_Donation where

Donor.donor_no=Blood_Donation.donor_no and

Blood_Donation. donor_no>2;

***Displaythe donor information with blood group whose**

city name contains “sh” in it.

SQL>Select donor_name,city,blood_group from

Donor,Blood_Donation where

Donor.donor_no=Blood_Donation.donor_no and city like

"%Msh%';

Slip18

Consider the following Entities and

Relationships

Bus (bus_no, capacity, depot_no)

Route (rout_no, source, destination, no_of_stations)

Relation between Bus and Route is Many to One.

Constraint: Primary key.

Solution:-

Create a Database in 3NF & write queries for following.

-Find out the route details on which buses whose capacity is 20

runs.

SQL>Select * from Bus,Route where Bus. route_no=Route.

rout_no and capacity =20;

-Display number of stations from 'Chinchwad' to ‘ Katraj’.

SQL> Select no_of_stations from Route where source='

Chinchwad ' and destination=' Katraj ';

Display the route on which more than 3 buses runs.

SQL> Select Route.rout_no,source, destination from Bus,Route

where Route.rout_no=Bus.rout_no group by Route.rout_no,source,

destination having count (Bus.rout_no)>3;

-Display number of buses of route ‘Swargate’ to ‘Hadapsar’.

SQL>select count(bus_no)from Bus,Route where

Route.rout_no=Bus.rout_no and

source=' Swargate ' and destination=' Hadapsar ';

-Findthe bust having maximum capacity from ‘Nigadi’ to

‘Kothrud'..

SQL>Select bus_no,capacity from Bus,Route where

Route.rout_no=Bus.rout_no

and source='Nigadi’ and destination='Kothrud' group by

bus_no,capacity

having capacity=(select max(capacity)) from Bus,Route;

Slip19

Consider the following Entities and

Relationships

Person (driver_id, driver_name, address)

Car (license_no, model, year)

Relation between Person and Car is Many to Many with date and

time as

descriptive attribute.Constraint: Primary key.

Solution:-

Create a Database in 3NF & write queries for following.

+Display details of all persons who are driving ‘Alto’ car.

SQL>Select dname from person,car ,pc where person.did=pc.did

and car.lno=pc.lno and model='alto’;

Count the number of cars driven by each driver.

SQL>Select count(model),dname from person,car ,pc where

person.did=pe.did and car.lno=pc.lno group by dname;

Display car details manufactured before year 2000.

SQL> Select * from car where year<2000;

«In which day ‘Mr. Ram’ drives maximum number of cars.

SQL>select count(car.model),dates,dname from person,car ,pc

where person.did=pc.did and car.lno=pc.lno and dname='ram’

group by dates,dname;

Display total number of persons who drives car in each year.

SQL>Select count(person.did),year,dname from person,car,pc

where person.did=pc.did and car.lno=pc.lno group by year,dname;

Slip20

Consider the following Entities and

Relationships

Person (pno, person_name, birthdate, income)

Area (area_name, area_type)

Relation between Person and area is Many to One.

Constraint: Primary key, income should be > 0, area_type should be

rural or urban.

Solution:-

Create a Database in 3NF & write queries for following.

-Display persons having income less than 1 lakhs in PCMC Area.

SQL> select pname from persons,area where persons.ano=

area.ano and aname='pcmc’

and income <100000;

-Display population of each area.

SQL>select population,aname from persons, area where persons.

ano= area. ano group by aname,population;

Display persons details from ‘Urban’ area.

SQL> select pname from persons,area where person.s ano= area.

ano and atype='urban’;

Display the details of area having population greater than that of

in Pune.

SQL>select max(population),aname from area where

aname='pune’ group by aname;

‘Display details of person from each area having minimum

income.

SQL> select pname,birthdate,income from persons where

income=(select min(income) from persons);