Project Objectives

- The Main target of this project is to make a organized database of a university.
- To insert new student or Instructor data in the database and retrieve them for various purpose.
- Another objective of the system is to maintain the university information like student's result, admission etc.
- It can also maintain the Instructor's information for administrative purpose.

Features of the project

- Show the Instructor's name according to their department.
- Show the Course title, course credit and department name for all department.
- Show the existing departments of the university.
- Show all the courses of CSE department
- Show all the courses of EEE department
- Show all the 3 credit courses of CSE department
- Login of normal user
- Changing the password of a registered user.
- Show the total credit of CSE department
- Show the total credit of all department.
- Show the result of student.
- Show the result of a student in a particular semester
- Show the semester wise Courses they have taken

- Show the course Teacher students have in their course
- Student’s login
- Student’s password change
- Show all the information of Students
- Show all the information of Department
- Show all the information of Instructors.
- Show all the information of Instructors of EEE department.
- Show all the information of Courses
- Show all the information of Sections
- Show all the information of Enrollment.
- Show the name and salary of the Instructors whose salary is 50000 or greater.
- Show the name and salary of the Instructors whose salary is 50000 or greater and department is CSE.
- Show all the information of Courses which are taken by a particular instructor.

- Show the name of the Instructor who are currently in service.
- Show the name of the Instructor who have left the university.
- Show the serving time and name of a particular Instructor who have left.
- Increase Instructor's salary by 10 percent whose salary is lower or equal to 30000
- Show the courses taken by a student.
- Show the courses of CSE department.
- Show the highest Salary among the Instructors
- Show the lowest Salary among the Instructors
- Show the courses that were offered in Spring 16 but not Fall 15.
- Show the department wise Maximum Salary
- Show the total credit hour of each department
- Show the section wise student in Spring 16 semester
- Show the students with name starting with B

- Show the student whose address is Sylhet
- Show the students who have age between 18 and 22
- Show the student who use Robi Sim
- Show the student whose phone number ends with 2
- Show the student who have got A+
- Show the instructor who take more than 2 courses
- Admin’s login in the database.
- Show the number of registered User
- Show the number of registered Student
- Show all the information about the registered User
- Show the registered User
- Show the student who attend class at a particular room

Types of Users

Normal Users

In this university management system a normal user can on ly see some limited information like Instructor's list, department list, Courses according to the departments etc.

Students (Private User)

Students are the private user of this system. They can know their result of different semester, their marks, their course teacher etc. We can say that they are the users who only can see their own information and all the information a general user can get.

ADMIN

Admin is the controller of the university management system. An administrative person will act as an admin. He can perform a lot of action on the database. Everything is under control of him. He can enter new Instructor, new Students, New sections and what not.

Feature grouping according to the Users

Normal User

- Show the Instructor’s name according to their department.
- Show the Course title ,course credit and department name for all department.
- Show the existing departments of the university.
- Show all the courses of CSE department
- Show all the courses of EEE department
- Show all the 3 credit courses of CSE department
- Login of normal user
- Changing the password of a registered user.
- Show the total credit of CSE department
- Show the total credit of all department.

Students (Private Users)

- Show the result of student.
- Show the result of a student in a particular semester

- Show the semester wise Courses they have taken
- Show the course Teacher students have in their course
- Student's login
- Student's password change

 # Admin

- Show all the information of Students
- Show all the information of Department
- Show all the information of Instructors.
- Show all the information of Instructors of EEE department.
- Show all the information of Courses
- Show all the information of Sections
- Show all the information of Enrollment.
- Show the name and salary of the Instructors whose salary is 50000 or greater.
- Show the name and salary of the Instructors whose salary is 50000 or greater and department is CSE.

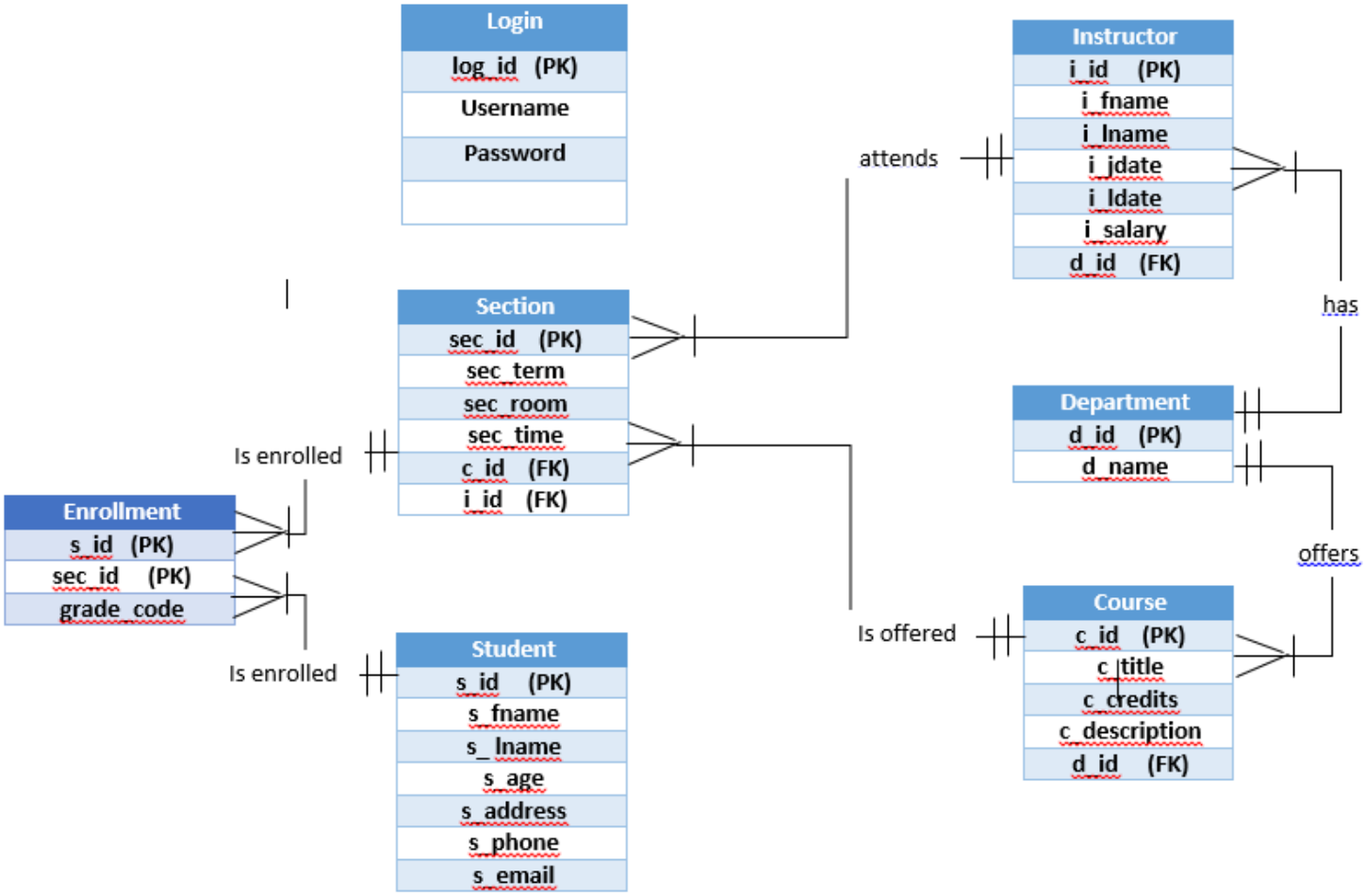
- Show all the information of Courses which are taken by a particular instructor.
- Show the name of the Instructor who are currently in service.
- Show the name of the Instructor who have left the university.
- Show the serving time and name of a particular Instructor who have left.
- Increase Instructor's salary by 10 percent whose salary is lower or equal to 30000
- Show the courses taken by a student.
- Show the courses of CSE department.
- Show the highest Salary among the Instructors
- Show the lowest Salary among the Instructors
- Show the courses that were offered in Spring 16 but not Fall 15.
- Show the department wise Maximum Salary
- Show the total credit hour of each department

- Show the section wise student in Spring 16 semester
- Show the students with name starting with B
- Show the student whose address is Sylhet
- Show the students who have age between 18 and 22
- Show the student who use Robi Sim
- Show the student whose phone number ends with 2
- Show the student who have got A+
- Show the instructor who take more than 2 courses
- Admin's login in the database.
- Show the number of registered User
- Show the number of registered Student
- Show all the information about the registered User
- Show the registered User
- Show the student who attend class at a particular room.

Names of the Entities with primary key

Table Name	Primary Key
Login	log_id
Instructor	i_id
Student	s_id
Section	sec_id
Department	d_id
Course	c_id
Enrollment	s_id, c_id

Entity Relationship Diagram (ERD)

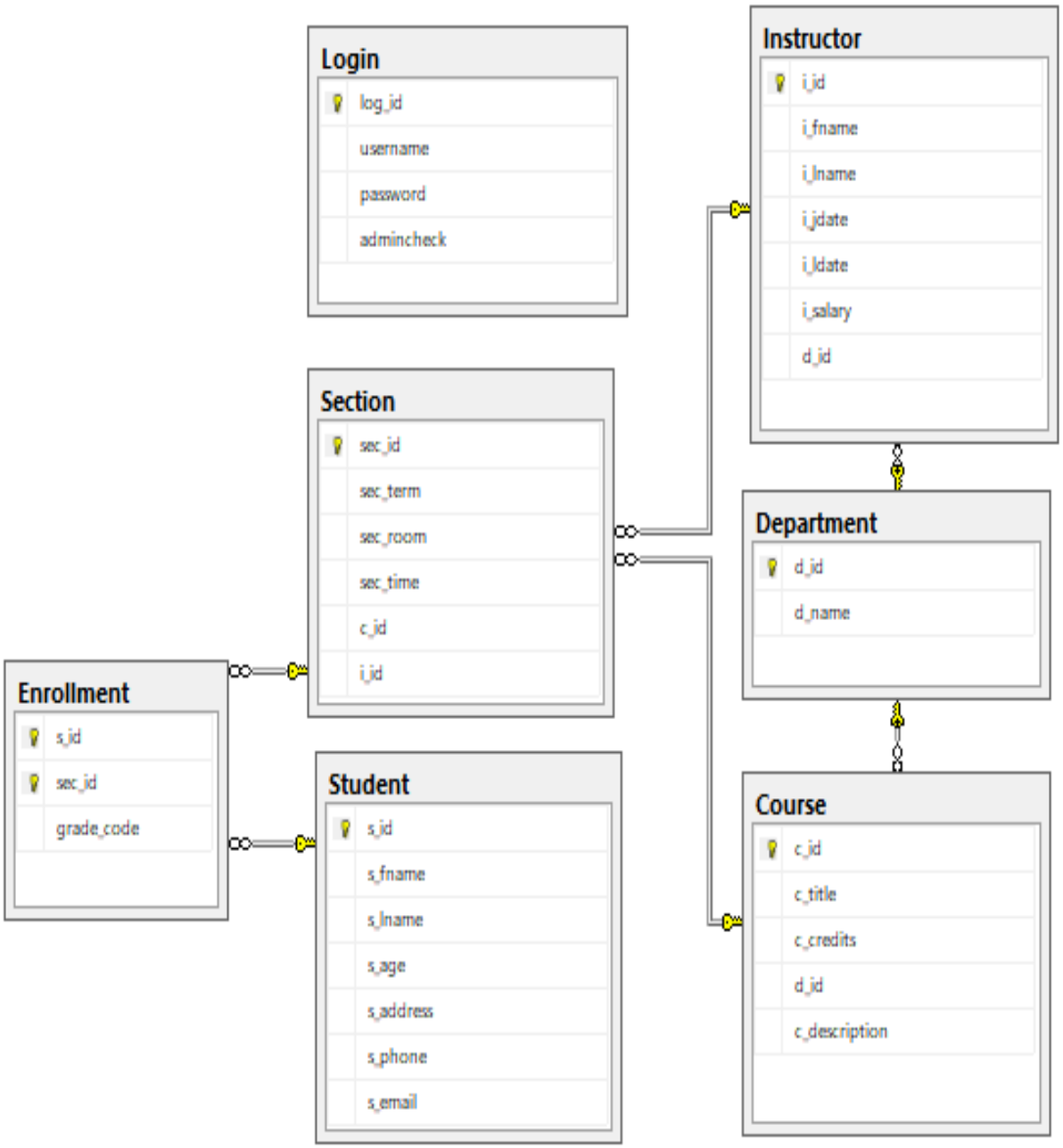


Relational Model

```
CREATE TABLE Student (
    s_id    int IDENTITY(10001,1)  PRIMARY KEY,
    s_fname varchar(20)  NOT NULL,
    s_lname varchar(20)  NOT NULL,
     s_age int  NOT NULL,
     s_address varchar(100)  NOT NULL,
     s_phone varchar(20)  NOT NULL,
     s_email varchar(50)  NOT NULL,
);

CREATE TABLE Department (
    d_id int IDENTITY(20001,1)  PRIMARY KEY,
    d_name varchar(40) NOT NULL UNIQUE
);

CREATE TABLE Instructor (
    i_id    int IDENTITY(30001,1)  PRIMARY KEY,
    i_fname varchar(20) NOT NULL,
    i_lname varchar(20) NOT NULL,
      i_jdate date NOT NULL,
      i_ldate date NULL,
      i_salary int NOT NULL,
    d_id  int  NOT NULL REFERENCES Department(d_id)
);


CREATE TABLE Course (
    c_id        int IDENTITY(40001,1)     PRIMARY KEY,
    c_title       varchar(100) NOT NULL,
    c_credits     tinyint      NOT NULL,
    d_id        int      NOT NULL REFERENCES Department(d_id),
    c_description varchar(255) NOT NULL
);

CREATE TABLE Section (
    sec_id   int IDENTITY(50001,1)      PRIMARY KEY,
    sec_term varchar(8)  NOT NULL,
    sec_room varchar(4),
    sec_time varchar(10),
    c_id int NOT NULL REFERENCES Course(c_id),
    i_id int  REFERENCES Instructor(i_id)
);
```

```
CREATE TABLE Enrollment (
    s_id     int  REFERENCES Student(s_id),
    sec_id     int      REFERENCES Section(sec_id),
    grade_code varchar(2),
    PRIMARY KEY (s_id, sec_id)
);

CREATE TABLE Login (
    log_id   int IDENTITY(60001,1)      PRIMARY KEY,
    username varchar(50),
    password varchar(100),
     admincheck int not null
);
```

Database Diagram



SQL Queries Grouped Under Different Types Of Users:

```
Insert Into Login
Values ('Jamal','12345',1),
       ('Kamal','123',0),
       ('Chapel','123',0),
       ('Rahim','1234',2);
Insert Into Student
Values ('Jack', 'Johnson',21,'Dhaka', '01723423423',
'jack@gmail.com'),
       ('Dolly', 'Denver',22,'Dhaka', '01712345678',
'dolly@gmail.com'),
       ('Helmut', 'Ziegler',21,'Comilla','01898765432',
'helmut@yahoo.com'),
       ('Robert', 'Thompson',24,'Khulna','01808975643',
'robert@gmail.com'),
       ('Jeffrey', 'Petersen',23,'Sylhet', '01773439315',
'petersen@yahoo.com'),
       ('Jack', 'Pirate',22,'Chittagong','01712321124',
'pirate@gmail.com'),
       ('Barb', 'Wire',21,
'Comilla','01709876789','barb@yahoo.com'),
       ('Roberta', 'Strong',24, 'Sylhet','01827645238',
'strong@gmail.com'),
       ('Heather', 'Black',22,'Dhaka',
'01921674523','black@yahoo.com'),
       ('Erik', 'Bjornsen',23,'Khulna', '01990675634',
'erik@yahoo.com');

Insert Into Department
Values ('CSE'),
       ('CIVIL'),
       ('EEE'),
       ('MECHA'),
       ('IPE'),
       ('BBA'),
       ('TEX');
```

```
Insert Into Instructor
Values ('Bill', 'Smith', '1995-12-21','2014-07-
02',50000,20001),
       ('James', 'Peterson', '1997-03-12','2018-06-
12',35000,20001),
       ('Adam', 'Thompson', '1998-04-15',null,60000,20002),
       ('Janetta', 'Oakley', '2001-02-13',null,50000,20003),
       ('Robin', 'Dexter', '2003-05-23',null,45000,20002),
       ('Annie', 'Jackson', '2004-06-02',null,30000,20004),
       ('Philip', 'Petrovsky', '2006-02-27',null,50000,20005),
       ('Anastasia', 'Scott', '2013-03-12',null,60000,20001);

Insert Into Course
Values ('EEE201',3, 20003, 'Microeconomics'),
       ('CIVIL301',3, 20002, 'Building Materials'),
       ('EEE302',3,20003, 'Power Electronics'),
       ('EEE303',4,20003 , 'Circuit Setup'),
       ('CSE201', 4,20001, 'Algorithms I'),
       ('CSE202', 3,20001, 'Data Stucture'),
       ('MECHA203',3,20004, 'Fluid Mechanics'),
       ('MECHA302',2,20004, 'Statistical Mathematics');

Insert Into Section
Values ('Spring16','7A05','10.30', 40001,30005),
       ('Fall16','7A05','01.00',40002,30004),
       ('Fall17','6A03', '12.30',40007,30007),
       ('Spring18','6A02', '12.30',40008,30007),
       ('Fall17','5A04', '11.00',40006,30002),
       ('Spring16','5A04', '11.00',40003,30005),
       ('Spring18','5A02', '11.00',40005,30003);

Insert Into Enrollment
Values  (10001,50001,'A+'),
        (10001,50006,'A'),
        (10002,50001,'B-'),
        (10002,50006,'A'),
        (10003,50003,'A+'),
        (10003,50005,'A+'),
        (10004,50003,'B+'),
```

```
(10004,50005,'B'),
(10005,50002,'A-'),
(10005,50004,'B-'),
(10006,50004,'C'),
(10006,50007,'F'),
(10008,50003,'D'),
(10009,50003,'F'),
(10007,50001,'A-');
```

Normal User

```
--> 1
SELECT i_fname +' '+ i_lname as Name_of_the_Faculties,d_name as
Department_Name
from Instructor I,Department D where I.d_id=D.d_id
-->2
SELECT c_title as Course_Title,c_description as Course_Name,
c_credits as Course_credit, d_name as Department_Name
from Course C,Department D where C.d_id=D.d_id
-->3
SELECT d_name as Department_Name FROM Department
-->4
SELECT c_title as Course_Name,c_credits as Credit
FROM Course C,Department D
where C.d_id=D.d_id and d_name='CSE'
-->5
SELECT c_title as Course_Name,c_credits as Credit
FROM Course C,Department D
where C.d_id=D.d_id and d_name='EEE'
-->6
SELECT c_title as Course_Name,c_credits as Credit
FROM Course C,Department D
where C.d_id=D.d_id and d_name='CSE' and c_credits=3
-->7
Select username from Login where username='Kamal' and password
='123' and admincheck=0
-->8
Update Login set password='1234' where username='Chapel'
```

```
-->9
Select sum(C.c_credits) as Total_Credit_Of_CSE from Course
C,Department D
where C.d_id=D.d_id and D.d_name='CSE'
-->10
Select D.d_name as Departpent_Name, sum(C.c_credits) as
Total_Credit
from Course C,Department D
where C.d_id=D.d_id group by D.d_name
```

 # Students (Private Users)

```
-->1 Result of a student
SELECT st.s_id as Student_ID, c_description as
Course_Name,grade_code as Grade
FROM Enrollment E,Section S,Course C,Student St
where E.sec_id=S.sec_id and S.c_id=C.c_id and E.s_id=St.s_id
and st.s_id = 10001
-->2 Result of a student in a particuler semester
SELECT st.s_id as Student_ID, c_description as
Course_Name,grade_code as Grade
FROM Enrollment E,Section S,Course C,Student St
where E.sec_id=S.sec_id and S.c_id=C.c_id and E.s_id=St.s_id
and st.s_id = 10007 and S.sec_term='Spring16'
-->3 Semester wise Courses they have taken
SELECT c_description as Course_Name,S.sec_term as Semester
FROM Enrollment E,Section S,Course C,Student St
where E.sec_id=S.sec_id and S.c_id=C.c_id and E.s_id=St.s_id
and st.s_id = 10001
-->4 Course Teacher they have in their course
SELECT c_description as Course_Name,I.i_fname as Instructor
FROM Enrollment E,Section S,Course C,Student St,Instructor
I,Department D
where E.sec_id=S.sec_id and S.c_id=C.c_id and E.s_id=St.s_id
and I.d_id=D.d_id and D.d_id=C.d_id and st.s_id = 10001
-->5 Student Login
Select username from Login where username='Jamal' and password
='12345' and admincheck=1
-->6
Update Login set password='1234' where username='Chapel'
```

ADMIN

```
-->1
SELECT * from Student
-->2
SELECT * from Department
-->3
SELECT * from Instructor
-->5
SELECT * from Instructor I,Department D where I.d_id=D.d_id and
D.d_name='EEE'
-->6
SELECT * from Course
-->7
SELECT * from Section
-->8
SELECT * from Enrollment
-->9
SELECT i_fname+' '+ i_lname AS Name_Of_The_Instructor,i_salary
AS Salary FROM Instructor WHERE i_salary>=50000
-->10
SELECT i_fname+' '+ i_lname AS Name_Of_The_Instructor,i_salary
AS Salary,d_name As Department_name
FROM Instructor I,Department D
WHERE i_salary>=50000 and I.d_id=D.d_id and d_name='CSE'
-->11
SELECT c_description as Course_Name FROM Course C,Section
S,Instructor I
where S.c_id=C.c_id and S.i_id=I.i_id and i_fname='Bill'
-->12 Current Instructor
SELECT i_fname+' '+ i_lname AS Name_Of_The_Instructor FROM
Instructor where i_ldate is NULL
-->13 Instructors who have left
SELECT i_fname+' '+ i_lname AS Name_Of_The_Instructor FROM
Instructor where i_ldate is not NULL
-->14 Serving Time of a teacher
SELECT DATEDIFF(YEAR,i_jdate,i_ldate) FROM Instructor where
i_fname='Bill'
```

```
-->15 Increse Instructor's salary by 10 percent whose salary is
lower or equal to 30000
UPDATE Instructor set i_salary=i_salary+i_salary*.10 where
i_salary<=30000
-->16 Course That are taken by a Student
SELECT c_description as Course_Name FROM Enrollment E,Section
S,Course C,Student St
where E.sec_id=S.sec_id and S.c_id=C.c_id and E.s_id=St.s_id
and st.s_fname='Jack'
-->17 Course of CSE
SELECT c_description FROM Course where c_title like 'CSE%'
-->18 Highest Salary among the Instructors
SELECT max(i_salary) from Instructor
-->19 Lowest Salary among the Instructors
SELECT min(i_salary) from Instructor
-->20 Courses that were offered in Spring 16 but not Fall 15\
SELECT c_description as Course_Name FROM Section S,Course C
where  S.c_id=C.c_id and S.sec_term='Spring16' and S.sec_term
!='Fall15'
-->21 Department wise Maximum Salary
SELECT D.d_name as Department, max(i_salary) from Instructor
I,Department D
where I.d_id=D.d_id Group by D.d_name
-->22 Total credit hour of each department
SELECT D.d_name as Department, sum(c_credits) from Course
C,Department D
where C.d_id=D.d_id Group by D.d_name
-->23 Section wise student in Spring 16 semister
SELECT S.sec_id,count(St.s_id) from Student St,Enrollment E,
Section S
where S.sec_term='Spring16' and St.s_id=E.s_id and
E.sec_id=S.sec_id
group by S.sec_id
-->24 Student with name starting with B
Select s_fname+' '+s_lname as Name from Student where s_fname
like 'B%'
-->25 Student whose address is Sylhet
Select s_fname+' '+s_lname as Name from Student where
s_address='sylhet'
```

```
-->26 Student who have age between 18 and 22
Select s_fname+' '+s_lname as Name from Student where s_age
between 18 and 22
-->27 Student who use Robi Sim
Select s_fname+' '+s_lname as Name from Student where s_phone
like '018%'
-->26 Student whose phone number ends with 2
Select s_fname+' '+s_lname as Name from Student where s_phone
like '%2'
-->27 Student who have got A+
Select distinct s_fname+' '+s_lname as Name  from Student St,
Enrollment E, Section S
where St.s_id=E.s_id and E.sec_id=S.sec_id and
E.grade_code='A+'
-->28 Instructor who take more than 2 courses
SELECT I.i_fname as Instructor_Name, count(S.c_id) from
Instructor I,Section S
where I.i_id=S.i_id  Group by I.i_fname having count(S.c_id)>=2
-->29 Admin Login
Select username from Login where username='Rahim' and password
='1234' and admincheck=2
-->30 Number of  registered User
Select count(log_id) from Login where admincheck=0 or
admincheck=1
-->31 Number of registered Student
Select count(log_id) from Login where admincheck=1
--> 32 Know all the information about the registered User
Select * from Login
-->33 Show the registered User
Select username from Login where admincheck=1 or admincheck=0
-->34 Student who attend class at a particular room
select s_fname from Student where s_id in
(select s_id from Enrollment where sec_id in
(Select sec_id from Section where sec_room='7A05'))
```

Project Limitations

- The system is not perfect for an university. It needs some improvement to use perfectly.
- The employee table is not added in this database.
- Instead of date of birth age is given in Student table. So we have to increase this column by one every year.
- Result is only given in grade system but marks should be included for better understanding.

Conclusion:

As this project is a MSSql Database project. It can be used in different types of User interfaced project like Java. This is a portable project as it is only created by some sql code. This project is also efficient as we have attached all the query possible for an university management system. We have used a low amount of resources as we have not used anything without SQL

server management studio. This is also a user friendly system for all types of user. At the end we can say that it is a complete system for a university.