**PROJECT REPORT**

##### DBMS PROJECT

On

STUDENT MANAGEMENT SYSTEM

###### ***Submitted by***

##### SAYANDIP PAUL

RA1711008010322

***in partial fulfillment for the award of the degree of***

##### BACHELOR OF TECHNOLOGY

IN

**INFORMATION TECHNOLOGY**

****

OCTOBER 2019

S.NO TITLES PAGE

1. ER diagram 3-8

2. DDL/DML 9

3. SQL constraints 10

4. Retrieving data using select 11-12

5. Restrictions and sorting 13-14

6. Aggregate and grouping 15-16

7. Single row function 17-19

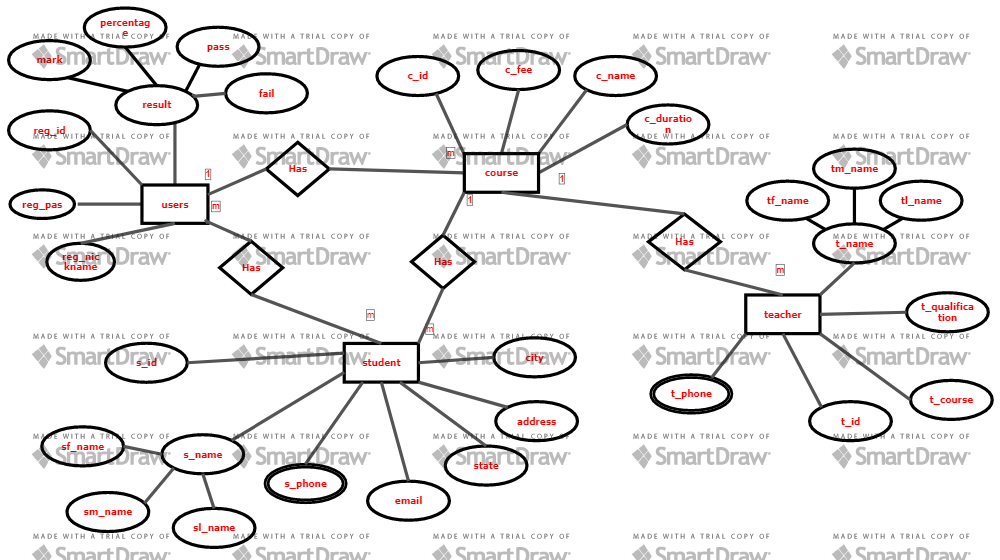
8. Joins 20-22

9. Subquery 23-24

10. Normalisation 25-28

###### **(1)ER DIAGRAM**

###### **STUDENT MANAGEMENT SYSTEM**



###### **ALL TABLES OF STUDENT MANAGEMENT SYSTEM**

SQL> create table student(s\_id int NOT NULL PRIMARY KEY, f\_name varchar(20),m\_name varchar(20),l\_name varchar(20),address varchar(20),city varchar(20),state varchar(20),email varchar(20));

Table created

SQL> create table s\_phone(s\_id int , FOREIGN KEY(s\_id) REFERENCES student(s\_id),phone int);

Table created

SQL> create table course(c\_id int NOT NULL PRIMARY KEY,s\_id int,FOREIGN KEY(s\_id) REFERENCES student (s\_id),c\_fee int,c\_name varchar(255),c\_duration varchar(255));

Table created

SQL> create table teacher(t\_id int NOT NULL PRIMARY KEY,tf\_name varchar(20),tm\_name varchar(20),tl\_name varchar(20),c\_id int, FOREIGN KEY (c\_id) REFERENCES cousre(c\_id),t\_course varchar(255),t\_qualification varchar(255));

Table created

SQL> create table t\_phone(t\_id int,FOREIGN KEY(t\_id) REFERENCES teacher(t\_id),t\_phonenumber int);

Table created

SQL> create table users (s\_id int,reg\_id int not null,reg\_nickname varchar(20),reg\_password varchar(20),mark int,per int,pass varchar(20),fail varchar(20),FOREIGN KEY(s\_id) REFERENCES student(s\_id));

Table created

SQL> desc student;

Name Null? Type

----------------------------------------- -------- -------------------- S\_ID NOT NULL NUMBER(38) F\_NAME VARCHAR2(255) M\_NAME VARCHAR2(255) L\_NAME VARCHAR2(255) ADDRESS VARCHAR(255) CITY VARCHAR2(255) STATE VARCHAR2(255)

SQL> desc s\_phone;

Name Null? Type

----------------------------------------- -------- --------------------

S\_ID NUMBER(38)

PHONE NUMBER(38)

SQL> desc course;

Name Null? Type

----------------------------------------- -------- --------------------

C\_ID NOT NULL NUMBER(38)

S\_ID NUMBER(38)

C\_FEE NUMBER(38)

C\_NAME VARCHAR2(255)

C\_DURATION VARCHAR2(255)

SQL> desc teacher;

Name Null? Type ----------------------------------------- -------- ------------------ T\_ID NOT NULL NUMBER(38)

C\_ID NUMBER(38)

TF\_NAME VARCHAR2(255)

TM\_NAME VARCHAR2(255)

TL\_NAME VARCHAR2(255)

T\_COURSE VARCHAR2(255)

T\_QUALIFICATION VARCHAR2(255) SQL> desc t \_phone; Name Null? Type ----------------------------------------- -------- -------------------- T\_ID NUMBER(38) T\_PHONENUMBER NUMBER(38)

SQL> desc users;

Name Null? Type ----------------------------------------- -------- -------------------- S\_ID NUMBER(38)

REG\_ID NOT NULL NUMBER(38)

REG\_NICKNAME VARCHAR2(255)

REG\_PASSWORD VARCHAR2(255)

MARK NUMBER(38)

PER VARCHAR2(255)

PASS VARCHAR2(255)

FAIL VARCHAR2(255)

###### **INSERT DATA TO TABLE**

###### STUDENT TABLE

SQL> insert into student(s\_id,f\_name,m\_name,l\_name,address,city,state)values(001,'sayan','dip','paul','alipurduar','chennai','tamilnadu'); 1 row created. SQL> insert into student(s\_id,f\_name,m\_name,l\_name,address,city,state)values(002,'alok','kumar','gupta','siliguri','alipur','westbengal'); 1 row created. SQL> insert into student(s\_id,f\_name,m\_name,l\_name,address,city,state)values(003,'amal','jit','mr','nandi','kochi','kerala'); 1 row created. SQL> insert into student(s\_id,f\_name,m\_name,l\_name,address,city,state)values(004,'arko','duti','mitra','balguria','kolkata','westbengal'); 1 row created. SQL> insert into student(s\_id,f\_name,m\_name,l\_name,address,city,state)values(005,'anish','kkr','thakur','dumdum','kolkata','westbengal'); 1 row created.

###### S\_PHONE TABLE

SQL> insert into s\_phone (s\_id,phone) values (001,9944870714);

SQL> insert into s\_phone (s\_id,phone) values (002,7063101929);

SQL> insert into s\_phone (s\_id,phone) values (003, 9515438334);

SQL> insert into s\_phone (s\_id,phone) values (004, 7895463153);

SQL> insert into s\_phone (s\_id,phone) values (005, 9955446428);

###### COURSE TABLE

SQL> insert into course (c\_id,s\_id,c\_fee,c\_name,c\_duration) values (11,001,25000

,'dbms','45\_hr');

SQL> insert into course (c\_id,s\_id,c\_fee,c\_name,c\_duration) values (12,002,45000

,'c++','45\_hr');

SQL> insert into course (c\_id,s\_id,c\_fee,c\_name,c\_duration) values (13,003,10200

0,'java','22\_hr');

SQL> insert into course (c\_id,s\_id,c\_fee,c\_name,c\_duration) values (14,004,10200

0,'python','52\_hr');

SQL> insert into course (c\_id,s\_id,c\_fee,c\_name,c\_duration) values (15,005,10200

,'pos','10\_hr');

###### TEACHER TABLE

SQL> insert into teacher (t\_id,c\_id,tf\_name,tm\_name,tl\_name,t\_course,t\_qualifica

tion) values (112,11,'sri','\_\_','vidhya','dbms','phd');

SQL> insert into teacher (t\_id,c\_id,tf\_name,tm\_name,tl\_name,t\_course,t\_qualifica

tion) values (115,12,'savri','\_\_','dasan','c++','btech');

SQL> insert into teacher (t\_id,c\_id,tf\_name,tm\_name,tl\_name,t\_course,t\_qualifica

tion) values (113,13,'jousaf','\_\_','paul','java','btech');

SQL> insert into teacher (t\_id,c\_id,tf\_name,tm\_name,tl\_name,t\_course,t\_qualifica

tion) values (114,14,'siva','\_\_','jos','python','bsc');

SQL> insert into teacher (t\_id,c\_id,tf\_name,tm\_name,tl\_name,t\_course,t\_qualifica

tion) values (111,15,'syu','mala','susi','pos','phd');

###### T\_PHONE TABLE

SQL> insert into t\_phone (t\_id,phonenumber) values (111,9944875714);

SQL> insert into t\_phone (t\_id,phonenumber) values (112,7063101549);

SQL> insert into t\_phone (t\_id,phonenumber) values (113, 9515438224);

SQL> insert into t\_phone (t\_id,phonenumber) values (114, 7895415213);

SQL> insert into t\_phone (t\_id,phonenumber) values (115, 9215146428);

###### USER TABLE

SQL> insert into users (s\_id,reg\_id,reg\_nickname,reg\_password,mark,per,pass,fail

) values (001,20,'pala','xyz',900,90,'y','n');

1 row created.

SQL> insert into users (s\_id,reg\_id,reg\_nickname,reg\_password,mark,per,pass,fail

) values (002,21,'onis','zxy',800,80,'y','n');

1 row created.

SQL> insert into users (s\_id,reg\_id,reg\_nickname,reg\_password,mark,per,pass,fail

) values (003,22,'arko','zx',700,70,'y','n');

1 row created.

SQL> insert into users (s\_id,reg\_id,reg\_nickname,reg\_password,mark,per,pass,fail

) values (004,23,'baba','ons',300,30,'n','y');

1 row created.

SQL> insert into users (s\_id,reg\_id,reg\_nickname,reg\_password,mark,per,pass,fail

) values (005,24,'papa','rrv',400,40,'n','y');

###### **(2) DDL AND DML**

**1)Create a table name student and attribute of s\_id,first last and middle name,address,email,city,state?**

SQL> create table student(s\_id int NOT NULL PRIMARY KEY, f\_name varchar(20),m\_name varchar(20),l\_name varchar(20),address varchar(20),city varchar(20),state varchar(20),email varchar(20));

**2)Display the datatype of all attribute’s in student table?**

SQL> desc student;

Name Null? Type

----------------------------------------- -------- -------------------- S\_ID NOT NULL NUMBER(38) F\_NAME VARCHAR2(255) M\_NAME VARCHAR2(255) L\_NAME VARCHAR2(255) ADDRESS VARCHAR(255) CITY VARCHAR2(255) STATE VARCHAR2(255)

**3)Insert values into the table student?**

SQL> insert into student(s\_id,f\_name,m\_name,l\_name,address,city,state)values(001,'sayan','dip','paul','alipurduar','chennai','tamilnadu');

**4)Change the datatype of s\_id in student table?**

SQL> alter table student modify s\_id varchar(20);

**5)Update the city of s\_id=001 to bby?**

SQL>update student set city=’bby’ where s\_id=001;

###### **RESULT**

DDL and DML command executed successfully.

###### **(3) SQL CONSTRAINTS**

**1)Create a table name student and with a attribute s\_id which is not null?**

SQL>create table student(s\_id int not null,age int);

Table created

**2) Create a table name student and with a attribute s\_id which is unique?**

SQL>create table stydent(s\_id int unique,age int);

Table created

**3)Create a table name student and with a attributes s\_id which is a primary key?**

SQL>create table stydent(s\_id int primary key,age int);

Table created

**4)Create a table name teacher and with a attributes s\_id which is a foreign key and team name?**

SQL>create table teacher(s\_id int,foreign key(s\_id) referances (s\_id),t\_name varchar(20));

Table created

**5)Create a table name student and with a attributes city which a default constraints “KOLKATA”?**

SQL>create table student(s\_id int,city varchar(20) default ‘KOLKATA’);

Table created

###### **RESULT**

SQL constraints command executed successfully.

###### **(4) RETRIEVING DATA USING SELECT**

**1)Display all the columns of table name course?**

SQL> select \* from course;

## OUTPUT

C\_ID S\_ID C\_FEE C\_NAME C\_DURATION

---------- ---------- ---------- -------------------- --------------------

11 1 25000 dbms 45\_hr

12 2 45000 c++ 45\_hr

13 3 102000 java 22\_hr

**2)Display the columns c\_name aliases coursename of table name course?**

SQL>select c\_name as coursename from course;

## OUTPUT

COURSEFEE --------------------

dbms

c++

java

**3)Display the columns c\_fee and c\_name using concatenation and aliases coursedetails of table name course?**

SQL>select c\_fee || c\_name as coursedetails from course;

## OUTPUT

COURSEDETAILS --------------------------------------- 25000 dbms

45000 c++

102000 java

**4)Display the course fee and increase in course fee by 3000 where course duration is 45\_hr?**

SQL>select c\_fee,c\_fee+3000 from course where c\_duration = ‘45\_hr’;

## OUTPUT

C\_FEE C\_FEE+3000 C\_DURATION

---------- -------------------- --------------------

25000 28000 45\_hr

45000 48000 45\_hr

**5)Remove the duplicate value from course durartion with student ID from table name course?**

SQL>select distinct c\_duration,s\_id from course;

## OUTPUT

S\_ID C\_DURATION

------------- --------------------

1 45\_hr

3 22\_hr

###### **RESULT**

Retrieving data using select command executed successfully.

###### **(5) RESTRICTIONS AND SORTING**

**1)Display the student ID and first name from table name student who’s state is westbengal?**

SQL>select s\_id,f\_name from student where state = ‘westbengal’;

## OUTPUT

S\_ID F\_NAME

------------- ------------------

2 alok

4 arko

5 anish

**2)Display all the data of table course who’s course ID is ‘11’?**

SQL>select \* from course where c\_id = 11;

## OUTPUT

C\_ID S\_ID C\_FEE C\_NAME C\_DURATION

---------- ---------- ---------- -------------------- --------------------

11 1 25000 dbms 45\_hr

**3)Display all the data of table course who’s course fee range in less then 100000?**

SQL>select \* from course where c\_fee < 100000;

## OUTPUT

## C\_ID S\_ID C\_FEE C\_NAME C\_DURATION ---------- ---------- ---------- -------------------- ------------------- 11 1 25000 dbms 45\_hr

12 2 45000 c++ 45\_hr

15 5 10200 pos 10\_hr

**4)Display the lastname and city of table student where student first name start with ‘a’ consist of 4 character?**

SQL>select l\_name,city from student where f\_name = ‘a% \_ \_ \_ ‘;

## OUTPUT

## L\_NAME CITY

------------------- - --------------------

gupta alipur

amla kochi

arko Kolkata

**5)Display the course fee of table name course which in between 10000 to 50000?**

SQL>select s\_id,c\_id,c\_fee from course where c\_fee in between(10000,50000);

## OUTPUT

S\_ID C\_ID C\_FEE

---------- ------------- ------------------------------

1 11 25000

2 12 45000

5 15 10200

###### **RESULT**

RESTRICTIONS AND Sorting command executed successfully.

###### **(6) AGGREGATE AND GROUPING**

**1)Display the average course fee for all the course in the given table?**

SQL>select avg(c\_fee) from course;

## OUTPUT

AVG(C\_FEE) ---------------

56840

**2)Display the minimum and maximum course fee in the given table?**

SQL>select min(c\_fee),max(c\_fee) from course;

## OUTPUT

MIN(C\_FEE) MAX(C\_FEE) ---------------- ---------------

10200 102000

102000

**3)Display the count number of course name from table name course?**

SQL>select count(c\_name) from course;

## OUTPUT

COUNT(C\_NAME) ----------------------------------- 5

**4)Display the total of course fee in the table name course?**

SQL>select sum(c\_fee) from course;

## OUTPUT

SUM(C\_NAME -----------------

284200

**5)Display the student ID and minimum course fee from coure table and group by student ID having course fee less then 102000 and sort the minimum course fee in descending order?**

SQL>select s\_id,min(c\_fee) from course group by s\_id having c\_fee < 102000 order by min(c\_fee) desc;

## OUTPUT

S\_ID MIN(C\_FEE)

---------- -------------

5 10200

1 25000

2 45000

###### **RESULT**

Aggregate and grouping command executed successfully.

###### **(7) SINGLE ROW FUNCTION**

**1)Display the student ID and student first name and state from student table where convert student first name and state into uppercase?**

SQL>select s\_id,upper(f\_name),upper(l\_name) from student;

## OUTPUT

S\_ID UPPER(L\_NAME) UPPER(STATE)

------- --------------------------- --------------------

1 PAUL WESTBENGAL

2 GUPTA WESTBENGAL

3 MR KERALA

4 MITRA WESTBENGAL

5 THAKUR WESTBENGAL

**2)Display the student ID and student first name and state from student table where convert student first name and state into lowercase?**

SQL> select s\_id,lower(f\_name),lower(l\_name) from student;

## OUTPUT

S\_ID UPPER(L\_NAME) UPPER(STATE)

------- ------------------ --------------------

1 paul westbengal

2 gupta westbengal

3 mr kerala

4 mitra westbengal

5 thakur westbengal

**3)Display the student ID and student state from table student and change the first character of all state to uppercase?**

SQL>select s\_id,initcap(state) from student;

## OUTPUT

S\_ID INITCAP(STATE)

------- --------------------

1 Westbengal

2 Westbengal

3 Kerala

4 Westbengal

5 Westbengal

**4)Display the student ID and student first name middle name last name together from the table student use the character-manipulation function?**

SQL>select s\_id,concat(f\_name,m\_name,l\_name) student\_name from student;

## OUTPUT

S\_ID STUDENT\_NAME

------- --------------------

1 sayandippaul

2 alokkumargupta

3 amaljitmr

4 arkodutimitra

5 anishkkrthakur

**5)Display the student ID and the length of the first name and the last name from the table student use the character-manipulation function?**

SQL>select s\_id,f\_name,length(f\_name),l\_name,length(l\_name) from student;

## OUTPUT

S\_ID F\_NAME LENGTH(F\_NAME) L\_NMAE LENGTH(L\_NAME)

------- -------------- ------------------ --------------- -------------------

1 sayan 5 paul 4

2 alok 4 gupta 5

3 amal 4 mr 2

4 arko 4 mitra 5

5 anish 5 thakur 6

###### **RESULT**

Single row function command executed successfully.

###### **(8) JOINS**

**1)Display the registration number fo usres table and student first namefrom the table student use inner join?**

SQL>select users.reg\_id,student.f\_name from users inner join student on users.s\_id = student.s\_id;

## OUTPUT

REG\_ID F\_NAME

------------- ----------------------

20 sayan

21 alok

22 amal

23 arko

24 anish

**2) Display the registration number fo usres table and student first namefrom the table student use left join and order by first name?**

SQL>select student.f\_name,users.reg\_id from users left join student on users.s\_id = student.s\_id order by student.f\_name;

## OUTPUT

F\_NAME REG\_ID

---------------------- --------------------

alok 21

amal 22

anish 24

arko 23

sayan 20

**3) Display the registration number fo usres table and student first name and last name from the table student use right join and order by registration number?**

SQL>select users.reg\_id,student.f\_name,student.l\_name from student right join student on usres.s\_id = student.s\_id order by users.reg\_id;

## OUTPUT

REG\_ID F\_NAME L\_NAME

---------------- ------------------ ---- ------------------

20 sayan paul

21 alok gupta

22 amal mr

23 arko mitra

24 anish thakur

**4)Select all teacher and all course in the given table use full join and order by teacher ID?**

SQL>select teacher.t\_id,course.c\_name from course full join course on tescher.c\_id = course.c\_id order by teacher.t\_id;

## OUTPUT

T\_ID C\_NAME

----------- -----------------

111 pos

112 dbms

113 java

114 python

115 c++

**5) Select all student first name and last name and city in the given table use self join and order by city?**

SQL>select A.f\_name as s\_name1,B.l\_name as s\_name2,A.city from student A,student B where A.s\_id <> B.s\_id and A.city = B.city order by A.city;

## OUTPUT

S\_NAME1 S\_NAME2 CITY

---------- ---------- -----------

sayan gupta alipur

alok paul chennai

amal thakur kochi

anish mitra kolkata

arko mr kolkata

###### **RESULT**

**JIONS all command executed successfully.**

###### **(9) SUBQUERY**

**1)Display the first middle and last name of student where the course fee is minimum?**

SQL>select f\_name,m\_name,l\_name from student where s\_id =(select s\_id from course where s\_id = min(c\_fee));

## OUTPUT

F\_NAME M\_NAME L\_NAME

------------- ------------- --------------

anish kkr thakur

**2) Display the teacher ID and first middle and last name of teacher where the course name is dbms?**

SQL> select t\_id,tf\_name,tm\_name,tl\_name from teacher where c\_id =(select c\_id from course where t\_course = ‘dbms’);

## OUTPUT

T\_ID TF\_NAME TM\_NAME TL\_NAME

----------- ------------------ -------------- --------------

112 sri \_\_ vidhya

**3)Display the phone number of the student of who’s registration number is 20?**

SQL>select phone from s\_phone where s\_id = (select s\_id from users where r\_id = 20);

## OUTPUT

PHONE

--------------

9944870714

**4)Display registration number who live in alipur?**

SQL>select r\_id from users where s\_id = (select s\_id from student where city = ‘alipur’);

## OUTPUT

REG\_ID

----------------

21

**5)Display teacher ID first name and phonenumber who’s qualification is b.tech?**

SQL>select t\_id,tf\_name,phone from t\_phone where t\_id = (select t\_id from teacher where t\_qualification = ‘b.tech’);

## OUTPUT

T\_ID TF\_NAME PHONE

--------- ----------- ---------------

113 jousaf 6541239575

115 savri 8235658254

###### **RESULT**

**SUBQUERY all command executed successfully.**

###### **(10) NORMALISATION**

**NORMAL FORM (1NF,2NF,3NF)**

**TABLE STUDENT**

##### Annotation 2019-10-08 024334.png

TABLE S\_PHONE

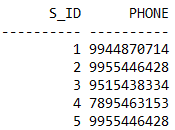


TABLE COURSE

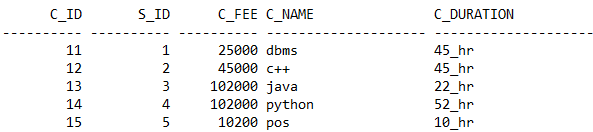


TABLE TEACHER

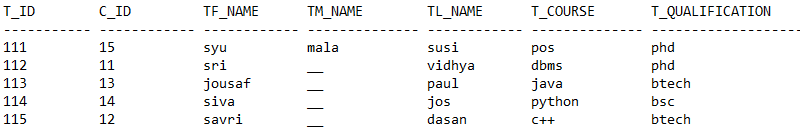


TABLE T\_PHONE

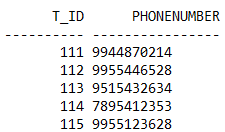
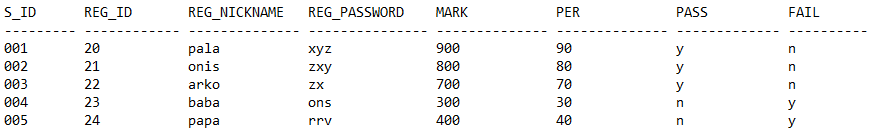
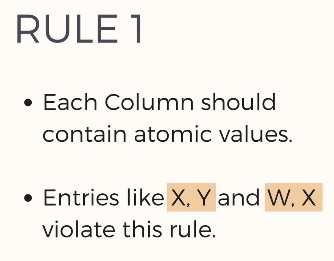
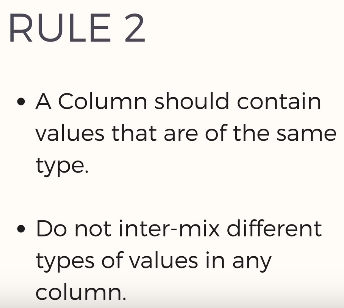


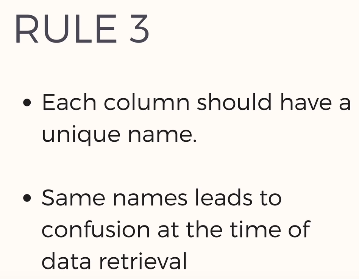
TABLE USRES

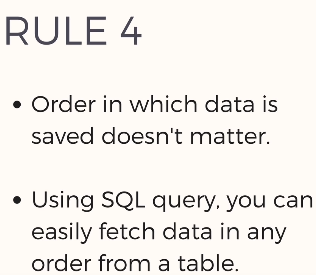


* This table satisfy all the rule of 1NF:-









* + This table satisfy all the rule of 2NF:-

RULE

* It should be in 1st normal form.
* It should not have partial dependencies.
* This table satisfy all the rule of 3NF:-

RULE

* + It should be in 2nd normal form.
  + It should not have transitive dependencies.

###### **RESULT**

The table’s of this student management system satisfy all the normal form(1NF,2NF,3NF).