Savitribai Phule Pune University

F. Y. B. B. A. (C. A.)

Semester I (CBCS 2019 Pattern)

Name: PATIL LALIT DEVIDAS

College Name: SINHGAD COLLEGE OF ARTS & COMMERCE NARHE PUNE

Roll No.: 106 **Division: B** **Seat No:**

Academic Year: 2022-23

CERTIFICATE

This is to certify that

Mr. PATIL LALIT DEVIDAS

Seat Number_____of F.Y.B.B.A. (C.A) Sem-I has Successfully Completed Laboratory course (Database Management System, C Language) in the year_____

He/she has scored mark out of 10 (For Lab Book).

Subject Teacher **H.O.D./Coordinator**

Internal Examiner **External Examiner**

Practical Slip:-2

Q1.(i)) Consider the following and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints

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

2 (cno varchar(10) primary key,

3 cname varchar(20),

4 addr varchar(20));

Table created.

SQL> desc client;

Name	Null?	Type
CNO	NOT NULL	VARCHAR2(10)
CNAME		VARCHAR2(20)
ADDR		VARCHAR2(20)

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

1 row created.

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

1 row created.

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

1 row created.

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

1 row created.

SQL> select * from client;

```
CNO       CNAME              ADDR
---------- -------------------- --------------------
CN001     Aashish            Pune
CN002     Patil              Mumbai
CN003     Mr.Roy             Pimpri
CN004     Mahesh             Mumbai
```

SQL> create table Sales_order

```
  2  (ordNo int primary key,
  3 ordDate varchar(23) not null,
  4  cno varchar(10) references client on delete cascade);
```

Table created.

SQL> desc Sales_order;

```
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ORDNO                                     NOT NULL NUMBER(38)
 ORDDATE                                   NOT NULL VARCHAR2(23)
 CNO                                                VARCHAR2(10)
```

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

1 row created.

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

1 row created.

SQL> insert into sales_order values(3,'09/03/2009','CN004');

1 row created.

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

1 row created.

select * from Sales_order;

ORDNO	ORDDATE	CNO
1	23/06/2015	CN001
2	09/03/2019	CN002
3	09/03/2009	CN004
4	09/08/2019	CN002

Consider the above table 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;

Name	Null?	Type
ORDNO	NOT NULL	NUMBER(38)
ORDDATE	NOT NULL	VARCHAR2(23)
CNO		VARCHAR2(10)
AMOUNT		NUMBER(38)

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

SQL> delete from client

2 where cname like'A%';

1 row deleted.

SQL> select * from client;

CNO	CNAME	ADDR
CN002	Patil	Mumbai
CN003	Mr.Roy	Pimpri
CN004	Mahesh	Mumbai

Q1.(ii) Consider the above tables and execute the following queries.

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;

ORDNO	ORDDATE	CNO	AMOUNT
2	09/03/2019	CN002	
3	09/03/2009	CN004	

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.

Practical Slip:-4

Q2.(i) Consider the following and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

Patient(Pcode,Name,Addr,Disease)

Bed(Bed_No,Room_No,loc)

Relatiionship:- 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
  2  (Pcode int primary key,
  3  name varchar(20)not null,
  4  addr varchar(20),
  5  disease varchar(10));
```

Table created.

```
SQL> desc Patient;
 Name                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 PCODE                     NOT NULL NUMBER(38)
 NAME                      NOT NULL VARCHAR2(20)
 ADDR                               VARCHAR2(20)
 DISEASE                            VARCHAR2(10)
```

```
SQL> insert into Patient values(11,'Lalit','Pimple gurav','listeria');
  1 row created.
SQL> insert into Patient values(12,'Makesh','Pune','norovirus');
  1 row created.
SQL> insert into Patient values(13,'Mr.Roy','Mumbai','cholera');
  1 row created.
SQL> insert into Patient values(14,'Rohan','Pimple gurav','dengue');
  1 row created.
```

SQL> insert into Patient values(15,'Sneha','Nashik','listeria');

1 row created.

SQL> insert into Patient values(16,'Ramesh','Gujrat','dengue');

1 row created.

SQL> select * from patient;

PCODE	NAME	ADDR	DISEASE
11	Lalit	pimple gurav	listeria
12	Makesh	pune	norovirus
13	Mr.Roy	Mumbai	cholera
14	Rohan	pimple gurav	dengue
15	sneha	nashik	listeria
16	Ramesh	Gujrat	dengue

6 rows selected.

SQL> create table Bed

2 (bno int primary key,rno int not null,loc varchar(10)not null,

3 pcode int references patient on delete cascade);

Table created

SQL> desc bed;

Name	Null?	Type
BNO	NOT NULL	NUMBER(38)
RNO	NOT NULL	NUMBER(38)
LOC	NOT NULL	VARCHAR2(10)
PCODE		NUMBER(38)

SQL> insert into bed values(1,105,'mumbai',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;

BNO	RNO	LOC	PCODE
1	105	mumbai	11
2	102	2nd floor	12
3	103	4th floor	13
4	104	1st floor	11
5	105	3rd floor	14
6	106	2nd floor	15

6 rows selected.

Consider the above table 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';

PCODE	NAME	ADDR	DISEASE
11	Lalit	pimple gurav	listeria
14	Rohan	pimple gurav	dengue

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;

PCODE	NAME	ADDR	DISEASE	BNO	RNO	LOC	PCODE
11	Lalit	pimple gurav	listeria	1	105	mumbai	11

Q2.(ii) Consider the above table and execute the following quries:

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;

COUNT(PATIENT.PCODE)

1

2

1

1

1

2. **Display the name of patient who are admitted in room no 101.**

SQL> select name from patient,bed

2 where patient.pcode=bed.pcode

3 and rno=102;

NAME

Makesh

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;
DISEASE
----------
Listeria
```

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';

RNO       BNO
---------- --------
 103         3
```

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

```
SQL> select * from patient,bed
  2  where patient.pcode=bed.pcode
  3  and loc='2nd floor'and rno=102;
PCODE NAME              ADDR              DISEASE         BNO       RNO    LOC         PCODE
---------- -------------------- -------------------- ----------          ----------   ---------- ----------     ----------
12       Makesh             pune              norovirus         2            102    2nd floor     12
```

Practical Slip:-6

Q3.(i) . Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

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

2 (pno int primary key,

3 pname varchar(20),

4 sdate date,

5 budget int,

6 status varchar(20) check(status in('c','i','p')));

Table created.

SQL> desc project;

Name	Null?	Type
PNO	NOT NULL	NUMBER(38)
PNAME		VARCHAR2(20)
SDATE		DATE
BUDGET		NUMBER(38)
STATUS		VARCHAR2(20)

SQL> insert into project values(1,'Poster Making','09/Mar/2019',23000,'c');

1 row created.

SQL> insert into project values(2,'Match of Health','01/Apr/2018',200000,'i');

1 row created.

SQL> insert into project values(3,'Moving Bird','23/Mar/2027',6000000,'p');

1 row created.

SQL> insert into project values(4,'The Blue Bird','12/Feb/2020',5000000,'c');

1 row created.

SQL> insert into project values(5,'Bharatmala','31/July/2015',6923240000000,'p');

1 row created.

SQL> select * from project;

PNO	PNAME	SDATE	BUDGET	STATUS
1	Poster Making	09-MAR-19	23000	c
2	Match of Health	01-APR-18	200000	i
3	Moving Bird	23-MAR-27	6000000	p
4	The Blue Bird	12-FEB-20	5000000	c
5	Bharatmala	31-JUL-15	6.9232E+12	p

SQL> create table department

2 (dno int primary key,

3 dname varchar(20),

4 hod varchar(20),

5 loc varchar(20),

6 pno int references project on delete cascade);

Table created.

SQL> desc department;

Name	Null?	Type
DNO	NOT NULL	NUMBER(38)
DNAME		VARCHAR2(20)
HOD		VARCHAR2(20)
LOC		VARCHAR2(20)
PNO		NUMBER(38)

SQL> insert into department values(101,'Computer','Desai','Pune',1);

1 row created.

SQL> insert into department values(102,'Commerce','Sarita','Pune',2);

1 row created.

SQL> insert into department values(103,'Computer','Mahesh','Pune',3);

1 row created.

SQL> insert into department values(104,'Engineering','Hitesh','Pune',4);

1 row created.

SQL> select * from department;

DNO	DNAME	HOD	LOC	PNO
101	Computer	Desai	Pune	1
102	Commerce	Sarita	Pune	2
103	Computer	Mahesh	Pune	3
104	Engineering	Hitesh	Pune	4

Consider the above tables and execute the following queries:

1. **Drop loc column from department table.**

SQL> alter table department

2 drop column loc;

Table altered.

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';

PNO	PNAME	SDATE	BUDGET	STATUS
3	Moving Bird	23-MAR-27	6000000	p

Q3.(ii) Consider the above tables and execute them following queries:

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

SQL> select pname,dname,hod from department,project

2 where department.pno=project.pno

3 and project.status='c';

```
PNAME              DNAME              HOD
-------------------- -------------------- --------------------
Poster Making       Computer           Desai
The Blue Bird       Engineering        Hitesh
```

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;

```
SUM(BUDGET)   DNAME
--------------------  ----------
    200000          Commerce
   5000000          Engineering
   6023000          Computer
```

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;

```
PNAME       STATUS       COUNT(DEPARTMENT.DNO)
-------------------- -------------------- --------------------
Match of Health     i                          1
```

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

SQL> select pname from department,project

2 where department.pno=project.pno

3 and hod='desai';

no rows selected.

5. Display department wise HOD.

SQL> select dname,hod from department,project

2 where department.pno=project.pno

3 order by dname;

DNAME	HOD
Commerce	Sarita
Computer	Mahesh
Computer	Desai
Engineering	Hitesh

Practical Slip:-8

Q4.(i) Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

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
  2  (bno int primary key,
  3  title varchar(10),
  4  author varchar(20),
  5  price int check(price>0),
  6  yp number);
```

Table created.

```
SQL> desc book;
 Name                          Null?          Type
 ------------------------------- ---------------- ------------------
 BNO                           NOT NULL NUMBER(38)
 TITLE                                  VARCHAR2(10)
 AUTHOR                                 VARCHAR2(20)
 PRICE                                  NUMBER(38)
 YP                                     NUMBER
```

```
SQL> insert into book values(101,'Dreams','Mr.Mahesh',150,2017);
    1 row created.
SQL> insert into book values(102,'Life','Mr.Lalit Patil',100,2019);
     1 row created.
SQL> insert into book values(103,'IT story','Mr.Gadhave',190,2011);
      1 row created.
SQL> insert into book values(104,'POM','Dr.Mangesh',200,2001);
```

1 row created.

SQL> insert into book values(105,'Statistics','Dr. P.G. Dixit',250,2017);

1 row created.

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

1 row created.

SQL> select * from book;

BNO	TITLE	AUTHOR	PRICE	YP
101	Dreams	Mr.Mahesh	150	2017
102	Life	Mr.Lalit Patil	100	2019
103	IT story	Mr.Gadhave	190	2011
104	POM	Dr.Mangesh	200	2001
105	Statistics	Dr. P.G. Dixit	250	2017
106	Joker	Mr. Talore	230	2011

6 rows selected

SQL> create table customer

2 (cid int primary key,

3 cname varchar(20),

4 addr varchar(20),

5 bno int references book);

Table created.

SQL> desc customer;

Name	Null?	Type
CID	NOT NULL	NUMBER(38)
CNAME		VARCHAR2(20)
ADDR		VARCHAR2(20)
BNO		NUMBER(38)

SQL> insert into customer values(1,'Ashok','Pune',101);

1 row created.

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

1 row created.

SQL> insert into customer values(3,'Rahul','Pimpri',103);

1 row created.

SQL> insert into customer values(4,'Lalit','Mumbai',104);

1 row created.

SQL> insert into customer values(5,'vikas','Pune',105);

1 row created.

SQL> insert into customer values(6,'Ganesh','Nashik',106);

1 row created.

SQL> select * from customer;

CID	CNAME	ADDR	BNO
1	Ashok	Pune	101
2	Mahesh	Mumbai	102
3	Rahul	Pimpri	103
4	Lalit	Mumbai	104
5	Vikas	Pune	105
6	Ganesh	Nashik	106

6 rows selected.

SQL> create table cust_book(bcid int primary key,

2 bno int references book,

3 cid int references customer);

Table created.

SQL> desc cust_book;

Name	Null?	Type
BCID	NOT NULL	NUMBER(38)

BNO	NUMBER(38)
CID	NUMBER(38)

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

1 row created.

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

1 row created.

SQL> insert into cust_book values(13,103,3);

1 row created.

SQL> insert into cust_book values(14,104,4);

1 row created.

SQL> insert into cust_book values(15,105,5);

1 row created.

SQL> insert into cust_book values(16,106,6);

1 row created.

SQL> select * from cust_book;

BCID	BNO	CID
11	101	1
12	102	2
13	103	3
14	104	4
15	105	5
16	106	6

6 rows selected.

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';

TITLE

IT story

2. Add column EMailId into customer table.

SQL> alter table customer

2 add Emailid varchar(20);

Table altered.

SQL> desc customer;

Name	Null?	Type
CID	NOT NULL	NUMBER(38)
CNAME		VARCHAR2(20)
ADDR		VARCHAR2(20)
BNO		NUMBER(38)
EMAILID		VARCHAR2(20)

Q4.(ii) Consider the above tables and execute the following queries:

1. Display customer details from 'Mumbai'.

SQL> select * from customer

2 where addr='Mumbai';

CID	CNAME	ADDR	BNO	EMAILID
2	Mahesh	Mumbai	102	
4	Lalit	Mumbai	104	

2. Display author wise details of book.

SQL> select author,title from book

2 order by author;

AUTHOR	TITLE
Dr. P.G. Dixit	Statistics
Dr.Mangesh	POM

Mr. Talore Joker
Mr.Gadhave IT story
Mr.Lalit Patil Life
Mr.Mahesh Dreams

6 rows selected.

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

SQL> select count(book.bno),cname from customer,book,cust_book

2 where customer.cid=cust_book.cid

3 and book.bno=cust_book.bno and book.bno>3

4 group by cname;

COUNT(BOOK.BNO) CNAME
--- ---
1 Lalit
1 Mahesh
1 Rahul
1 Ashok
1 Ganesh
1 vikas

6 rows selected.

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

SQL> select book.title from book,customer,cust_book

2 where customer.cid=cust_book.cid

3 and book.bno=cust_book.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
  2  set title='DBMS'
  3  where author='Mr.Talore';

1 row updated.

SQL> select * from book;

     BNO  TITLE      AUTHOR          PRICE      YP
----------  ---------- --------------------  ----------  ----------
     101  Dreams     Mr. Mahesh          150       2017
     102  Life       Mr. Lalit Patil     100       2019
     103  IT story   Mr. Gadhave         190       2011
     104  POM        Dr. Mangesh         200       2001
     105  Statistics Dr. P. G. Dixit     250       2017
     106  DBMS       Mr. Talore          230       2011

6 rows selected.
```

Practical Slip:-10

Q5.(i) Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

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.

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

Table created.

```
SQL> desc employee;
 Name                              Null?            Type
 ----------------------------------- ---------------- ------------------------
 ENO                               NOT NULL         NUMBER(38)
 NAME                                               VARCHAR2(20)
 SKILL                             NOT NULL         VARCHAR2(20)
 PAYRATE                                            NUMBER(38)
```

```
SQL> insert into employee values(1,'Mahesh','Manager',23000);
  1 row created.
SQL> insert into employee values(2,'Meghna','Waiter',23000);
  1 row created.
SQL> insert into employee values(3,'Sarita','Ceo',23000);
  1 row created.
SQL> insert into employee values(4,'Ram','Chef',23000);
  1 row created.
```

SQL> select * from employee;

ENO	NAME	SKILL	PAYRATE
1	Mahesh	Manager	23000
2	Meghna	Waiter	23000
3	Sarita	Ceo	23000
4	Ram	Chef	23000

SQL> create table position

2 (pno int primary key,

3 skill varchar(20),

4 eno int references employee);

Table created.

SQL> desc position;

Name	Null?	Type
PNO	NOT NULL	NUMBER(38)
SKILL		VARCHAR2(20)
ENO		NUMBER(38)

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;

PNO	SKILL	ENO
201	mg	1
203	ceo	2
202	wt	3
205	wdf	4
204	whd	2

SQL> create table Emp_p

2 (epno int primary key,

3 eno int references employee,

4 pno int references position);

Table created.

SQL> desc emp_p;

Name	Null?	Type
EPNO	NOT NULL	NUMBER(38)
ENO		NUMBER(38)
PNO		NUMBER(38)

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

1 row created.

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

1 row created.

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

1 row created.

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

1 row created.

SQL> insert into emp_p values(15,1,204); 1 row created.

SQL> select * from emp_p;

EPNO	ENO	PNO
11	1	201
12	2	202
13	2	203
14	3	202
15	1	204

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;

NAME	SKILL
Mahesh	Manager
Meghna	Waiter
Ram	Chef
Sarita	Ceo

2. **Update the posting of employee to 220 whose skill is "Manager".**

SQL> update position set pno=220

2 where skill='mg';

SQL> select * from position;

PNO	SKILL	ENO
220	mg	1
203	ceo	2
202	wt	3
205	wdf	4
204	whd	2

Q5(ii). Consider the above tables and execute the following queries:

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

SQL> select name,payrate from employee;

```
NAME                 PAYRATE
-------------------- ----------
Mahesh                 23000
Meghna                 23000
Sarita                 23000
Ram                    23000
```

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,emp_p

```
  2  where employee.eno=emp_p.eno and position.pno=emp_p.pno 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.**

SQL> select name from employee,position,emp_p

```
  2  where employee.eno=emp_p.eno and position.pno=emp_p.pno and employee.skill='chef';

no rows selected
```

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

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

```
  2  where employee.eno=emp_p.eno and position.pno=emp_p.pno group by employee.skill,name;

NAME                 SKILL
-------------------- --------------------
Mahesh               Manager
Meghna               Waiter
Sarita               Ceo
```

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

SQL> update employee set payrate=20000

2 where skill='waiter';

1 row updated.

SQ L> select * from employee;

ENO	NAME	SKILL	PAYRATE
1	Mahesh	Manager	23000
2	Meghna	Waiter	20000
3	Sarita	Ceo	23000
4	Ram	Chef	23000

Practical Slip:-12

Q6(i). Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

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.

```
SQL> create table movie
  2  (m_name varchar(20) primary key,
  3  r_year number(4) check(r_year>2000),
  4  budget number(12));
Table created.
SQL> desc movie;
 Name                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 M_NAME                    NOT NULL VARCHAR2(20)
 R_YEAR                             NUMBER(4)
 BUDGET                             NUMBER(12)

SQL> insert into movie values('ramsetu',2022,30000000);
    1 row created.
SQL> insert into movie values('rockery',2022,2500000000);
   1 row created.
SQL> insert into movie values('dangal',2020,1500000000);
    1 row created.
```

SQL> insert into movie values('joker',2019,2500000000);

1 row created.

SQL> insert into movie values('titanic',2015,2000000000);

1 row created.

SQL> select * from movie;

```
M_NAME          R_YEAR     BUDGET
-------------------- ----------     ----------
ramsetu             2022      30000000
rockery             2022      2500000000
dangal              2020      1500000000
joker               2019      2500000000
titanic             2015      2000000000
```

SQL> create table actor

```
  2  (a_name varchar(20) primary key,
  3  role varchar(20),
  4  charges number(10),
  5  a_address varchar(20),
  6  m_name varchar(25),
  7  foreign key(m_name) references movie);
```

Table created.

SQL> desc actor;

```
 Name                          Null?    Type
 ------------------------------- -------------- ----------------------------
 A_NAME                        NOT NULL VARCHAR2(20)
 ROLE                                   VARCHAR2(20)
 CHARGES                                NUMBER(10)
 A_ADDRESS                               VARCHAR2(20)
 M_NAME                                 VARCHAR2(25)
```

SQL> insert into actor2 values('amir','lead',25000000,'mumbai',' dangal');

1 row created.

SQL> insert into actor2 values('akshay','lead',150000000,'mumbai','ramsetu');

1 row created.

SQL> insert into actor2 values('leonardo','lead',100000000,'UK','titanic');

1 row created.

SQL> insert into actor2 values('robert de niro','villan',18000000,'new york','joker');

1 row created.

SQL> insert into actor2 values('joaquin','lead',30000000,'new york','joker');

1 row created.

SQL> select * from actor;

A_NAME	ROLE	CHARGES	A_ADDRESS	M_NAME
amir	lead	25000000	mumbai	dangal
akshay	lead	150000000	mumbai	ramsetu
leonardo	lead	100000000	UK	titanic
robert de niro	villan	18000000	new york	joker
joaquin	lead	30000000	new york	joker

SQL> create table producer

2 (p_id varchar(20) primary key,

3 name varchar(20),

4 p_address varchar(20),

5 m_name varchar(25),

6 foreign key(m_name) references movie);

Table created.

SQL> desc producer;

Name	Null?	Type
P_ID	NOT NULL	VARCHAR2(20)
NAME		VARCHAR2(20)
P_ADDRESS		VARCHAR2(20)
M_NAME		VARCHAR2(25)

SQL> insert into producer values('P001','rohit','hydrabad','dangal');

1 row created.

SQL> insert into producer values('P002','akash','chennai','ramsetu');

1 row created.

SQL> insert into producer values('P003','stanley','UK','joker');

1 row created.

SQL> insert into producer values('P004','nayro','new york','titanic');

1 row created.

SQL> insert into producer values('P005','advani','delhi','dangal');

1 row created.

SQL> select * from producer;

P_ID	NAME	P_ADDRESS	M_NAME
P001	rohit	hydrabad	dangal
P002	akash	chennai	ramsetu
P003	stanley	UK	joker
P004	nayro	new york	titanic
P005	advani	delhi	dangal

Consider the above tables and execute the following queries:

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

SQL> select m_name from movie where movie.budget in (select max(movie.budget) from movie);

M_NAME

---

rockery

joker

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

SQL> select p_id,name,p_address from producer,movie

2 producer.m_name=movie.m_name

3 group by name,r_year

4 having count(m_name>1);

P_ID	NAME	P_ADDRESS
P001	rohit	hydrabad
P005	advani	delhi

Q6(ii). Consider the above tables and execute the following queries:

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

SQL> select m_name from(select m_name,row_number() over (order by budget desc) serial from movie) where serial=2;

M_NAME

Joker

2. **List the names of actors who have acted in the maximum number of movies.**

SQL> select a_name from(select a_name, count(a_name)as val from actor2 group by a_name)

2 where val in (select max(val) from

3 (select a_name, count(a_name) as val from actor2 group by a_name));

A_NAME

Joaquin

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

SQL> select m_name from producer

2 group by m_name

3 having count(*)>1;

M_NAME

Dangal

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

SQL> select distinct(a_name) from actor

2 where charges in (select max(charges) from actor);

A_NAME

Akshay

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

SQL> select a_name from actor2

2 where m_name in (select m_name from actor2 where a_name='akshay') and a_name!='akshay';

no rows selected

Practical Slip:-14

Q7(i). Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

Student (stud_reg_no, stud_name, class)

Competition (cno, cname, ctype)

Relation between Student and Competition is Many to Many with rank and year as descriptive attribute.

Constraint: Primary key, class must be("FY,SY,TY").

```
SQL>  create table competition
  2  (cno varchar(20) primary key,
  3  cname varchar(30),
  4  ctype varchar(20));
```

Table created.

```
SQL> desc competition;
 Name                      Null?    Type
 ----------------------------------- -------- ----------------------------
 CNO                       NOT NULL VARCHAR2(20)
 CNAME                              VARCHAR2(30)
 CTYPE                              VARCHAR2(20)
SQL> insert into competition values('C101','mehandi','monopoly');
   1 row created.
SQL> insert into competition values('C102','E-Rangoli','perfect');
  1 row created.
SQL> insert into competition values('C103','programming','oligopoly');
  1 row created.
SQL> insert into competition values('C104','kabaddi','monopoly');
  1 row created.
```

SQL> select * from competition;

CNO	CNAME	CTYPE
C101	mehandi	monopoly
C102	E-Rangoli	perfect
C103	programming	oligopoly
C104	kabaddi	monopoly

SQL> create table student

2 (Stud_reg_no varchar(10) primary key,

3 stud_name varchar(20),

4 class varchar(5) check (class in('FY','ST','TY')),

5 rank varchar(10),

6 year number(4),

7 cno varchar(20),

8 foreign key(cno) references competition);

Table created.

SQL> desc student;

Name	Null?	Type
STUD_REG_NO	NOT NULL	VARCHAR2(10)
STUD_NAME		VARCHAR2(20)
CLASS		VARCHAR2(5)
RANK		VARCHAR2(10)
YEAR		NUMBER(4)
CNO		VARCHAR2(20)

SQL> insert into student values('S01','Meghna','FY','1st',2014,'C101');

1 row created.

SQL> insert into student values('S02','Mahesh','FY','1st',2020,'C102');

1 row created.

SQL> insert into student values('S03','Mahavir','TY','3rd',2015,'C103');

1 row created.

SQL> insert into student values('S04','Mahi','TY','1st',2021,'C104');

1 row created.

SQL> select * from student;

STUD_REG_N	STUD_NAME	CLASS	RANK	YEAR	CNO
S01	Meghna	FY	1st	2014	C101
S02	Mahesh	FY	1st	2020	C102
S03	Mahavir	TY	3rd	2015	C103
S04	Mahi	TY	1st	2021	C104

Consider the above tables and execute the following queries:

1. **Count total no students class wise**.

SQL> select class,count(*)as total_student from student group by class;

CLASS	TOTAL_STUDENT
TY	2
FY	2

2. **Delete the details of student who has participated in "Mehandi" competation.**

SQL> delete from student09 where cno=(select cno from competition where cname='mehandi');

1 row deleted.

SQL> select * from student09;

STUD_REG_N	STUD_NAME	CLASS	RANK	YEAR	CNO
S02	Mahesh	FY	1st	2020	C102

S03 Mahavir TY 3rd 2015 C103

S04 Mahi TY 1st 2021 C104

Q7(ii). Consider the above tables and execute the following queries:

1. **Display students from class 'FY' and participated in 'E-Rangoli ' Competition.**

SQL> select stud_name from student

2 where class='FY' and cno=(select cno from competition where cname='E-Rangoli');

STUD_NAME

Mahesh

2. **Find the number of student for programming competition**

SQL> select count(stud_name) from student

2 where cno=(select cno from competition where cname='programming');

COUNT(STUD_NAME)

1

3. **Display the names of first three winners of each competition.**

SQL> select stud_name,com.cname,rank from student join competition com on student.cno=com.cno order by cname;

STUD_NAME	CNAME	RANK
Mahesh	E-Rangoli	1st
Mahi	kabaddi	1st
Mahavir	programming	3rd

4. **Display the names of first three winners of each competition.**

SQL> select cname,count(*)/(select sum(count(distinct(stud_reg_no))) as num from student

2 group by stud_reg_no) as average from student

3 join competition comp on student.cno=comp.cno group by cname;

CNAME	AVERAGE
E-Rangoli	.333333333
programming	.333333333
kabaddi	.333333333

5. **Display total number of competition held in the year 2014.**

SQL> select count(cname) from competition

2 where cno in (select cno from student where year='2014');

COUNT(CNAME)
0

Practical Slip:-16

Q8(i). Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

Employee (emp_id, emp_name, address)

Investment (inv_no, inv_name, inv_date, inv_amount)

Relation between Employee and Investment is One to Many.

Constraint: Primary key, inv_amount should be > 0.

```
SQL> create table employee
  2  (emp_id varchar(10) primary key,
  3  emp_name varchar(20),
  4  address varchar(20));
SQL> insert into employee values('E101','mahesh','pune');
 1 row created.
SQL> insert into employee values('E102','lalit','surat');
 1 row created.
SQL> insert into employee values('E103','meghna','pune');
 1 row created.
SQL> insert into employee values('E104','rukum','mumbai');
 1 row created.
SQL> select * from employee;
EMP_ID     EMP_NAME    ADDRESS
---------- -------------------- --------------------
E101       mahesh              pune
E102       lalit               surat
E103       meghna              pune
E104       rukum               Mumbai
SQL> create table investment
  2  (inv_no varchar(10) primary key,
```

```
  3  inv_name varchar(20),
  4  inv_date date,
  5  inv_amount number(7),
  6  emp_id varchar(10),
  7  foreign key(emp_id) references employee1);
Table created.
SQL> insert into investment values('I01','stock market','07-mar-2019',200000,'E102');
  1 row created.
SQL> insert into investment values('I02','mutual funds','08-jun-2019',150000,'E103');
  1 row created.
SQL> insert into investment values('I03','real estate','14-dec-2019',300000,'E102');
  1 row created.
SQL> insert into investment values('I04','gold','2-jan-2013',500000,'E104');
  1 row created.
SQL> select * from investment;
INV_NO    INV_NAME   INV_DATE    INV_AMOUNT  EMP_ID
---------- -------------------- ----------------   ----------------   ----------
I01       stock market      07-MAR-19     200000        E102
I02       mutual funds      08-JUN-19     150000        E103
I03       real estate          14-DEC-19     300000        E102
I04       gold                  02-JAN-13     500000        E104
```

Consider the above tables and execute the following queries:

1. **Display the details of employee who has invested amount in "Mutual Fund".**

```
SQL> select * from employee
  2  where emp_id=(select emp_id from investment where inv_name='mutual funds');
 EMP_ID    EMP_NAME       ADDRESS
 ---------- -------------------- --------------------
 E103      meghna             pune
```

2. **Add column Phone_No in Employee table.**

SQL> alter table employee1

2 add phone_no number(12);

Table altered.

Q8(ii). Consider the above tables and execute the following queries:

1. **Display employee details who have invested more than 100000.**

SQL> select * from employee1

2 where emp_id in (select emp_id from investment where inv_amount>100000);

EMP_ID	EMP_NAME	ADDRESS	PHONE_NO
E102	lalit	surat	
E103	meghna	pune	
E104	rukum	Mumbai	

2. **Display employee wise total investment amount.**

SQL> select emp_name,sum(inv.inv_amount) from employee1

2 join investment inv on employee1.emp_id=inv.emp_id group by emp_name;

EMP_NAME	SUM(INV.INV_AMOUNT)
meghna	150000
rukum	500000
lalit	500000

3. **Display the employee names who invest on date 2nd Jan 2013.**

SQL> select emp_name from employee1

2 where emp_id in (select emp_id from investment where inv_date='02-jan-2013');

EMP_NAME
rukum

4. **Display employee whose investment are more than 3.**

SQL> select emp_name from employee1

2 where emp_id in (select emp_id from investment group by emp_id having count(*)>3);

no rows selected

5. **Find average investment of employees of Pune.**

SQL> select avg(inv_amount) from investment

2 where emp_id in (select emp_id from employee1 where address='pune');

AVG(INV_AMOUNT)

150000

Practical Slip:-18

Q9(i). Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

Game (game_name, no_of_players,coach_name)
Player (pid, pname, address, club_name)

Relation between Game and Player is Many to Many.
Constraint: Primary key, no_of_players should be > 0.

```
SQL> create table game
  2  (game_name varchar(20) primary key,
  3  no_of_players number(2)check(no_of_players>0),
  4  coach_name varchar(15));

Table created.

SQL> insert into game values('cricket',11,' mr. sehwag');
  1 row created.

SQL> insert into game values('cycling',20,'lalit');
  1 row created.

SQL> insert into game values('kabaddi',7,'rakesh');
  1 row created.

SQL> insert into game values('footboll',12,'rukum');
  1 row created.

SQL> select * from game;

GAME_NAME   NO_OF_PLAYERS COACH_NAME
-------------------- ------------- ---------------
cricket                  11 mr. sehwag
cycling                  20 lalit
kabaddi                    7 rakesh
footboll                  12 rukum

SQL> create table player
  2  (pid varchar(10),
  3  pname varchar(20),
```

```
  4  address varchar(20),
  5  club_name varchar(15),
  6  gname varchar(20),
  7  foreign key(gname) references game(game_name));
```

Table created.

SQL> insert into player values('P01','mahesh','pune','unity sport','cricket');

1 row created.

SQL> insert into player values('P02','rajesh','surat','motion sport','kabaddi');

1 row created.

SQL> insert into player values('P03','rukum','mumbai','future sport','footboll');

1 row created.

SQL> insert into player values('P04','nayara','delhi','vmax sport','cycling')

1 row created.

SQL> select * from player;

PID	PNAME	ADDRESS	CLUB_NAME	GNAME
P01	mahesh	pune	unity sport	cricket
P02	rajesh	surat	motion sport	kabaddi
P03	rukum	mumbai	future sport	footboll
P04	nayara	delhi	vmax sport	Cycling

Consider the above tables and execute the following queries:

1. **Display the name of club whose coach is "Mr. Sehwag".**

SQL> select club_name from player

2 where gname in (select game_name from game where coach_name='mr. sehwag');

CLUB_NAME

---

unity sport

2. **Update the game name of player to cricket whose name is “Mr Rahane”.**

```
SQL> update player set gname='criket'
  2  where pname='Mr. Rahane';
0 rows updated.
```

Q9(ii). Consider the above tables and execute the following queries:

1. **Display players from ‘Delhi’.**

```
SQL> select pname from player
  2  where address='delhi';
PNAME
--------------------
Nayara
```

2. **List all games which require more than 4 players.**

```
SQL> select game_name from game where no_of_players>4;
GAME_NAME
--------------------
cricket
cycling
kabaddi
footboll
```

3. **Find the total number of cricket players of 'sports club”.**

```
SQL> select  count(*) as total_player from player
  2  where gname='cricket' and club_name='sports club';
 TOTAL_PLAYER
------------
         0
```

4. Display games having more number of players than that of football.

SQL> select game_name from game

2 where no_of_players>(select no_of_players from game where game_name='footboll');

GAME_NAME

Cycling

5. Display coach wise player details.

SQL> select coach_name,p. * from game join player p on p.gname=game.game_name

2 order by coach_name;

COACH_NAME	PID	PNAME	ADDRESS	CLUB_NAME	GNAME
lalit	P04	nayara	delhi	vmax sport	cycling
mr. sehwag	P01	mahesh	pune	unity sport	cricket
rakesh	P02	rajesh	surat	motion sport	kabaddi
rukum	P03	rukum	mumbai	future sport	footboll

Practical Slip:-20

Q10(i). Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

Wholesaler (w_no, w_name, address, city)

Product (product_no, product_name, rate)

Relation between Wholesaler and Product is Many to Many with quantity as descriptive attribute.

Constraint: Primary key, rate should be > 0.

```
SQL>create table wholesaler
  2  (w_no varchar(20) primary key,
  3  w_name varchar(20),
  4 address varchar(20), city varchar(10));
```

Table created.

```
SQL> insert into wholesaler values('W101','mr.patil','mans nagar','pune');
  1 row created.
SQL> insert into wholesaler values('W102','mr.khabia','narhe','pune');
  1 row created.
SQL> insert into wholesaler values('W103','mukesh','pimpri','pune');
   1 row created.
SQL> insert into wholesaler values('W104','rakesh','ghandhi nagar','mumbai');
   1 row created.
SQL> select * from wholesaler;
```

W_NO	W_NAME	ADDRESS	CITY
W101	mr.patil	mans nagar	pune
W102	mr.khabia	narhe	pune
W103	mukesh	pimpri	pune
W104	rakesh	ghandhi nagar	Mumbai

```
SQL> create table product
  2  (p_no varchar(5),
  3  p_name varchar(20),
  4  rate number(6) check (rate>0),
  5  quantity number(4),
  6  wno varchar(10),
  7  primary key (p_no,wno),
  8  foreign key(wno) references wholesaler(w_no));
Table created.
SQL> insert into product values('P01','Monitor',5000,20,'W101');
 1 row created.
SQL> insert into product values('P02','keyboard',500,30,'W102');
  1 row created.
SQL> insert into product values('P03','mouse',300,40,'W103');
 1 row created.
SQL> insert into product values('P04','glass',100,50,'W104');
 1 row created.
SQL> select * from product;
P_NO  P_NAME        RATE   QUANTITY   WNO
----- -------------------- ---------- ----------   -------
P01   Monitor          5000       20        W101
P02   keyboard         500        30        W102
P03   mouse            300        40        W103
P04   glass            100        50        W104
```

Consider the above tables and execute the following queries:

1. **Update product_name to “Monitor” whose supplier is “Mr. Patil”.**

```
SQL> update product set p_name='monitor'
  2 where wno in (select w_no from wholesaler where w_name='mr. patil');
0 rows updated.
```

2 Display product wise rate.

SQL> select p_name,rate from product;

P_NAME	RATE
Monitor	5000
keyboard	500
mouse	300
glass	100

Q10(ii). Consider the above tables and execute the following queries:

1. **Display wholesaler from 'Pune' city and supplying 'Monitor'.**

SQL> select w_name from wholesaler where city='pune'

2 and w_no in(select wno from product where p_name='monitor');

no rows selected

2. **Display total number of wholesaler of each product.**

SQL> select p_name,count(*) from product join wholesaler on product.wno=wholesaler.w_no group by p_name;

P_NAME	COUNT(*)
keyboard	1
glass	1
Monitor	1
mouse	1

3. **Display all wholesalers who are supplying ‘Keyboard’ with maximum price.**

```
SQL> select w_no from wholesaler w join product p on w.w_no=p.wno
  2  where w_no in (select wno from product where p_name='keyboard')
  3  and rate=(select max(rate) from product where p_name='keyboard');

W_NO
--------------------
W102
```

4. **Display total quantity of each product sold by ‘Mr. Khabia’.**

```
SQL> select p_name,quantity from product
  2  where wno=(select w_no from wholesaler where w_name='mr. khabia');

no rows selected
```

5. **Decrement rate of all products by 5% supplied by wholesaler from 'Pune ' city.**

```
SQL> select p_name, rate-0.05*rate from product
  2  where wno in (select w_no from wholesaler where ciry='pune');

P_NAME             RATE-0.05*RATE
-------------------- --------------
Monitor                    4750
keyboard                    475
mouse                       285
```

Practical Slip:-22

Q11(i). Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

Train (train_no, train_name, depart_time, arrival_time,source_stn**,** dest_stn)

Passenger (p_id,p_name,address,age, gender)

Relation between Train and Passenger is Many to Many with seat_no, amount and date as descriptive attribute.

Constraint: Primary key, seat_no should not be null.

```
SQL> create table train
  2  (t_no number(6) primary key,
  3  t_name varchar(20),
  4  depart_time varchar(10),
  5  arrival_time varchar(10),
  6  source_stn varchar(25),
  7  dest_stn varchar(15));

Table created.

SQL> insert into train values(12267,'sahyadri express','11:25 PM','05:55 AM','mumbai','ahmedabad');

  1 row created.

SQL> insert into train values(22201,'kokan express','08:15 PM','06:35 AM','pune','mumbai');

  1 row created.

SQL> insert into train values(22293,'shatabdi express','07:54 PM','09:39 AM','pune','uruli');

  1 row created.

SQL> insert into train values(12426,'rajdhani express','07:00 PM','05:39 AM','jammu','new delhi');

    0    row created.

SQL> select * from train;

      T_NO    T_NAME          DEPART_TIM    ARRIVAL_TI   SOURCE_STN    DEST_STN
---------- ---------------------- ------------------- ----------------  -----------------  -----------------
     12267    sahyadri express     11:25 PM     05:55 AM        mumbai       ahmedabad
     22201    kokan express         08:15 PM   06:35 AM          pune           mumbai
     22293    shatabdi express     07:54 PM   09:39 AM          pune           uruli
```

```
 T_NO    T_NAME              DEPART_TIM   ARRIVAL_TI SOURCE_STN     DEST_STN
---------- ---------------------  ---------- ----------  -------------       --------------
  12426  rajdhani express     07:00 PM   05:39 AM   jammu        new delhi
```

```
SQL> create table passenger
  2  (p_id varchar(10) primary key,
  3  p_name varchar(20),
  4  address varchar(10),
  5  age number(3),
  6  gender varchar(6),
  7  seat_no number(4) check (seat_no>0),
  8  amount number(6),
  9  tdate varchar(15),
 10  t_no number(6),
 11  foreign key(t_no) references train);

Table created.

SQL>insert into passenger values('P01','mahesh kumar','pune',19,'male',192,230,'26-jan-2023',12267);

1 row created.

SQL> insert into passenger values('P02','rukum','mumbai',20,'male',187,278,'5-jan-2019',22201);

1 row created.

SQL> insert into passenger values('P03','mr. roy','pune',65,'male',197,459,'21-jan-2019',22293);

1 row created.

SQL> insert into passenger values('P04','nayara','jammu',76,'female',357,7812,'27-jan-2023',12426);

1 row created.

SQL> insert into passenger values('P05','meghna','pune',19,'female',545,123,'19-jan-2019',12267);

1 row created.

SQL> insert into passenger values('P06','nitiya','mumbai',34,'female',845,823,'14-mar-2023',22293);

1 row created.
```

SQL> select * from passenger;

P_ID	P_NAME	ADDRESS	AGE	GENDER	SEAT_NO	AMOUNT	TDATE	T_NO
P01	mahesh kumar	pune	19	male	192	230	26-jan-2023	12267
P02	rukum	mumbai	20	male	187	278	5-jan-2019	22201
P03	mr. roy	pune	65	male	197	459	21-jan-2019	22293
P04	nayara	jammu	76	female	357	7812	27-jan-2023	12426
P05	meghna	pune	19	female	545	123	19-jan-2019	12267
P06	nitiya	mumbai	34	female	845	823	14-mar-2023	22293

6 rows selected.

Consider the above tables and execute the following queries:

1. **Display passenger names and their seat no's of train "sahyadri express".**

SQL> select p_name,seat_no,t_name from passenger,train

2 where t_name='sahyadri express';

P_NAME	SEAT_NO	T_NAME
mahesh kumar	192	sahyadri express
rukum	187	sahyadri express
mr. roy	197	sahyadri express
nayara	357	sahyadri express
meghna	545	sahyadri express
nitiya	845	sahyadri express

6 rows selected.

2. **Display details of train in which "Mr. Roy" is travelling from "Pune" to "Uruli Kanchan".**

SQL> select t_name from train,passenger

2 where p_name='mr. roy' and SOURCE_STN='pune' and DEST_STN='uruli';

T_NAME

---

shatabdi express

Q11(ii). Consider the above tables and execute the following queries:

1. **Display passenger details having age>50**

```
SQL> select * from passenger
  2  where age>50;

P_ID   P_NAME   ADDRESS    AGE   GENDER   SEAT_NO   AMOUNT   TDATE     T_NO
---------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- --------
P03      mr. roy      pune       65    male       197     459       21-jan-2019   22293
P04      nayara       jammu      76    female     357     7812      27-jan-2023   12426
```

2. **Display total amount collected for "Kokan Express" on 5th January 2019.**

```
SQL> select sum(amount) from passenger,train
  2  where t_name='kokan express' and tdate='5-jan-2019';

SUM(AMOUNT)
-----------
        278
```

3. **Find total number of passenger of "Pune to Mumbai" route.**

```
SQL> select count(p_id) from passenger,train
  2  where source_stn='pune' and dest_stn='mumbai';

COUNT(P_ID)
-----------
          6
```

4. **Cancel all the trains of 21st January 2019.**

```
SQL> select t_name from train, passenger
  2  where passenger.t_no=train.t_no
  3  and tdate='21-jan-2019';
```

T_NAME

---

shatabdi express

5. **Calculate total number of male passenger in “Shatabdi Express”.**

SQL> select count(gender) from passenger,train

1 where gender='male' and t_name='shatabdi express';

COUNT(GENDER)

3

Practical Slip:-24

Q12(i). Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

Bus(bus_no, capacity, depo_name)

Route(Route_no, source, destination, no_of_stations)

Relation between Bus and Route is Many to One.

Constraint: Bus capacity should be greater than 0, depo_name should not be null.

```
SQL> create table bus
  2  (bus_no number(10) primary key,
  3  capacity number(3) check(capacity>0),
  4 depo_name varchar(15));
SQL> insert into bus values('RJ04QL8902',90,'kothrud');
 1 row created.
SQL> insert into bus values('MH12HL7812',100,'bhor');
  1 row created.
SQL> insert into bus values('MH11HL9852',100,'shirur');
  1 row created.
SQL> insert into bus values('GJ15OR8720',80,'indapur');
  1 row created.
SQL> select * from bus;
    BUS_NO      CAPACITY    DEPO_NAME
    ----------          ---------     ---------------
       RJ04QL8902      90      kothrud
       MH12HL7812     100    bhor
       MH11HL9852     100    shirur
       GJ15OR8720      80     indapur
SQL> create table route
  2  (route_no number(3) primary key,
  3  source varchar(10),
  4  dest_name varchar(15),
```

```
5     no_of_stations number(3),
6    bus_no varchar(10),
6     foreign key(bus_no) references bus);
Table created.
SQL> insert into route values(41,'petrol','Barmer(RJ)',10,'RJ04QL8902');
   1 row created.
SQL> insert into route values(34,'diesel','Maharashtra',15,'MH12HL7812');
   1 row created.
SQL> insert into route values(35,'diesel','Maharashtra',89,'MH12HL7812');
   1 row created.
SQL> insert into route values(78,'electricty','Maharashtra',90,'MH11HL9852');
  1 row created.
SQL> insert into route values(61,'electricty','indapur',190,'GJ15OR8720');
  1 row created.
SQL> SELECT * FROM ROUTE;
 ROUTE_NO  SOURCE   DEST_NAME   NO_OF_STATIONS   BUS_NO
---------- ---------- ---------- ---------- ----------
    41          petrol      Barmer(RJ)          10          RJ04QL8902
    34          diesel      Maharashtra         15          MH12HL7812
    35          diesel      Maharashtra         89          MH12HL7812
    78          electricty  Maharashtra         90          MH11HL9852
    61          electrict   indapur             190         GJ15OR8720
```

Consider the above tables and execute the following queries:

1. **List all buses at depo “kothrud”.**

```
SQL> select bus_no from bus where depo_name='kothrud';
BUS_NO
----------
RJ04QL8902
```

2. **Delete bus details whose number is “MH12HL7812”.**

SQL> delete bus where bus_no='MH12HL7812';

1 row deleted.

Q12(ii). Consider the above tables and execute the following queries: [25 Marks]

1. **List all buses on route no 41.**

SQL> select bus_no,route_no from route

2 where route_no='41';

```
BUS_NO      ROUTE_NO
----------          ----------
RJ04QL8902        41
```

2. **List the route details having number of stations > 10.**

SQL> select * from route

2 where no_of_stations>10;

```
ROUTE_NO SOURCE    DEST_NAME   NO_OF_STATIONS  BUS_NO
---------- ---------- ---------- ----------  ----------
      35        diesel          Maharashtra     89                MH12HL7812
      78        electricty      Maharashtra     90                MH11HL9852
      61        electricty      indapur         190               GJ15OR8720
      34         diesel          Maharashtra      15                MH12HL7812
```

3. **Delete all buses having capacity < 20.**

SQL> delete bus where capacity<20;

0 rows deleted.

4. **Find the maximum number of stations.**

SQL> select max(no_of_stations) from route;

MAX(NO_OF_STATIONS)

190

5. **List all routes starting from “station”.**

SQL> select * from route

2 where DEST_NAME='indapur';

ROUTE_NO	SOURCE	DEST_NAME	NO_OF_STATIONS	BUS_NO
61	electricty	indapur	190	GJ15OR8720

Practical Slip:-26

Q13(i). Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

Gym (Name, city, charges, scheme)

Member (ID, Name, phoneNo, address)

Relation between Gym and member is one to many.
Constraint: Primary Key, charges must be greater than0.

```
SQL> create table gym
  2  (gym_id varchar(5) primary key,
  3  name varchar(20),
  4  city varchar(10),
  5  charges number(7),
  6  scheme varchar(40);
Table created.
SQL> insert into gym values('G101','sinhgad fitness','narhe',5000,'barbell squats: 5 sets of 5 reps.');
  1 row created.
SQL> insert into gym values('G102','gold gym','pimpri',15000,'barbell deadlifts: 3 sets of 3 reps.');
  1 row created.
SQL> insert into gym values('G103','lifetime','pune',25000,'push-up: 3 sets of 15 reps.');
   1 row created.
SQL> insert into gym values('G104','lifetime','mumbai',20000,'planks: 3 sets, 1 minute hold each.');
   1 row created.
SQL> select * from gym;
GYM_I   NAME            CITY     CHARGES   SCHEME
 -------  --------------------  ---------- ----------  --------------------------------
G101   sinhgad fitness    narhe     5000     barbell squats: 5 sets of 5 reps.
G102   gold gym           pimpri    15000    barbell deadlifts: 3 sets of 3 reps.
G103   lifetime           pune      25 000   push-up: 3 sets of 15 reps.
```

GYM_I NAME CITY CHARGE SCHEME

------ ---------- ---------- ---------- ----------

G104 lifetime mumbai 20000 planks: 3 sets, 1 minute hold each.

SQL> create table member

2 (id varchar(5) primary key,

3 m_name varchar(15),

4 phone_no number(10),

5 address varchar(25),

6 gym_id varchar(5),

7 foreign key(gym_id) references gym);

Table created.

SQL> insert into member values('m101','mr. patil',123456789,'manaji nagar pune','G101');

1 row created.

SQL> insert into member values('m102','mahesh',7425985294,'dhayari pune','G102');

1 row created.

SQL> insert into member values('m103','rukum',2123445547,'narhe pune','G103');

1 row created.

SQL> insert into member values('m104','vicky',9087984567,'ambegoav pune','G103');

1 row created.

SQL> insert into member values('m105','lait',8949678478,'wadgoav pune','G104');

1 row created.

SQL> insert into member values('m106','sahil',7563749834,'dhayari pune','G101');

1 row created.

SQL> select * from member;

ID M_NAME PHONE_NO ADDRESS GYM_I

----- ---------- ---------- ---------- ---------

m101 mr. patil 123456789 pune G101

m102 mahesh 7425985294 dhayari pune G102

m103 rukum 2123445547 narhe pune G103

```
ID    M_NAME      PHONE_NO    ADDRESS         GYM_I
----- ---------- ------------------ ----------------    ---------
m104    vicky     9087984567   ambegoav pune   G103
m105   lalit       8949678478    wadgoav pune    G104
m106   sahil      7563749834 dhayari pune        G101
rows selected.
```

Consider the above tables and execute the following queries:

1. **Display the scheme details of "Gold Gym".**

```
SQL> select scheme from gym
  2  where name='gold gym';
SCHEME
----------------------------------------
barbell deadlifts: 3 sets of 3 reps.
```

2. **List out all the gym of "Pimpri" city.**

```
SQL> select * from gym
  2  where city='pimpri';
GYM_I  NAME        CITY     CHARGES   SCHEME
------ ---------- ------------ ---------- ----------
G102   gold gym      pimpri   15000   barbell deadlifts: 3 sets of 3 reps.
```

Q13(ii). **Consider the above tables and execute the following queries:**

1. **Give the details of scheme to which "Mr. Patil" is admitted.**

```
SQL> select scheme from gym,member
  2  where gym.gym_id=member.gym_id
  3  and member.m_name='mr. patil';
SCHEME
----------------------------------------
barbell squats: 5 sets of 5 reps.
```

2. List out gym wise members.

```
SQL> select name,m_name from gym,member
  2  where member.gym_id=gym.gym_id
  3  order by name,m_name desc;

NAME                M_NAME
------------------- ---------------
gold gym            mahesh
lifetime            vicky
lifetime            rukum
lifetime            lalit
sinhgad fitness     sahil
sinhgad fitness     mr. patil
6 rows selected.
```

3. List out scheme wise charges.

```
SQL> select scheme, charges from gym;

SCHEME                                   CHARGES
---------------------------------------- ----------
barbell squats: 5 sets of 5 reps.        5000
barbell deadlifts: 3 sets of 3 reps.     15000
push-up: 3 sets of 15 reps.              25000
planks: 3 sets, 1 minute hold each.      20000
```

4. **Display all the members from pune city who have paid Gym charges more than 10000.**

SQL> select m_name from member,gym

2 where member.gym_id=gym.gym_id

and city='pune' and charges>10000;

M_NAME

---

rukum

Vicky

5. **Drop column address from Member table.**

SQL> alter table member

2 drop column address;

Table altered.

Practical Slip:-28

Q14(i). Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

Blood (blood_id, blood_group, cholesterol_level, storage)

Donar (donar_id, name, address, gender, phone_no)

Relation between Blood and Donar is Many to one.

Constraint: Primary Key , Blood_group should not be null.

```
SQL> create table blood
  2  (blood_id varchar(5) primary key,
  3  blood_group varchar(10),
  4  bc_level varchar(10),
  5  storange number(5));
Table created.
SQL> insert into blood values('bl001','o+ve','200ml/dl',3)
  1 row created.
SQL> insert into blood values('bl002','o+ve','190ml/dl',4);
  1 row created.
SQL> insert into blood values('bl003','A+ve','290ml/dl',5);
  1 row created.
SQL> insert into blood values('bl004','B+ve','290ml/dl',5);
  1 row created.
SQL> insert into blood values('bl005','AB+ve','239ml/dl',3);
  1 row created.
SQL> select * from blood;
BLOOD_ID  BLOOD_GROU   BC_LEVEL    STORANGE
--------- ----------  ----------  ----------
bl001         o+ve                200ml/dl          3
bl002          o+ve               190ml/dl          4
```

BLOOD_ID BLOOD_GROU BC_LEVEL STORANGE

--- --- --- ---

bl003 A+ve 290ml/dl 5

bl004 B+ve 290ml/dl 5

bl005 AB+ve 239ml/dl 3

SQL> create table donar

2 (donar_id varchar(5) primary key,

3 name varchar(10),

4 address varchar(15),

5 gender varchar(6),

6 phone_no number(10),

7 blood_id varchar(5),

8 foreign key(blood_id) references blood);

Table created.

SQL> insert into donar values('do001','mahesh','pune','male',7425985294,'bl001');

1 row created.

SQL> insert into donar values('do002','mahandule','mumbai','male',8495965292,'bl002');

1 row created.

SQL> insert into donar values('do003','lalit','surat','male',6495965692,'bl002');

1 row created.

SQL> insert into donar values('do004','rukum','mumbai','male',6695965692,'bl003');

1 row created.

SQL> insert into donar values('do005','rahul','mumbai','male',6605965692,'bl004');

1 row created.

SQL> insert into donar values('do006','nayara','primri','female',6805967695,'bl005');

1 row created.

SQL> insert into donar values('do007','nitiya','barmer','female',7895267695,'bl001');

1 row created.

SQL> select * from donar;

DONAR_ID NAME ADDRESS GENDER PHONE_NO BLOOD

--- --- --- --- --- ---

do001 mahesh pune male 7425985294 bl001

do002 mahandule mumbai male 8495965292 bl002

do003 lalit surat male 6495965692 bl002

do004 rukum mumbai male 6695965692 bl003

do005 rahul mumbai male 6605965692 bl004

do006 nayara primri female 6805967695 bl005

do007 nitiya barmer female 7895267695 bl001

7 rows selected.

Consider the above tables and execute the following queries:

1. **List out Donar wise Blood_group.**

SQL> select blood_group from blood,donar

2 where blood.blood_id=donar.blood_id

3 group by blood_group;

BLOOD_GROU

AB+ve

B+ve

o+ve

A+ve

2. **Display cholesterol_level of blood which is given by 'Mr. Mahandule'.**

SQL> select bc_level from blood,donar

2 where blood.blood_id=donar.blood_id

and name='mahandule';

BC_LEVEL

190ml/dl

Q14(ii) Consider the above tables and execute the following queries:

1. **Display the stock of blood whose group is “O+ve”.**

SQL> select storange from blood

2 where blood_group='o+ve';

```
STORANGE
----------
         3
         4
```

2. **Delete gender column from Donar table.**

SQL> alter table donar

2 drop column gender;

Table altered.

3. **Update storage of blood to 100 bottles whose blood_group is “B+ve”.**

SQL> update blood set storange= 100 where blood_group='B+ve';

1 row updated.

SQL> select * from blood;

BLOOD_ID	BLOOD_GROU	BC_LEVEL	STORAGE
bl001	o+ve	200ml/dl	3
bl002	o+ve	190ml/dl	4
bl003	A+ve	290ml/dl	5
bl004	B+ve	290ml/dl	100
bl005	AB+ve	239ml/dl	3

4. **Display blood group having maximum storage.**

SQL> select blood_group, max(storange) from blood

2 group by blood_group;

BLOOD_GROU MAX(STORANGE)

---------- ----------

B+ve 100

5. **Display the details of donar in ascending order of donar_name.**

SQL> select name from donar

2 order by name asc;

NAME

---

lalit

mahandule

mahesh

nayara

nitiya

rahul

rukum

Practical Slip:-30

Q15(i). Consider the following entities and their relationships. Create a RDB in 3 NF with appropriate data types and Constraints.

Library(Lno, Lname, Location, Librarian, no_of_books)

Book(Bid, Bname, Author_Name, Price, publication)

Relation between Library and Book is one to many.
Constraint: Primary key, Price should not be null.

```
SQL> create table library
  2  (lno number(5) primary key,
  3  lname varchar(20),
  4  loc varchar(10),
  5  lbr_name varchar(15),
  6  no_of_books number(7));
Table created.
SQL> insert into library values(18491,'sinhgad college edu','narhe,pune','rajesh',10000);
  1 row created.
SQL> insert into library values(18529,'DPU Library','mumbai','rakesh',15000);
  1 row created.
SQL> insert into library values(18782,'DYP Library','Hyderabad','MD.mahesh',200000);
  1 row created.
SQL> insert into library values(17782,'ashiwarya college','jodhapur','Navin parohit',20000);
  1 row created.
SQL> select * from library;
     LNO    LNAME                 LOC              LBR_NAME      NO_OF_BOOKS
---------- ---------------------------  ----------------- ------------------ ------------------
   18491   sinhgad college edu    narhe,pune    rajesh             10000
   18529   DPU Library              mumbai       rakesh             15000
   18782   DYP Library             Hyderabad    MD.mahesh       200000
```

```
 17782  ashiwarya college    jodhapur   Navin parohit        20000
SQL> create table book1
  2  (bid varchar(7),
  3  bname varchar(25),
  4  author varchar(20),
  5  price number(5),
  6  publication varchar(20),
  7  lno number(5),
  8  foreign key(lno) references library);
Table created.
SQL> insert into book1 values('A198263','C programming','Dr. Sunita D. patil',235,'nirali',18491);
 1 row created.
SQL> insert into book1 values('A178349','DBMS','Dr. Ms. Manisha',265,'nirali',18491);
  1 row created.
SQL> insert into book1 values('B142345','POM','Dr. Mangesh',190,'vision',18529);
 1 row created.
SQL> insert into book1 values('B142345','POM','Dr. Mangesh',199,'BPB',17782);
   1 row created.
SQL> insert into book1 values('B192245','Maths','Dr. mahesh Suthar',599,'Sanjivni',18529);
  1 row created.
SQL> insert into book1 values('C150242','Chemisrty','Dr. Pawan sharma',299,'Sanjivni',18782);
   1 row created.
SQL> select * from book1;
BID        BNAME                AUTHOR              PRICE     PUBLICATION        LNO
------- ------------------------- -------------------------  --------- -------------------- ----------
A198263   C programming   Dr. Sunita D. patil        235    nirali                18491
A178349 DBMS                 Dr. Ms. Manisha          265   nirali                18491
B142345 POM                  Dr. Mangesh              190     vision                18529
B142345 POM                  Dr. Mangesh              199     BPB                   17782
```

BID BNAME AUTHOR PRICE PUBLICATION LNO

------- ------------------------ -------------------- ---------- -------------------- ----------

B192245 Maths Dr. mahesh Suthar 599 Sanjivni 18529

C150242 Chemisrty Dr. Pawan sharma 299 Sanjivni 18782

6 rows selected.

Consider the above tables and execute the following queries:

1. **Display library details having number of books greater than 10000.**

SQL> select * from library

2 where no_of_books>10000;

LNO LNAME LOC LBR_NAME NO_OF_BOOKS

---------- -------------------- ---------------------- -------------- ------------------------

18529 DPU Library mumbai rakesh 15000

18782 DYP Library Hyderabad MD.mahesh 200000

17782 ashiwarya college jodhapur Navin parohit 20000

2. **Display the list of books of 'BPB' publication.**

SQL> select * from book1

2 where publication='BPB';

BID BNAME AUTHOR PRICE PUBLICATION LNO

------- ------------------------ -------------------- ---------- -------------------- ----------

B142345 POM Dr. Mangesh 190 BPB 18529

B142345 POM Dr. Mangesh 199 BPB 17782

Q15(ii). Consider the above tables and execute the following queries:

1. Display publication wise number of books.

SQL> select publication, no_of_books from book1,library

2 order by publication,no_of_books;

PUBLICATION NO_OF_BOOKS

---------- ----------

BPB 10000

BPB 10000

BPB 15000

BPB 15000

BPB 20000

BPB 20000

BPB 200000

BPB 200000

Sanjivni 10000

Sanjivni 10000

Sanjivni 15000

PUBLICATION NO_OF_BOOKS

---------- ----------

Sanjivni 15000

Sanjivni 20000

Sanjivni 20000

Sanjivni 200000

Sanjivni 200000

nirali 10000

nirali 10000

nirali 15000

nirali 15000

nirali 20000

nirali 20000

PUBLICATION NO_OF_BOOKS

---------- ----------

nirali 200000

nirali 200000

24 rows selected.

2. Display total price of books of 'Nirali' publication.

SQL> select sum(price) from book1

2 where publication='nirali';

SUM(PRICE)

500

3. Delete the book details of Vision publication from 'DPU' Library.

SQL> delete book1 where publication='vision';

2 row deleted.

4. Display all books of author whose initial character of name is 'A'.

SQL> select author from book1 where author like 'a%';

no rows selected

5. Update number of books of success publication to 1000 from DYP library.

SQL> update library set no_of_books=1000 where lname='DYP Library';

1 row updated.

*****