BACHELOR OF COMPUTER APPLICATION LAB MANUAL

3rd Semester

Prepared By

Pure and Applied Science Dept.

Computer Application

MIDNAPORE CITY COLLEGE

INSTRUCTIONS TO STUDENTS

- Before entering the lab, the student should carry the following things (MANDATORY)

  1. Identity card issued by the college.
  2. Class notes
  3. Lab observation book
  4. Lab Manual
  5. Lab Record

- Student must sign in and sign out in the register provided when attending the lab session without fail.
- Come to the laboratory in time. Students, who are late more than 10 min., will not be allowed to attend the lab.
- Students need to maintain 80% attendance in lab if not a strict action will be taken.
- All students must follow a Dress Code while in the laboratory.
- Foods, drinks are NOT allowed.
- All bags must be left at the indicated place.
- Refer to the lab staff if you need any help in using the lab.
- Respect the laboratory and its other users.
- Workspace must be kept clean and tidy after experiment is completed.
- Read the Manual carefully before coming to the laboratory and be sure about what you are supposed to do.
- Do the experiments as per the instructions given in the manual.
- Copy all the programs to observation which are taught in class before attending the lab session.
- Students are not supposed to use floppy disks, pen drives without permission of lab- in charge.
- Lab records need to be submitted on or before the date of submission.

DATABASE MANAGEMENT SYSTEM LABORATORY MANUAL (Course Code: BCA-2196)

OVERVIEW OF Oracle 10g -- Installation & study about creating, inserting and storing record CREATE TABLE

```
create table msc(roll numeric(2), fname varchar(20), lname varchar(15));
```

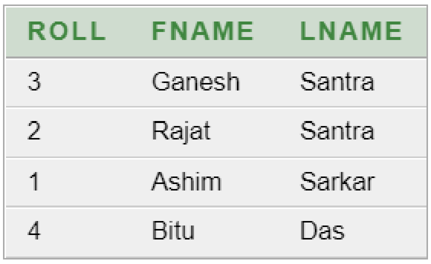
INSERT INTO TABLE

```
insert into msc values(3,'Ganesh','Santra');
insert into msc values(2,'Rajat','Santra'); insert into msc
values(1,'Ashim','Sarkar'); insert into msc values(4,'Bitu','Das');
```

VIEW THE TABLE

```
select * from msc;
```

OUTPUT



ROLL	FNAME	LNAME
3	Ganesh	Santra
2	Rajat	Santra
1	Ashim	Sarkar
4	Bitu	Das

1. **Create Student Table (Create, Insert, view, add, update, delete and conditions) CREATE**

```
create table student2(name varchar(15),roll numeric(3), addr
varchar(25),phn_no numeric(10),email varchar(20),dob varchar(8),age
numeric(3),marks numeric(5),aadhar_no numeric(15));
```

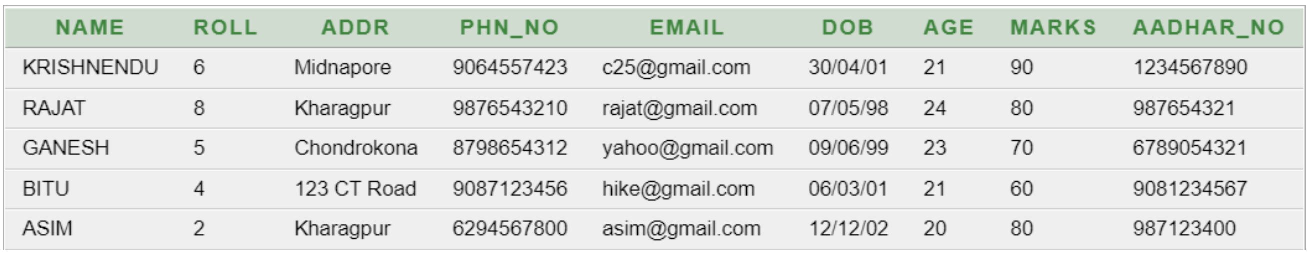
INSERT

```
insert into student2
values('KRISHNENDU',06,'Midnapore',9064557423,'c25@gmail.com','30/04/0
1',21,90, 1234567890);
insert into student2
values('RAJAT',08,'Kharagpur',9876543210,'rajat@gmail.com','07/05/98',24,80,
0987654321);
insert into student2
values('GANESH',05,'Chondrokona',8798654312,'yahoo@gmail.com','09/06/99'
,23,70, 6789054321);
insert into student2 values('BITU',04,'123 CT
Road',9087123456,'hike@gmail.com','06/03/01',21,60,9081234567); insert into
student2
```

```
values('ASIM',02,'Kharagpur',6294567800,'asim@gmail.com','12/12/02',20,80,0
987123400);
insert into student2
values('ARPAN',03,'Ghatal',7865432190,'arpan@gmail.com','08/11/97',23,70,5
678094312, 'Botany');
```

VIEW

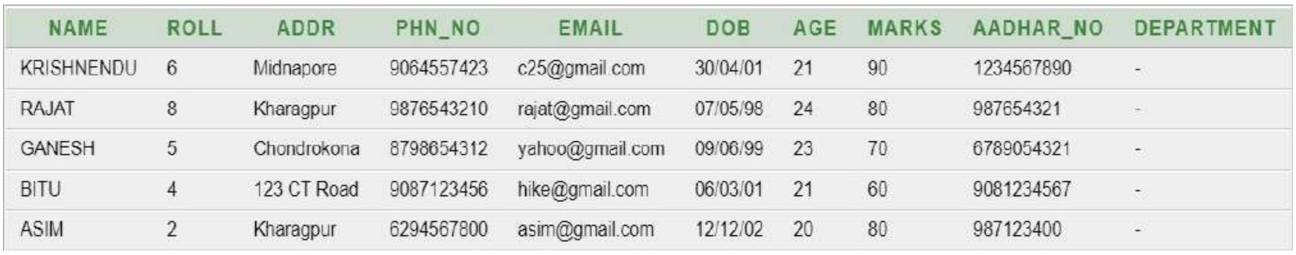
```
select * from student2;
```



NAME	ROLL	ADDR	PHN_NO	EMAIL	DOB	AGE	MARKS	AADHAR_NO
KRISHNENDU	6	Midnapore	9064557423	c25@gmail.com	30/04/01	21	90	1234567890
RAJAT	8	Kharagpur	9876543210	rajat@gmail.com	07/05/98	24	80	987654321
GANESH	5	Chondrokona	8798654312	yahoo@gmail.com	09/06/99	23	70	6789054321
BITU	4	123 CT Road	9087123456	hike@gmail.com	06/03/01	21	60	9081234567
ASIM	2	Kharagpur	6294567800	asim@gmail.com	12/12/02	20	80	987123400

ADD

```
alter table student2 add department varchar(20);
```

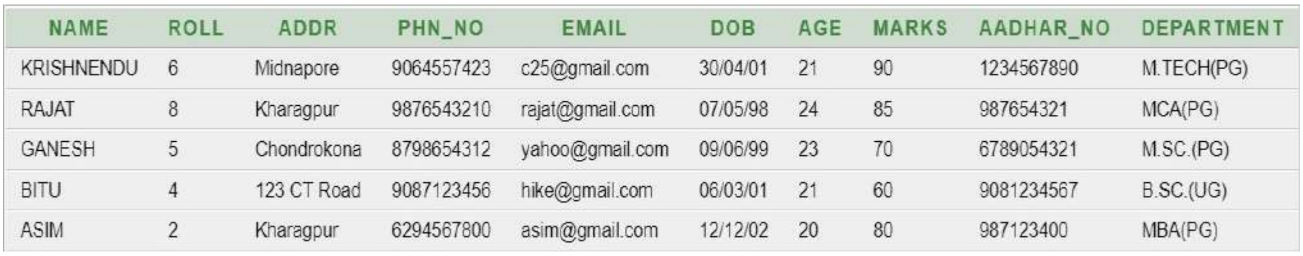


NAME	ROLL	ADDR	PHN_NO	EMAIL	DOB	AGE	MARKS	AADHAR_NO	DEPARTMENT
KRISHNENDU	6	Midnapore	9064557423	c25@gmail.com	30/04/01	21	90	1234567890	-
RAJAT	8	Kharagpur	9876543210	rajat@gmail.com	07/05/98	24	80	987654321	-
GANESH	5	Chondrokona	8798654312	yahoo@gmail.com	09/06/99	23	70	6789054321	-
BITU	4	123 CT Road	9087123456	hike@gmail.com	06/03/01	21	60	9081234567	-
ASIM	2	Kharagpur	6294567800	asim@gmail.com	12/12/02	20	80	987123400	-

UPDATE

```
update student2 set marks=85 where name='RAJAT'; update student2 set
department='M.SC.(PG)' where roll=05;
update student2 set department='M.TECH.(PG)' where roll=06; update student2
set department='MBA(PG)' where roll=02; update student2 set
department='B.SC.(UG)' where roll=04; update student2 set
department='MCA(PG)' where roll=08;
```

DELETE

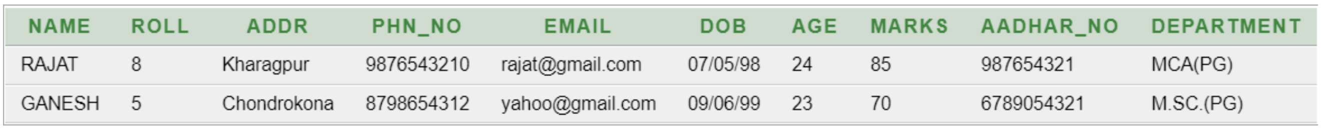


NAME	ROLL	ADDR	PHN_NO	EMAIL	DOB	AGE	MARKS	AADHAR_NO	DEPARTMENT
KRISHNENDU	6	Midnapore	9064557423	c25@gmail.com	30/04/01	21	90	1234567890	M.TECH(PG)
RAJAT	8	Kharagpur	9876543210	rajat@gmail.com	07/05/98	24	85	987654321	MCA(PG)
GANESH	5	Chondrokona	8798654312	yahoo@gmail.com	09/06/99	23	70	6789054321	M.SC.(PG)
BITU	4	123 CT Road	9087123456	hike@gmail.com	06/03/01	21	60	9081234567	B.SC.(UG)
ASIM	2	Kharagpur	6294567800	asim@gmail.com	12/12/02	20	80	987123400	MBA(PG)

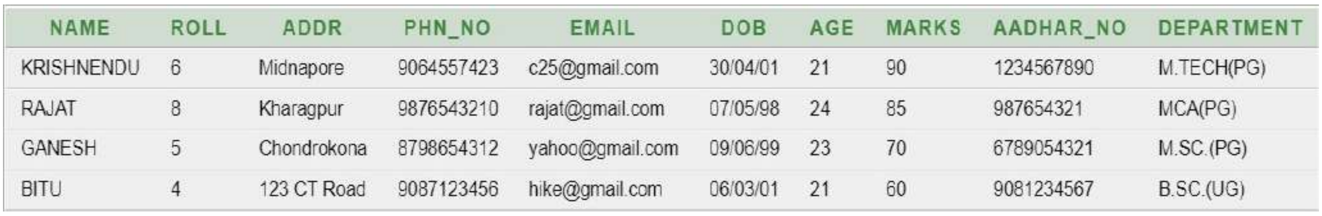
delete from student2 where roll=3;

CONDITION (and, or, where)

select * from student2 where age>=22 and age<=25;

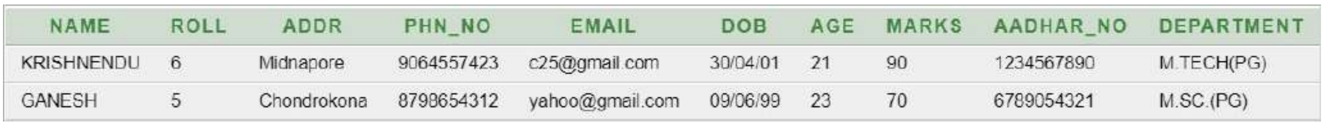


NAME	ROLL	ADDR	PHN_NO	EMAIL	DOB	AGE	MARKS	AADHAR_NO	DEPARTMENT
RAJAT	8	Kharagpur	9876543210	rajat@gmail.com	07/05/98	24	85	987654321	MCA(PG)
GANESH	5	Chondrokona	8798654312	yahoo@gmail.com	09/06/99	23	70	6789054321	M.SC.(PG)



NAME	ROLL	ADDR	PHN_NO	EMAIL	DOB	AGE	MARKS	AADHAR_NO	DEPARTMENT
KRISHNENDU	6	Midnapore	9064557423	c25@gmail.com	30/04/01	21	90	1234567890	M.TECH(PG)
RAJAT	8	Kharagpur	9876543210	rajat@gmail.com	07/05/98	24	85	987654321	MCA(PG)
GANESH	5	Chondrokona	8798654312	yahoo@gmail.com	09/06/99	23	70	6789054321	M.SC.(PG)
BITU	4	123 CT Road	9087123456	hike@gmail.com	06/03/01	21	60	9081234567	B.SC.(UG)

select * from student2 where marks=70 or marks=90;

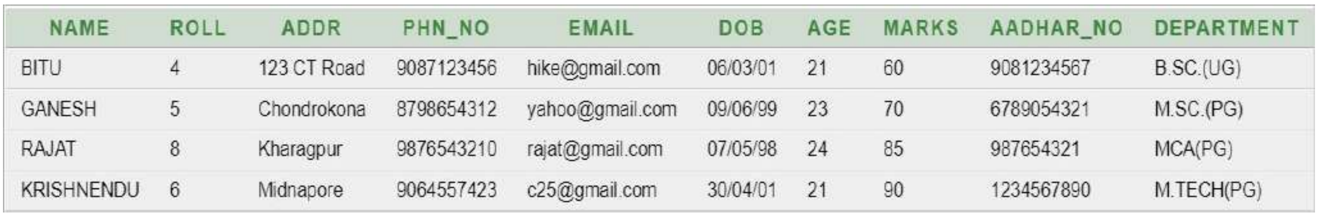


NAME	ROLL	ADDR	PHN_NO	EMAIL	DOB	AGE	MARKS	AADHAR_NO	DEPARTMENT
KRISHNENDU	6	Midnapore	9064557423	c25@gmail.com	30/04/01	21	90	1234567890	M.TECH(PG)
GANESH	5	Chondrokona	8798654312	yahoo@gmail.com	09/06/99	23	70	6789054321	M.SC.(PG)

2. **Implementation of student table by using (ascending, descending, group by, order by, distinct, count, max, min, average, character recognition)**

ASCENDIING ORDER BY ROLL

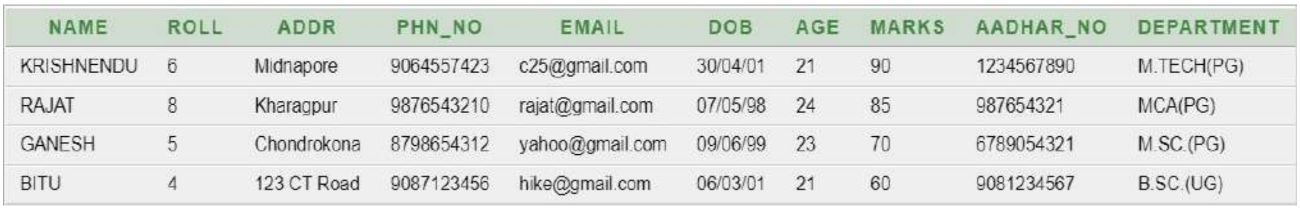
select * from student2 order by roll asc;



NAME	ROLL	ADDR	PHN_NO	EMAIL	DOB	AGE	MARKS	AADHAR_NO	DEPARTMENT
BITU	4	123 CT Road	9087123456	hike@gmail.com	06/03/01	21	60	9081234567	B.SC.(UG)
GANESH	5	Chondrokona	8798654312	yahoo@gmail.com	09/06/99	23	70	6789054321	M.SC.(PG)
RAJAT	8	Kharagpur	9876543210	rajat@gmail.com	07/05/98	24	85	987654321	MCA(PG)
KRISHNENDU	6	Midnapore	9064557423	c25@gmail.com	30/04/01	21	90	1234567890	M.TECH(PG)

DESCENDIING ORDER BY MARKS

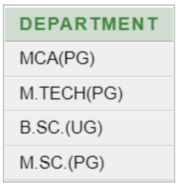
select * from student2 order by marks desc;



NAME	ROLL	ADDR	PHN_NO	EMAIL	DOB	AGE	MARKS	AADHAR_NO	DEPARTMENT
KRISHNENDU	6	Midnapore	9064557423	c25@gmail.com	30/04/01	21	90	1234567890	M.TECH(PG)
RAJAT	8	Kharagpur	9876543210	rajat@gmail.com	07/05/98	24	85	987654321	MCA(PG)
GANESH	5	Chondrokona	8798654312	yahoo@gmail.com	09/06/99	23	70	6789054321	M.SC.(PG)
BITU	4	123 CT Road	9087123456	hike@gmail.com	06/03/01	21	60	9081234567	B.SC.(UG)

GROUP BY DEPARTMENT

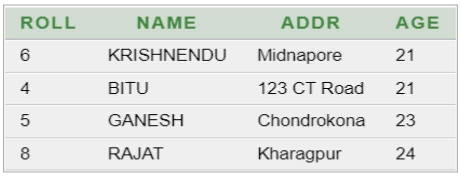
select department from student2 group by department;



DEPARTMENT
MCA(PG)
M.TECH(PG)
B.SC.(UG)
M.SC.(PG)

ORDER BY AGE

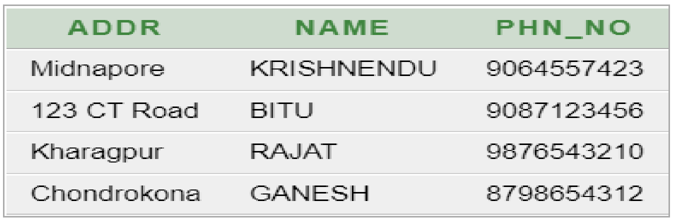
select roll, name, addr, age from student2 order by age;



ROLL	NAME	ADDR	AGE
6	KRISHNENDU	Midnapore	21
4	BITU	123 CT Road	21
5	GANESH	Chondrokona	23
8	RAJAT	Kharagpur	24

DISTINCT ADDRESS OF STUDENTS

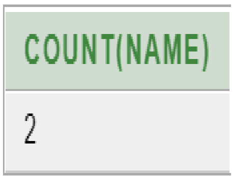
select distinct(addr), name, phn_no from student2;



ADDR	NAME	PHN_NO
Midnapore	KRISHNENDU	9064557423
123 CT Road	BITU	9087123456
Kharagpur	RAJAT	9876543210
Chondrokona	GANESH	8798654312

COUNT NAME WHOES AGE IS LESS THAN 23

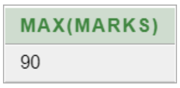
select count(name) from student2 where age<23;



COUNT(NAME)
2

COUNT MAXIMUM MARKS

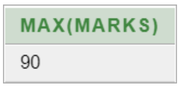
select max(marks) from student2;



MAX(MARKS)
90

COUNT MINIMUM MARKS

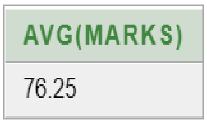
select min(marks) from student2;



MAX(MARKS)
90

COUNT AVERAGE MARKS

select avg(marks) from student2;

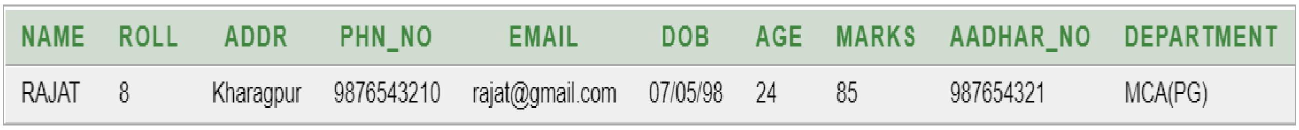


AVG(MARKS)
76.25

CHARACTER RECOGNIZATIOIN

- **LAST CHARACTER**

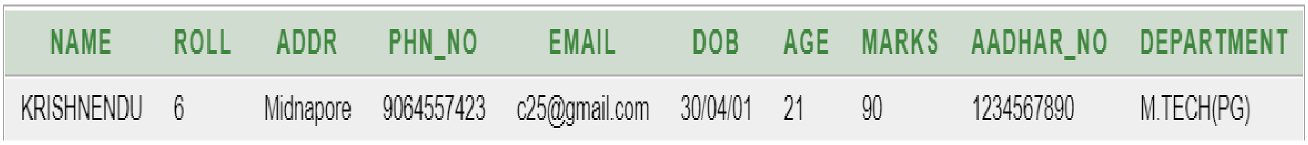
select * from student2 where name like '%T';



NAME	ROLL	ADDR	PHN_NO	EMAIL	DOB	AGE	MARKS	AADHAR_NO	DEPARTMENT
RAJAT	8	Kharagpur	9876543210	rajat@gmail.com	07/05/98	24	85	987654321	MCA(PG)

- **FIRST CHARACTER**

select * from student2 where name like 'K%';



NAME	ROLL	ADDR	PHN_NO	EMAIL	DOB	AGE	MARKS	AADHAR_NO	DEPARTMENT
KRISHNENDU	6	Midnapore	9064557423	c25@gmail.com	30/04/01	21	90	1234567890	M.TECH(PG)

3. **Creating Employee table with various queries**
   i) **Create a table employee (e_id, f_name, l_name, address, phone_no, email, salary, age, gender).**
   ii) **Insert 5 rows into this table of employee.**
   iii) **View the employee table. Create:**

create table employee2(e_id varchar(3),f_name varchar(15),l_name varchar(15),address varchar(25),phone_no numeric(10),email varchar(25),salary numeric(10),age numeric(3),gender varchar(8));

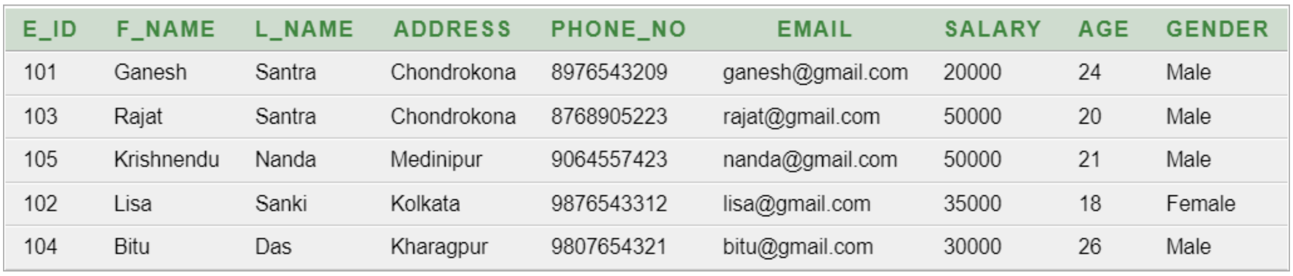
Insert:

insert into employee2 values(101,'Ganesh','Santra','Chondrokona',8976543209,'ganesh@gmail.com',20000,24,'Male'); insert into employee2 values(103,'Rajat','Santra','Chondrokona',8768905223,'rajat@gmail.com',50000,

```
20,'Male'); insert into employee2
values(105,'Krishnendu','Nanda','Medinipur',9064557423,'nanda@gmail.com',5
0000,21,'Male'); insert into employee2
values(102,'Lisa','Sanki','Kolkata',9876543312,'lisa@gmail.com',35000,18,'Fem
ale');
insert into employee2
values(104,'Bitu','Das','Kharagpur',9807654321,'bitu@gmail.com',30000,26,'Ma
le');
```

View:

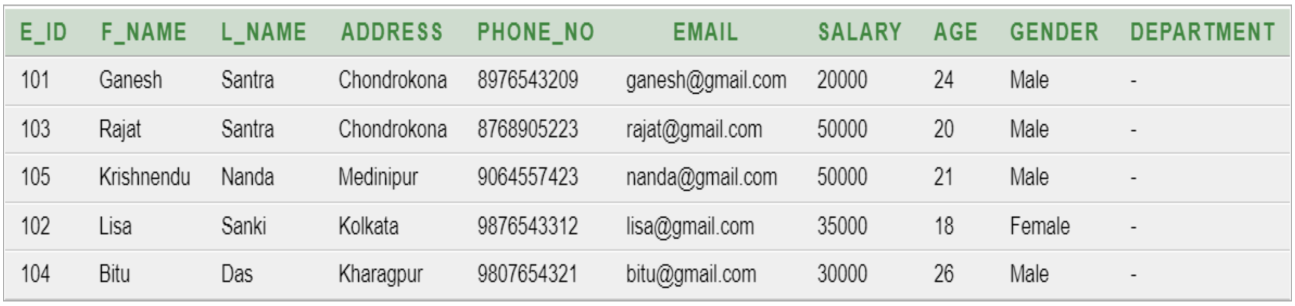
```
select * from employee2;
```



E_ID	F_NAME	L_NAME	ADDRESS	PHONE_NO	EMAIL	SALARY	AGE	GENDER
101	Ganesh	Santra	Chondrokona	8976543209	ganesh@gmail.com	20000	24	Male
103	Rajat	Santra	Chondrokona	8768905223	rajat@gmail.com	50000	20	Male
105	Krishnendu	Nanda	Medinipur	9064557423	nanda@gmail.com	50000	21	Male
102	Lisa	Sanki	Kolkata	9876543312	lisa@gmail.com	35000	18	Female
104	Bitu	Das	Kharagpur	9807654321	bitu@gmail.com	30000	26	Male

iv) Add a column Department.

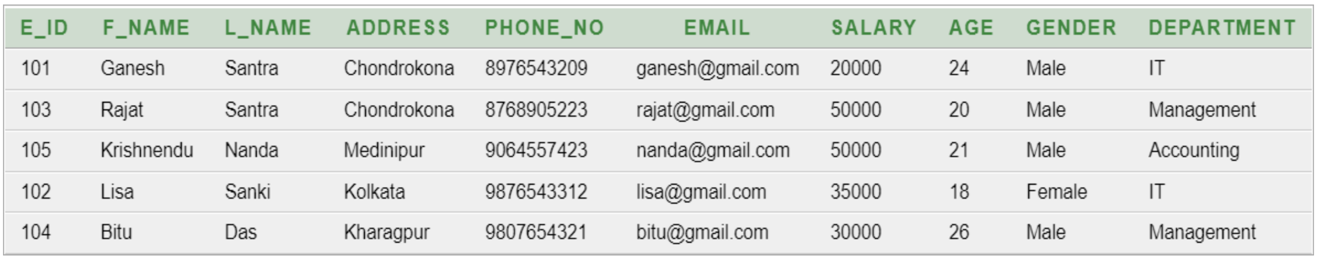
```
alter table employee2 add department varchar(15);
```



E_ID	F_NAME	L_NAME	ADDRESS	PHONE_NO	EMAIL	SALARY	AGE	GENDER	DEPARTMENT
101	Ganesh	Santra	Chondrokona	8976543209	ganesh@gmail.com	20000	24	Male	-
103	Rajat	Santra	Chondrokona	8768905223	rajat@gmail.com	50000	20	Male	-
105	Krishnendu	Nanda	Medinipur	9064557423	nanda@gmail.com	50000	21	Male	-
102	Lisa	Sanki	Kolkata	9876543312	lisa@gmail.com	35000	18	Female	-
104	Bitu	Das	Kharagpur	9807654321	bitu@gmail.com	30000	26	Male	-

v) Update Department details.

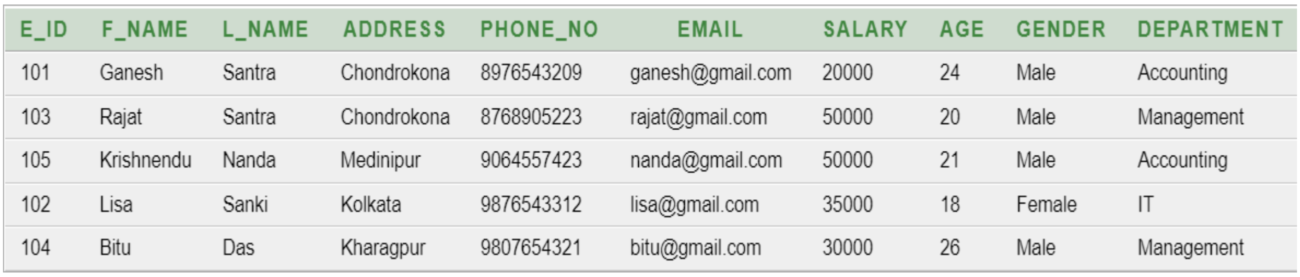
```
update employee2 set department='Management' where e_id=104; update
employee2 set department='Management' where e_id=103; update employee2
set department='Accounting' where e_id=105; update employee2 set
department='IT' where e_id=102;
update employee2 set department='IT' where e_id=101;
```



E_ID	F_NAME	L_NAME	ADDRESS	PHONE_NO	EMAIL	SALARY	AGE	GENDER	DEPARTMENT
101	Ganesh	Santra	Chondrokona	8976543209	ganesh@gmail.com	20000	24	Male	IT
103	Rajat	Santra	Chondrokona	8768905223	rajat@gmail.com	50000	20	Male	Management
105	Krishnendu	Nanda	Medinipur	9064557423	nanda@gmail.com	50000	21	Male	Accounting
102	Lisa	Sanki	Kolkata	9876543312	lisa@gmail.com	35000	18	Female	IT
104	Bitu	Das	Kharagpur	9807654321	bitu@gmail.com	30000	26	Male	Management

vi) Modify employee details, department = ‘Accounting’ whose id = 101.

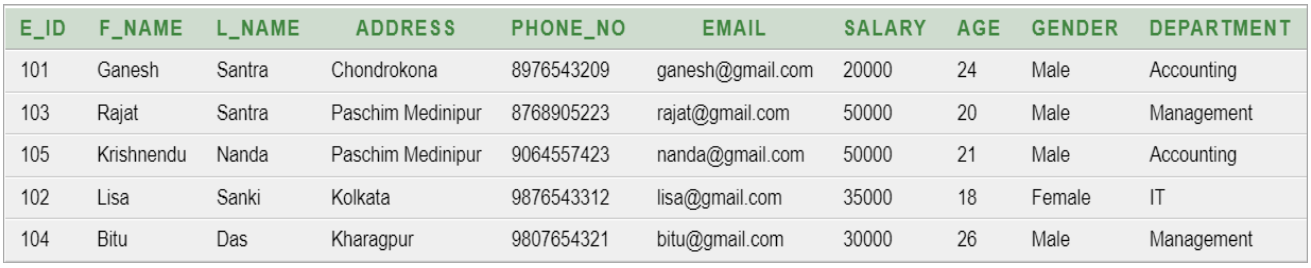
update employee2 set department='Accounting' where e_id=101;



E_ID	F_NAME	L_NAME	ADDRESS	PHONE_NO	EMAIL	SALARY	AGE	GENDER	DEPARTMENT
101	Ganesh	Santra	Chondrokona	8976543209	ganesh@gmail.com	20000	24	Male	Accounting
103	Rajat	Santra	Chondrokona	8768905223	rajat@gmail.com	50000	20	Male	Management
105	Krishnendu	Nanda	Medinipur	9064557423	nanda@gmail.com	50000	21	Male	Accounting
102	Lisa	Sanki	Kolkata	9876543312	lisa@gmail.com	35000	18	Female	IT
104	Bitu	Das	Kharagpur	9807654321	bitu@gmail.com	30000	26	Male	Management

vii) Modify employee address whose id = 103 and 105.

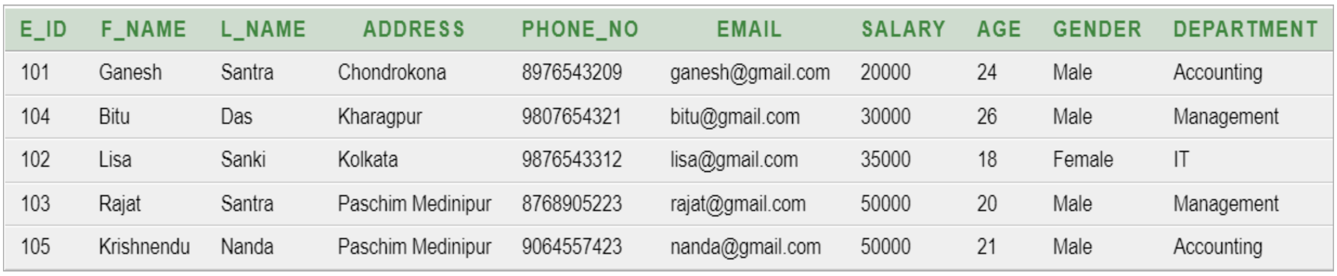
update employee2 set address='Paschim Medinipur' where e_id=103 and e_id=105;



E_ID	F_NAME	L_NAME	ADDRESS	PHONE_NO	EMAIL	SALARY	AGE	GENDER	DEPARTMENT
101	Ganesh	Santra	Chondrokona	8976543209	ganesh@gmail.com	20000	24	Male	Accounting
103	Rajat	Santra	Paschim Medinipur	8768905223	rajat@gmail.com	50000	20	Male	Management
105	Krishnendu	Nanda	Paschim Medinipur	9064557423	nanda@gmail.com	50000	21	Male	Accounting
102	Lisa	Sanki	Kolkata	9876543312	lisa@gmail.com	35000	18	Female	IT
104	Bitu	Das	Kharagpur	9807654321	bitu@gmail.com	30000	26	Male	Management

viii) Find the employee details by their salary by ascending order.

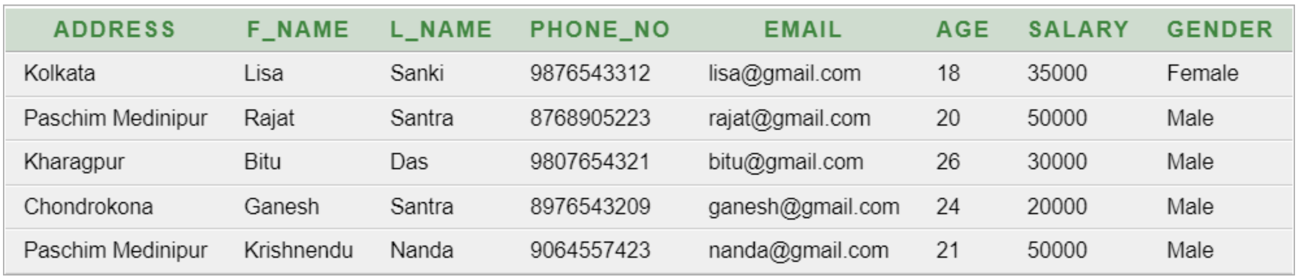
select * from employee2 order by salary asc;



E_ID	F_NAME	L_NAME	ADDRESS	PHONE_NO	EMAIL	SALARY	AGE	GENDER	DEPARTMENT
101	Ganesh	Santra	Chondrokona	8976543209	ganesh@gmail.com	20000	24	Male	Accounting
104	Bitu	Das	Kharagpur	9807654321	bitu@gmail.com	30000	26	Male	Management
102	Lisa	Sanki	Kolkata	9876543312	lisa@gmail.com	35000	18	Female	IT
103	Rajat	Santra	Paschim Medinipur	8768905223	rajat@gmail.com	50000	20	Male	Management
105	Krishnendu	Nanda	Paschim Medinipur	9064557423	nanda@gmail.com	50000	21	Male	Accounting

ix) Find the employees details by their address.

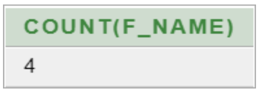
select distinct(address),f_name,l_name,phone_no,email,age,salary,gender from employee2;



ADDRESS	F_NAME	L_NAME	PHONE_NO	EMAIL	AGE	SALARY	GENDER
Kolkata	Lisa	Sanki	9876543312	lisa@gmail.com	18	35000	Female
Paschim Medinipur	Rajat	Santra	8768905223	rajat@gmail.com	20	50000	Male
Kharagpur	Bitu	Das	9807654321	bitu@gmail.com	26	30000	Male
Chondrokona	Ganesh	Santra	8976543209	ganesh@gmail.com	24	20000	Male
Paschim Medinipur	Krishnendu	Nanda	9064557423	nanda@gmail.com	21	50000	Male

x) Count the employee’s name whose salary > 20,000.

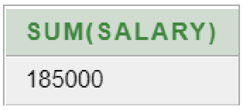
```
select count(f_name) from employee2 where salary>20000;
```



COUNT(F_NAME)
4

xi) Calculate the total salary of all employees.

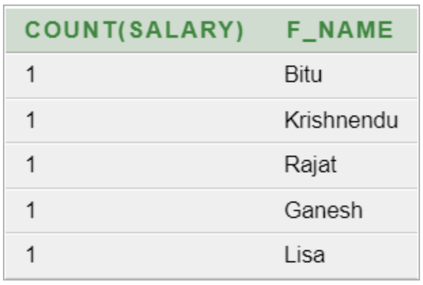
```
select sum(salary) from employee2;
```



SUM(SALARY)
185000

xii) Count salary, name of employees group by their name.

```
select count(salary),f_name from employee2 group by f_name;
```



COUNT(SALARY)	F_NAME
1	Bitu
1	Krishnendu
1	Rajat
1	Ganesh
1	Lisa

4. **Joining two tables using cross join, inner join, left join, right join**

Create two tables name movie(movie_id,title,category) and another is member(member_id,name, movie_id)

----- Movie Table -----

Create:

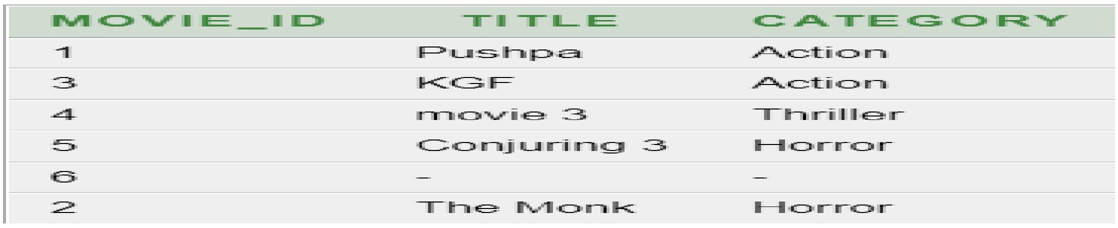
```
create table movie(movie_id numeric(2),title varchar(15),category varchar(15));
```

Insert:

```
insert into movie values(1,'Pushpa','Action'); insert into movie values(2,'The
Monk','Horror'); insert into movie values(3,'KGF','Action'); insert into movie
values(4,'movie 3','Thriller');
insert into movie values(5,'Conjuring 3','Horror'); insert into movie
values(6,",");
```

```
select * from movie;
```

View:



MOVIE_ID	TITLE	CATEGORY
1	Pushpa	Action
3	KGF	Action
4	movie 3	Thriller
5	Conjuring 3	Horror
6	-	-
2	The Monk	Horror

-----Member Table -----

Create:

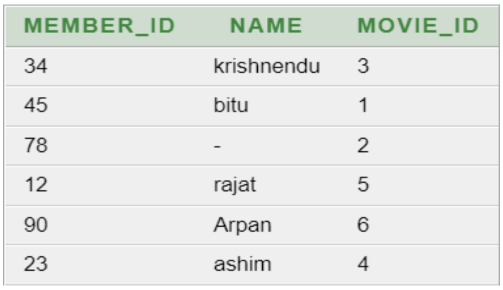
create table member(member_id numeric(2),name varchar(15),movie_id numeric(2));

Insert:

insert into member values(34,'krishnendu',3); insert into member values(78,'',2); insert into member values(45,'bitu',1); insert into member values(23,'ashim',4); insert into member values(12,'rajat',5); insert into member values(90,'Arpan',6);

View:

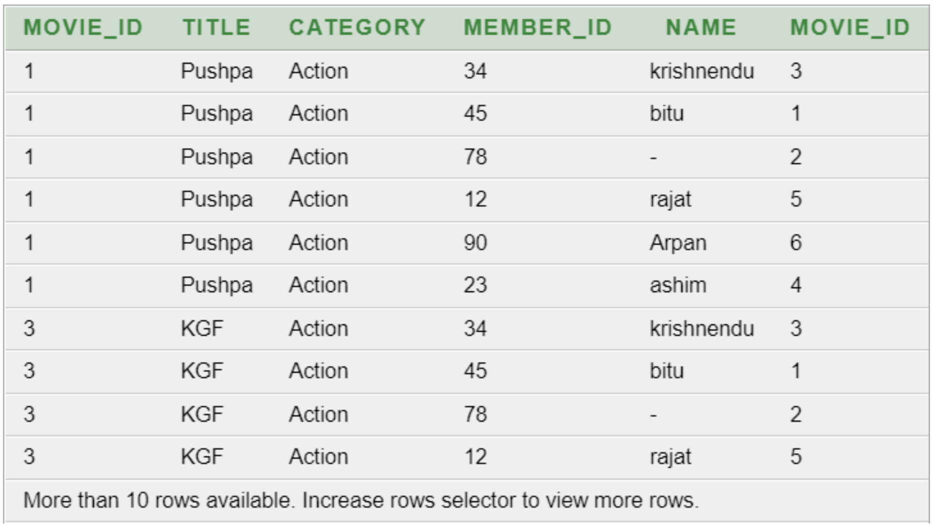
select * from member;



MEMBER_ID	NAME	MOVIE_ID
34	krishnendu	3
45	bitu	1
78	-	2
12	rajat	5
90	Arpan	6
23	ashim	4

------ Cross Join -----

select * from movie cross join member;

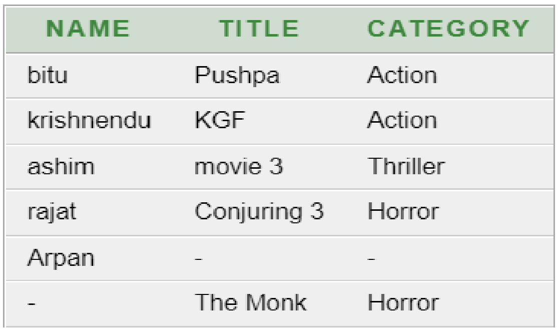


MOVIE_ID	TITLE	CATEGORY	MEMBER_ID	NAME	MOVIE_ID
1	Pushpa	Action	34	krishnendu	3
1	Pushpa	Action	45	bitu	1
1	Pushpa	Action	78	-	2
1	Pushpa	Action	12	rajat	5
1	Pushpa	Action	90	Arpan	6
1	Pushpa	Action	23	ashim	4
3	KGF	Action	34	krishnendu	3
3	KGF	Action	45	bitu	1
3	KGF	Action	78	-	2
3	KGF	Action	12	rajat	5

More than 10 rows available. Increase rows selector to view more rows.

------ Inner Join -----

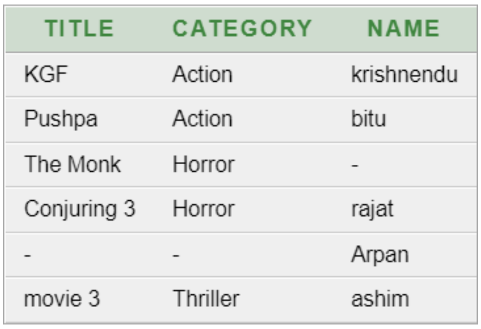
select member.name,movie.title,movie.category from member,movie where member.movie_id=movie.movie_id;



NAME	TITLE	CATEGORY
bitu	Pushpa	Action
krishnendu	KGF	Action
ashim	movie 3	Thriller
rajat	Conjuring 3	Horror
Arpan	-	-
-	The Monk	Horror

------ Left Join -----

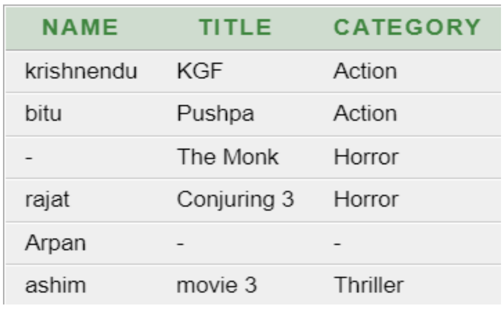
select movie.title,movie.category,member.name from movie left join member on movie.movie_id=member.movie_id;



TITLE	CATEGORY	NAME
KGF	Action	krishnendu
Pushpa	Action	bitu
The Monk	Horror	-
Conjuring 3	Horror	rajat
-	-	Arpan
movie 3	Thriller	ashim

------ Right Join -----

select member.name,movie.title,movie.category from member right join movie on member.movie_id = movie.movie_id;



NAME	TITLE	CATEGORY
krishnendu	KGF	Action
bitu	Pushpa	Action
-	The Monk	Horror
rajat	Conjuring 3	Horror
Arpan	-	-
ashim	movie 3	Thriller

5. **Creating three tables sailors, boat and reserve joining these tables with various query**

   i) **Create three tables name sailor(s_id,s_name,s_rating,s_age), boat(b_id,b_name,color,s_id) and reserve(s_id,b_id,hire_date)**

   Inserting 5 rows into sailor, boat and reserve tables View these tables

---- SAILOR ----

Create:

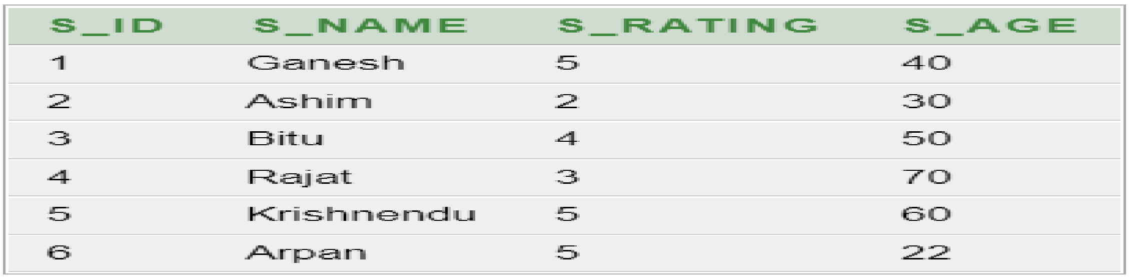
create table sailor(s_id numeric(2),s_name varchar(25),s_rating numeric(10),s_age numeric(3));

Insert:

insert into sailor values(01,'Ganesh',5,40); insert into sailor values(02,'Ashim',2,30); insert into sailor values(03,'Bitu',4,50); insert into sailor values(04,'Rajat',3,70);
insert into sailor values(05,'Krishnendu',5,60); insert into sailor values(06,'Arpan',5,22);

```
select * from sailor;
```

View:



S_ID	S_NAME	S_RATING	S_AGE
1	Ganesh	5	40
2	Ashim	2	30
3	Bitu	4	50
4	Rajat	3	70
5	Krishnendu	5	60
6	Arpan	5	22

---- BOAT ----

Create:

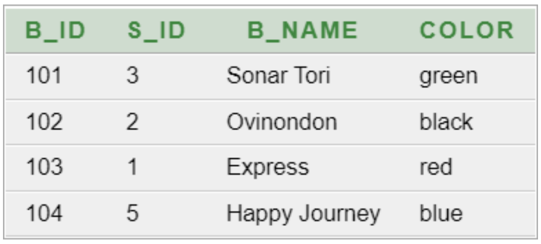
```
create table boat(b_id numeric(5),s_id numeric(2),b_name
varchar(20),color varchar(10));
```

Insert:

```
insert into boat values(101,03,'Sonar Tori','green'); insert into boat
values(102,02,'Ovinondon','black'); insert into boat
values(103,01,'Express','red');
insert into boat values(104,05,'Happy Journey','blue');
```

View:

```
select * from boat;
```



B_ID	S_ID	B_NAME	COLOR
101	3	Sonar Tori	green
102	2	Ovinondon	black
103	1	Express	red
104	5	Happy Journey	blue

--- RESERVE ---

Create:

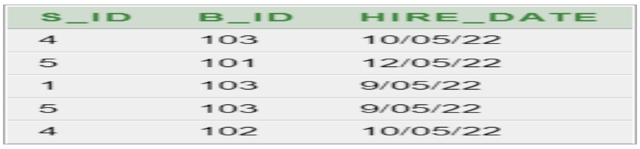
```
create table reserve(s_id numeric(2),b_id numeric(5),hire_date varchar(20));
```

Insert:

```
insert into reserve values(04,103,'10/05/22'); insert into reserve
values(05,101,'12/05/22'); insert into reserve values(01,103,'9/05/22'); insert
into reserve values(05,103,'9/05/22'); insert into reserve
values(04,102,'10/05/22');
```

select * from reserve;* from boat;

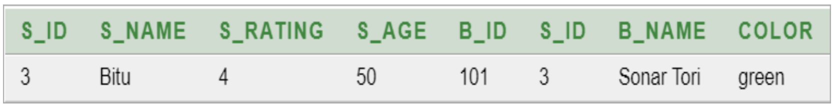
view:



S_ID	B_ID	HIRE_DATE
4	103	10/05/22
5	101	12/05/22
1	103	9/05/22
5	103	9/05/22
4	102	10/05/22

i) Find all information of sailor who have reserve boat number 101.

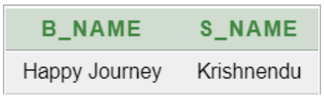
select * from sailor join boat on sailor.s_id=boat.s_id where b_id = 101;



S_ID	S_NAME	S_RATING	S_AGE	B_ID	S_ID	B_NAME	COLOR
3	Bitu	4	50	101	3	Sonar Tori	green

ii) Find the name of boat reserve by sailor name.

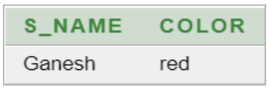
select boat.b_name,sailor.s_name from boat join sailor on boat.s_id=sailor.s_id where s_name='Krishnendu';



B_NAME	S_NAME
Happy Journey	Krishnendu

iii) Find the name of the sailor who have reserve a red boat and list in the order of age.

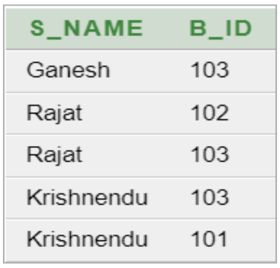
select sailor.s_name,boat.color from sailor join boat on sailor.s_id=boat.s_id where boat.color='red' order by sailor.s_age asc;



S_NAME	COLOR
Ganesh	red

iv) Find the name of sailor who have reserved at least one boat.

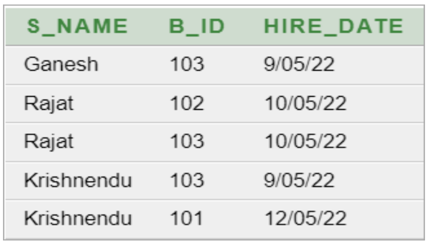
select sailor.s_name,reserve.b_id from sailor join reserve on sailor.s_id=reserve.s_id;



S_NAME	B_ID
Ganesh	103
Rajat	102
Rajat	103
Krishnendu	103
Krishnendu	101

v) **Find the name of sailor who have reserved difference boat on the same date.**

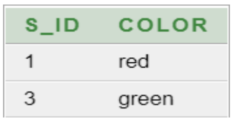
```
select sailor.s_name,reserve.b_id,reserve.hire_date from sailor
join reserve on sailor.s_id=reserve.s_id;
```



S_NAME	B_ID	HIRE_DATE
Ganesh	103	9/05/22
Rajat	102	10/05/22
Rajat	103	10/05/22
Krishnendu	103	9/05/22
Krishnendu	101	12/05/22

vi) **Find the id of sailor who have reserved a red boat or green boat.**

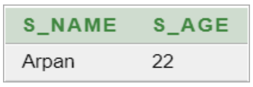
```
select sailor.s_id,boat.color from sailor join boat on
sailor.s_id=boat.s_id where boat.color='red' or boat.color='green';
```



S_ID	COLOR
1	red
3	green

vii)**Find the name and age of youngest sailor.**

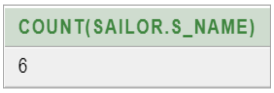
```
select sailor.s_name,sailor.s_age from sailor where
sailor.s_age>=18 and sailor.s_age<=25;
```



S_NAME	S_AGE
Arpan	22

viii) **Count the number of different sailor name.**

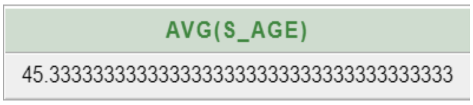
```
select count(sailor.s_name) from sailor;
```



COUNT(SAILOR.S_NAME)
6

ix) **Find the average age of sailor for each rating levels.**

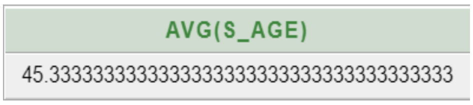
```
select avg(s_age) from sailor order by s_rating;
```



AVG(S_AGE)
45.333333333333333333333333333333333333

x) **Find the average age of sailor for each rating levels that at least two sailor.**

select avg(s_age) from sailor order by s_rating;



AVG(S_AGE)
45.33333333333333333333333333333333333333

6. **Creating four tables movie, actor, acts and director joining these tables with various query.**

i) **Create four tables name movie (movie_id,title,dir_name,release_year,rating), actor(act_id,act_name, movie_id), acts(act_name,movie_name) and director(dir_name,movie_id,year)**

Inserting 5 rows into movie,actor,acts and director tables View these tables

--- MOVIE ---

Create:

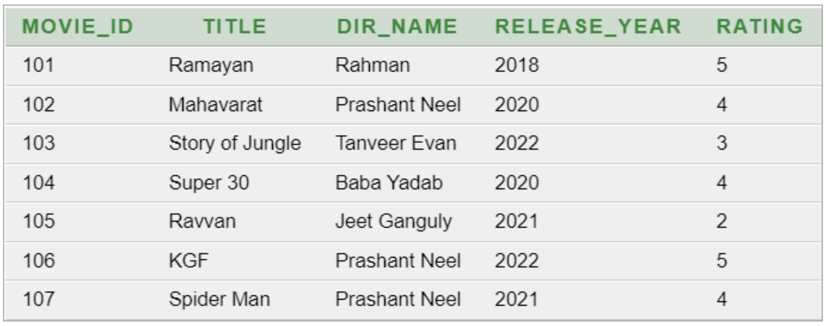
create table movie1(movie_id numeric(5),title varchar(20),dir_name varchar(25),release_year numeric(6),rating numeric(3));

Insert:

insert into movie1 values(101,'Ramayan','Rahman',2018,5);
insert into movie1 values(102,'Mahavarat','Prashant Neel',2020,4); insert into movie1 values(103,'Story of Jungle','Tanveer Evan',2022,3); insert into movie1 values(104,'Super 30','Baba Yadab',2020,4);
insert into movie1 values(105,'Ravvan','Jeet Ganguly',2021,2); insert into movie1 values(106,'KGF','Prashant Neel',2022,5);
insert into movie1 values(107,'Spider Man','Prashant Neel',2021,4);

View:

select * from movie1;



MOVIE_ID	TITLE	DIR_NAME	RELEASE_YEAR	RATING
101	Ramayan	Rahman	2018	5
102	Mahavarat	Prashant Neel	2020	4
103	Story of Jungle	Tanveer Evan	2022	3
104	Super 30	Baba Yadab	2020	4
105	Ravvan	Jeet Ganguly	2021	2
106	KGF	Prashant Neel	2022	5
107	Spider Man	Prashant Neel	2021	4

--- ACTOR ---

Create:

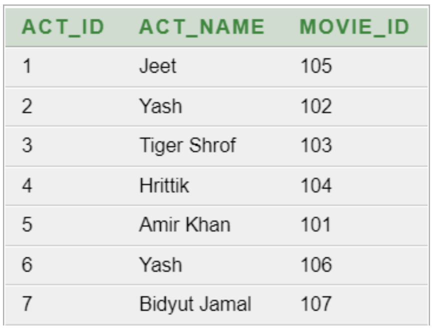
```
create table actor(act_id numeric(5),act_name varchar(25),movie_id
numeric(5));
```

Insert:

```
insert into actor values(001,'Jeet',105); insert into actor values(002,'Yash',102);
insert into actor values(003,'Tiger Shrof',103); insert into actor
values(004,'Hrittik',104); insert into actor values(005,'Amir Khan',101); insert
into actor values(006,'Yash',106);
insert into actor values(007,'Bidyut Jamal',107);
```

View:

```
select * from actor;
```



ACT_ID	ACT_NAME	MOVIE_ID
1	Jeet	105
2	Yash	102
3	Tiger Shrof	103
4	Hrittik	104
5	Amir Khan	101
6	Yash	106
7	Bidyut Jamal	107

--- ACTS ---

Create:

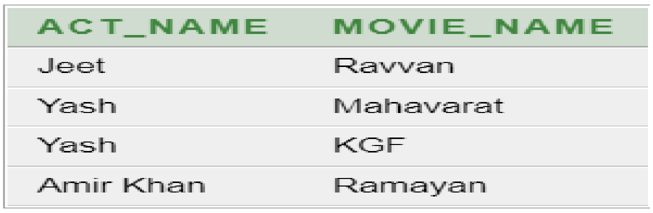
```
create table acts(act_name varchar(25),movie_name varchar(20));
```

View:

Insert:

```
insert into acts values('Amir Khan','Ramayan'); insert into acts
values('Yash','KGF');
insert into acts values('Yash','Mahavarat'); insert into acts
values('Jeet','Ravvan');

select * from acts;
```



ACT_NAME	MOVIE_NAME
Jeet	Ravvan
Yash	Mahavarat
Yash	KGF
Amir Khan	Ramayan

--- DIRECTOR ---

Create:

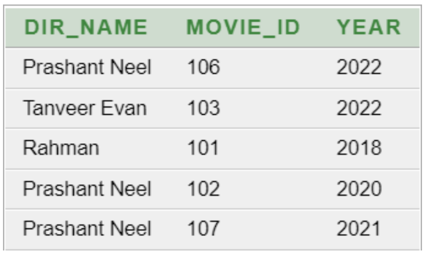
```
create table director1(dir_name varchar(25),movie_id numeric(5),year
numeric(6));
```

Insert:

```
insert into director1 values('Prashant Neel',106,2022); insert into director1
values('Tanveer Evan',103,2022); insert into director1
values('Rahman',101,2018); insert into director1 values('Prashant
Neel',102,2020); insert into director1 values('Prashant Neel',107,2021);
```

View:

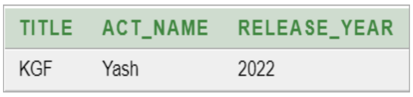
```
select * from director1;
```



DIR_NAME	MOVIE_ID	YEAR
Prashant Neel	106	2022
Tanveer Evan	103	2022
Rahman	101	2018
Prashant Neel	102	2020
Prashant Neel	107	2021

ii) Find the movie name made after 2020 acts by ‘Yash’.

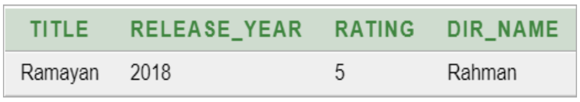
```
select movie1.title,actor.act_name,movie1.release_year from movie1 join actor
on movie1.movie_id=actor.movie_id where movie1.release_year>2020 and
actor.act_name='Yash';
```



TITLE	ACT_NAME	RELEASE_YEAR
KGF	Yash	2022

iii) Find the movies made by ‘Prasanth Neel’ in the year 2018 which is rating above 4 stars.

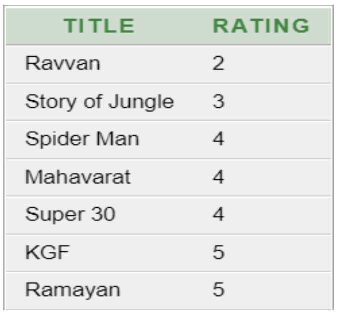
```
select movie1.title,movie1.release_year,movie1.rating,director1.dir_name from
movie1 join director1 on movie1.movie_id=director1.movie_id where
movie1.release_year=2018 and movie1.rating>4 and
director1.dir_name='Rahman';
```



TITLE	RELEASE_YEAR	RATING	DIR_NAME
Ramayan	2018	5	Rahman

iv) **Find all movies with their ratings in ascending order.**

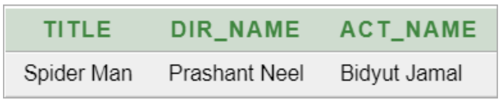
select movie1.title, movie1.rating from movie1 order by movie1.rating asc;



TITLE	RATING
Ravvan	2
Story of Jungle	3
Spider Man	4
Mahavarat	4
Super 30	4
KGF	5
Ramayan	5

v) **Find movies name which is directed by ‘Prashant Neel’ and doesn’t acts by ‘Yash’.**

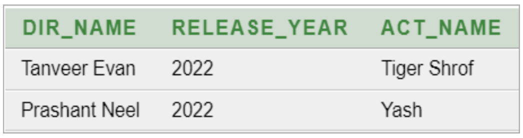
select movie1.title,movie1.dir_name,actor.act_name from movie1 join actor on movie1.movie_id=actor.movie_id where movie1.dir_name='Prashant Neel' and actor.act_name!='Yash';



TITLE	DIR_NAME	ACT_NAME
Spider Man	Prashant Neel	Bidyut Jamal

vi) **Find all actor and director name who are combine the same movie in the year 2022.**

select movie1.dir_name, movie1.release_year, actor.act_name from movie1 join actor on movie1.movie_id= actor.movie_id where movie1.release_year=2022;

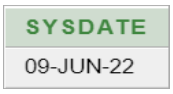


DIR_NAME	RELEASE_YEAR	ACT_NAME
Tanveer Evan	2022	Tiger Shrof
Prashant Neel	2022	Yash

7. SQL queries of find the System date and time.

i) How to find system date from database.

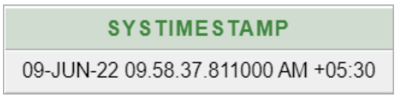
select SYSDATE from dual;



SYSDATE
09-JUN-22

ii) How to find system time from database.

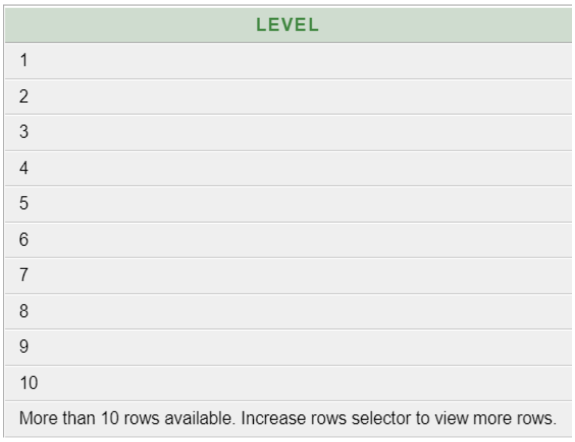
select SYSTIMESTAMP from dual;



SYSTIMESTAMP
09-JUN-22 09.58.37.811000 AM +05:30

iii) How to display 1 to 100 number with using query.

select level from dual connect by level<=100



LEVEL
1
2
3
4
5
6
7
8
9
10
More than 10 rows available. Increase rows selector to view more rows.

8. **Creating six tables book, author, publisher, book-copies, book-lending and library-branch and joining these tables with various queries.**

i) Create four tables name – book(b_id,title,pub_name,pub_year),
author(a_id,b_id,a_name),
publisher(pub_id,pub_name,address,phone_no),
bookcopies(b_id,branch_id,number_of_copies),
booklending(b_id,branch_id,card_no,purchase_date),
librarybranch(branch_id,branch_name,address)
Inserting 5 rows into these tables View these tables

--- BOOK ---

Create:

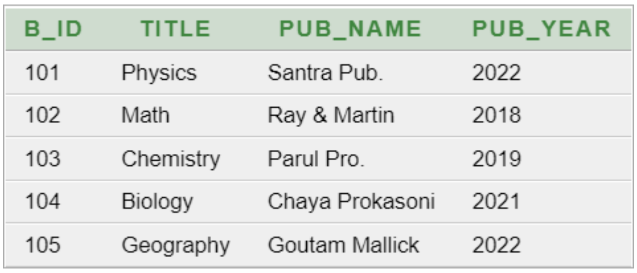
```
create table book(b_id numeric(3),title varchar(20),pub_name
varchar(15),pub_year numeric(4));
```

Insert:

```
insert into book values(101,'Physics','Santra Pub.',2022); insert into book
values(102,'Math','Ray & Martin',2018); insert into book
values(103,'Chemistry','Parul Pro.',2019); insert into book
values(104,'Biology','Chaya Prokasoni',2021);
insert into book values(105,'Geography','Goutam Mallick',2022);
```

View:

```
select * from book;
```



B_ID	TITLE	PUB_NAME	PUB_YEAR
101	Physics	Santra Pub.	2022
102	Math	Ray & Martin	2018
103	Chemistry	Parul Pro.	2019
104	Biology	Chaya Prokasoni	2021
105	Geography	Goutam Mallick	2022

--- AUTHOR ---

Create:

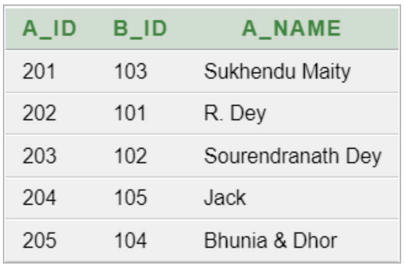
```
create table author(a_id numeric(3),b_id numeric(3),a_name varchar(25));
```

Insert:

```
insert into author values(201,103,'Sukhendu Maity'); insert into author
values(202,101,'R. Dey');
insert into author values(203,102,'Sourendranath Dey'); insert into author
values(204,105,'Jack');
insert into author values(205,104,'Bhunia & Dhor');
```

View:

```
select * from author;
```



A_ID	B_ID	A_NAME
201	103	Sukhendu Maity
202	101	R. Dey
203	102	Sourendranath Dey
204	105	Jack
205	104	Bhunia & Dhor

--- PUBLISHER ---

Create:

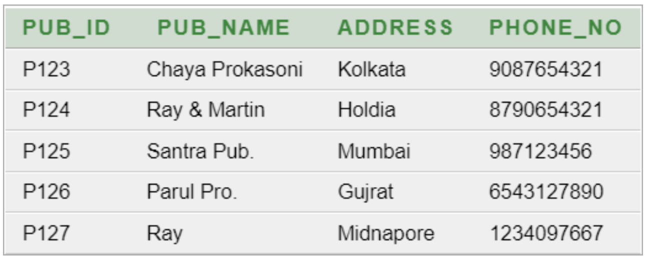
```
create table publisher(pub_id varchar(5),pub_name varchar(20),address
varchar(25),phone_no numeric(10));
```

Insert:

```
insert into publisher values('P123','Chaya Prokasoni','Kolkata',9087654321);
insert into publisher values('P124','Ray & Martin','Holdia',8790654321); insert
into publisher values('P125','Santra Pub.','Mumbai',0987123456); insert into
publisher values('P126','Parul Pro.','Gujrat',6543127890);
insert into publisher values('P127','Ray','Midnapore',1234097667);
```

View:

```
select * from publisher;
```



PUB_ID	PUB_NAME	ADDRESS	PHONE_NO
P123	Chaya Prokasoni	Kolkata	9087654321
P124	Ray & Martin	Holdia	8790654321
P125	Santra Pub.	Mumbai	987123456
P126	Parul Pro.	Gujrat	6543127890
P127	Ray	Midnapore	1234097667

--- BOOKCOPIES ---

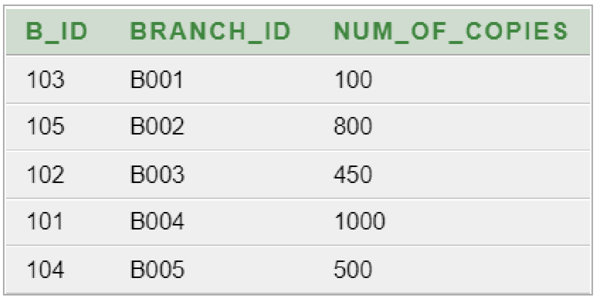
Create:

```
create table bookcopies(b_id numeric(3),branch_id varchar(4),num_of_copies
numeric(5));
```

Insert:

```
insert into bookcopies values(103,'B001',100); insert into bookcopies
values(105,'B002',800); insert into bookcopies values(102,'B003',450); insert
into bookcopies values(101,'B004',1000); insert into bookcopies
values(104,'B005',500);
```

View:
select * from bookcopies;

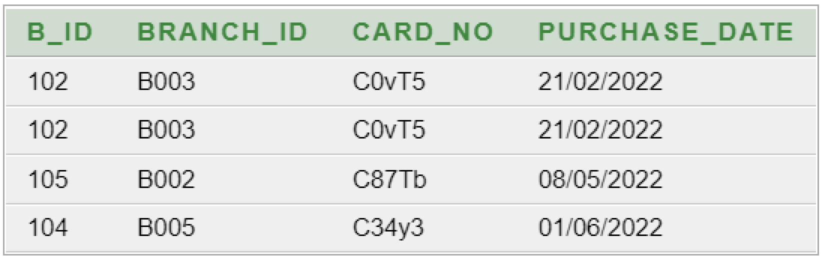


B_ID	BRANCH_ID	NUM_OF_COPIES
103	B001	100
105	B002	800
102	B003	450
101	B004	1000
104	B005	500

--- BOOKLENDING ---

Create:
create table booklending(b_id numeric(3),branch_id varchar(4),card_no varchar(5),purchase_date varchar(12));

Insert:
insert into booklending values(102,'B003','C0vT5','21/02/2022'); insert into booklending values(105,'B002','C87Tb','08/05/2022'); insert into booklending values(104,'B005','C34y3','01/06/2022'); **View:**
select * from booklending;



B_ID	BRANCH_ID	CARD_NO	PURCHASE_DATE
102	B003	C0vT5	21/02/2022
102	B003	C0vT5	21/02/2022
105	B002	C87Tb	08/05/2022
104	B005	C34y3	01/06/2022

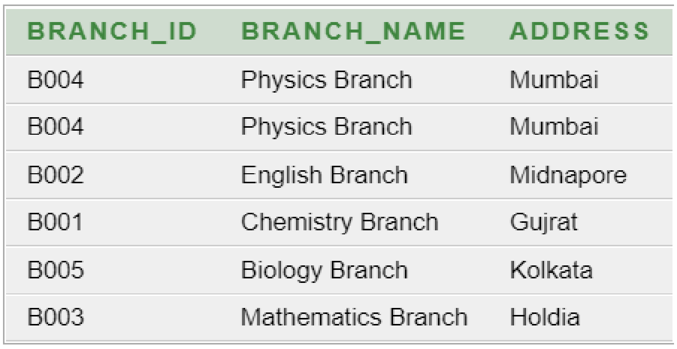
--- LIBRARY BRANCH ---

Create:
create table librarybranch(branch_id varchar(4),branch_name varchar(25),address varchar(25));

Insert:
insert into librarybranch values('B004','Physics Branch','Mumbai'); insert into librarybranch values('B002','English Branch','Midnapore'); insert into librarybranch values('B001','Chemistry Branch','Gujrat'); insert into librarybranch values('B005','Biology Branch','Kolkata'); insert into librarybranch values('B003','Mathematics Branch','Holdia');

View:

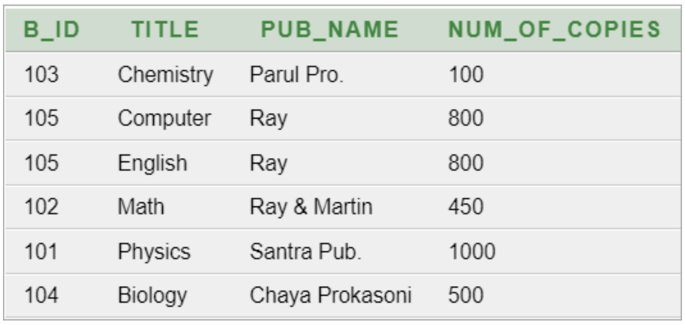
select * from librarybranch;



BRANCH_ID	BRANCH_NAME	ADDRESS
B004	Physics Branch	Mumbai
B004	Physics Branch	Mumbai
B002	English Branch	Midnapore
B001	Chemistry Branch	Gujrat
B005	Biology Branch	Kolkata
B003	Mathematics Branch	Holdia

ii) Find the details of all books in the library with their id, title, name of publisher, author and number of copies in each branch.

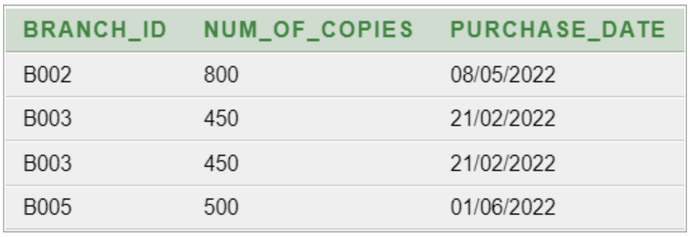
select book.b_id,book.title,book.pub_name,bookcopies.num_of_copies from book join bookcopies on book.b_id= bookcopies.b_id;



B_ID	TITLE	PUB_NAME	NUM_OF_COPIES
103	Chemistry	Parul Pro.	100
105	Computer	Ray	800
105	English	Ray	800
102	Math	Ray & Martin	450
101	Physics	Santra Pub.	1000
104	Biology	Chaya Prokasoni	500

iii) Get the particular borrower who have borrow more than 3 books from 27th january to 27th june.

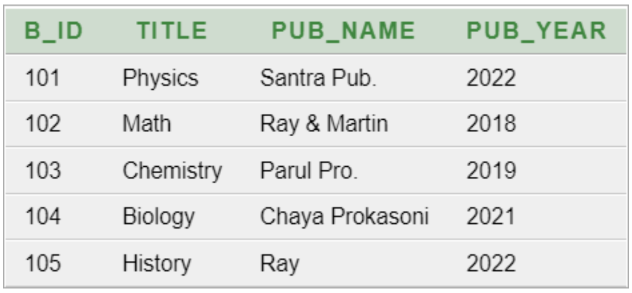
select book.b_id,book.title,book.pub_name,bookcopies.num_of_copies from book join bookcopies on book.b_id= bookcopies.b_id;



BRANCH_ID	NUM_OF_COPIES	PURCHASE_DATE
B002	800	08/05/2022
B003	450	21/02/2022
B003	450	21/02/2022
B005	500	01/06/2022

iv) Delete a book name from the book table and update this

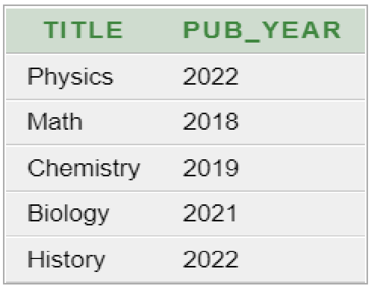
delete name from book where title=Geography; update book set title= 'History' where b_id=105;



B_ID	TITLE	PUB_NAME	PUB_YEAR
101	Physics	Santra Pub.	2022
102	Math	Ray & Martin	2018
103	Chemistry	Parul Pro.	2019
104	Biology	Chaya Prokasoni	2021
105	History	Ray	2022

v)Partition the book table based on year of publication

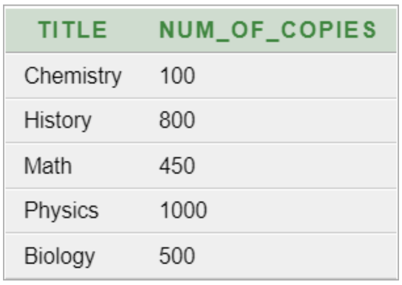
```
select book.title,pub_year from book;
```



TITLE	PUB_YEAR
Physics	2022
Math	2018
Chemistry	2019
Biology	2021
History	2022

vi) Create a view of all books and its number of copies that are currently available in the library

```
select book.title,bookcopies.num_of_copies from book join bookcopies on
book.b_id=bookcopies.b_id;
```



TITLE	NUM_OF_COPIES
Chemistry	100
History	800
Math	450
Physics	1000
Biology	500

9. **Creating three tables Salesman, Customers and orders and joining these tables with various queries.**

i) Create four tables name salesman(s_id,s_same,s_city,commission),customers(c_id,c_name,c_city, grade,s_id), orders(order_num,purchase_amount,order_date,c_id,s_id)

Inserting 5 rows into these tables and view these tables

--- SALESMAN ---

Create:

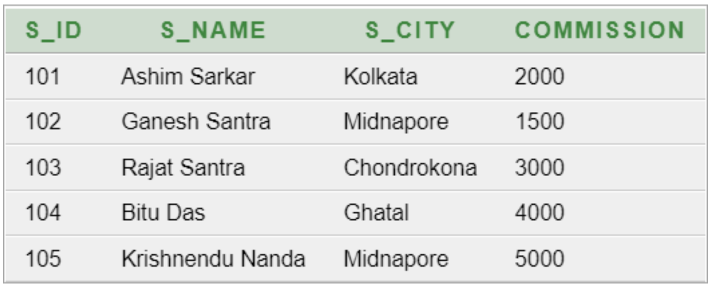
```
create table salesmans(s_id numeric(3),s_name varchar(25),s_city
varchar(30),commission numeric(5));
```

Insert:

```
insert into salesmans values(101,'Ashim Sarkar','Kolkata',2000); insert into
salesmans values(102,'Ganesh Santra','Midnapore',1500); insert into salesmans
values(103,'Rajat Santra','Chondrokona',3000); insert into salesmans
values(104,'Bitu Das','Ghatal',4000);
insert into salesmans values(105,'Krishnendu Nanda','Midnapore',5000);
```

View:

```
select * from salesmans;
```



S_ID	S_NAME	S_CITY	COMMISSION
101	Ashim Sarkar	Kolkata	2000
102	Ganesh Santra	Midnapore	1500
103	Rajat Santra	Chondrokona	3000
104	Bitu Das	Ghatal	4000
105	Krishnendu Nanda	Midnapore	5000

--- CUSTOMERS ---

Create:

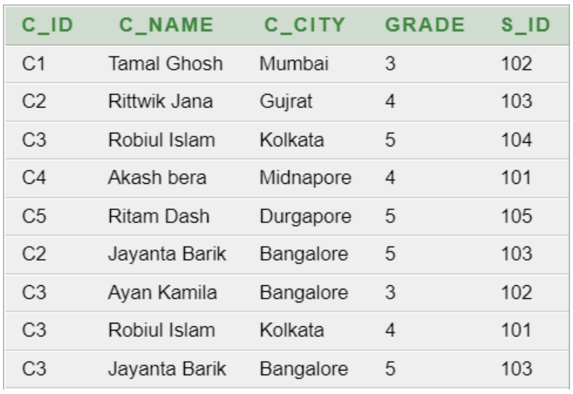
```
create table customers(c_id varchar(3),c_name varchar(20),c_city
varchar(25),grade numeric(2),s_id numeric(3));
```

Insert:

```
insert into customers values('C1','Tamal Ghosh','Mumbai',3,102); insert into
customers values('C2','Rittwik Jana','Gujrat',4,103); insert into customers
values('C3','Robiul Islam','Kolkata',5,104); insert into customers
values('C4','Akash bera','Midnapore',4,101); insert into customers
values('C5','Ritam Dash','Durgapore',5,105); insert into customers
values('C2','Jayanta Barik','Bangalore',5,103); insert into customers
values('C3','Ayan Kamila','Bangalore',3,102); insert into customers
values('C3','Robiul Islam','Kolkata',4,101);
```

View:

select * from customers;



C_ID	C_NAME	C_CITY	GRADE	S_ID
C1	Tamal Ghosh	Mumbai	3	102
C2	Rittwik Jana	Gujrat	4	103
C3	Robiul Islam	Kolkata	5	104
C4	Akash bera	Midnapore	4	101
C5	Ritam Dash	Durgapore	5	105
C2	Jayanta Barik	Bangalore	5	103
C3	Ayan Kamila	Bangalore	3	102
C3	Robiul Islam	Kolkata	4	101
C3	Jayanta Barik	Bangalore	5	103

--- ORDERS ---

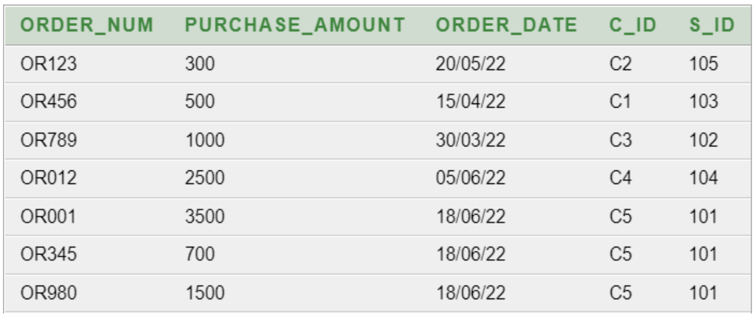
Create:

create table orders(order_num varchar(5),purchase_amount numeric(5),order_date varchar(10),c_id varchar(3),s_id numeric(3));

Insert:

insert into orders values('OR123',300,'20/05/22','C2',105); insert into orders values('OR456',500,'15/04/22','C1',103); insert into orders values('OR789',1000,'30/03/22','C3',102); insert into orders values('OR012',2500,'05/06/22','C4',104); insert into orders values('OR001',3500,'18/06/22','C5',101); insert into orders values('OR345',700,'18/06/22','C5',101); insert into orders values('OR980',1500,'18/06/22','C5',101); **View:**

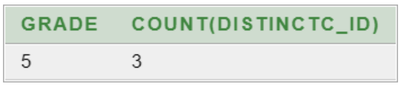
select * from orders;



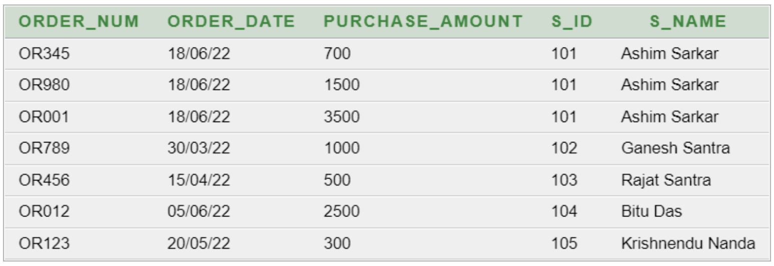
ORDER_NUM	PURCHASE_AMOUNT	ORDER_DATE	C_ID	S_ID
OR123	300	20/05/22	C2	105
OR456	500	15/04/22	C1	103
OR789	1000	30/03/22	C3	102
OR012	2500	05/06/22	C4	104
OR001	3500	18/06/22	C5	101
OR345	700	18/06/22	C5	101
OR980	1500	18/06/22	C5	101

ii) Count the customers with their grades above the average who lives in 'Bangalore'.

select grade,count(distinct c_id) from customers group by grade having grade>(select avg(grade) from customers where c_city='Bangalore');



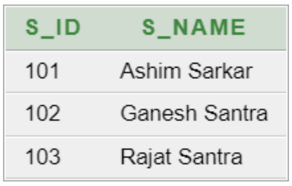
GRADE	COUNT(DISTINCTC_ID)
5	3



ORDER_NUM	ORDER_DATE	PURCHASE_AMOUNT	S_ID	S_NAME
OR345	18/06/22	700	101	Ashim Sarkar
OR980	18/06/22	1500	101	Ashim Sarkar
OR001	18/06/22	3500	101	Ashim Sarkar
OR789	30/03/22	1000	102	Ganesh Santra
OR456	15/04/22	500	103	Rajat Santra
OR012	05/06/22	2500	104	Bitu Das
OR123	20/05/22	300	105	Krishnendu Nanda

iii) Find the name and numbers of all salesman who have more than one customer.

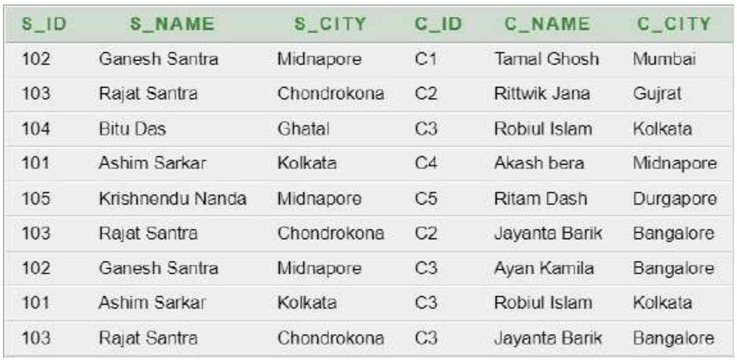
select s_id,s_name from salesmans where 1<(select count(*) from customers where customers.s_id=salesmans.s_id);



S_ID	S_NAME
101	Ashim Sarkar
102	Ganesh Santra
103	Rajat Santra

iv) List all salesman and indicate those who have and don’t have customer in their cities.

select salesmans.s_id,salesmans.s_name,salesmans.s_city,customers.c_id,customers.c_name,customers.c_city from salesmans join customers on salesmans.s_id=customers.s_id where salesmans.s_city=customers.c_city or salesmans.s_city!=customers.c_city;



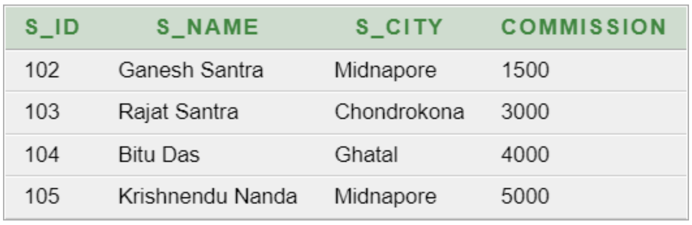
S_ID	S_NAME	S_CITY	C_ID	C_NAME	C_CITY
102	Ganesh Santra	Midnapore	C1	Tamal Ghosh	Mumbai
103	Rajat Santra	Chondrokona	C2	Rittwik Jana	Gujrat
104	Bitu Das	Ghatal	C3	Robiul Islam	Kolkata
101	Ashim Sarkar	Kolkata	C4	Akash bera	Midnapore
105	Krishnendu Nanda	Midnapore	C5	Ritam Dash	Durgapore
103	Rajat Santra	Chondrokona	C2	Jayanta Barik	Bangalore
102	Ganesh Santra	Midnapore	C3	Ayan Kamila	Bangalore
101	Ashim Sarkar	Kolkata	C3	Robiul Islam	Kolkata
103	Rajat Santra	Chondrokona	C3	Jayanta Barik	Bangalore

v) Create a view that finds the salesman who has the customer with the highest order of a day.

select orders.order_num,orders.order_date,orders.purchase_amount,salesmans.s_id,salesmans.s_name from orders join salesmans on orders.s_id=salesmans.s_id order by orders.s_id asc;

vi) Delete the salesman with id ‘101’.

```
delete from salesmans where s_id=101;
```



S_ID	S_NAME	S_CITY	COMMISSION
102	Ganesh Santra	Midnapore	1500
103	Rajat Santra	Chondrokona	3000
104	Bitu Das	Ghatal	4000
105	Krishnendu Nanda	Midnapore	5000

GROUP-B: NUMERICAL LABORATORY MANUAL
(Course Code: BCA-2197)

S. No.	Experiments
1.	Solving various problems related programme with C.
2.	Implement Numerical problems Using C.
3.	Assignment on Interpolation: Newton forward, Newton Backward & Lagrange.
4.	Assignments on Numerical Integration: Trapezoidal Rule, Simpson's 1/3 Rule.
5.	Assignments on Numerical solution of a system of liner equation: Gauss elimination, Gauss Jacobi, Gauss Seidel.
6.	Assignments on Algebraic Equation: Bisection, Regula-falsi, Newton Raphson.
7.	Assignments on Ordinary Differential Equation: Euler's method, RungeKutta.

1. **Solving various problems related programme with C.**

   a. ***Write a program to print the Fibonacci series for 1 to n value.***

   Program:

```
#include<stdio.h>
#include<conio.h>
int main()
{
int a=0,b=1,c,n,i;
clrscr();
printf("Enter the number of terms \n");
scanf("%d",&n);
for(i=1;i<=n;i++){
    printf("%d ,",a);
    c=a+b;
    a=b;
    b=c;
}
getch();
return 0;
}
```

   Input and Output Section:

```
Enter the number of terms
10
0 ,1 ,1 ,2 ,3 ,5 ,8 ,13 ,21 ,34 ,
Enter the number of terms
3
0, 1, 1,
```

b. Write a program to print a prime number up to 1 to n.

Program:

```
#include<stdio.h>
#include<conio.h>
int main()
{
int n, i, count=0, j;
clrscr();
printf("enter the number: \n");
scanf("%d",&n);
for(i=1; i<=n; i++)
  {
    for(j=1; j<=i; j++)
    {
      if(i%j==0)
        count++;
    }
if(count==2)
  printf(" %d ",i);
count=0;
}
return 0;
getch( );
}
```

Input and Output Section:

```
enter the number:
10
```

```
2  3  5  7

enter the number:

100

2  3  5  7  11  13  17  19  23  29  31  37  41  43  47  53  59  61  67
71  73  79  83  89  97
```

c. ***Write a program to display Fibonacci series (i) using recursion (ii) using iteration.***

Program:

```
#include <stdio.h>

#include<conio.h>

//Iterative function definition

void ifib(int n){

    int a=0,b=1,c,i;

    if(n<1){

        printf("Fibonacci series : %d ",a);

    }

    else if(n==1){

        printf("Fibonacci series : %d %d ",a,b);

                 }

        else{

            printf("Fibonacci series are: %d %d",a,b);

            for(i=2;i<=n;i++){

                c=a+b;

                a=b;

                b=c;


            printf(" %d",c);
            }
        }
    }
//Recursion function definition
int rfib(int n){
    int a=0,b=1;
//base condition
    if(n<=1){
        return n;
    }
//Recursive Procedure
    else{
        return rfib(n-2)+rfib(n-1);
    }
}
int main()
{
    int n,result;
    printf("Enter the number \n");
    scanf("%d",&n);
    //iterative function call
    ifib(n);
    //recursion function call
    result=rfib(n);
    printf("\n Last value of Fibonacci series %d",result);
    getch();
    return 0;
```

```
}
```

Input and Output Section:

```
Enter the number
1
Fibonacci series : 0 1
 Last value of Fibonacci series 1
Enter the number
10
Fibonacci series are : 0 1 1 2 3 5 8 13 21 34 55
 Last value of Fibonacci series 55
```

d. ***Write a program to calculate GCD of 2 number (i) with recursion (ii) without recursion.***

Program:

```
#include <stdio.h>

#include<conio.h>

//Iterative function definition

int igcd(int n,int m){

    while(m!=n){

        if(m>n){

            m=m-n;

        }

        else{

            n=n-m;

        }

            }

          return m;

    }

//Recursion function definition

int rgcd(int n,int m){

    //base condition

    if(n==m){

        return n;

    }

    // Recrsive Procedure

    if(m>n){

        return rgcd(m-n,n);

    }

        rgcd(m,n-m);
}
int main()
{
    int a,b;
    printf("Enter the two number \n");
    scanf("%d%d",&a,&b);
    //iterative function call
    printf("%d and %d GCD is %d ",a,b,igcd(a,b));
    //recursion function call
    printf("\n%d and %d GCD is %d",a,b,rgcd(a,b));
    getch();
    return 0;
}
```

Input and Output Section:

```
Enter the two number
101 202
101 and 202 GCD is 101
101 and 202 GCD is 101
Enter the two number
13 91
13 and 91 GCD is 13
13 and 91 GCD is 13
Enter the two number
19 17
19 and 17 GCD is 1
19 and 17 GCD is 1
```

e. ***Write a program to calculate factorial of a number (i) using recursion (ii) using iteration***

Program:

```
#include <stdio.h>
#include<conio.h>
//Iterative function definition
long ifact(int n){
 long fact=1,i;
 for(i=1;i<=n;i++){
     fact=fact*i;
 }
 return fact;
}
//Recursion function definition
long rfact(int n){
    //base condition
    if(n<=1){
        return n;
    }
    // Recrsive Procedure
    else{
        return n*rfact(n-1);
    }
}
int main()
{
    int n;
    printf("Enter the number \n");
    scanf("%d",&n);
    //iterative function call
    printf("%d Factorial is %ld",n,ifact(n));
    //recursion function call
    printf("\n%d Factorial is %ld",n,rfact(n));
    getch();
    return 0;
}
```

Input and Output Section:

```
Enter the number
5
5 Factorial is 120
5 Factorial is 120
Enter the number
9
9 Factorial is 362880
9 Factorial is 362880
```

f. ***Write a program to convert the given binary number to 2‘s complement.***

Program:

```
#include <stdio.h>
#include<conio.h>
int main()
{
   int n;
   printf("Enter the number of bits do you want to enter :");
   scanf("%d",&n);
   // binary array declaration;
   char binary[n+1];
   // onescomplement array declaration
   char onescomplement[n+1];
   // twoscomplement array declaration
   char twoscomplement[n+1];
   int carry=1;
   printf("\nEnter the binary number : ");
   scanf("%s", binary);
   printf("%s", binary);
   printf("\nThe ones complement of the binary number is :");

   // Finding onescomplement in C
   for(int i=0;i<n;i++)
   {
      if(binary[i]=='0')
      onescomplement[i]='1';
      else if(binary[i]=='1')
      onescomplement[i]='0';
   }
   onescomplement[n]='\0';
   printf("%s",onescomplement);
printf("\nThe twos complement of a binary number is : ");
// Finding twoscomplement in C
for(int i=n-1; i>=0; i--)
    {
       if(onescomplement[i] == '1' && carry == 1)
```

```
        {
            twoscomplement[i] = '0';
        }
        else if(onescomplement[i] == '0' && carry == 1)
        {
            twoscomplement[i] = '1';
            carry = 0;
        }
        else
        {
            twoscomplement[i] = onescomplement[i];
        }
    }
twoscomplement[n]='\0';
printf("%s",twoscomplement);
getch();
return 0;
}
```

Input and Output Section:

```
Enter the number of bits do you want to enter :5
Enter the binary number : 10101
10101
The ones complement of the binary number is :01010
The twos complement of a binary number is : 01011
Enter the number of bits do you want to enter :4
Enter the binary number : 0001
0001
The ones complement of the binary number is :1110
The twos complement of a binary number is : 1111
```

2. Implement Numerical problems Using C.

3. Assignment on Interpolation:

(i) Newton Forward Interpolation:

(ii) Newton Backward Interpolation:

(iii) Lagrange Interpolation:

(i) Newton Forward Interpolation:

Evaluate f (2.5) from the table using Newton forward formula.

X	0	1	2	3	4	5
f(x)	41	43	47	53	61	71

Program:

```
#include<stdio.h>
#include<conio.h>
int main(){
float x[10],y[20],s,u,sum,p=1,h;
int i,j,n;
clrscr();
printf("Enter the how many point \n");
scanf("%d",&n);
printf("Enter the value of x and y \n");
for(i=1;i<=n;i++)
     scanf("%f%f",&x[i],&y[i]);
h=x[2]-x[1];
printf("Enter which value is to be computed \n");
scanf("%f",&u);
s=(u-x[1])/h;
sum=y[1];
    for(i=1;i<=n-1;i++){
        for(j=1;j<=n-i;j++)
                 {
                          y[j]=y[j+1]-y[j];
                 }
        p=p*(s-i+1)/i;
        sum=sum+p*y[1];
        }
        printf("The result=%f",sum);
        getch();
return 0;
}
```

Input and Output Section:

```
Enter the how many point
6
Enter the value of x and y
0 41
1 43
2 47
3 53
4 61
5 71
Enter which value is to be computed
2.5
The result=49.750000
```

(ii) Newton Backward Interpolation:

Evaluate f (6) from the table using Newton forward formula.

X	0	1	2	3	4	5
f(x)	1	5	31	121	341	781

Program:

```
#include<stdio.h>
#include<conio.h>
int main(){
        float x[10],y[20],s,u,sum,p=1,h;
        int i,j,n;
        clrscr();
        printf("Enter the how many point \n");
        scanf("%d",&n);
        printf("Enter the value of x and y \n");
        for(i=1;i<=n;i++)
        scanf("%f%f",&x[i],&y[i]);
        h=x[2]-x[1];
        printf("Enter which value is to be computed \n");
        scanf("%f",&u);
        s=(u-x[n])/h;
        sum=y[n];
        for(i=1;i<=n-1;i++){
                for(j=1;j<=n-i;j++)
                {
                        y[j]=y[j+1]-y[j];
                }
               p=p*(s+i-1)/i;
               sum=sum+p*y[j-1];
        }
        printf("The result=%f",sum);
        getch();
return 0;
}
```

Input and Output Section:

```
Enter the how many point
6
Enter the value of x and y
0 1
1 5
2 31
3 121
4 341
5 781
Enter which value is to be computed
6
The result=1555.000000
```

(iii) Lagrange Interpolation:

Write a program to find the value of a function f(x) using given tabular values by Lagrange Interpolation method. Test the program to find f(2.18) using the following:

X	2	2.2	2.4	2.6	2.8	3
f(x)	0.30103	0.34242	0.38041	0.41497	0.44716	0.47721

Program:

```
#include<stdio.h>
#include<conio.h>
int main(){
float x[10],y[10],u,p,sum=0;
int i,j,n;
clrscr();
printf("Enter the how many points \n");
scanf("%d",&n);
printf("Enter the x and y values \n");
    for(i=0;i<n;i++){
         scanf("%f%f",&x[i],&y[i]);
                 }
printf("Which value is to be computed \n");
scanf("%f",&u);
for(i=0;i<n;i++){
     p=y[i];
     for(j=0;j<n;j++){
         if(i!=j){
             p=p*(u-x[j])/(x[i]-x[j]);
                 }
             }
         sum=sum+p;
        }
printf("The result y(%.2f)= %f",u,sum);
getch();
return 0;
}
```

Input and Output Section:

```
Enter the how many points
6
Enter the x and y values
2       0.30103
2.2     0.34242
2.4     0.38041
2.6     0.41497
2.8     0.44716
3       0.47721
Which value is to be computed
```

2.18
The result y(2.18)=0.338414

4. Assignments on Numerical Integration:
(i)Trapezoidal Rule
(ii) Simpson’s 1/3 Rule.

(i) Trapezoidal Rule

Evaluate the integral by Trapezoidal rule with sub-interval n=10

$$\int_0^{\frac{\pi}{4}} \sqrt{1-0.154 sin\ ^2 x}\ \ dx$$

Solution: Here a=0, b=$\frac{\pi}{4}$ = $\frac{22}{28}$ = 0.7857, n=10,

Then, h=$\frac{b-a}{n}$ = $\frac{45^0-0}{10}$ = 0.07857

Program:

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
int main(){
float a,b,h,sum=0,val,x,y[10];
int i,n;
clrscr();
printf("Enter the upper limit \n");
scanf("%f",&b);
printf("Enter the lower limit \n");
scanf("%f",&a);
printf("Enter the subInterval \n");
scanf("%d",&n);
h=(b-a)/n;
        for(i=0;i<=n;i++){
                x=a+i*h;
                y[i]=sqrt(1-0.154*sin(x)*sin(x));
        }
        for(i=1;i<=n-1;i++){
                sum=sum+y[i];
    }
    val=h/2*(y[0]+y[n]+2*sum);
printf("The result = %f",val);
getch();
return 0;
}
```

Input and Output Section:

```
Enter the upper limit
0.7857
Enter the lower limit
0
Enter the subInternal
10
The result = 0.774523
```

(ii) Simpson’s 1/3 Rule.

Evaluate $\int_0^4 (3x - 2x^2)dx$, taking 10 intervals by Simpson’s 1/3 rule.

Program:

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
int main(){
float a,b,h,sum=0,sum1=0,val,x,y[10];
int i,n;
clrscr();
printf("Enter the upper limit \n");
scanf("%f",&b);
printf("Enter the lower limit \n");
scanf("%f",&a);
printf("Enter the subInterval \n");
scanf("%d",&n);
h=(b-a)/n;
        for(i=0;i<=n;i++){
                x=a+i*h;
                y[i]=(3*x-2*pow(x,2));
                //y[i]=(3*x-2*(x*x));
        }
        for(i=1;i<=n-1;i+=2){
                sum1=sum1+y[i];
```

```
         }
        for(i=2;i<=n-2;i+=2){
                sum=sum+y[i];
        }
    val=h/3*(y[0]+y[n]+2*sum+4*sum1);
printf("The result = %f",val);
getch();
return 0;
}
```

Input and Output Section:

```
Enter the upper limit
4
Enter the lower limit
0
Enter the subInternal
10
The result = -18.666668
```

5. *Assignments on Numerical solution of a system of liner equation:*

(i)Gauss elimination

(ii)Gauss Jacobi

(iii)Gauss Seidel

(i) Gauss elimination

Write a program to solve following system of equation by Gauss-elimination method:

$x+2y+3z=7$

$2x+7y+15z=26$

$3x+15y+41z=26$

Solution:

Augmented Matrix:

$$\begin{bmatrix} 1 & 2 & 3 \dots \dots .7 \\ 2 & 7 & 15 \dots ..26 \\ 3 & 15 & 41 \dots ..26 \end{bmatrix}$$

Program:

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
int main()
{
    int i,j,k,n;
    float A[20][20],c,x[10],sum=0.0;
    clrscr();
    printf("\nEnter the order of matrix: ");
    scanf("%d",&n);
    printf("\nEnter the elements of augmented matrix row-wise:\n\n");
    for(i=1; i<=n; i++)
    {
          for(j=1; j<=(n+1); j++)
          {
              printf("A[%d][%d] : ", i,j);
              scanf("%f",&A[i][j]);
          }
    }
    /* loop for the generation of upper triangular matrix*/
    for(j=1; j<=n; j++)
    {
          for(i=1; i<=n; i++)
          {
              if(i>j)
              {
                    c=A[i][j]/A[j][j];
                    for(k=1; k<=n+1; k++)
                    {
                        A[i][k]=A[i][k]-c*A[j][k];
                    }
              }
          }
    }
    x[n]=A[n][n+1]/A[n][n];
    /* this loop is for backward substitution*/
    for(i=n-1; i>=1; i--)
    {
        sum=0;
        for(j=i+1; j<=n; j++)
        {
            sum=sum+A[i][j]*x[j];
        }
        x[i]=(A[i][n+1]-sum)/A[i][i];
    }
    printf("\nThe solution is: \n");
    for(i=1; i<=n; i++)
    {
        printf("\nx%d=%.2f\t",i,x[i]);
    }
    getch();
    return(0);
}
```

Input and Output Section:

```
Enter the order of matrix: 3
Enter the elements of augmented matrix row-wise:
A[1][1] : 1
A[1][2] : 2
A[1][3] : 3
A[1][4] : 7
A[2][1] : 2
A[2][2] : 7
A[2][3] : 15
A[2][4] : 26
A[3][1] : 3
A[3][2] : 15
A[3][3] : 41
A[3][4] : 26
The solution is:
x1=-19.60
x2=22.60
x3=-6.20
```

(ii) Gauss Jacobi

Solve following system of equation by Gauss Jacobi iteration method (Iteration method)

$$20x_1+x_2+x_3=23.28$$

$$x_1+15x_2-x_3=29.92$$

$$2x_1+x_2-20x_3=-55.64$$

Program:

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
#define eps 0.0001
#define x1(x2,x3)  ((23.28-x2-x3)/20)
#define x2(x1,x3)  ((29.92-x1+x3)/15)
#define x3(x1,x2)  ((55.64+x2+2*x1)/20)
int main(){
float x1=0,x2=0,x3=0,y1,y2,y3;
int flag=0;
clrscr();
printf("\n \t x1 \t\t x2 \t\t x3");
printf("\n \t %f \t %f \t %f",x1,x2,x3);
do{
        y1=x1(x2,x3);
        y2=x2(x1,x3);
        y3=x3(x1,x2);
     if(fabs(x1-y1)<eps && fabs(x2-y2)<eps && fabs(x3-y3)<eps){
        printf("\n x1=%.3f",y1);
        printf("\n x2=%.3f",y2);
        printf("\n x3=%.3f",y3);
          flag=1;
     }
    else{
      x1=y1;
      x2=y2;
      x3=y3;
    printf("\n \t %f \t %f \t %f",x1,x2,x3);
    }
}while(flag!=1);
getch();
return 0;
}
```

Input and Output Section:

```
X1              X2              X3
0.000000        0.000000        0.000000
1.164000        1.994667        2.782000
0.925167        2.102533        2.998133
0.908967        2.132864        2.979643
0.908375        2.132712        2.979540
X1=0.908

X2=2.133

X3=2.979
```

(iii) Gauss Seidel

Compute the solution of the system by Gauss-Seidel method:

$$5.2x_1+1.2x_2+2.3x_3=18.2$$

$$3.2x_1+9.4x_2-1.5x_3=22.8$$

$$2.1x_1-1.6x_2+8.6x_3=28.4$$

Program:

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
#define eps 0.0001
#define x1(x2,x3)  ((18.2-1.2*x2-2.3*x3)/5.2)
#define x2(x1,x3)  ((22.8-3.2*x1+1.5*x3)/9.4)
#define x3(x1,x2)  ((28.4-2.1*x1+1.6*x2)/8.6)
int main(){
float x1=0,x2=0,x3=0,y1,y2,y3;
int flag=0;
clrscr();
printf("\n \t x1 \t\t x2 \t\t x3");
printf("\n \t %f \t %f \t %f",x1,x2,x3);
do{
```

```
          y1=x1(x2,x3);
          y2=x2(y1,x3);
          y3=x3(y1,y2);
      if(fabs(x1-y1)<eps && fabs(x2-y2)<eps && fabs(x3-y3)<eps){
          printf("\n x1=%.3f",y1);
          printf("\n x2=%.3f",y2);
          printf("\n x3=%.3f",y3);
            flag=1;
      }
     else{
       x1=y1;
       x2=y2;
       x3=y3;
     printf("\n \t %f \t %f \t %f",x1,x2,x3);
     }
}while(flag!=1);
getch();
return 0;
}
```

Input and Output Section:

```
 X1                 X2                 X3
 0.000000           0.000000           0.000000
 3.500000           1.234043           2.677264
 2.031047           2.161335           3.208481
 1.582095           2.398938           3.362314
 1.459221           2.465315           3.404667
 1.425171           2.483665           3.416396
 1.415748           2.488745           3.419642
 1.413140           2.490150           3.420540
 1.412419           2.490540           3.420789
 1.412219           2.490647           3.420858
X1=1.412

X2=2.491

X3=3.421
```

6. *Assignments on Algebraic Equation:*

(i) Bisection Method

(ii) Regula-falsi Method

(iii) Newton Raphson Method

(iv) Iteration Method

(i) Bisection Method

Compute the root of the equation x^3-4x-9=0 correct to 3 decimal places, using Bisection method.

Solution:

Let f(x)= x^3-4x-9

f (1) =1-4-9=-12 i.e. (-ve)

f (2) =8-8-9=-9 i.e. (-ve)

f (3) =27-12-9=6 i.e. (+ve)

Hence the root lies between 2 and 3.

Program:

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
float Bisection(float x){
return (pow(x,3)-4*x-9);
}
int main(){
        float a,b,x,eps;
        clrscr();
        printf("Enter the value of a,b and eps \n");
        scanf("%f%f%f",&a,&b,&eps);
        x=(a+b)/2;
        while(fabs(x-b)>eps){
        if(Bisection(a)*Bisection(x)>0)
                a=x;
         else
                b=x;
        // printf("The middle point = %f",x);
         x=(a+b)/2;
        }
        printf("The approximate root is =%.3f",x);
```

```
          getch();
return 0;
}
```

Input and Output Section:

```
Enter the value of a, b and eps
2  3   0.00001
The approximate root is =2.707
```

(ii) Regula-falsi Method

Find a root of the equation 3x-cosx-1=0, by Regula_Falsi method.

Solution:

Let f(x)= 3x-cosx-1

f(0)=0-cos0-1=-2 i.e. (-ve)

f(1)=3-cos1-1=1.0001 i.e.(+ve)

Hence the root lies between 0 and 1.

Program:

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
float Regula_falsi(float x){
return 3*x-cos(x)-1;
}
int main(){
clrscr();
float a,b,x,eps;
printf("Enter the value of a,b,eps \n");
scanf("%f%f%f",&a,&b,&eps);
while(fabs (Regula_falsi(b) )> eps)
{
x=(a*Regula_falsi(b)-b*Regula_falsi(a))/(Regula_falsi(b)-Regula_falsi(a));
        if(Regula_falsi(a)*Regula_falsi(x)>0)
                b=x;
        else
                a=x;
}
        printf("The real root is =%f",x);
        getch();
return 0;
}
```

Input and Output Section:

```
Enter the value of a, b and eps
0   1   0.00001
The approximate root is =0.607102
```

(iii) Newton Raphson Method

Find the root of x^3-8x-4=0 which lies between 3 and 4, by Newton-Raphson Method.

Solution:

Let f(x)= x^3-8x-4

f '(x)=$3x^2$-8

f (3) =-1 i.e. (-ve)

f (4) =28 i.e. (+ve)

Hence the root lies between 3 and 4.

Let the initial guess value x0=3.

Program:

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
float f(float x){
    return x*x*x-8*x-4.0;
}
float df(float x){
return 3.0*x*x-8.0;
}
int main(){
        float x0,x1,q,eps;
        int k=1;
        printf("Enter the initial guess value and eps \n");
        scanf("%f%f",&x0,&eps);
        do{
        q=x0;
        x1=x0-(f(x0)/df(x0));
        x0=x1;
        k=k+1;
        }while(fabs(x1-q)>eps);
        printf("Real root is=%f \n",x1);
        printf("Number of steps=%d",k);
getch();
return 0;
}
```

Input and Output Section:

```
Enter the initial guess value and eps
3 0.00001
Real root is=3.051374
Number of steps=4
```

(iv) Iteration Method

Find the real root of the equation x^3-x-1=0 using iteration method.

Program:

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
float f(float x){
return pow(((1+x)/x),.5);
}
int main(){
float x1,x2,r;
clrscr();
printf("Enter the initial guess value \n");
scanf("%f",&x1);
x2=f(x1);
do{
x1=x2;
x2=f(x1);
r=fabs((x2-x1)/x2);
}while(r>0.001);
printf("The real root=%f",x2);
getch();
return 0;
}
```

Input and Output Section:

```
Enter the initial guess value
1
The real root=1.324901
```

7. Assignments on Ordinary Differential Equation:

(i) Euler's Method

(ii) RungeKutta Method

(i) Euler's Method

Compute y (0.03) where $\frac{dy}{dx} = 2x^2 + 3y$, y (0) =1 by Euler's method.

Solution:

Here initial value x_0=0, y_0=1 and last value of x=0.03

Let h=0.*03* $n = \frac{x-x0}{h} = \frac{0.03-0}{0.01} = 3$

Program:

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
float f(float x,float y){
return (2*x*x+3*y);
}
int main(){
        float x0,y0,x,h,n;
        int i;
        clrscr();
        printf("Enter the initial value of x0 y0 and h \n");
        scanf("%f%f%f",&x0,&y0,&h);
        printf("Enter the last value of x \n");
        scanf("%f",&x);
        n=(x-x0)/h;
        for(i=1;i<=n;i++){
        y0=y0+h*f(x0,y0);
        x0=x0+h;
        }
        printf("The result y(%.2f)=%.3f ",x0,y0);
getch();
return 0;
}
```

Input and Output Section:

```
Enter the initial value of x0 y0 and h
0 1 0.01
Enter the last value of x
0.03
The result y (0.03) =1.093
```

(ii) RungeKutta Method

RKM (Second Order):

Compute y (0.8) by Runge-Kutta method from the equation.

$$\frac{dy}{dx} = xy, \quad y(0) = 1, \quad taking\ h = 0.2$$

Solution:

Here initial value of x_0=0, y_0=1

Last value of x=0.8 and h=0.2 then, n=$\frac{x-x0}{h} = \frac{0.8-0}{0.2} = 4$

Program:

```
#include<stdio.h>
#include<conio.h>
float f(float x,float y){
        return (x*y);
}
int main(){
            float x0,y0,x,h,k1,k2;
            int i,n;
            clrscr();
            printf("Enter the initial value of x0, y0 and h \n");
            scanf("%f%f%f",&x0,&y0,&h);
```

```
                printf("Enter the last value of x \n");
                scanf("%f",&x);
                n=(x-x0)/h;
                for(i=1;i<=n;i++){
                    k1=h*f(x0,y0);
                    k2=h*f(x0+h,y0+k1);
                    y0=y0+(k1+k2)/2;
                    x0=x0+h;
                }
                printf("The result y(%.1f)=%f",x0,y0);
                getch();
return 0;
}
```

Input and Output Section:

```
Enter the initial value of x0, y0 and h
0 1 0.2
Enter the last value of x
0.8
The result y (0.8) =1.320000
```

RKM (Fourth Order):

Calculate y (2.5) using Runge-Kutta method of order four of the initial value problem

$$\frac{dy}{dx} = cos^2x + 2x - y^2, \text{ y (1) =0 using h=0.1}$$

Solution:

Here initial value of x_0=1, y_0=0

Last value of x=2.5, h=0.1 then, n=$\frac{x-x0}{h}$ =$\frac{2.5-1}{0.1}$=15

Program:

```
#include<stdio.h>
#include<conio.h>
float f(float x, float y){
        return (cos(x)*cos(x)+2*x-y*y);
}
int main(){
                float x0,y0,x,h,k1,k2,k3,k4,n;
                int i;
                clrscr();
                printf("Enter the initial value of x0, y0 and h \n");
                scanf("%f%f%f",&x0,&y0,&h);
                printf("Enter the last value of x \n");
                scanf("%f",&x);
                n=(x-x0)/h;
                for(i=1;i<=n;i++){
                    k1=h*f(x0,y0);
                    k2=h*f(x0+h/2,y0+k1/2);
                    k3=h*f(x0+h/2,y0+k2/2);
                    k4=h*f(x0+h,y0+k3);
                    y0=y0+(k1+2*k2+2*k3+k4)/6;
                    x0=x0+h;
                }
                printf("The result y(%.1f)=%f",x0,y0);
                getch();
return 0;
}
```

Input and Output Section:

```
Enter the initial value of x0, y0 and h
1 0 0.1
Enter the last value of x
2.5
The result y (2.5) =2.210078
```