**Prince Chaudhary**

**1NT22CS143**

**2.Create a relational database schema for a Project, described by the following relations. STUDENT (Rollno: integer, Name: String, Sem: integer, Degree: String, Contact no: integer, Guide\_No: integer) GUIDE (Guide\_name: String, Guide\_No: integer, Guide\_reserach\_domain: String, Contact\_No: integer, Email\_Id: String) PROJECT (Project\_No: Integer, Project\_title: String, Project\_Area: String, Start\_dt, date, Guide\_No: integer) GROUP (Group\_Code: integer, Roll\_No:integer ) PROJECT\_GROUP (Group\_Code: integer, Project\_No: integer, no\_of\_students:integer) For the above schema, perform the following.**

**a) Create the tables with the appropriate integrity constraints b) Insert around 10 records in each of the tables**

**c) Find the list of guide, who are guiding more than two student groups.**

**d) Find the list of project no, project name & name of guide, in domain of Data Base.**

**e) Create a view as student\_project details that lists student name, project name and guide name**

query:

create database studentdatabase;

use studentdatabase;

CREATE TABLE GUIDE (

Guide\_No INTEGER PRIMARY KEY,

Guide\_name VARCHAR(100),

Guide\_reserach\_domain VARCHAR(100) ,

Contact\_No INTEGER ,

Email\_Id VARCHAR(100)

);

CREATE TABLE STUDENT (

Rollno INTEGER PRIMARY KEY,

Name VARCHAR(100),

Sem INTEGER,

Degree VARCHAR(50) ,

Contact\_no VARCHAR(50) ,

Guide\_No INTEGER,

FOREIGN KEY (Guide\_No) REFERENCES GUIDE(Guide\_No)

);

CREATE TABLE PROJECT (

Project\_No INTEGER PRIMARY KEY,

Project\_title VARCHAR(200) ,

Project\_Area VARCHAR(100) ,

Start\_dt DATE ,

Guide\_No INTEGER,

FOREIGN KEY (Guide\_No) REFERENCES GUIDE(Guide\_No)

);

CREATE TABLE STUDENT\_GROUP (

Group\_Code INTEGER,

Roll\_No INTEGER,

PRIMARY KEY (Group\_Code, Roll\_No),

FOREIGN KEY (Roll\_No) REFERENCES STUDENT(Rollno)

);

CREATE TABLE PROJECT\_GROUP (

Group\_Code INTEGER,

Project\_No INTEGER,

no\_of\_students INTEGER,

PRIMARY KEY (Group\_Code, Project\_No),

FOREIGN KEY (Group\_Code) REFERENCES STUDENT\_GROUP(Group\_Code),

FOREIGN KEY (Project\_No) REFERENCES PROJECT(Project\_No)

);

-- Insert records into GUIDE

INSERT INTO GUIDE (Guide\_No, Guide\_name, Guide\_reserach\_domain, Contact\_No, Email\_Id) VALUES

(1, 'Dr. Smith', 'Data Base', 1234567890, 'smith@example.com'),

(2, 'Dr. Johnson', 'Machine Learning', 1234567891, 'johnson@example.com'),

(3, 'Dr. Williams', 'Data Base', 1234567892, 'williams@example.com'),

(4, 'Dr. Brown', 'Networking', 1234567893, 'brown@example.com');

ALTER TABLE STUDENT MODIFY Contact\_no varchar(100);

-- Insert records into STUDENT

INSERT INTO STUDENT (Rollno, Name, Sem, Degree, Contact\_no, Guide\_No) VALUES

(1, 'Alice', 6, 'B.Tech', 9876543210, 1),

(2, 'Bob', 6, 'B.Tech', 9876543211, 2),

(3, 'Charlie', 6, 'B.Tech', 9876543212, 1),

(4, 'David', 6, 'B.Tech', 9876543213, 3),

(5, 'Eve', 6, 'B.Tech', 9876543214, 1),

(6, 'Frank', 6, 'B.Tech', 9876543215, 4),

(7, 'Grace', 6, 'B.Tech', 9876543216, 2),

(8, 'Hank', 6, 'B.Tech', 9876543217, 3),

(9, 'Ivy', 6, 'B.Tech', 9876543218, 1),

(10, 'Jack', 6, 'B.Tech', 9876543219, 4);

-- Insert records into PROJECT

INSERT INTO PROJECT (Project\_No, Project\_title, Project\_Area, Start\_dt, Guide\_No) VALUES

(1, 'Database Optimization', 'Data Base', '2023-01-01', 1),

(2, 'AI in Healthcare', 'Machine Learning', '2023-02-01', 2),

(3, 'Network Security', 'Networking', '2023-03-01', 4),

(4, 'Big Data Analytics', 'Data Base', '2023-04-01', 3),

(5, 'Deep Learning Models', 'Machine Learning', '2023-05-01', 2);

-- Insert records into STUDENT\_GROUP

INSERT INTO STUDENT\_GROUP (Group\_Code, Roll\_No) VALUES

(1, 1), (1, 2), (2, 3), (2, 4), (3, 5), (3, 6), (4, 7), (4, 8), (5, 9), (5, 10);

-- Insert records into PROJECT\_GROUP

INSERT INTO PROJECT\_GROUP (Group\_Code, Project\_No, no\_of\_students) VALUES

(1, 1, 2),

(2, 2, 2),

(3, 3, 2),

(4, 4, 2),

(5, 5, 2);

SELECT g.Guide\_name

FROM GUIDE g

JOIN PROJECT p ON g.Guide\_No = p.Guide\_No

JOIN PROJECT\_GROUP pg ON p.Project\_No = pg.Project\_No

GROUP BY g.Guide\_name

HAVING COUNT(DISTINCT pg.Group\_Code) > 2;

SELECT p.Project\_No, p.Project\_title, g.Guide\_name

FROM PROJECT p

JOIN GUIDE g ON p.Guide\_No = g.Guide\_No

WHERE p.Project\_Area = 'Data Base';

CREATE VIEW student\_project\_details AS

SELECT s.Name AS Student\_Name, p.Project\_title AS Project\_Name, g.Guide\_name AS Guide\_Name

FROM STUDENT s

JOIN STUDENT\_GROUP sg ON s.Rollno = sg.Roll\_No

JOIN PROJECT\_GROUP pg ON sg.Group\_Code = pg.Group\_Code

JOIN PROJECT p ON pg.Project\_No = p.Project\_No

JOIN GUIDE g ON p.Guide\_No = g.Guide\_No;

SELECT \* FROM student\_project\_details;

Screenshots: