PROJECT REPORT

Project Name:SCHOOL MANAGEMENT SYSTEM

Course Name :ADVANCE DATABASE MANAGEMENT SYSTEM

Database Schema Diagram:

Table 1: STUDENT_INFO

Number	Column	Type	Length	Nullable
01	ST_ID	NUMBER	10	NO
02	SNAME	VARCHAR2	20	YES
03	FNAME	VARCHAR2	20	YES
04	MNAME	VARCHAR2	20	YES
05	T_ID	NUMBER	10	YES
06	PASSING_YEAR	NUMBER	4	YES

Table 2: TEACHER_INFO

Number	Column	Type	Length	Nullable
01	T_ID	NUMBER	10	NO
02	TNAME	VARCHAR2	20	YES
03	CNAME	VARCHAR2	20	YES

Table 3: STUDENT_COURSE

Number	Column	Type	Length	Nullable
01	ST_ID	NUMBER	10	YES
02	CNAME	VARCHAR2	20	YES
03	T_ID	NUMBER	10	YES
04	C_RESULT	NUMBER	10	YES

Table 4: GRADE

Number	Column	Type	Length	Nullable
01	LOW_NUMBER	NUMBER	10	YES
02	HIGH_NUMBER	NUMBER	10	YES
03	GPA	NUMBER	10	YES
04	C_GRADE	VARCHAR2	10	YES

Table 5: LOGIN

Number	Column	Type	Length	Nullable
01	USER_ID	NUMBER	10	NO
02	PASSWORD	VARCHAR2	10	YES
03	STATUS	VARCHAR2	10	YES

SQL commands for table creation:

For STUDENT _INFO Table:

```
CREATE TABLE STUDENT_INFO
    (ST_ID NUMBER(10) PRIMARY KEY,
      SNAME VARCHAR2(20),
      FNAME VARCHAR2(20),
      MNAME VARCHAR2(20),
      T_ID NUMBER(10),
```

```
    PASSING_YEAR NUMBER(4));


INSERT INTO STUDENT_INFO
VALUES(10001,'ALICE','BOB','GANIKA',20002,2015);

INSERT INTO STUDENT_INFO
VALUES(10002,'NAYEM','HASEM','NAZMA',20001,2014);

INSERT INTO STUDENT_INFO

VALUES(10003,'NADIA','AMIR','SETU',20006,2015);

INSERT INTO STUDENT_INFO
VALUES(10004,'ARIF','KARIM','SANJIDA',20004,2018);

INSERT INTO STUDENT_INFO
VALUES(10005,'JANNAT','RAJ','TULEI',20002,2017);

INSERT INTO STUDENT_INFO
VALUES(10006,'SOHID','MONCUR','KHADIJA',20007,2015);

INSERT INTO STUDENT_INFO
VALUES(10007,'RAHIM','KARIM','ROHIMA',20001,2018);

INSERT INTO STUDENT_INFO
VALUES(10008,'ASHIK','NOZRUL','AMINA',20003,2017);

INSERT INTO STUDENT_INFO
VALUES(10009,'OLIVA','JOY','EVANA',20005,2014);

INSERT INTO STUDENT_INFO
VALUES(10010,'SOJIB','SHIBLO','MITU',20006,2015);
```

For TEACHER_INFO Table:

```
CREATE TABLE TEACHER_INFO
    (T_ID NUMBER(10) PRIMARY KEY,
      TNAME VARCHAR2(20),
      CNAME VARCHAR2(20));

INSERT INTO TEACHER_INFO VALUES(20001,'PRODIP','BANGLA');
INSERT INTO TEACHER_INFO VALUES(20002,'MASUM','BANGLA');
INSERT INTO TEACHER_INFO VALUES(20003,'BILLAH','MATH');
INSERT INTO TEACHER_INFO VALUES(20004,'SHAKIB','MATH');
INSERT INTO TEACHER_INFO VALUES(20005,'RAKIB','BANGLA');
INSERT INTO TEACHER_INFO VALUES(20006,'TANJIL','ENGLISH');
INSERT INTO TEACHER_INFO VALUES(20007,'HASIB','MATH');
INSERT INTO TEACHER_INFO VALUES(20008,'AMIN','ENGLISH');
INSERT INTO TEACHER_INFO VALUES(20009,'FATIMA','BANGLA');
INSERT INTO TEACHER_INFO VALUES(20010,'KANIJ','ENGLISH');
```

For STUDENT_COURSE Table:

```
CREATE TABLE STUDENT_COURSE
    (ST_ID NUMBER(10),
      CNAME VARCHAR2(20),
      T_ID NUMBER(10),
      C_RESULT NUMBER(10));
```

```
INSERT INTO STUDENT_COURSE VALUES(10001,'BANGLA',20001,88);
INSERT INTO STUDENT_COURSE VALUES(10001,'MATH',20003,93);
INSERT INTO STUDENT_COURSE VALUES(10001,'ENGLISH',20006,67);
INSERT INTO STUDENT_COURSE VALUES(10002,'BANGLA',20002,90);
INSERT INTO STUDENT_COURSE VALUES(10002,'MATH',20007,78);
INSERT INTO STUDENT_COURSE VALUES(10002,'ENGLISH',20008,95);
INSERT INTO STUDENT_COURSE VALUES(10003,'BANGLA',20009,30);
INSERT INTO STUDENT_COURSE VALUES(10003,'MATH',20003,80);
INSERT INTO STUDENT_COURSE VALUES(10003,'ENGLISH',20010,49);
INSERT INTO STUDENT_COURSE VALUES(10004,'BANGLA',20001,90);
INSERT INTO STUDENT_COURSE VALUES(10004,'MATH',20004,49);
INSERT INTO STUDENT_COURSE VALUES(10004,'ENGLISH',20006,37);
INSERT INTO STUDENT_COURSE VALUES(10005,'BANGLA',20009,83);
INSERT INTO STUDENT_COURSE VALUES(10005,'MATH',20003,93);
INSERT INTO STUDENT_COURSE VALUES(10005,'ENGLISH',20010,89);
INSERT INTO STUDENT_COURSE VALUES(10006,'BANGLA',20002,80);
INSERT INTO STUDENT_COURSE VALUES(10006,'MATH',20007,40);
INSERT INTO STUDENT_COURSE VALUES(10006,'ENGLISH',20008,87);
INSERT INTO STUDENT_COURSE VALUES(10007,'BANGLA',20001,98);
INSERT INTO STUDENT_COURSE VALUES(10007,'MATH',20003,49);
INSERT INTO STUDENT_COURSE VALUES(10007,'ENGLISH',20010,99);
INSERT INTO STUDENT_COURSE VALUES(10008,'BANGLA',20009,87);
```

```
INSERT INTO STUDENT_COURSE VALUES(10008,'MATH',20003,58);
INSERT INTO STUDENT_COURSE VALUES(10008,'ENGLISH',20006,87);
INSERT INTO STUDENT_COURSE VALUES(10009,'BANGLA',20002,89);
INSERT INTO STUDENT_COURSE VALUES(10009,'MATH',20007,70);
INSERT INTO STUDENT_COURSE VALUES(10009,'ENGLISH',20008,86);
INSERT INTO STUDENT_COURSE VALUES(10010,'BANGLA',20002,87);
INSERT INTO STUDENT_COURSE VALUES(10010,'MATH'20004,38);
INSERT INTO STUDENT_COURSE VALUES(10010,'ENGLISH',20008,88);
```

For GRADE Table:

```
CREATE TABLE GRADE
( LOW_NUMBER NUMBER(10),
        HI_NUMBER NUMBER(10),
        GPA NUMBER(10),
GRADE VARCHAR2(10));

INSERT INTO GRADE VALUES (80,100,5.00,'A+');
INSERT INTO GRADE VALUES (70,79,4.00,'A');
INSERT INTO GRADE VALUES (60,69,3.50,'A-');
INSERT INTO GRADE VALUES (50,59,3.00,'B');
INSERT INTO GRADE VALUES (40,49,2.00,'C');
INSERT INTO GRADE VALUES (33,39,1.00,'D');
INSERT INTO GRADE VALUES (0,32,0.00,'F');
```

For LOGIN Table:

```
CREATE TABLE LOGIN
( USER_ID NUMBER(10) PRIMARY KEY,
        PASSWORD VARCHAR2(10),
        STATUS VARCHAR2(10));
INSERT INTO LOGIN VALUES (10001,'PS001','ACTIVE');
INSERT INTO LOGIN VALUES (10002,'PS002','ACTIVE');
INSERT INTO LOGIN VALUES (10003,'PS003','ACTIVE');
INSERT INTO LOGIN VALUES (10004,'PS004','ACTIVE');
INSERT INTO LOGIN VALUES (10005,'PS005','ACTIVE');
INSERT INTO LOGIN VALUES (10006,'PS006','ACTIVE');
INSERT INTO LOGIN VALUES (10007,'PS007','ACTIVE');
INSERT INTO LOGIN VALUES (10008,'PS008','ACTIVE');
INSERT INTO LOGIN VALUES (10009,'PS009','ACTIVE');
INSERT INTO LOGIN VALUES (10010,'PS010','ACTIVE');
INSERT INTO LOGIN VALUES (20001,'PS011','ACTIVE');
INSERT INTO LOGIN VALUES (20002,'PS012','ACTIVE');
INSERT INTO LOGIN VALUES (20003,'PS013','ACTIVE');
INSERT INTO LOGIN VALUES (20004,'PS014','ACTIVE');
INSERT INTO LOGIN VALUES (20005,'PS014','ACTIVE');
INSERT INTO LOGIN VALUES (20006,'PS015','ACTIVE');
INSERT INTO LOGIN VALUES (20007,'PS016','ACTIVE');
INSERT INTO LOGIN VALUES (20008,'PS017','ACTIVE');
INSERT INTO LOGIN VALUES (20009,'PS018','ACTIVE');
```

INSERT INTO LOGIN VALUES (20010,'PS019','ACTIVE');

Screenshot of Sample Data:

STUDENT_INFO:

ST_ID	SNAME	FNAME	MNAME	T_ID	PASSING_YEAR
10001	ALICE	BOB	GANIKA	20002	2015
10002	NAYEM	HASEM	NAZMA	20001	2014
10003	NADIA	AMIR	SETU	20006	2015
10005	JANNAT	RAJ	TULEI	20002	2017
10006	SOHID	MONCUR	KHADIJA	20007	2015
10007	RAHIM	KARIM	ROHIMA	20001	2018
10004	ARIF	KARIM	SANJIDA	20004	2018
10008	ASHIK	NOZRUL	AMINA	20003	2017
10009	OLIVA	JOY	EVANA	20005	2014
10010	SOJIB	SHIBLO	MITU	20006	2015

10 rows returned in 0.14 seconds CSV Export

TEACHER_INFO:

T_ID	TNAME	CNAME
20001	PRODIP	BANGLA
20002	MASUM	BANGLA
20003	BILLAH	MATH
20004	SHAKIB	MATH
20006	TANJIL	ENGLISH
20007	HASIB	MATH
20008	AMIN	ENGLISH
20009	FATIMA	BANGLA
20005	RAKIB	BANGLA
20010	KANIJ	ENGLISH

10 rows returned in 0.00 seconds

STUDENT_COURSE:

ST_ID	CNAME	T_ID	C_RESULT
10001	BANGLA	20001	88
10001	ENGLISH	20006	67
10003	BANGLA	20009	30
10003	MATH	20003	80
10004	BANGLA	20001	90
10004	ENGLISH	20006	37
10005	ENGLISH	20010	89
10006	MATH	20007	40
10006	ENGLISH	20008	87
10007	BANGLA	20001	98
10007	MATH	20003	49
10009	MATH	20007	70
10009	ENGLISH	20008	86
10010	BANGLA	20002	87
10010	MATH	20004	38
10001	MATH	20003	93
10002	BANGLA	20002	90
10002	MATH	20007	78
10002	ENGLISH	20008	95
10003	ENGLISH	20010	49
10004	MATH	20004	49
10005	BANGLA	20009	83
10005	MATH	20003	93
10006	BANGLA	20002	80
10007	ENGLISH	20010	99
10008	BANGLA	20009	87
10008	MATH	20003	58
10008	ENGLISH	20006	87
10009	BANGLA	20002	89
10009	ENGLISH	20008	86

More than 30 rows available. Increase rows selector to view more rows.

30 rows returned in 0.01 seconds CSV Export

GRADE:

LOW_NUMBER	HI_NUMBER	GPA	GRADE
70	79	4	A
60	69	4	A-
40	49	2	C
33	39	1	D
80	100	5	A+
50	59	3	B
0	32	0	F

7 rows returned in 0.02 seconds CSV Export

LOGIN:

USER_ID	PASSWORD	STATUS
10001	PS001	ACTIVE
10002	PS002	ACTIVE
10003	PS003	ACTIVE
10008	PS008	ACTIVE
10010	PS010	ACTIVE
20001	PS011	ACTIVE
20003	PS013	ACTIVE
20006	PS015	ACTIVE
20008	PS017	ACTIVE
20009	PS018	ACTIVE
20010	PS019	ACTIVE
10004	PS004	ACTIVE
10005	PS005	ACTIVE
10006	PS006	ACTIVE
10007	PS007	ACTIVE
10009	PS009	ACTIVE
20002	PS012	ACTIVE
20004	PS014	ACTIVE
20005	PS014	ACTIVE
20007	PS016	ACTIVE

20 rows returned in 0.06 seconds

6. View

1 . Student Information

```
CREATE VIEW stview1
AS SELECT *
FROM student_info
WHERE st_id = '10002';
```

2. Teacher Information

```
CREATE VIEW tview1
AS SELECT *
FROM teacher_info
WHERE t_id = '20001';
```

3. Result View

```
CREATE VIEW stcview1
```

```
AS SELECT *
FROM STUDENT_COURSE
WHERE st_id = '10004';
```

4. GPA Calculate

```
CREATE VIEW gpaview1
AS SELECT ROUND(avg(gpa),2) as AVG_GPA
FROM student_info,student_course,grade
WHERE c_result between low_number and hi_number
AND student_info.st_id=student_course.st_id
AND sname='NAYEM';
```

5. All Student's Gpa

```
CREATE VIEW gpaview2
AS SELECT student_info.st_id,round(avg(gpa),2) as cgpa
FROM student_info,student_course,grade
WHERE student_info.st_id=student_course.st_id and c_result between
low_number and hi_number
GROUP BY student_info.st_id order by student_info.st_id;
```

7. Procedures and Functions

PROCEDURE

1.Teacher Registration

```
create or replace procedure teacher_reg(name teacher_info.tname%type, cname
teacher_info.cname%type, pass login.password%type)
is
num teacher_info.t_id%type;
begin
Insert into teacher_info values (t_sq.nextval,name,cname);
select max(t_id) into num from teacher_info;
insert into login values(num,pass,'ACTIVE');
end;
```

2.Update Result

```
create or replace procedure Update_result (student_id
student_course.st_id%type, result student_course.c_result%type)
is
begin
update student_course set c_result=result where st_id=student_id;
end;
```

3.Student Registration

```
create or replace procedure student_reg(name student_info.sname%type, fname
student_info.fname%type, mname student_info.mname%type, tname
student_info.t_id%type,passing student_info.passing_year%type, pass
login.password%type)
is
num student_info.st_id%type;
t1 teacher_info.t_id%type;
t2 teacher_info.t_id%type;
t3 teacher_info.t_id%type;

begin
Insert into student_info values
(st_sq.nextval,name,fname,mname,tname,passing);
select max(st_id) into num from student_info;
insert into login values(num,pass,'ACTIVE');

select t_id into t1 from  (select t_id from teacher_info where cname='MATH'
Order by dbms_random.value)  where rownum =1;
select t_id into t2 from  (select t_id from teacher_info where cname='BANGLA'
Order by dbms_random.value)  where rownum =1;
select t_id into t3 from  (select t_id from teacher_info where cname='ENGLISH'
Order by dbms_random.value)  where rownum =1;

insert into student_course values(num,'MATH',t1,'');
insert into student_course values(num,'BANGLA',t2,'');
insert into student_course values(num,'ENGLISH',t3,'');
end;
```

4.Get bonus

```
create or replace procedure Bonus(teacher_id student_course.t_id%type)
is
i number(4);
result student_course.c_result%type;
```

```
Cursor c Is select st_id from student_course where t_id=teacher_id;
begin
for i in c loop
select c_result into result from student_course where st_id=i.st_id and
t_id=teacher_id;
if(result>70) then
   result:=result+5;
else
   result:=result+10;
end if;
update student_course set c_result=result where st_id=i.st_id and
t_id=teacher_id;
end loop;
end;
```

Function

1. Total Mark Calculator

```
create or replace function total_number(id student_info.st_id%type)
return number
is
total number(5);
begin
select sum(c_result) into total from student_course where st_id=id;
return total;
end;
```

2. Grade Calculator

```
create or replace function grade_check(num student_course.c_result%type)
return float
is
rgrade float(5);
begin
select grade into rgrade from Grade where num between low_number and
hi_number;
return rgrade;
end;
```

3. Check Password

```
create or replace function CheckPassword(uname login.user_id%type,pass
login.password%type)
return boolean
is
c number(2);
begin
c:=0;
select count(*) into c from login where user_id=uname and password=pass;
if c!=0 then
return true;
else
return false;
end if;
end;
```

8. Triggers

TRIGGER

1. Validate Info to add a new Teacher.

```
create or replace trigger t_id_tigger

before insert on teacher_info

for each row

declare

c number(2):=0;
```

```
begin
select count(*) into c from teacher_info where :old.t_id=:new.t_id;
if c>0 then
Raise_application_error(-20111, 'Duplicate Id');
end if;
end;
```

2. Check Activity Time

```
create or replace trigger time_check
before insert or delete or update on login
begin
if ( (to_char(SYSDATE,'D') not between '1' and '5') )
then
Raise_application_error(-20754,'Not working Day');
end if;
end;
```

3. Validate Info to add a new Student.

```
create or replace trigger st_id_tigger
before insert on student_info
for each row
declare
c number(2):=0;
begin
```

```
select count(*) into c from student_info where :old.st_id=:new.st_id;
if c>0 then
Raise_application_error(-20111, 'Duplicate Id');
end if;
end;
```

4. Result insert check

```
create or replace trigger result_check
before insert or update on student_course
for each row
Begin
if (:new.c_result <0) then
 Raise_application_error(-20115,'Marks can not be negetive');
else
 dbms_output.put_line('insert Successfully');
end if;
end;
```

9. Package and exception handling

1 . Package

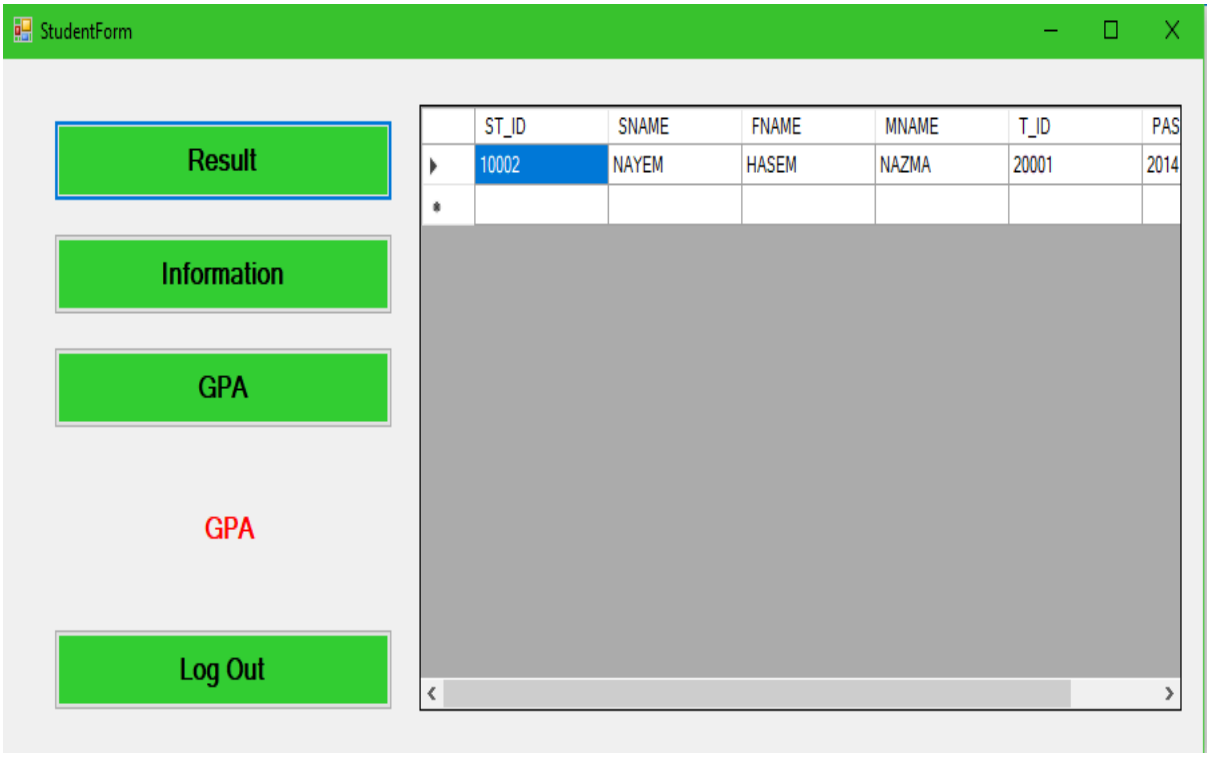
```
CREATE PACKAGE st_password AS
   PROCEDURE find_password(password login.password%type);
```

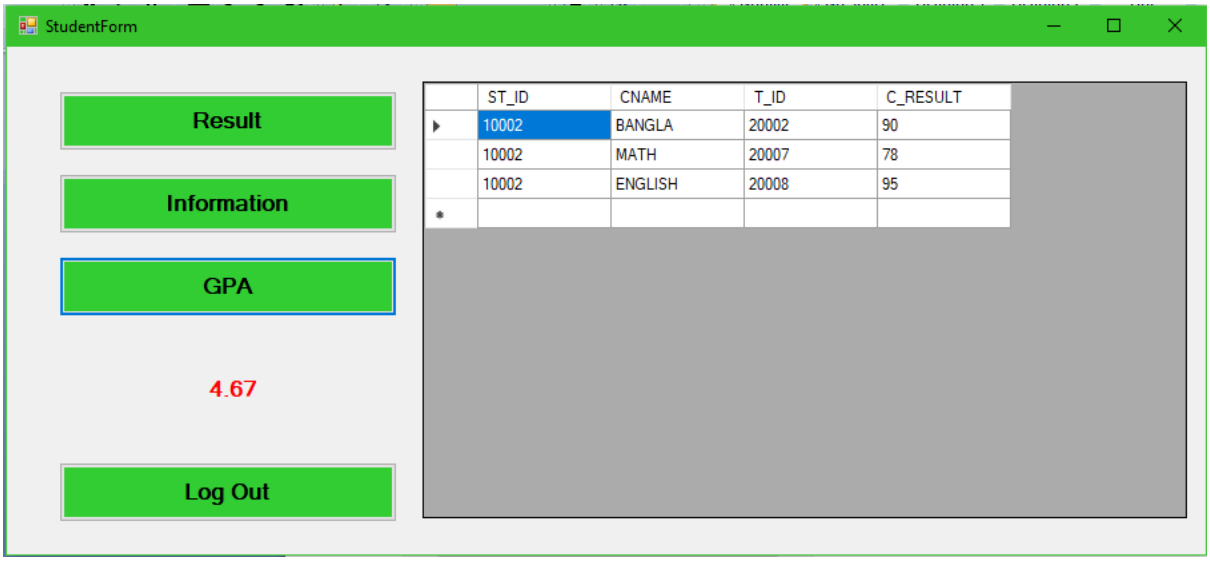
```
END st_password;

CREATE OR REPLACE PACKAGE BODY st_password AS
PROCEDURE find_password(password login.password%type) IS
 st_user login.user_id%type;
 BEGIN
  SELECT user_id INTO  st_user
  FROM login
  WHERE password = password;
  dbms_output.put_line(user_id);
 END find_password;
END st_password;
```

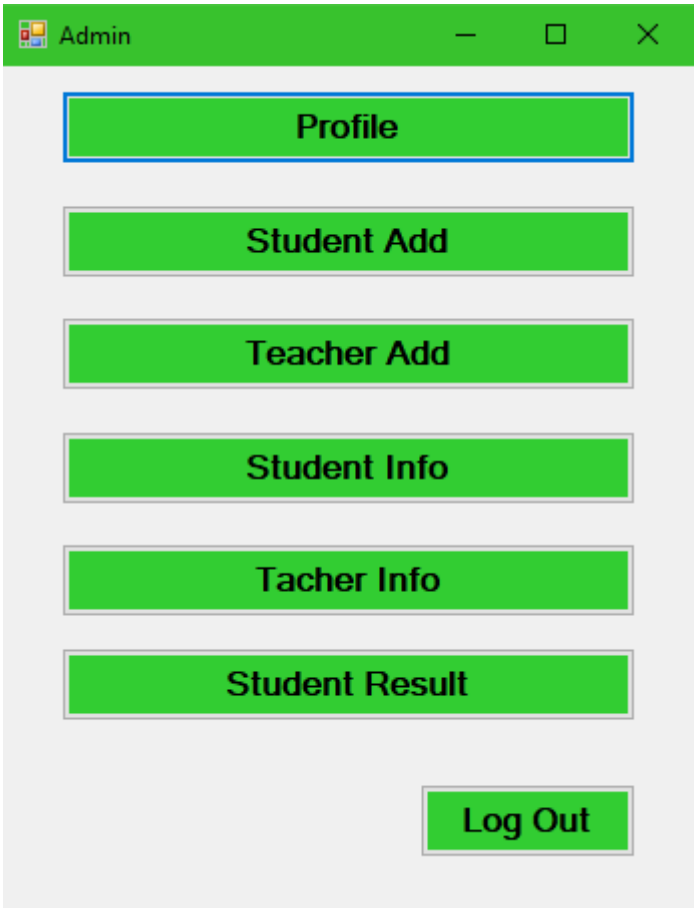
2. Exception handling

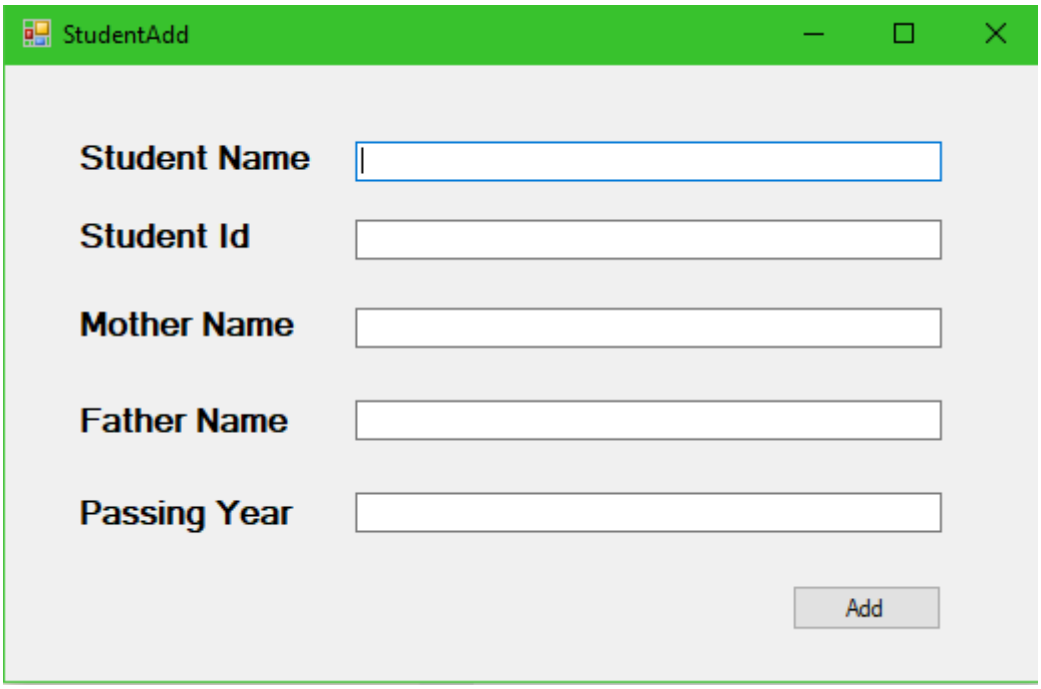
```
DECLARE
  s_id student_info.st_id%type := 10002;
  st_name student_info.SName%type;
  st_passingyear student_info.passing_year%type;
BEGIN
  SELECT  sname, passing_year INTO  st_name , st_passingyear
  FROM student_info
  WHERE st_id = s_id;
  DBMS_OUTPUT.PUT_LINE ('Name: '||  st_name);
  DBMS_OUTPUT.PUT_LINE ('Passing Year : ' || st_passingyear);

EXCEPTION
  WHEN no_data_found THEN
    dbms_output.put_line('No such Student!');
  WHEN others THEN
    dbms_output.put_line('Error!');
END;
```

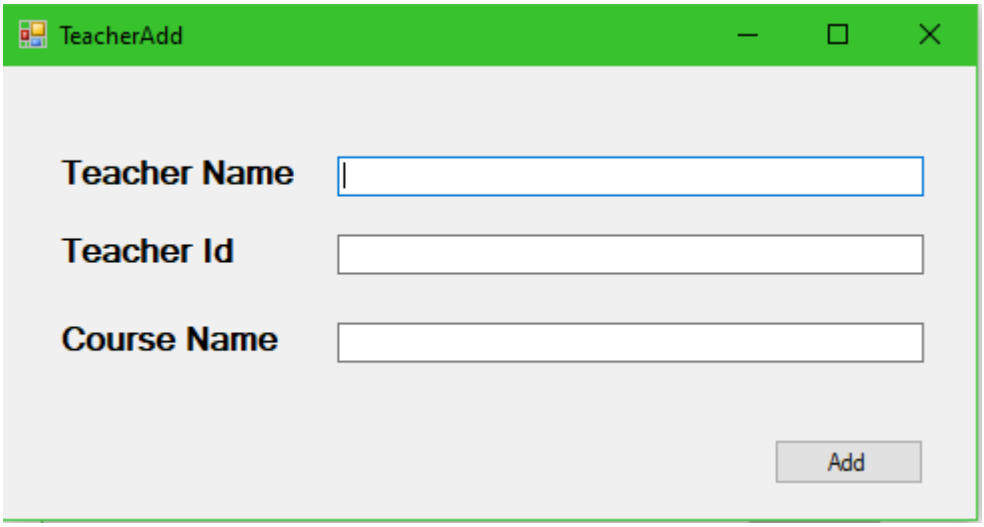
Interface


StudentForm
Result
Information
GPA
GPA
Log Out
ST_ID
SNAME
FNAME
MNAME
T_ID
PAS
10002
NAYEM
HASEM
NAZMA
20001
2014


StudentForm
Result
Information
GPA
4.67
Log Out
ST_ID
CNAME
T_ID
C_RESULT
10002
BANGLA
20002
90
10002
MATH
20007
78
10002
ENGLISH
20008
95


Admin
Profile
Student Add
Teacher Add
Student Info
Tacher Info
Student Result
Log Out

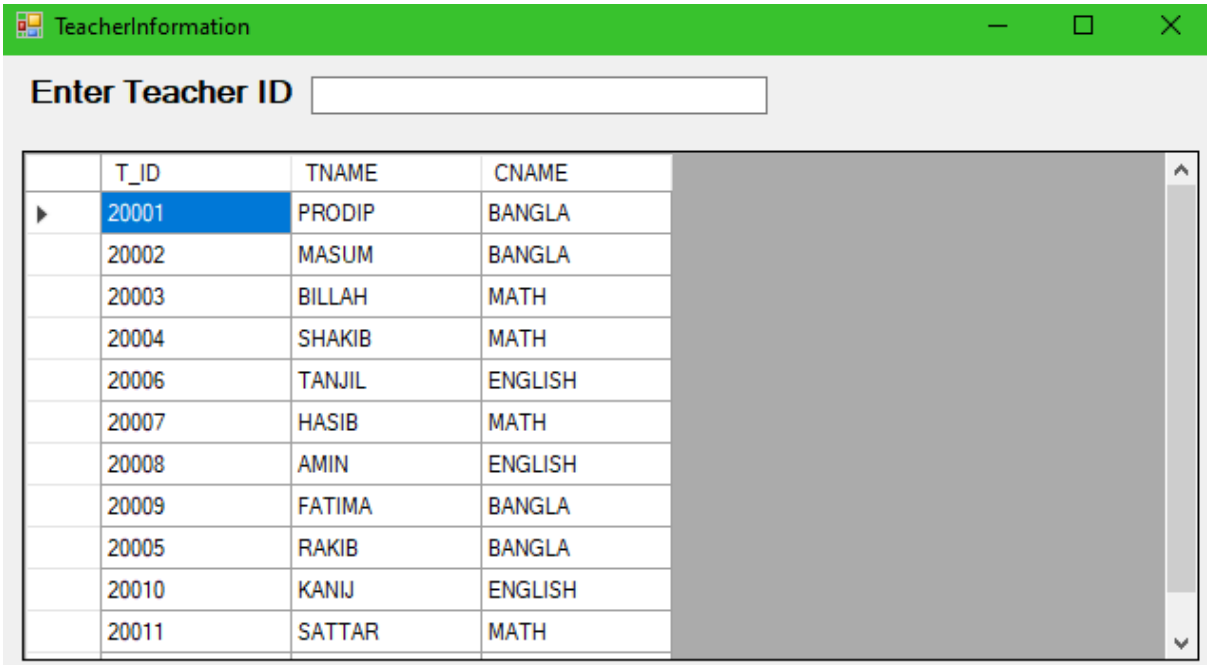

StudentAdd
Student Name
Student Id
Mother Name
Father Name
Passing Year
Add



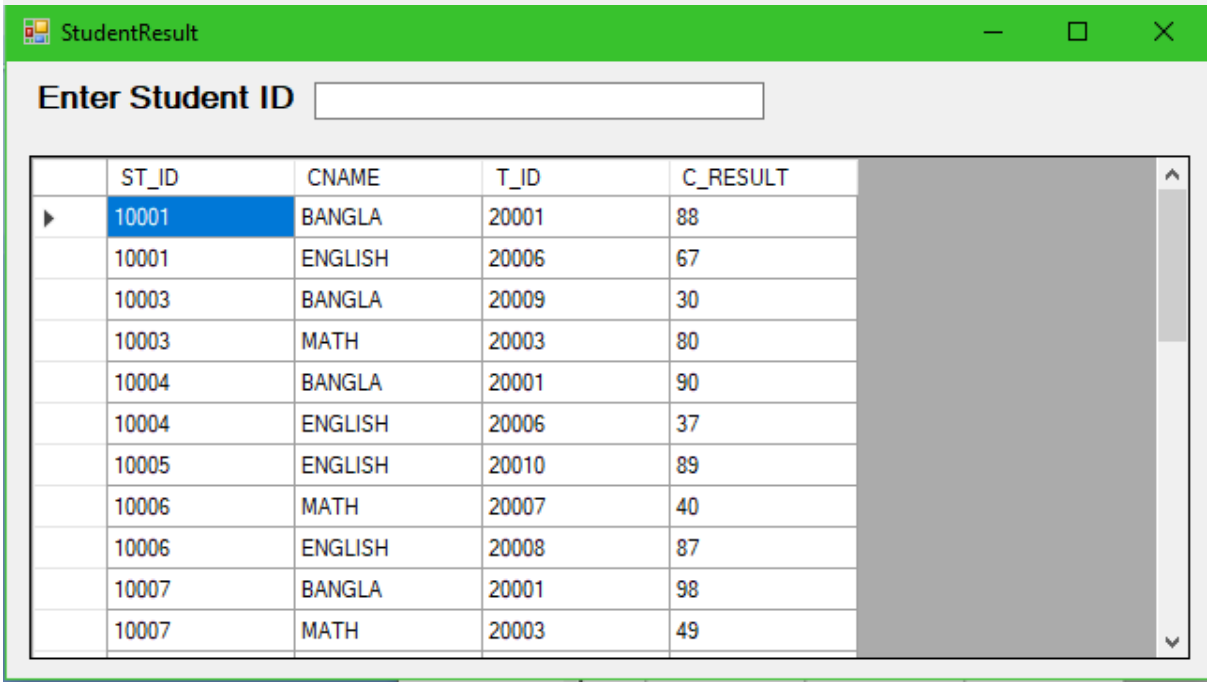
StudentInfomation

Enter Student ID

	ST_ID	SNAME	FNAME	MNAME	T_ID	PASSING_
▶	10001	ALICE	BOB	GANIKA	20002	2015
	10002	NAYEM	HASEM	NAZMA	20001	2014
	10003	NADIA	AMIR	SETU	20006	2015
	10005	JANNAT	RAJ	TULEI	20002	2017
	10006	SOHID	MONCUR	KHADIJA	20007	2015
	10007	RAHIM	KARIM	ROHIMA	20001	2018
	10011	SUBON	RIJON	MITHILA	20001	2020
	10012	NIBIR	ARIF	NARGIS	20001	2021
	10004	ARIF	KARIM	SANJIDA	20004	2018
	10008	ASHIK	NOZRUL	AMINA	20003	2017



T_ID	TNAME	CNAME
20001	PRODIP	BANGLA
20002	MASUM	BANGLA
20003	BILLAH	MATH
20004	SHAKIB	MATH
20006	TANJIL	ENGLISH
20007	HASIB	MATH
20008	AMIN	ENGLISH
20009	FATIMA	BANGLA
20005	RAKIB	BANGLA
20010	KANIJ	ENGLISH
20011	SATTAR	MATH



ST_ID	CNAME	T_ID	C_RESULT
10001	BANGLA	20001	88
10001	ENGLISH	20006	67
10003	BANGLA	20009	30
10003	MATH	20003	80
10004	BANGLA	20001	90
10004	ENGLISH	20006	37
10005	ENGLISH	20010	89
10006	MATH	20007	40
10006	ENGLISH	20008	87
10007	BANGLA	20001	98
10007	MATH	20003	49