PRACTICAL ASSIGNMENT- 7

* Name: Hardik Arora
* Branch : Btech - CS
* Program: AIML
* University Roll No. : 2215500071
* Section: 2AC
* Class Roll No. : 28

CREATE DATABASE P6;

USE P6;

CREATE TABLE IF NOT EXISTS Student (

    sID INT PRIMARY KEY,

    sName VARCHAR(50),

    GPA FLOAT,

    sizeHS INT NOT NULL,

    DoB VARCHAR(50)

);

INSERT INTO student(sID, sName, GPA, sizeHS, DoB) VALUES ('123', 'Amy', '3.9', '1000', '1996-06-26');

INSERT INTO student(sID, sName, GPA, sizeHS, DoB) VALUES ('234', 'Bob', '3.6', '1500', '1995-04-07');

INSERT INTO student(sID, sName, GPA, sizeHS, DoB) VALUES ('345', 'Craig', '3.5', '500', '1995-02-04');

INSERT INTO student(sID, sName, GPA, sizeHS, DoB) VALUES ('456', 'Doris', '3.9', '1000', '1997-07-24');

INSERT INTO student(sID, sName, GPA, sizeHS, DoB) VALUES ('567', 'Edward', '2.9', '2000', '1996-12-21');

INSERT INTO student(sID, sName, GPA, sizeHS, DoB) VALUES ('678', 'Fay', '3.8', '200', '1996-08-27');

INSERT INTO student(sID, sName, GPA, sizeHS, DoB) VALUES ('789', 'Gary', '3.4', '800', '1996-10-08');

INSERT INTO student(sID, sName, GPA, sizeHS, DoB) VALUES ('987', 'Helen', '3.7', '800', '1997-03-27');

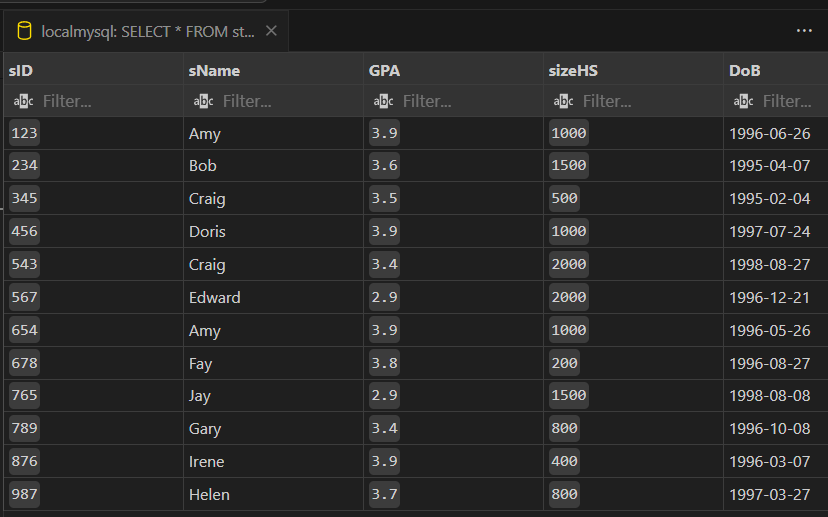
INSERT INTO student(sID, sName, GPA, sizeHS, DoB) VALUES ('876', 'Irene', '3.9', '400', '1996-03-07');

INSERT INTO student(sID, sName, GPA, sizeHS, DoB) VALUES ('765', 'Jay', '2.9', '1500', '1998-08-08');

INSERT INTO student (sID, sName, GPA, sizeHS, DoB) VALUES ('654', 'Amy', '3.9', '1000', '1996-05-26');

INSERT INTO student (sID, sName, GPA, sizeHS, DoB) VALUES ('543', 'Craig', '3.4', '2000', '1998-08-27');

SELECT \* FROM student;



CREATE TABLE IF NOT EXISTS College(

    cName VARCHAR(50) PRIMARY KEY,

    State VARCHAR(50),

    enrollment INT NOT NULL

);

INSERT INTO college(cName, State, enrollment) VALUES('Stanford', 'CA', '15000');

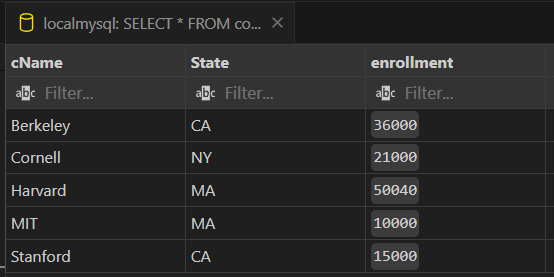
INSERT INTO college(cName, State, enrollment) VALUES('Berkeley', 'CA', '36000');

INSERT INTO college(cName, State, enrollment) VALUES('MIT', 'MA', '10000');

INSERT INTO college(cName, State, enrollment) VALUES('Cornell', 'NY', '21000');

INSERT INTO college(cName, State, enrollment) VALUES('Harvard', 'MA', '50040');

SELECT \* FROM college;



CREATE TABLE IF NOT EXISTS Applied(

    sID INT NOT NULL,

    cName VARCHAR(50) NOT NULL,

    major VARCHAR(50) NOT NULL,

    decision VARCHAR(1) NOT NULL

);

INSERT INTO Applied(sID, cName, major, decision) VALUES('123', 'Stanford', 'CS', 'Y');

INSERT INTO Applied(sID, cName, major, decision) VALUES('123', 'Stanford', 'EE', 'N');

INSERT INTO Applied(sID, cName, major, decision) VALUES('123', 'Berkeley', 'CS', 'Y');

INSERT INTO Applied(sID, cName, major, decision) VALUES('123', 'Cornell', 'EE', 'Y');

INSERT INTO Applied(sID, cName, major, decision) VALUES('234', 'Berkeley', 'biology', 'N');

INSERT INTO Applied(sID, cName, major, decision) VALUES('345', 'MIT', 'bioengineering', 'Y');

INSERT INTO Applied(sID, cName, major, decision) VALUES('345', 'Cornell', 'bioengineering', 'N');

INSERT INTO Applied(sID, cName, major, decision) VALUES('345', 'Cornell', 'CS', 'Y');

INSERT INTO Applied(sID, cName, major, decision) VALUES('345', 'Cornell', 'EE', 'N');

INSERT INTO Applied(sID, cName, major, decision) VALUES('678', 'Stanford', 'history', 'Y');

INSERT INTO Applied(sID, cName, major, decision) VALUES('987', 'Stanford', 'CS', 'Y');

INSERT INTO Applied(sID, cName, major, decision) VALUES('987', 'Berkeley', 'CS', 'Y');

INSERT INTO Applied(sID, cName, major, decision) VALUES('876', 'Stanford', 'CS', 'N');

INSERT INTO Applied(sID, cName, major, decision) VALUES('876', 'MIT', 'biology', 'Y');

INSERT INTO applied(sID, cName, major, decision) VALUES('876', 'MIT', 'marine biology', 'N');

INSERT INTO Applied(sID, cName, major, decision) VALUES('765', 'Stanford', 'history', 'Y');

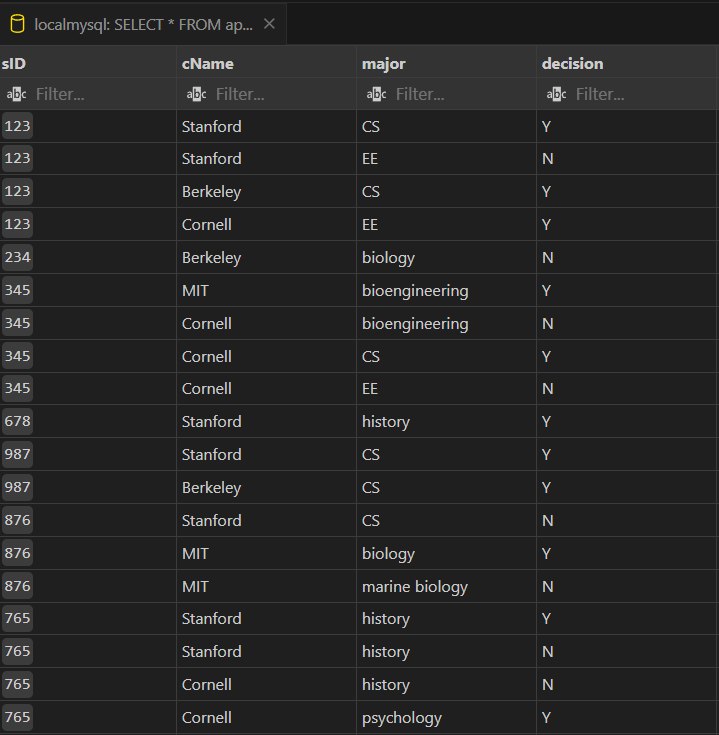
INSERT INTO applied(sID, cName, major, decision) VALUES('765', 'Stanford', 'history', 'N');

INSERT INTO applied(sID, cName, major, decision) VALUES('765', 'Cornell', 'history', 'N');

INSERT INTO applied(sID, cName, major, decision) VALUES('765', 'Cornell', 'psychology', 'Y');

INSERT INTO applied(sID, cName, major, decision) VALUES('543', 'MIT', 'CS', 'N');

SELECT \* FROM applied;



# Write SQL queries for the following:

-- Q1. Create view WeakStudent on Student whose GPA is less than 3.7.

CREATE VIEW WeakStudent AS

SELECT \*

FROM Student

WHERE GPA < 3.7;

-- Q2. Create a view cView on college (containing all the columns) and rename the column cName as collegeName and enrollment as seats respectively.

CREATE VIEW cView AS

SELECT cName AS collegeName, enrollment AS seats, state, enrollment

FROM college;

-- Q3. Create view CSaccept having IDs and college of students who applied to CS major and their application is accepted.

CREATE VIEW CSaccept AS

SELECT sID, cName

FROM Apply

WHERE major = 'CS' AND decision = 'Y';

-- Q4. Create view CSberkeley having IDs, name, GPA, sizeHS of those students who are accepted in CS at Berkeley and come from a High School with more than 500 students.

CREATE VIEW CSberkeley AS

SELECT s.sID, s.sName, s.GPA, s.sizeHS

FROM Student s

JOIN CSaccept c ON s.sID = c.sID

WHERE c.cName = 'Berkeley' AND s.sizeHS > 500;

-- Q5. Display information about students in CSberk having GPA greater than 3.8.

SELECT \*

FROM CSberkeley

WHERE GPA > 3.8;

-- Q6. Drop view CSaccept.

DROP VIEW CSaccept;

-- Q7. Display all students in CSberkeley.

SELECT \*

FROM CSberkeley;

-- Q8. Update GPA by 0.8 of students in view WeakStudent who have a high school size greater than 1000.

UPDATE Student

SET GPA = GPA + 0.8

WHERE sID IN (SELECT sID FROM WeakStudent WHERE sizeHS > 1000);

-- Q9. Create a view AppCount which contains sID of Student and the number of applications they filed. Name the column sID and NoOfApp.

CREATE VIEW AppCount AS

SELECT sID, COUNT(\*) AS NoOfApp

FROM Apply

GROUP BY sID;

-- Q10. Update NoOfApp so that sID 234 contains 4 applications.

UPDATE AppCount

SET NoOfApp = 4

WHERE sID = 234;

-- Q11. Create a view StuName containing student names and their GPA. Is this View Updatable? If not, specify why.

CREATE VIEW StuName AS

SELECT sName, GPA

FROM Student;

-- This view is not updatable because it contains columns from the Student table that are not unique, making it ambiguous for the database to determine which records to update if changes are made.

-- Q12. Create view studentHS having details of students who come from a High School of size more than 1000 using with check option.

CREATE VIEW studentHS AS

SELECT \*

FROM Student

WHERE sizeHS > 1000

WITH CHECK OPTION;

-- Q13. Try to insert details of a new student Ram with sID 999 having GPA 9.9 and sizeHS 9999 to the newly created view studentHS. (Comment on whether this view is updatable.)

-- This insert will fail because the check option is set, and the sizeHS value for Ram (9999) does not meet the condition specified in the view (sizeHS > 1000). Therefore, the view is not updatable in this context.

-- Q14. Clerk realizes he wrongly typed sizeHS of Ramu as 9999; it is actually 999. Can you help him update the value of sizeHS of Ramu?

UPDATE studentHS

SET sizeHS = 999

WHERE sID = 999;

-- Q15. Now, another boy registered in our system named Ramu with sID 998 having GPA 9.8 and sizeHS 989.

INSERT INTO studentHS (sID, sName, GPA, sizeHS)

VALUES (998, 'Ramu', 9.8, 989);

* Name: Hardik Arora
* Branch : Btech - CS
* Program: AIML
* University Roll No. : 2215500071
* Section: 2AC
* Class Roll No. : 28
* Submitted to: Ayushi Mam
* Submitted on: 06-05-2024