

Name of Experiment : Student database

Date : 9/11/2024

Page No. 01

Experiment No : 01

Experiment Result :

## Database Management Systems - Lab

### LAB CYCLE - I.

Consider a structure named student with attributes as SID, NAME, BRANCH, SEMESTER, ADDRESS. Write a program in c/c++ and perform the following operations using the concept of files.

- a. Insert a new student
- b. Modify the address of the student based on SID.
- c. Delete a student
- d. List all the students
- e. List all the students of CSE branch.
- f. List all the students of CSE branch and reside in Kuvempunagar.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

typedef struct student {
    char SID[20];
    char name[30];
    char branch[25];
    char semester[10];
    char address[25];
} stu;

stu st;
```

void insert\_new() {
 printf ("Enter SID: ");
}

Teacher's Signature : \_\_\_\_\_

Name of Experiment : .....

Date : 09/11/2024 .....

Page No. 02

Experiment No : 01 .....

Experiment Result : .....

```
scanf ("%s", &st.SID);
printf ("Enter name : ");
scanf ("Enter name %s", &st.name);
printf ("Enter branch : ");
scanf ("%s", &st.branch);
printf ("Enter the semester : ");
scanf ("%s", &st.semester);
printf ("Enter the address : ");
fgets (&st.address, sizeof (st.address), stdin);
st.address [strcmp (&st.address, "\n")] = '\0';
FILE *fp;
fp = fopen ("student_info.txt", "a");
if (fp == NULL) {
    printf ("Error can't open the file now !\n");
    return;
}
fprintf (fp, "%s\t%s\t%s\t%s\n", &st.SID, &st.name, &st.branch,
        &st.semester, &st.address);
fclose (fp);
printf ("The student data has been added to the file !\n");
}

void modify_add() {
char new_ID [20];
int flag = 0;
printf ("Enter the student ID whose address need to be modified : ");
scanf ("%s", new_ID);
FILE *fp = fopen ("student_info.txt", "r+");
```

Teacher's Signature : \_\_\_\_\_

Name of Experiment :.....

Date : 09/11/2024 .....

Page No. 03

Experiment No : 01.....

Experiment Result :.....

```
if (fp == NULL) {
    printf("Error opening the file\n");
    return;
}

FILE *fp1 = fopen ("sample.txt", "a");
if (fp1 == NULL) {
    printf("Error opening the file\n");
    return;
}

while (fscanf(fp, "%s %s %s %s", &st.SID, &st.name, &st.branch,
              &st.semester, &st.address) != EDF) {
    if (strcmp(st.SID, new.SID) == 0) {
        printf("Enter the new address of the student : ");
        scanf("%s", &st.address);
        flag = 1;
    }

    fprintf(fp1, "%s\t%s\t%s\t%s\n", st.SID, st.name,
            st.branch, st.semester, st.address);
}

if (flag) {
    printf("The address is successfully modified\n");
}
else
    printf("The student with the given student ID do not exist\n");

fclose(fp);
fclose(fp1);
remove("student-info.txt");
rename("sample.txt", "student-info.txt");
}
```

Teacher's Signature : \_\_\_\_\_

Name of Experiment : .....

Date : ... 09/11/2024 .....

Page No. 04

Experiment No : 01 .....

Experiment Result : .....

```
void delete_st() {  
    char new_ID[20];  
    int flag = 0;  
    printf ("Enter the student ID whose entry in the file is to be deleted\n");  
    scanf ("%s", new_ID);  
    FILE *fp = fopen ("student_info.txt", "r");  
    if (fp == NULL) {  
        printf ("Error opening the file !!\n");  
        return;  
    }  
    FILE *fp1 = fopen ("sample.txt", "a");  
    if (fp1 == NULL) {  
        printf ("Error opening the file !!\n");  
        return;  
    }  
    while (fscanf (fp, "%-8%-S%-S%-S%-S", &st.SID, &st.name, &st.branch,  
           &st.semester, &st.address) != EOF) {  
        if (strcmp (st.SID, new_ID) != 0) {  
            fprintf (fp1, "%-8t%-St%-St%-St%-St\n", st.SID, st.name,  
                    st.branch, st.semester, st.address);  
        }  
        else  
            flag = 1;  
    }  
    if (flag)  
        printf ("The student with the given ID is deleted successfully\n");  
    else  
        printf ("The given student do not exist\n");
```

Teacher's Signature : \_\_\_\_\_

Name of Experiment :.....

Date :..09/11/2024.....

Experiment No : 01.....

Experiment Result :.....

Page No. 05

```
fclose(fp);  
fclose(fp);  
remove("student-info.txt");  
rename("sample.txt", "student-info.txt");  
}  
  
void list_all(){  
    int flag = 0;  
    FILE *fp = fopen("student-info.txt", "r");  
    if (fp == NULL){  
        printf("Error opening the file !!\n");  
        return;  
    }  
  
    printf("The list of all students :\n");  
    while (fscanf(fp, "%s/%s/%s/%s/%s", &st.SID, &st.name, &st.branch, &st.semester,  
              &st.address) != EOF){  
        printf("Student ID : %s\n", st.SID);  
        printf("Student name : %s\n", st.name);  
        printf("Student branch : %s\n", st.branch);  
        printf("Student semester : %s\n", st.semester);  
        printf("Student address : %s\n", st.address);  
        printf("-----\n");  
        flag = 1;  
    }  
  
    if (flag == 0)  
        printf("There is no student present in the given file !!\n");  
    fclose(fp);  
}
```

Teacher's Signature : \_\_\_\_\_

Name of Experiment : .....  
Experiment No : 01 .....

Date : 09/11/2024 .....  
Experiment Result : .....

Page No. 06

```
void list_cse() {
    int flag=0;
    FILE *fp = fopen ("student_info.txt", "r");
    if (fp == NULL) {
        printf ("Error opening the file !!\n");
        return;
    }
    printf ("The list of all the students of CSE branch\n");
    while (fscanf (fp, "%10s%10s%10s%10s", &st.ID, &st.name, &st.branch,
              &st.semester, &st.address) != EOF) {
        if (strcmp(st.branch, "CSE") == 0) {
            printf ("Student ID : %s\n", st.ID);
            printf ("Student name : %s\n", st.name);
            printf ("Student branch : %s\n", st.branch);
            printf ("Student semester : %s\n", st.semester);
            printf ("Student address : %s\n", st.address);
            printf ("-----\n");
            flag = 1;
        }
    }
    if (flag == 0)
        printf ("There is no student present in CSE in the file !!\n");
    fclose (fp);
}
```

```
void list_cse_kuv() {
    int flag=0;
    FILE *fp = fopen ("student_info.txt", "r");
}
```

Teacher's Signature : \_\_\_\_\_

Name of Experiment : .....

Date : 09/11/2024 .....

Experiment No : 01 .....

Experiment Result : .....

Page No. 09

```
if (fp == NULL) {
    printf("Error opening the file\n");
    return;
}

printf("In ----- CSE students in Kuvempunagar ----- \n");
while (fscanf(fp, "%s %s %s %s", &st.SID, &st.name, &st.branch,
              &st.semester, &st.address) != EDF) {
    if (strcmp(st.branch, "CSE") == 0 && strcmp(st.address, "Kuvempunagar") == 0) {
        printf("Student SID      : %s\n", st.SID);
        printf("Student name     : %s\n", st.name);
        printf("Student branch   : %s\n", st.branch);
        printf("Student semester : %s\n", st.semester);
        printf("Student address  : %s\n", st.address);
        printf("Student ----- \n");
        flag = 1;
    }
}

if (flag == 0)
    printf("There is no student of whose branch is CSE and resides
          in Kuvempunagar\n");
fclose(fp);

int main() {
    int choice;
    while(1) {
        printf("In ----- Menu ----- \n");
        printf("1. Insert a new student\n");
    }
}
```

Teacher's Signature : \_\_\_\_\_

Name of Experiment : .....

Date : 09/11/2024 .....

Experiment No : 01 .....

Experiment Result : .....

Page No. 08

```
printf ("2. Modify the address of a student based on SID\n");
printf ("3. Delete a student\n");
printf ("4. List all students\n");
printf ("5. List all students of CSE branch\n");
printf ("6. List all students of CSE branch residing in kuvempunagar\n");
printf ("7. Exit\n\n");
printf ("Enter your choice : ");
scanf ("%d", &choice);
switch(choice) {
    case 1 : insert_new();
    printf ("\n"); break;
    case 2 : modify_addr();
    printf ("\n"); break;
    case 3 : delete_st();
    printf ("\n"); break;
    case 4 : list_all();
    printf ("\n"); break;
    case 5 : list_cse();
    printf ("\n"); break;
    case 6 : list_cse_kuv();
    printf ("\n"); break;
    case 7 : exit(0);
    printf ("\n"); break;
    default : printf ("Invalid choice !!\n");
    printf ("\n"); break;
}
```

}

}

return 0;

J.S.N  
10/11/2024

Teacher's Signature : \_\_\_\_\_

## Output:

----- Menu -----

1. Insert a new student
2. Modify the address of a student based on SID
3. Delete a student
4. List all students
5. List all students of CSE branch
6. List all students of CSE branch residing in kuvempunagar.

Enter your choice : 3

Error opening the file !!

Enter your choice : 3

Error opening the file !!

Enter your choice : 4

Error opening the file !!

Enter your choice : 5

Error opening the file !!

Enter your choice : 6

Error opening the file !!

Enter your choice : 1

Enter SID : 02JST22UCS035

Enter name : Shanyashree

Enter branch : CSE

Enter semester : 5

Enter address : Agrahara

The student data has been added to the file.

Enter your choice : 1

Enter SID : 02JST22VCS015

Enter name : Archana

Enter branch : CSE

Enter semester : 5

Enter address : Bogadi

The student data has been added to the file.

Enter your choice : 1

Enter SID : 02JST22VCS025

Enter name : Chandana

Enter branch : CSE

Enter semester : 5

Enter address : kuvempunagar

The student data has been added to the file.

Enter your choice : 1

Enter SID : 02JST22VLS143

Enter name : Seetha

Enter branch : ISE

Enter semester : 5

Enter address : Romakrishnanagar

The student data has been added to the file.

Enter your choice : 1

Enter SID : 02JST22VLS045

Enter name : Ram

Enter branch : CSE

Enter semester : 5

Enter address : kuvempunagar

The student data has been added to the file.

Enter your choice : 1

Enter SID : 01JST22ULS030 MEDGS

Enter name : Shyam

Enter branch : Mechanical

Enter semester : 5

Enter address : kuvempunagar

The student data has been added to the file.

Enter your choice : 4

----- List of students -----

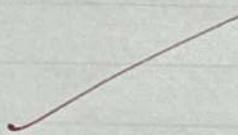
Student ID : 02JST22ULS035

Student name : Amanyathree

Student branch : CSE

Student semester : 5

Student address : Agrahara



-----

Student ID : 02JST22ULS015

Student name : Aechana

Student branch : CSE

Student semester : 5

Student address : Bogadi

-----

Student ID : 02JST22VCS025

Student name : chandana

Student branch : CSE

Student semester : 5

Student address : kuvempunagar

-----  
Student ID : 02JST22VJS143

Student name : Seetha

Student branch : ISE

Student semester : 5

Student address : somakalathmanagar

-----  
Student ID : 02JST22VCS045

Student name : Ram

Student branch : CSE

Student semester : 5

Student address : kuvempunagar

-----  
Student ID : 01JST22VME065

Student name : Shyam

Student branch : Mechanical

Student semester : 5

Student address : kuvempunagar



Enter your choice : 2

Enter the student ID whose address need to be modified : 02JST22VCS025

Enter the new address of the student : Vijaynagar

The address is successfully modified.

Enter your choice & 5

The list of all students of CSE branch :

|                  |   |               |
|------------------|---|---------------|
| Student ID       | : | O2JST22UCS035 |
| Student name     | : | Dhanyaathree  |
| Student branch   | : | CSE           |
| Student semester | : | 5             |
| Student address  | : | Agrahara      |

|                  |   |               |
|------------------|---|---------------|
| Student ID       | : | O2JST22UCS015 |
| Student name     | : | Archana       |
| Student branch   | : | CSE           |
| Student semester | : | 5             |
| Student address  | : | Bogadi        |

|                  |   |                         |
|------------------|---|-------------------------|
| Student ID       | : | O2JST22UCS025           |
| Student name     | : | Chandana                |
| Student branch   | : | CSE                     |
| Student semester | : | 5                       |
| Student address  | : | Kunempunagat Vijaynagar |

|                  |   |               |
|------------------|---|---------------|
| Student ID       | : | O2JST22UCS045 |
| Student name     | : | Ram           |
| Student branch   | : | CSE           |
| Student semester | : | 5             |
| Student address  | : | Kuvempunagar  |

Enter your choice : 6

----- CSE students in Kuvempunagal -----

Student ID : OJST22UCS045  
Student name : Ram  
Student branch : CSE  
Student semester : 5  
Student address : Kuvempunagal

---

Enter your choice : 3

Enter the student ID whose entry in the file to be deleted

OJST22UME065

The student with the given ID is deleted successfully.

Enter your choice : f .

Name of Experiment : student database

Experiment No : 02

Date : 16/11/2024

Page No. 09

Experiment Result :

Create a table for the structure student with attributes as SID, NAME, BRANCH, SEMESTER, ADDRESS, PHONE, EMAIL. Insert atleast 10 tuples and perform the following operations using SQL.

- a. Insert a new student
- b. Modify the address of the student based on SID.
- c. Delete a student
- d. List all the students
- e. List all the students of CSE branch
- f. List all the students of CSE branch and resides in kuvempunagar.

Step 0 : Create the database

```
CREATE DATABASE STUDENT;
```

Step 1 : Use the database

```
USE STUDENT;
```

Step 2 : Create the table

```
create table STUDENT (  
    SID int primary key,  
    Name varchar(50),  
    Branch varchar(10),  
    Semester int,  
    Address varchar(100),  
    Phone varchar(15),  
    Email varchar(50)  
)
```

Step 3 : Insert initial tuples

Teacher's Signature : \_\_\_\_\_

Name of Experiment : .....

Date :..16/11/24.....

Experiment No : 02 .....

Experiment Result : .....

Page No. 10

```
insert into STUDENT (SID, Name, Branch, Semester, Address, Phone, Email) VALUES  
(1, 'John Doe', 'CSE', 3, 'Kuvempunagar', '9876572340', 'john@example.com'),  
(2, 'Jane Smith', 'ECE', 5, 'Saravananipalam', '8776124350', 'jane@example.com'),  
(3, 'Sam Wilson', 'CSE', 3, 'Vijaynagar', '8715432782', 'sam@example.com'),  
(4, 'Lisa Ray', 'EEE', 5, 'Kuvempunagar', '7768923152', 'lisa@example.com'),  
(5, 'Tom Lee', 'CSE', 1, 'Kuvempunagar', '8215098615', 'tom@example.com'),  
(6, 'Anna Brown', 'Mech', 1, 'Hobloal', '6360098760', 'anna@example.com'),  
(7, 'Mark Taylor', 'CSE', 7, 'Saravananipalam', '9860057812', 'mark@example.com'),  
(8, 'Emily Clark', 'CSE', 5, 'Kuvempunagar', '8150057125', 'emily@example.com'),  
(9, 'James Bond', 'Civil', 1, 'Jayalakshmi pucram', '9812578230', 'salah@example.com'),  
(10, 'Sarah Connor', 'CSE', 5, 'Kuvempunagar', '6150057125', 'sarah@example.com');
```

Step 4 : Perform the operations

-- A. Insert New Student

```
insert into STUDENT (SID, Name, Branch, Semester, Address, Phone, Email) VALUES  
(11, 'Bob Smith', 'CSE', 3, 'Kuvempunagar', '8182572180', 'bob@example.com');
```

-- B. Modify the address of the student based on SID

Update STUDENT

set address = 'Gokulam'

where SID = 3 ;

-- C. Delete a student

Delete from STUDENT

where SID = 2 ;

-- D. List All students

Select \* from STUDENT ;

Teacher's Signature : \_\_\_\_\_

Name of Experiment : .....

Experiment No : 02 .....

Date : 16/11/2024 .....

Experiment Result : .....

Page No. 11

-- E. List all students of CSE branch

Select \* from student

where Branch = 'CSE' ;

-- F. List All students of CSE Branch and Reside in kuvempunagar

Select \* from STUDENT

where Branch = 'CSE' and address = 'kuvempunagar' ;

*J.N  
23/11/2024*

Teacher's Signature : \_\_\_\_\_

Output:

A. After inserting new students :

| SID | Name         | Branch | Semester | Address           | Phone      | Email             |
|-----|--------------|--------|----------|-------------------|------------|-------------------|
| 1   | John Doe     | CSE    | 3        | Kuvempunagar      | 9876512340 | john@example.com  |
| 2   | Jane Smith   | ECE    | 5        | Salaswathipulam   | 8776124350 | jane@example.com  |
| 3   | Sam Wilson   | CSE    | 3        | Vijaynagar        | 8715432782 | sam@example.com   |
| 4   | Lisa Ray     | EEE    | 5        | Kuvempunagar      | 7768923152 | lisa@example.com  |
| 5   | Tom Lee      | CSE    | 1        | Kuvempunagar      | 8215098615 | tom@example.com   |
| 6   | Anna Brown   | Mech   | 1        | Hebbal            | 6360098160 | anna@example.com  |
| 7   | Mark Taylor  | CSE    | 7        | Salaswathipulam   | 9860057812 | mark@example.com  |
| 8   | Emily Clark  | CSE    | 5        | Kuvempunagar      | 8150057125 | emily@example.com |
| 9   | James Bond   | Civil  | 1        | Jayalakshmiipulam | 9812578230 | james@example.com |
| 10  | Sarah Connor | CSE    | 5        | Kuvempunagar      | 6150057125 | sarah@example.com |
| 11  | Bob Smith    | CSE    | 3        | Kuvempunagar      | 8182572180 | bob@example.com   |

B. After updating the address of the student :

| SID | Name         | Branch | Semester | Address           | Phone       | Email             |
|-----|--------------|--------|----------|-------------------|-------------|-------------------|
| 1   | John Doe     | CSE    | 3        | Kuvempunagar      | 98761512340 | john@example.com  |
| 2   | Jane Smith   | ECE    | 5        | Salaswathipulam   | 8776124350  | jane@example.com  |
| 3   | Sam Wilson   | CSE    | 3        | Gokulam           | 8715432782  | sam@example.com   |
| 4   | Lisa Ray     | EEE    | 5        | Kuvempunagar      | 7768923152  | lisa@example.com  |
| 5   | Tom Lee      | CSE    | 1        | Kuvempunagar      | 8215098615  | tom@example.com   |
| 6   | Anna Brown   | Mech   | 1        | Hebbal            | 6360098160  | anna@example.com  |
| 7   | Mark Taylor  | CSE    | 7        | Salaswathipulam   | 9860098160  | mark@example.com  |
| 8   | Emily Clark  | CSE    | 5        | Kuvempunagar      | 8150057125  | emily@example.com |
| 9   | James Bond   | Civil  | 1        | Jayalakshmiipulam | 9812578230  | james@example.com |
| 10  | Sarah Connor | CSE    | 5        | Kuvempunagar      | 6150057125  | sarah@example.com |
| 11  | Bob Smith    | CSE    | 3        | Kuvempunagar      | 8182572180  | bob@example.com   |

C. After deleting a student's

| SID | Name         | Branch | Semester | Address           | Phone       | Email             |
|-----|--------------|--------|----------|-------------------|-------------|-------------------|
| 1   | John Doe     | CSE    | 3        | Kuvempunagar      | 98761512340 | john@example.com  |
| 3   | Sam Wilson   | CSE    | 3        | Gokulam           | 8715432782  | sam@example.com   |
| 4   | Lisa Ray     | EEE    | 5        | Kuvempunagar      | 7768923152  | lisa@example.com  |
| 5   | Tom Lee      | CSE    | 1        | Kuvempunagar      | 8215098615  | tom@example.com   |
| 6   | Anna Brown   | Mech   | 1        | Hobbal            | 6360098160  | anna@example.com  |
| 7   | Mark Taylor  | CSE    | 7        | Sarathwathiapuram | 9860057812  | mark@example.com  |
| 8   | Emily Clark  | CSE    | 5        | Kuvempunagar      | 8150057125  | emily@example.com |
| 9   | James Bond   | Civil  | 1        | Jayalakshmiapuram | 9812578230  | james@example.com |
| 10  | Sarah Connor | CSE    | 5        | Kuvempunagar      | 6150057125  | sarah@example.com |
| 11  | Bob Smith    | CSE    | 3        | Kuvempunagar      | 8182572180  | bob@example.com   |

E. List of all students of CSE branch:

| SID | Name         | Branch | Semester | Address           | Phone       | Email             |
|-----|--------------|--------|----------|-------------------|-------------|-------------------|
| 1   | John Doe     | CSE    | 3        | Kuvempunagar      | 98761512340 | john@example.com  |
| 3   | Sam Wilson   | CSE    | 3        | Gokulam           | 8715432782  | sam@example.com   |
| 5   | Tom Lee      | CSE    | 1        | Kuvempunagar      | 8215098615  | tom@example.com   |
| 7   | Mark Taylor  | CSE    | 7        | Sarathwathiapuram | 9860057812  | mark@example.com  |
| 8   | Emily Clark  | CSE    | 5        | Kuvempunagar      | 8150057125  | emily@example.com |
| 10  | Sarah Connor | CSE    | 5        | Kuvempunagar      | 6150057125  | sarah@example.com |
| 11  | Bob Smith    | CSE    | 3        | Kuvempunagar      | 8182572180  | bob@example.com   |

F. List of all students of CSE branch residing in Kuvempunagar:

| SID | Name         | Branch | Semester | Address      | Phone       | Email             |
|-----|--------------|--------|----------|--------------|-------------|-------------------|
| 1   | John Doe     | CSE    | 3        | Kuvempunagar | 98761512340 | john@example.com  |
| 5   | Tom Lee      | CSE    | 1        | Kuvempunagar | 8215098615  | tom@example.com   |
| 8   | Emily Clark  | CSE    | 5        | Kuvempunagar | 8150057125  | emily@example.com |
| 10  | Sarah Connor | CSE    | 5        | Kuvempunagar | 6150057125  | sarah@example.com |
| 11  | Bob Smith    | CSE    | 3        | Kuvempunagar | 8182572180  | bob@example.com   |

Name of Experiment : Sailors database  
Experiment No : 03

Date : 23/11/2024

Experiment Result :

Page No. 12

Data Definition Language (DDL) commands in RDBMS.

- Consider the database schemas given below. write ER diagram and schema diagram. The primary keys are underlined and the data types are specified. Create tables for the following schema listed below by properly specifying the primary keys and foreign keys.
- Enter all five tuples for each relation.
- Altering tables, adding and dropping different types of constraints. Also adding and dropping fields in to the relational schemas of the listed problems. Delete, update operations.

A. Sailors database

SAILORS (sId, sname, rating, age)

BOAT (bId, bname, color)

RESERVES (sId, bId, date)

Create database Sailors ;

Use Sailors ;

create table Sailors (

    sId int primary key,

    sname varchar(50),

    rating float,

    age int

);

Create table Boat (

    bId int primary key,

    bname varchar(50),

    color varchar(20)

);

Teacher's Signature : \_\_\_\_\_

Name of Experiment :

Experiment No : 03

Date : 23/11/2024

Page No. 13

Experiment Result :

create table Reserves (

sid int,

bid int,

date date,

primary key (sid, bid),

foreign key (sid) references Sailors (sid) on update cascade on delete cascade,

foreign key (bid) references Boat (bid) on update cascade on delete cascade

);

-- Insert into Sailors

insert into Sailors (sid, sname, rating, age) values

(601, "John", 4.3, 30),

(602, "James", 4.1, 31),

(603, "Peter", 3.5, 29),

(604, "Rock", 4.2, 26),

(605, "Kevin", 3.9, 28);

-- Insert into Boat

insert into Boat (bid, bname, color) values

(701, 'Boat1', 'Blue'),

(702, 'Boat2', 'Red'),

(703, 'Boat3', 'Green'),

(704, 'Boat4', 'Yellow'),

(705, 'Boat5', 'White');

-- Insert into Reserves

insert into reserves (sid, bid, date) values

(601, 701, "2024-07-01"),

Teacher's Signature :

Name of Experiment : .....

Date : 23/11/2024 .....

Experiment No : 03 .....

Experiment Result : .....

Page No. 14

(602, 702, '2024-05-02'),

(603, 703, '2024-03-04'),

(604, 704, '2024-02-05'),

(605, 705, '2024-05-09');

-- alter the sailor table

alter table Sailors

add email varchar(100);

update Sailors set email = 'john@gmail.com' where sid = 601;

update Sailors set email = 'james@gmail.com' where sid = 602;

update Sailors set email = 'peter@gmail.com' where sid = 603;

update Sailors set email = 'rock@gmail.com' where sid = 604;

update Sailors set email = 'kevin@gmail.com' where sid = 605;

-- check if the table is altered or not

select \* from Sailors;

-- alter the boat table

alter table Boat add model int;

update Boat set model = 1995 where bid IN (701, 702, 703, 704, 705);

-- check if the table is altered or not

select \* from Boat;

-- alter the reserves table

alter table Reserves add departure\_time time;

Teacher's Signature : \_\_\_\_\_

Name of Experiment :.....

Date : 23/11/2024.....

Page No. 15

Experiment No : A3.....

Experiment Result :.....

update Reserves set departure\_time = '10:30:00'

where bid IN (f01, f02, f03, f04, f05);

-- check if the table is updated

select \* from Reserves;

-- adding and dropping constraints

-- ensure that no two sailors have the same email ID

alter table Sailors

add constraint unique\_email unique(email);

-- ensure that no boat have the model year less than 1980

alter table Boat

add constraint check\_model\_year check(model >= 1980);

-- dropping constraints

-- dropping the unique email id constraint

alter table Sailors

drop constraint unique\_email;

-- dropping the model year constraint

-- drop the check constraint from the model column in boat table.

alter table Boat drop constraint check\_model\_year;

-- adding and dropping fields in to the relational schemas

-- adding fields

alter table Boat add location varchar(100);

Teacher's Signature : \_\_\_\_\_

Name of Experiment : .....

Date :.. 23/11/2024 .....

Experiment No :... Q.3 .....

Experiment Result : .....

Page No. 16

-- dropping fields

-- dropping the 'location' field from the Boat table  
alter table Boat drop column location ;

-- delete and update operations

-- update rating of John

update sailors set rating = 4.9 where sid = 601 ;

-- update color of boat 1

update Boat set color = 'Green' where bname = 'Boat 1' ;

-- delete operations

delete from Sailors where sid = 603 ;

delete from Boat where bid = 105 ;

-- update operations

update sailors set sid = 603 where sid = 604 ;

update Boat set bname = 'Boat5' where bname = 'Boat4' ;

Teacher's Signature : \_\_\_\_\_

Output :

1. Sailors table :

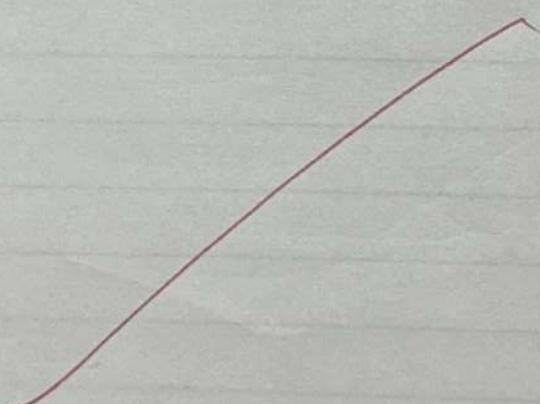
| SID | Sname | Rating | Age |
|-----|-------|--------|-----|
| 601 | John  | 4.3    | 30  |
| 602 | James | 4.1    | 31  |
| 603 | Peter | 3.5    | 29  |
| 604 | Rock  | 4.2    | 26  |
| 605 | Kevin | 3.9    | 28  |

2. Boat table :

| BId | Bname | color  |
|-----|-------|--------|
| f01 | Boat1 | Blue   |
| f02 | Boat2 | Red    |
| f03 | Boat3 | Green  |
| f04 | Boat4 | Yellow |
| f05 | Boat5 | white  |

3. Reserves table :

| Sid | Bid | Date       |
|-----|-----|------------|
| 601 | f01 | 2024-07-01 |
| 602 | f02 | 2024-05-02 |
| 603 | f03 | 2024-03-04 |
| 604 | f04 | 2024-02-05 |
| 605 | f05 | 2024-05-09 |



4. Sailors table after alteration :

| SID | Sname | Rating | Age | Email           |
|-----|-------|--------|-----|-----------------|
| 601 | John  | 4.3    | 30  | john@gmail.com  |
| 602 | James | 4.1    | 31  | james@gmail.com |
| 603 | Peter | 3.5    | 29  | peter@gmail.com |
| 604 | Rock  | 4.2    | 26  | rock@gmail.com  |
| 605 | Kevin | 3.9    | 28  | kevin@gmail.com |

5. Boat table after updating :

| BID | Bname | Color  | Model |
|-----|-------|--------|-------|
| F01 | Boat1 | Blue   | 1995  |
| F02 | Boat2 | Red    | 1995  |
| F03 | Boat3 | Green  | 1995  |
| F04 | Boat4 | Yellow | 1995  |
| F05 | Boat5 | white  | 1995  |

6. Reserves table after updating :

| SID | BID | Date       | Departure-time |
|-----|-----|------------|----------------|
| 601 | F01 | 2024-07-01 | 10:30:00       |
| 602 | F02 | 2024-05-02 | 10:30:00       |
| 603 | F03 | 2024-03-04 | 10:30:00       |
| 604 | F04 | 2024-02-05 | 10:30:00       |
| 605 | F05 | 2024-05-09 | 10:30:00       |

7. Sailors table after updating rating :

| SID | Sname | rating | Age | Email           |
|-----|-------|--------|-----|-----------------|
| 601 | John  | 4.9    | 30  | john@email.com  |
| 602 | James | 4.1    | 31  | james@email.com |
| 603 | Pete  | 3.5    | 29  | pete@email.com  |
| 604 | Rock  | 4.2    | 26  | rock@email.com  |
| 605 | Kevin | 3.9    | 28  | kevin@email.com |

8. Boat table after updating color :

| BID | Bname | Color  | Model |
|-----|-------|--------|-------|
| F01 | Boat1 | Green  | 1995  |
| F02 | Boat2 | Red    | 1995  |
| F03 | Boat3 | Green  | 1995  |
| F04 | Boat4 | Yellow | 1995  |
| F05 | Boat5 | white  | 1995  |

9. Sailors table after deleting sailor :

| SID | Sname | rating | Age | Email           |
|-----|-------|--------|-----|-----------------|
| 601 | John  | 4.9    | 30  | john@email.com  |
| 602 | James | 4.1    | 31  | james@email.com |
| 604 | Rock  | 4.2    | 26  | rock@email.com  |
| 605 | Kevin | 3.9    | 28  | kevin@email.com |

10. Boat table after deleting boat :

| BID | Bname | Color  | Model |
|-----|-------|--------|-------|
| F01 | Boat1 | Green  | 1995  |
| F02 | Boat2 | Red    | 1995  |
| F03 | Boat3 | Green  | 1995  |
| F04 | Boat4 | Yellow | 1995  |

11. Reserves table after deleting :

| SID | BID | Date       | departure-time |
|-----|-----|------------|----------------|
| 601 | f01 | 2024-07-01 | 10:30:00       |
| 602 | f02 | 2024-05-02 | 10:30:00       |
| 604 | f04 | 2024-02-05 | 10:30:00       |

Schema Diagram :

Sailors

| <u>SID</u> | Sname | Rating | Age | Email |
|------------|-------|--------|-----|-------|
|            |       |        |     |       |

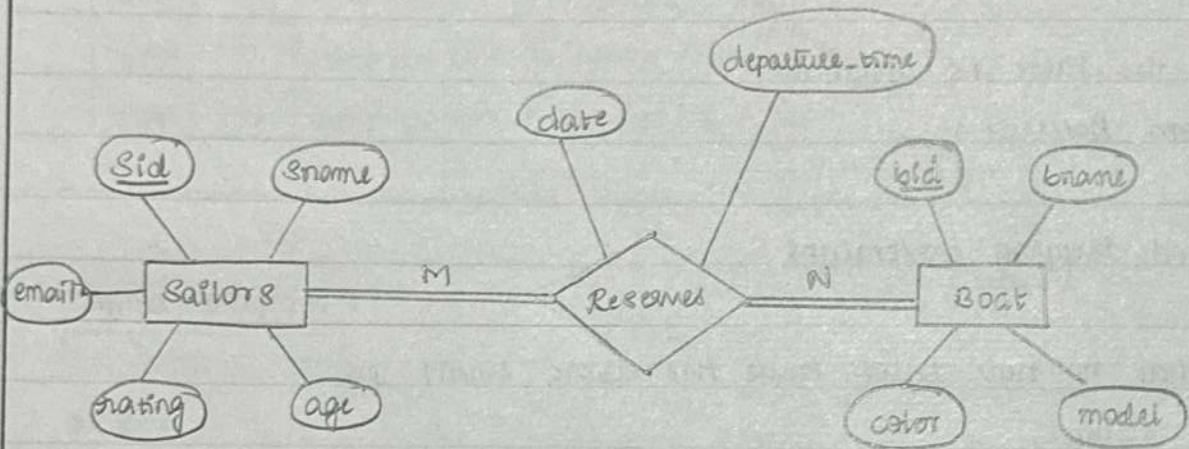
Boat

| <u>BID</u> | Bname | color | model |
|------------|-------|-------|-------|
|            |       |       |       |

Reserves

| <u>SID</u> | <u>BID</u> | Date | departure-time |
|------------|------------|------|----------------|
|            |            |      |                |

ER diagram :



B. Insurance database :

PERSON (driver\_id# : string, name : string, address : string)

CAR (regno : string, model : string, year : int)

ACCIDENT (report\_number : int, acc\_date : date, location : string)

OWNS (driver\_id# : string, regno : string)

PARTICIPATED (driver\_id# : string, regno : string, report\_number : int, damage\_amount : int)

Create database Insurance;

use Insurance;

Create table Person (

driver\_id varchar(50) primary key,

name varchar(100),

address varchar(100)

);

Create table CAR (

regno varchar(20) primary key,

model varchar(50),

year int

);

Create table Accident (

report\_number int primary key,

acc\_date date,

location varchar(100)

);

Name of Experiment :.....

Date :..30/11/2024.....

Page No. 18

Experiment No :..... 4.....

Experiment Result :.....

```
create table Owns (
    driver_id varchar(50),
    regno varchar(20),
    foreign key (driver_id) references Person (driver_id) on update cascade on delete cascade,
    foreign key (regno) references car (regno) on update cascade on delete cascade
);

create table Participated (
    driver_id varchar(50),
    regno varchar(20),
    report_number int,
    damage_amount int,
    foreign key (driver_id) references Person (driver_id) on update cascade, on delete cascade,
    foreign key (regno) references car (regno) on update cascade on delete cascade,
    foreign key (report_number) references Accident (report_number) on update cascade, on delete cascade
);
```

insert into Person values

("D111", "Driver-1", "Kuvempunagar, Mysuru"),  
("D222", "Smith", "JP Nagar, Mysuru"),  
("D333", "Driver-3", "Vdaygiri, Mysuru"),  
("D444", "Driver-4", "Lajvannagar, Mysuru"),  
("D555", "Driver-5", "Vijaynagar, Mysuru");

insert into car values

("KA-20-AB-4223", "Swift", 2020),  
("KA-20-Bc-5674", "Mazda", 2017),

Teacher's Signature : \_\_\_\_\_

Name of Experiment :.....

Date :.. 30/11/2024 .....

Experiment No : 4 .....

Experiment Result :.....

Page No. 19

( "KA-21-AC-5473", "Alto", 2015),

( "KA-21-BD-4728", "Tribes", 2019),

( "KA-09-MA-1234", "Tiago", 2018);

insert into Accident values

( 43627 , "2020-04-05" , "Nashabad, Mysuru"),

( 56345 , "2019-12-16" , "Gokulam, Mysuru"),

( 63744 , "2020-05-14" , "Vijaynagar, Mysuru").

( 54634 , "2019-08-30" , "Kuvempunagar, Mysuru"),

( 65738 , "2021-01-21" , "JSS Layout, Mysuru");

insert into Car values

( "D111" , "KA-20-AB-4223"),

( "D222" , "KA-20-BC-5674"),

( "D333" , "KA-21-AC-5473"),

( "D444" , "KA-21-BD-4728"),

( "D555" , "KA-09-MA-1234");

insert into Person values

( "D111" , "KA-20-AB-4223" , 43627 , 20000),

( "D222" , "KA-20-BC-5674" , 56345 , 49500),

( "D333" , "KA-21-AC-5473" , 63744 , 15000),

( "D444" , "KA-21-BD-4728" , 54634 , 5000),

( "D555" , "KA-09-MA-1234" , 65738 , 25000);

select \* from Person;

select \* from car;

select \* from Accident;

Teacher's Signature : \_\_\_\_\_

Name of Experiment :.....

Date : 20/11/2024 .....

Experiment No : 4 .....

Page No. 20

Experiment Result :.....

Select \* from Owns ;

Select \* from Participated ;

1. Find the total number of people who owned a car that were involved in accidents in 2021

Select COUNT (driver\_id)

from Participated p, Accident a

where p.report-number = a.report-number and a.acc-date like "2021%";

2. Find the number of accident in which cars belonging to Smith were involved.

Select COUNT (distinct a.report-number)

from Accident a

where exists (Select \* from Person p, participated ptd

where p.driver\_id = ptd.driver\_id and p.name = "Smith" and

a.report-number = ptd.report-number);

3. Add a new accident to the database

insert into Accident values (45562, "2024-04-05", "Mandya");

insert into Participated values ("D222", "KA-21-BD-4728", 45562, 50000);

4. Delete the Mazda belonging to Smith.

delete from car

where model = "Mazda" and regno in (select car.regno

from Person p, Owns o

where p.driver\_id = o.driver\_id and

o.regno = car.regno and p.name =  
"Smith");

Teacher's Signature : \_\_\_\_\_

Name of Experiment : .....

Date : 30/11/2024 .....

Page No. 21

Experiment No : 04 .....

Experiment Result : .....

5. Update the damage amount for the car with regno of KA-09-MA-1234 in the accident with report number 65738.

Update Participated

set damage\_amount = 1000

where report\_number = 65738 and regno = "KA-09-MA-1234";

6. View that shows models and years of car that are involved in accident

create view CarsInAccident as

select distinct model, year

from car c, participated p

where c.regno = p.regno ;

7. Create a view that shows name and address of drivers who own a car.

create view DriversWithCar as

select name, address

from Person p, Owns o

where p.driver\_id = o.driver\_id ;

select \* from DriversWithCar ;

8. Create a view that shows the names of the drivers who participated in an accident in a specific place.

create view DriversWithAccidentInPlace as

select name

from Person p, Accident a, Participated ptd

where p.driver\_id = ptd.driver\_id and a.report\_number = ptd.report\_number

and a.location = "Vijaynagar, Mysore" ;

(D) 12/11/24

Teacher's Signature : \_\_\_\_\_

### Output :

#### Person table :

| driver_id | name     | address              |
|-----------|----------|----------------------|
| D111      | Driver-1 | Kuvempunagar, Mysuru |
| D222      | Smith    | JP Nagar, Mysuru     |
| D333      | Driver-3 | Udaygiri, Mysuru     |
| D444      | Driver-4 | Rajivnagar, Mysuru   |
| D555      | Driver-5 | Vijaynagar, Mysuru   |

#### Car table :

| regno         | model  | year |
|---------------|--------|------|
| KA-09-MA-1234 | Tiago  | 2018 |
| KA-20-AB-4223 | Swift  | 2020 |
| KA-20-BC-5674 | Mazda  | 2017 |
| KA-21-AC-5473 | Alto   | 2015 |
| KA-21-BD-4728 | Triber | 2019 |

#### Accident table :

| report-number | acc-date   | location             |
|---------------|------------|----------------------|
| 43627         | 2020-04-05 | Nazarbad, Mysuru     |
| 54634         | 2019-08-30 | Kuvempunagar, Mysuru |
| 56345         | 2019-12-16 | Gokulam, Mysuru      |
| 63744         | 2020-05-14 | Vijaynagar, Mysuru   |
| 65738         | 2021-01-21 | JSS Layout, Mysuru   |

#### Overs table :

| driver_id | regno         |
|-----------|---------------|
| D111      | KA-20-AB-4223 |
| D222      | KA-20-BC-5674 |
| D333      | KA-21-AC-5473 |
| D444      | KA-21-BD-4728 |
| D555      | KA-09-MA-1234 |

Participated table :

| driver-id | regno               | report-number | damage-amount |
|-----------|---------------------|---------------|---------------|
| D111      | KA - 20 - AB - 4223 | 43627         | 20000         |
| D222      | KA - 20 - BC - 5674 | 56345         | 49500         |
| D333      | KA - 21 - AC - 5473 | 63744         | 15000         |
| D444      | KA - 21 - BD - 4728 | 54634         | 5000          |
| D555      | KA - 09 - MA - 1234 | 65738         | 25000         |

1.  $\text{count}(\text{driver-id})$

1

2.  $\text{count}(\text{distinct a.report-number})$

2

3. Accident table :

| report-number | acc-date   | location             |
|---------------|------------|----------------------|
| 43627         | 2020-04-05 | Nazarbad, Mysuru     |
| 54634         | 2019-08-30 | Kuvempunagar, Mysuru |
| 56345         | 2019-12-16 | Gokulam, Mysuru      |
| 63744         | 2020-05-14 | Vijaynagar, Mysuru   |
| 65738         | 2021-01-21 | JSS layout, Mysuru   |
| 45562         | 2024-04-05 | Mandyra              |

Participated table :

| driver-id | regno               | report-number | damage-amount |
|-----------|---------------------|---------------|---------------|
| D111      | KA - 20 - AB - 4223 | 43627         | 20000         |
| D222      | KA - 20 - BC - 5674 | 56345         | 49500         |
| D333      | KA - 21 - AC - 5473 | 63744         | 15000         |
| D444      | KA - 21 - BD - 4728 | 54634         | 5000          |
| D222      | KA - 09 - MA - 1234 | 65738         | 25000         |
| D222      | KA - 21 - BD - 4728 | 45562         | 50000         |

A. Car tables

| regno         | model  | year |
|---------------|--------|------|
| KA-09-MA-1234 | Tiago  | 2018 |
| KA-20-AB-4223 | Swift  | 2020 |
| KA-21-BC-5473 | Alto   | 2015 |
| KA-21-BD-4728 | Tribel | 2019 |

Owner tables:

| driver_id | regno         |
|-----------|---------------|
| D111      | KA-20-AB-4223 |
| D333      | KA-21-AC-5473 |
| D444      | KA-21-BD-4728 |
| D555      | KA-09-MA-1234 |

Participated table 6

| driver_id | regno         | report-number | damage-amount |
|-----------|---------------|---------------|---------------|
| D111      | KA-20-AB-4223 | 43627         | 20000         |
| D333      | KA-21-AC-5473 | 63744         | 15000         |
| D444      | KA-21-BD-4728 | 54634         | 5000          |
| D222      | KA-09-MA-1234 | 65738         | 25000         |
| D222      | KA-21-BD-4728 | 45562         | 50000         |

5. Participated table:

| driver_id | regno         | report-number | damage-amount |
|-----------|---------------|---------------|---------------|
| D111      | KA-20-AB-4223 | 43627         | 20000         |
| D333      | KA-21-AC-5473 | 63744         | 15000         |
| D444      | KA-21-BD-4728 | 54634         | 5000          |
| D222      | KA-09-MA-1234 | 65738         | 1000          |
| D222      | KA-21-BD-4728 | 45562         | 50000         |

6. Cars In Accident table :

| model  | year |
|--------|------|
| Trago  | 2018 |
| Swift  | 2020 |
| Astro  | 2015 |
| Tribal | 2019 |

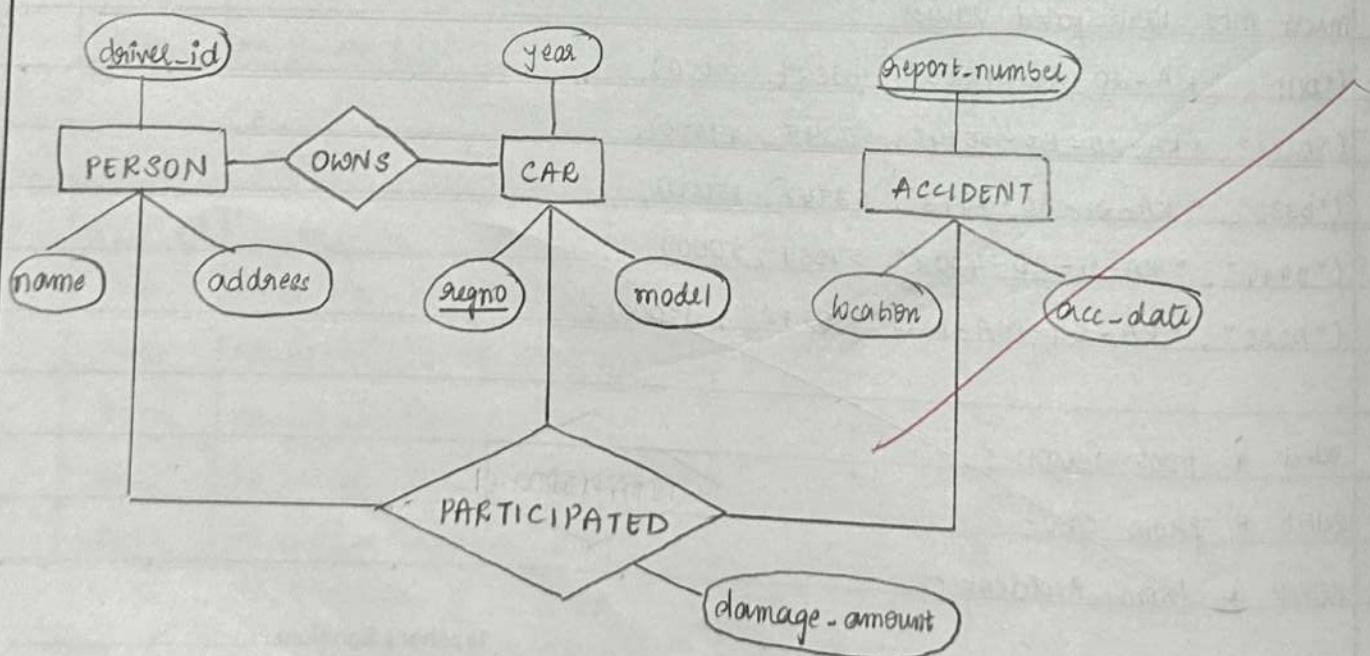
7. DriversWithCar table :

| name     | address              |
|----------|----------------------|
| Driver-1 | Kurempunagar, Mysore |
| Driver-3 | Vdaygiri, Mysore     |
| Driver-4 | Rajivnagar, Mysore   |
| Driver-5 | Vijaynagar, Mysore   |

8. DriversWithAccidentInPlace table :

| name     |
|----------|
| Driver-3 |

ER diagram :



schema diagram:

Person

|           |      |         |
|-----------|------|---------|
| driver_id | name | address |
|-----------|------|---------|



CAR

|       |       |      |
|-------|-------|------|
| regno | model | year |
|-------|-------|------|



Accident

|               |          |          |
|---------------|----------|----------|
| report-number | acc-date | location |
|---------------|----------|----------|



Owns

|           |       |
|-----------|-------|
| driver_id | regno |
|-----------|-------|

Participated

|           |       |               |               |
|-----------|-------|---------------|---------------|
| driver_id | regno | report-number | damage-amount |
|-----------|-------|---------------|---------------|

Name of Experiment :.....

Date :.....

Experiment No :.....

Experiment Result :.....

Page No.

C. Order processing database

Customer (cust# : int, cname = string, city : string)

Order (order# : int, odate = date, order-amt = int, cust# = int)

Order-item (Order # : int, item# : int, qty : int)

Item (item# : int, unitprice = int)

Shipment (order# : int, warehouse# = int, ship-date = date)

Warehouse (warehouse# : int, city : string)

Create database order-processing;

use order-processing;

Create table if not exists Customers (

cust-id int primary key,

cname varchar(35) not null,

city varchar(35) not null

);

Create table if not exists Orders (

order-id int primary key,

odate date not null,

order-amt int not null,

foreign key (cust-id) references Customers (cust-id) on delete cascade on

update cascade

);

Create table if not exists Items (

item-id int primary key,

unitprice int not null

);

Teacher's Signature : \_\_\_\_\_

Name of Experiment :.....

Date :.....

Experiment No :.....

Experiment Result :.....

Page No.

Create table if not exists OrderItems (

qty int not null,

order\_id int not null,

item\_id int not null,

foreign key (order\_id) references Orders (order\_id) on update cascade on delete cascade,

foreign key (item\_id) references Items (item\_id) on update cascade on delete cascade

);

Create table if not exists warehouses (

warehouse\_id int primary key,

city varchar (35) not null

);

Create table if not exists Shipments (

ship\_date date not null,

order\_id int not null,

warehouse\_id int not null,

foreign key (order\_id) references Orders (order\_id) on update cascade on delete cascade,

foreign key (warehouse\_id) references Warehouses (warehouse\_id) on update cascade on delete cascade

);

insert into customers values

(0001, "Ravi", "Mysuru"),

(0002, "Ram", "Bengaluru"),

(0003, "Tumai", "Mumbai"),

(0004, "Kiran", "Delhi"),

(0005, "Chetan", "Bengaluru");

Teacher's Signature : \_\_\_\_\_

Name of Experiment :.....

Date :.....

Page No.

Experiment No :.....

Experiment Result :.....

insert into OrderLine values

(001 , "2020-01-14" , 0001 , 2000),  
(002 , "2021-04-13" , 0002 , 500),  
(003 , "2019-10-02" , 0003 , 2500),  
(004 , "2019-05-12" , 0005 , 1000),  
(005 , "2020-12-13" , 0007 , 1200);

insert into Items values

(0001 , 400),  
(0002 , 200),  
(0003 , 1000),  
(0004 , 100),  
(0005 , 500);

insert into Warehouses values

(0001 , "Mysuru"),  
(0002 , "Bengaluru"),  
(0003 , "Mumbai"),  
(0004 , "Delhi").  
(0005 , "Chennai");

insert into ordertems values

(001 , 0001 , 5),  
(002 , 0005 , 1),  
~~(003 , 0005 , 5),~~  
(004 , 0003 , 1),  
(005 , 0004 , 12);

Teacher's Signature : \_\_\_\_\_

Name of Experiment : .....

Date : .....

Experiment No : .....

Experiment Result : .....

Page No.

Insert into shipments values

(001, 0002, "2020-01-16"),

(002, 0001, "2021-04-14"),

(003, 0004, "2019-10-07"),

(004, 0003, "2019-05-16"),

(005, 0005, "2020-12-23");

Select \* from customers;

Select \* from orders;

Select \* from orderitems;

Select \* from items;

Select \* from shipments;

Select \* from warehouses;

1. List the order# and ship-date for all orders shipped from warehouse# "0001"

Select order-id, ship-date

from shipments

where warehouse-id = 0001;

2. List the warehouse information from which the customer named "Kumar" supplied his orders. Produce a listing of Order#, warehouse#

Select order-id, warehouse-id

from warehouses natural join shipments

where order-id in (Select order-id

from orders

where cust-id in (Select cust-id

from customers

where cname like "% Kumar %") ;

Teacher's Signature : \_\_\_\_\_

Name of Experiment : .....

Experiment No : .....

Date : .....

Experiment Result : .....

Page No.

3. Produce a listing of cname , #ofOrder , Avg\_Order\_Amt where the middle column is the total number of orders by the customer and the last column is the average order amount for that customer (use aggregate functions).
- Select cname , COUNT(\*) as no-of-orders , AVG(order-amount) as avg-order-amount  
from customers c , Orders o  
where c.cust-id = o.cust-id group by cname ;

4. Find the item with the maximum unit price.

select max(unitprice)  
from items ;

5. Create a view to display orderid and shipment date of all orders shipped from a warehouse 2.

Create view ShipmentDateFromWarehouse2 as

select order-id , shipdate  
from shipments

where warehouse-id = 2 ;

select \* from ShipmentDateFromWarehouse2 ;

6. A view that shows the warehouse ids from where the kumar's orders are being shipped.

Create view WarehouseWithKumarOrders as

select s.warehouse-id

from warehouse w , customers c , Orders o , Shipment s

where w.warehouse-id = s.warehouse-id and s.order-id = o.order-id and

o.cust-id = c.cust-id and c.cname = "kumar" ;

select \* from WarehouseWithKumarOrders ;

Teacher's Signature : \_\_\_\_\_

Name of Experiment : .....

Date : .....

Experiment No : .....

Experiment Result : .....

Page No.

7. Delete all orders for customer named "Kumar".

delete from orders

where cust\_id = (select cust\_id

from customers

where cname like "% Kumar %");

~~JN  
28/12/20~~

Teacher's Signature : \_\_\_\_\_

Output:

Customer table :

| cust-id | cname  | city      |
|---------|--------|-----------|
| 1       | Ravi   | Mysuru    |
| 2       | Ram    | Bengaluru |
| 3       | Kumar  | Mumbai    |
| 4       | Firan  | Delhi     |
| 5       | Chetan | Bengaluru |

Item table :

| item-id | unitprice |
|---------|-----------|
| 1       | 400       |
| 2       | 200       |
| 3       | 1000      |
| 4       | 100       |
| 5       | 500       |

Orders table :

| order-id | odate      | cust-id | order-amt |
|----------|------------|---------|-----------|
| 1        | 2020-01-14 | 1       | 2000      |
| 2        | 2021-04-13 | 2       | 500       |
| 3        | 2019-10-02 | 3       | 2500      |
| 4        | 2019-05-12 | 5       | 1000      |
| 5        | 2020-12-13 | 4       | 1200      |

Shipments table :

| order-id | warehouse-id | ship-date  |
|----------|--------------|------------|
| 1        | 2            | 2020-01-16 |
| 2        | 1            | 2021-04-14 |
| 3        | 4            | 2019-10-07 |
| 4        | 3            | 2019-05-16 |
| 5        | 5            | 2020-12-23 |

Order-items table:

| order-id | item-id | qty |
|----------|---------|-----|
| 1        | 1       | 5   |
| 2        | 5       | 1   |
| 3        | 5       | 5   |
| 4        | 3       | 1   |
| 5        | 4       | 12  |

Warehouses table:

| warehouse-id | city      |
|--------------|-----------|
| 1            | Mysuru    |
| 2            | Bengaluru |
| 3            | Mumbai    |
| 4            | Delhi     |
| 5            | Chennai   |

1.

| order-id | ship-date  |
|----------|------------|
| 2        | 2021-04-14 |

2.

| order-id | warehouse-id |
|----------|--------------|
| 3        | 4            |

3.

| cname  | no-of-orders | avg-order-amt |
|--------|--------------|---------------|
| Ram    | 1            | 2000.0000     |
| Ram    | 1            | 500.0000      |
| Kumar  | 1            | 2500.0000     |
| Akran  | 1            | 1200.0000     |
| chetan | 1            | 1000.0000     |

4.

|                 |
|-----------------|
| max(unit price) |
| 1000            |

5.

| order-id | ship-date  |
|----------|------------|
| 1        | 2020-01-16 |

6.

|               |
|---------------|
| warehouse-id. |
| 4             |

7. orders table

| order-id | odate      | cust-id | order-amt |
|----------|------------|---------|-----------|
| 1        | 2020-01-14 | 1       | 2000      |
| 2        | 2021-04-13 | 2       | 500       |
| 4        | 2019-05-12 | 5       | 1000      |
| 5        | 2020-12-13 | 4       | 1200      |

OrderItems table:

| order-id | item-id | qty |
|----------|---------|-----|
| 1        | 1       | 5   |
| 2        | 5       | 1   |
| 4        | 3       | 1   |
| 5        | 4       | 12  |

Shipments table

| order-id | warehouse-id | ship-date  |
|----------|--------------|------------|
| 1        | 2            | 2020-01-16 |
| 2        | 1            | 2021-04-14 |
| 4        | 3            | 2019-05-16 |
| 5        | 5            | 2020-12-23 |

### schema diagram

