**Login:**

SLECECT Username

FROM User

WHERE EXISTS (SELECT \*

FROM User

WHERE User.Username = $Username AND User.Password = $Password)

**Credentials:**

INSERT INTO User

VALUES ($Username, $Password)

WHERE NOT EXISTS (SELECT \*

FROM User

WHERE User.Username = $Username)

**Personal information:**

**Create student general info:**

INSERT INTO Regular\_User

VALUES ($Username, $Address, $Name, $Email\_Id, $DOB, $Permanent\_Address, $Gender, $Contact\_No)

WHERE NOT EXIST (SELECT Username

FROM Regular\_User

WHERE Regular\_User.Username = $Username)

**Bring courses the student registered before, and the Grades earned are B or higher:**

//first bring the drop down list of course codes that the students took the courses before, name it as SelectedTutorCourseCode

SELECT CCode.Code AS SelectedTutorCourseCode

FROM CCode

WHERE CCode.Title IN (SELECT Title

FROM Section JOIN Registers ON Section.CRN = Registers.CRN

WHERE Registers.Username = $Username

AND Registers.Grade IN (‘A’, ‘B’))

**Add apply\_to\_tutor info:**

//Select course titles corresponding to that course code for later insert, name as SelectedTutorTitle

SELECT Title AS SelectedTutorTitle

FROM CCode

WHERE CCode.Code IN $SelectedTutorCourseCode

//for each selectedTutorTitle

INSERT INTO Applied\_To\_Tutor\_For

VALUES ($Username, $SelectedTutorTitle)

**View major info:**

//bring the 5 majors for students to select

SELECT Department.Name AS MajorSelectionForStudents

FROM Department

**Create major info:**

INSERT INTO Student (Username, Major, Degree)

VALUES($Username, $SelectedMajor, $Degree)

**Create education history:**

INSERT INTO Education\_History

VALUES ($Username, $Year\_Of\_Grad, $Name\_Of\_School, $Major, $Degree, $GPA)

WHERE (SELECT COUNT(\*)

FROM Education\_History

WHERE Education\_History\_Username = $Username)<=3

**Update:**

**Bring general info:**

SELECT Address, Name, Email\_Id, DOB, Permanent\_Adrress, Gender, Contact\_No

FROM Regular\_User

WHERE Regular.Username = $Username

UPDATE Regular\_User

SET Address = $Address, Name = $Name , Email\_Id = $Email\_Id, DOB = $DOB, Permanent\_Address = $Permanent\_Address, Gender = $Gender, Contact\_No = $Contact\_No

WHERE Regular\_User.Username = $Username

**Bring courses the student registered before, and the Grades earned are B or higher:**

//first bring the drop down list of course codes that the students took the courses before, name it as SelectedTutorCourseCode

SELECT CCode.Code AS SelectedTutorCourseCode

FROM CCode

WHERE CCode.Title IN (SELECT Title

FROM Section JOIN Registers ON Section.CRN = Registers.CRN

WHERE Registers.Username = $Username

AND Registers.Grade IN (‘A’, ‘B’))

//Select course titles corresponding to that course code for later insert, name as SelectedTutorTitle

**Add apply\_to\_tutor info:**

SELECT Title AS SelectedTutorTitle

FROM CCode

WHERE CCode.Code IN $SelectedTutorCourseCode

//for each selectedTutorTitle

INSERT INTO Applied\_To\_Tutor\_For

VALUES ($Username, $SelectedTutorTitle)

//find the key “title” for oringinal added course codes, name as OriginalTitle

**Delete apply\_to\_tutor info:**

**Select changed applied tutor course:**

SELECT Title AS OriginalTitle

FROM CCode

WHERE CCode.Code = $SelectedTutorCourseCode

//find out newly applied tutor course title, name as changedTitle

**Selected original applied tutor course:**

SELECT Title AS changedTitle

FROM Applied\_To\_Tutor\_For

WHERE Applied\_To\_Tutor\_For.Username = $Username

DELETE FROM Applied\_To\_Tutor\_For

WHERE Applied\_To\_Tutor\_For.Title IN (‘OriginalTitle’ EXCEPT ‘changedTitle’)

**View major info:**

SELECT Major, Degree

FROM Student

WHERE Student.Username = $Username

**Update major info:**

UPDATE Student

SET Major = $Major, Degree = $Degree

WHERE Student.Username = $Username

**Update education history:**

//select the education history the user wants to change by selecting year and school name

SELECT Year\_Of\_Grad AS selectedYear, Name\_Of\_School AS selectedSchool, Major, Degree, GPA

FROM Education\_History

WHERE Education\_History.Username = $Username

AND Education\_History.Year\_Of\_Grad = $Year\_Of\_Grad

AND Education\_History.Name\_Of\_School = $Name\_Of\_School

//update old history by setting new values

UPDATE Education\_History

SET Year\_Of\_Grad = $Year\_Of\_Grad, Name\_Of\_School = $Name\_Of\_School, Major = $Major, Degree = $Degree, GPA = $GPA

WHERE Education\_History.Username = $Username

AND Education\_History.Year\_Of\_Grad = $selectedYear

AND Education\_History.Name\_Of\_School = $seletedSchool

**Faculty personal information:**

**Create general info:**

INSERT INTO Regular\_User

VALUES ($Username, $Address, $Name, $Email\_Id, $DOB, $Permanent\_Address, $Gender, $Contact\_No)

WHERE NOT EXIST (SELECT Username

FROM Regular\_User

WHERE Regular\_User.Username = $Username)

**View department info:**

**Bring department they can select**

//bring the dropdown list of departments, find the selected department ID according to the user’s selection, name it as selectedDepartment\_Id

SELECT Dept\_Id AS selectedDepartment\_Id

FROM Department

WHERE Department.Name = $Name

INSERT INTO Faculty(Username, Position, Dept\_Id)

VALUES ($Username, $Position, $selectedDepartment\_Id)

**view sections:**

**bring course lists provided by the department:**

SELECT Code

FROM CCode

WHERE CCode. Title IN (SELECT Title

FROM Department\_Offers\_Course

WHERE Department\_Offers\_Course.Dept\_Id = $selectedDepartment\_Id)

//bring section letters:

SELECR DISTINCT Letter

FROM Section

WHERE Section.Title IN (SELECT Title

FROM Department\_Offers\_Course

WHERE Department\_Offers\_Course.Dept\_Id = $selectedDepartment\_Id)

**Insert teaching info:**

INSERT INTO SECTION (Instructor\_Username)

VALUES ($Username)

WHERE Section.Title = (SELECT Title

FROM CCode

WHERE CCode.Code = $SelectedCode)

AND Section.Letter = $SelectedLetter

**Create research interests:**

INSERT INTO Research\_Interests

VALUES ($Username, $Interest)

**Update faculty general info:**

SELECT Address, Name, Email\_Id, DOB, Permanent\_Adrress, Gender, Contact\_No

FROM Regular\_User

WHERE Regular.Username = $Username

UPDATE Regular\_User

SET Address = $Address, Name = $Name , Email\_Id = $Email\_Id, DOB = $DOB, Permanent\_Address = $Permanent\_Address, Gender = $Gender, Contact\_No = $Contact\_No

WHERE Regular\_User.Username = $Username

**View department info:**

SELECT Posistion, Dept\_Id

FROM Faculty

WHERE Faculty.Username = $Username

SELECT Dept\_Id AS selectedDept\_Id

FROM Department

WHERE Department.Name = $Name

UPDATE Faculty

SET Position = $Position, Dept\_Id = $selectedDept\_Id

WHERE Faculty.Username = $Username

**view sections:**

**bring the course lists provided by the department, show the course he chose to teach:**

SELECT Code

FROM CCode

WHERE CCode. Title IN (SELECT Title

FROM Department\_Offers\_Course

WHERE Department\_Offers\_Course.Dept\_Id = $selectedDepartment\_Id)

SELECT Code

FROM CCode

WHERE CCode.Title = (SELECT Title

FROM Section

WHERE Section.Instructor\_Username = $Username)

//bring section letters:

SELECR DISTINCT Letter

FROM Section

WHERE Section.Title IN (SELECT Title

FROM Department\_Offers\_Course

WHERE Department\_Offers\_Course.Dept\_Id = $selectedDepartment\_Id)

//bring the section letter the instructor chose

SELECT Letter  
FROM Section

WHERE Section.Title IN (SELECT Title

FROM Department\_Offers\_Course

WHERE Department\_Offers\_Course.Dept\_Id = $selectedDepartment\_Id)

**Insert teaching info:**

INSERT INTO SECTION (Instructor\_Username)

VALUES ($Username)

WHERE Section.Title = (SELECT Title

FROM CCode

WHERE CCode.Code = $SelectedCode)

AND Section.Letter = $SelectedLetter

**Update research interests:**

**View original interest:**

SELECT Interest AS SelectedInterest

FROM Research\_Interests

WHERE Research\_Interests.Username = $Username

AND Research\_Interests.Interest = $SelectedInterest

**Change to new interest:**

UPDATE Research\_Interests

SET Interest = $NewInputInterest

WHERE Research\_Interests.Username = $Username

AND Research\_Interest.Interest = ‘SelectedInterest’

**Register:**

**Select department:**

SELECT Dept\_Id AS selectedDept\_Id

FROM Department

WHERE Department.Name = $SelectedDeptName

//know current semester term

**Retrieve course code and section info:**

SELECT \*

FROM Section JOIN CCode ON Section.Title = CCode.Title

WHERE Section.TItle IN (SELECT Title

FROM Department\_Offers\_Course

WHERE Department\_Offers\_Course.Dept\_Id = ‘selectedDept\_Id’)

**Insert register info:**

INSERT INTO Registers(Username, CRN, Grade\_Mode)

VALUES ($Username, $CRN, $Grade\_Mode)

**Registered courses:**

**Select registration info:**

//find all CRN the student registers, name as selectedCRN

SELECT CRN AS selectedCRN

FROM Registers

WHERE Registers.Username = $Username

**Select section and course code info:**

SELECT Code, Title, Letter, Grade\_Mode

FROM Section, Registers

WHERE Registers.Username = $Username AND Section.CRN IN $selectedCRN

**Assign tutors:**

**Select title from section info:**

//select the course title that the instructor teach and can assign tutors, name as selectedTitle

SELECT Title AS selectedTitle

FROM Section

WHERE Section.Instructor\_Username=$Username

**Bring apply\_to\_tutor info:**

CREATE VIEW Temp

SELECT Username, Name

FROM Regular\_User JOIN Applied\_To\_Tutor\_For

WHERE Applied\_To\_Tutor\_For.Title = $selectedTitle

**Add tutors from applicants:**

//name as selectedTutorUsername

SELECT Username AS selectedTutorUsername

FROM Temp

WHERE Temp.Name = $NameSelected

//for each selectedTutor

INSERT INTO Tutor

VALUES ($selectedTutorUsername)

//for each selectedTutor

INSEERT INTO Tutors\_For

VALUES

($selectedTutorUsername, $selectedTitle)

DROP VIEW Temp

**enter course code/keyword:**

**look up title:**

//name as selectedCourseTitle

SELECT Title AS selectedCourseTitle

FROM CCode

WHERE CCode.Title LIKE ‘%’+$Keyword+‘%’

**Bring course code list:**

SELECT Code

FROM CCode

WHERE CCode.Title IN (SELECT Title

FROM Section JOIN Registers ON CRN

WHERE Registers.Username = $Username)

**Look up tutors:**

//find the tutors available, name as tutorUsername

SELECT Username AS tutorUsername

FROM Tutors\_For

WHERE Tutors\_For.Title = $selectedCourseTitle

OR Tutors\_For.Title = (SELECT Title

FROM CCode

WHERE CCode.Code = $selectedCode)

**Look up name and Email\_Id:**

SELECT Name, Email\_Id

FROM Regular\_User

WHERE Regular\_User.Username IN $tutorUsername

**Visit details:**

**view course code drop down list:**

SELECT Code AS availableCourseCode

FROM CCode

WHERE CCode.Title IN (SELECT Title

FROM Tutors\_For

WHERE Tutors\_For.Username=$Username)

**Get the CRN for tutored course: logs\_visit:**

//get the CRN for tutored course code, name as tutorCRN

SELECT CRN

FROM Registers

WHERE Registers.CRN IN (SELECT CRN

FROM Section

WHERE Section.Title = (SELECT Title

FROM CCode

WHERE CCode.Code = $SelectedCodebyTutor))

**Selected student name according to student\_Id:**

//get the student usename who receives tutor service, name as StudentUsername

SELECT Username

FROM Student

WHERE Student.Student\_Id = $enteredStudent\_Id

**Insert logs\_visit info:**

INSERT INTO Logs\_Visit

VALUES

($TutorUsername, $StudentUsername, $tutorCRN, CURRENT\_TIMESTAMP)

**Admin report:**

**Show all courses:**

SELECT Code, Title, AVG(DECODE(Grade,’A’,4,’B’,3’C’,2’,’D’,1,’F’,0))

FROM CCode, Registers JOIN Section ON Registers.CRN = Section.CRN

WHERE CCode.Title = (SELECT Title

FROM Section)

GROUP BY Code

**Faculty report:**

//average grade for students have more than 3 meetings with tutors

SELECT Code, Title, AVG(DECODE(Grade,’A’,4,’B’,3’C’,2’,’D’,1,’F’,0))

FROM CCode, Section, Registers

WHERE CCode.Title = Section.Title

AND Section.CRN = Registers.CRN

AND Section.Instructor\_Username = $Username

AND Registers.Username = (SELECT Student

FROM (SELECT Student, COUNT(\*)

FROM Logs\_Visit

GROUP BY Student

HAVING COUNT(\*)>3))

GROUP BY Code

//average grade for students have 1~ 3 meetings with tutors

SELECT Code, Title, AVG(DECODE(Grade,’A’,4,’B’,3’C’,2’,’D’,1,’F’,0))

FROM CCode, Section, Registers

WHERE CCode.Title = Section.Title

AND Section.CRN = Registers.CRN

AND Section.Instructor\_Username = $Username

AND Registers.Username = (SELECT Student

FROM (SELECT Student, COUNT(\*)

FROM Logs\_Visit

GROUP BY Student

HAVING COUNT(\*)<=3 AND COUNT(\*) >=1))

GROUP BY Code

//average grade for students have no meeting with tutors

SELECT Code, Title, AVG(DECODE(Grade,’A’,4,’B’,3’C’,2’,’D’,1,’F’,0))

FROM CCode, Section, Registers

WHERE CCode.Title = Section.Title

AND Section.CRN = Registers.CRN

AND Section.Instructor\_Username = $Username

AND Registers.Username = (SELECT Username

FROM Registers

EXCEPT (SELECT Student

FROM Logs\_Vist))

GROUP BY Code

**Student report:**

SELECT Instructor\_Username, Code, Title, AVG(DECODE(Grade,’A’,4,’B’,3’C’,2’,’D’,1,’F’,0))

FROM Section JOIN Registers

GROUP BY Instructor\_Username