A Project

Report

On

STUDENT MANAGEMENT SYSTEM

By

DARSHAN JAIN

Batch: 2022 – 7888

Center: Bangalore TC Palya

Under the Guidance of,

Chittaranjan Ghosh.

Technical Trainer

EduBridge

Introduction:

The student management system project report is used to formulate the idea of minimizing administrations' workload in managing students' information. This project report simplifies the needed information, including tracking for both parents and administrative staff, by coordinating scheduling and communications between professors and students.

. **Modules:-**

- Login Module
- GeneratedID Module
- Enroll Module
- View Balance Module
- Pay Tuition Fees Module
- Check Status Module

I have implemented this project using java 8.0 and Mysql 8.0
It is a console based project .

OBJECTIVE

The main objective of the Student Management System is to provide information about generationId of unique students with their respective gradeYear ,login details,Enrollment of the course,Tuition Fees,Viewing balance of each student after payment and status of the student.This project is developed by console app through administrative end.It provides teachers easy access for the student information and manages their time easily.

MODULES

1. **LOGIN MODULE**- The student login module is used for the user to enter their correct username and password provided by the administrator,for performing StudentOperations .Providing wrong credentials will display a message in the console with username/password incorrect.

2. **GENERATE_ID MODULE-** This module is used for generating 5 digit unique_id for each student by combining their individual id's to the grade Year in which they are studying.This id is generated by allowing the user to enter the student individual id and the respective gradeYear of the student.

3. **ENROLL MODULE** - In this module students are enrolled for the course by allowing the user to enter course id,course fees for each course provided by the administrator .It calculates the total course fees based on the course enrolled by the student by cost of each course .The Remaining balance is calculated by the total course fees after enrolling.

4.VIEW BALANCE MODULE - The view balance module is used for checking the students remaining balance,allowing the user to input the courseId of the respective course for displaying the remaining balance after registering for the course.

5.**PAY TUITION FEES MODULE** - The module is used for allowing the user to enter the amount to be paid for the entire duration by deduction from its overall remaining balance.If the user enters the payment more than the remaining balance ,the message will display invalid payment amount from the user in the console.

6.CHECK STATUS MODULE - The status module is used to display the information about the student details such as studentName,student id ,course name and remaining balance of the student by allowing the user to enter the correct student id and course id associated with the student in the record.

Software Requirements:

Back end: MySQL workbench 8.0 CE,Java 8.0

Browser: Google Chrome

Operating System: Windows10

Data Dictionary:

- Create Database StudentDBMS

MySQL 8.0 Command Line Client

```
mysql> use studentdbms;
Database changed
mysql> show tables;
+----------------------+
| Tables_in_studentdbms |
+----------------------+
| course               |
| registration         |
| student              |
+----------------------+
3 rows in set (0.01 sec)

mysql> desc student;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| username    | varchar(20) | NO   |     | NULL    |       |
| pass_word   | varchar(20) | NO   |     | NULL    |       |
| studentName | varchar(50) | YES  |     | NULL    |       |
| gradeYear   | int         | NO   |     | NULL    |       |
| studentID   | int         | NO   | PRI | NULL    |       |
| generatedID | varchar(20) | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
6 rows in set (0.01 sec)

mysql> clear
mysql> desc course;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| courseID      | int         | NO   | PRI | NULL    |       |
| courseName    | varchar(20) | YES  | UNI | NULL    |       |
| courseFees    | bigint      | NO   |     | NULL    |       |
| courseDuration | int        | YES  |     | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
4 rows in set (0.01 sec)

mysql> desc registration;
+-----------------+--------+------+-----+---------+-------+
| Field           | Type   | Null | Key | Default | Extra |
+-----------------+--------+------+-----+---------+-------+
| registrationID  | int    | NO   | PRI | NULL    |       |
| studentID       | int    | YES  | MUL | NULL    |       |
| courseID        | int    | YES  | MUL | NULL    |       |
| TotalCourseFees | double | YES  |     | NULL    |       |
| RemainingBalance | bigint | YES |     | NULL    |       |
+-----------------+--------+------+-----+---------+-------+
5 rows in set (0.00 sec)

mysql>
```

Screenshots:-

Login Module:

1)

```
========================== WELCOME TO STUDENT MANAGEMENT SYSTEM ========================
================================================================================
                Username : Shantanul08

                Password : 12345
================================================================================
================================================================================
========================== STUDENT OPERATIONS ==========================
================================================================================
                Enter 1-> Generate ID of the student
                Enter 2-> Enroll
                Enter 3-> TutionBalance
                Enter 4-> payTutionfee.
                Enter 5-> status
                Enter 6-> logout
================================================================================
                Enter a valid input between 1 to 6:
```

2)

```
========================== WELCOME TO STUDENT MANAGEMENT SYSTEM ========================
================================================================================
                Username : Shantanul10

                Password : 12345
================================================================================
================================================================================
Incorrect username or password!!
================================================================================
```

GeneratedID Module:

1)

```
================================================================================
================================================================================
============================ STUDENT OPERATIONS ============================
================================================================================
                    Enter 1-> Generate ID of the student
                    Enter 2-> Enroll
                    Enter 3-> TutionBalance
                    Enter 4-> payTutionfee.
                    Enter 5-> status
                    Enter 6-> logout
================================================================================
                    Enter a valid input between 1 to 6:
1
Enter the student_id:
1000
Enter the gradeYear of student:
9
Generated id is: 91000
================================================================================
```

2)

```
================================================================================
============================ STUDENT OPERATIONS ============================
================================================================================
                    Enter 1-> Generate ID of the student
                    Enter 2-> Enroll
                    Enter 3-> TutionBalance
                    Enter 4-> payTutionfee.
                    Enter 5-> status
                    Enter 6-> logout
================================================================================
                    Enter a valid input between 1 to 6:
1
Enter the student_id:
1000
Enter the gradeYear of student:
10
Invalid student id or gradeYear
--------------------------------------------------------------------------------
```

Enroll Module:

1)

```
================================================================================
============================ STUDENT OPERATIONS ============================
================================================================================
                    Enter 1-> Generate ID of the student
                    Enter 2-> Enroll
                    Enter 3-> TutionBalance
                    Enter 4-> payTutionfee.
                    Enter 5-> status
                    Enter 6-> logout
================================================================================
                    Enter a valid input between 1 to 6:
2
Enter the courseID :
104
Enter the courseFees :
1200
Enter the TotalCourseFees :
9400
Enter the Remaining Balance :
20600
CourseEnrolled : DEVOPS
Total course fees is :10600.0
Remaining Balance :10000
true
--------------------------------------------------------------------------------
```

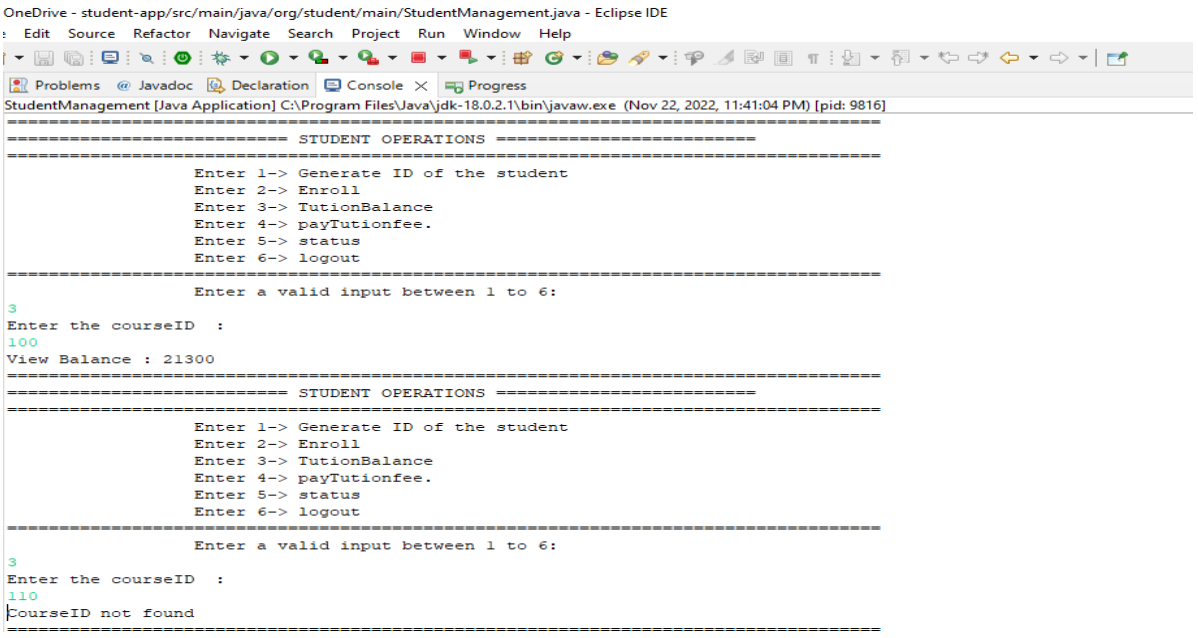
2)

```
--------------------------------------------------------------------------------
========================== STUDENT OPERATIONS ==========================
================================================================================
                  Enter 1-> Generate ID of the student
                  Enter 2-> Enroll
                  Enter 3-> TutionBalance
                  Enter 4-> payTutionfee.
                  Enter 5-> status
                  Enter 6-> logout
================================================================================
                  Enter a valid input between 1 to 6:
2
Enter the courseID :
110
Enter the courseFees :
4567
Enter the TotalCourseFees :
7890
Enter the Remaining Balance :
50000
Operation not performed
```

3)

```
================================================================================
========================== STUDENT OPERATIONS ==========================
================================================================================
                  Enter 1-> Generate ID of the student
                  Enter 2-> Enroll
                  Enter 3-> TutionBalance
                  Enter 4-> payTutionfee.
                  Enter 5-> status
                  Enter 6-> logout
================================================================================
                  Enter a valid input between 1 to 6:
2
Enter the courseID :
104
Enter the courseFees :
1200
Enter the TotalCourseFees :
9400
Enter the Remaining Balance :
10000
CourseEnrolled : DEVOPS
Total course fees is :10600.0
Remaining Balance not performed
--------------------------------------------------------------------------------
```

View Balance Module:



PayTuitionFees Module:

1)

```
================================================================================
========================== STUDENT OPERATIONS ==========================
================================================================================
                    Enter 1-> Generate ID of the student
                    Enter 2-> Enroll
                    Enter 3-> TutionBalance
                    Enter 4-> payTutionfee.
                    Enter 5-> status
                    Enter 6-> logout
================================================================================
                    Enter a valid input between 1 to 6:
4
Enter the studentID of the student  :
1000
Enter the payment:
5000
Tution balance fee is :16300
================================================================================
========================== STUDENT OPERATIONS ==========================
================================================================================
                    Enter 1-> Generate ID of the student
                    Enter 2-> Enroll
                    Enter 3-> TutionBalance
                    Enter 4-> payTutionfee.
                    Enter 5-> status
                    Enter 6-> logout
================================================================================
                    Enter a valid input between 1 to 6:
4
Enter the studentID of the student  :
1000
Enter the payment:
50000
Invalid Tution Amount !!!
================================================================================
```

2)

```
================================================================================
========================== STUDENT OPERATIONS ==========================
================================================================================
                    Enter 1-> Generate ID of the student
                    Enter 2-> Enroll
                    Enter 3-> TutionBalance
                    Enter 4-> payTutionfee.
                    Enter 5-> status
                    Enter 6-> logout
================================================================================
                    Enter a valid input between 1 to 6:
4
Enter the studentID of the student  :
1100
Enter the payment:
50000
Invalid studentID
```

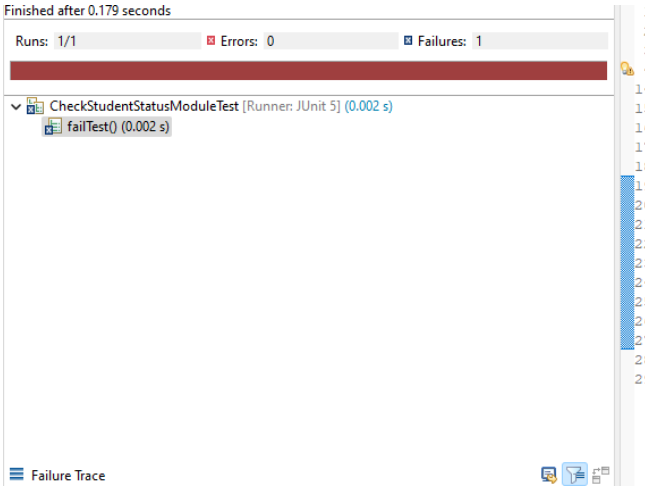
Status Module:

```
============================ STUDENT OPERATIONS ============================
=============================================================================================
                    Enter 1-> Generate ID of the student
                    Enter 2-> Enroll
                    Enter 3-> TutionBalance
                    Enter 4-> payTutionfee.
                    Enter 5-> status
                    Enter 6-> logout
=============================================================================================
                    Enter a valid input between 1 to 6:
5
Enter studentID:
1000
Enter courseID:
100
********************STUDENT DETAIL****************************************
*******************************************************************************
Student name : Shantanu
Student ID : 1000
Course name : DBMS
RemainingBalance : 16300
=============================================================================================
============================ STUDENT OPERATIONS ============================
=============================================================================================
                    Enter 1-> Generate ID of the student
                    Enter 2-> Enroll
                    Enter 3-> TutionBalance
                    Enter 4-> payTutionfee.
                    Enter 5-> status
                    Enter 6-> logout
=============================================================================================
                    Enter a valid input between 1 to 6:
5
Enter studentID:
1100
Enter courseID:
130
********************STUDENT DETAIL****************************************
*******************************************************************************
Invalid studentID/courseID
```

Logout:

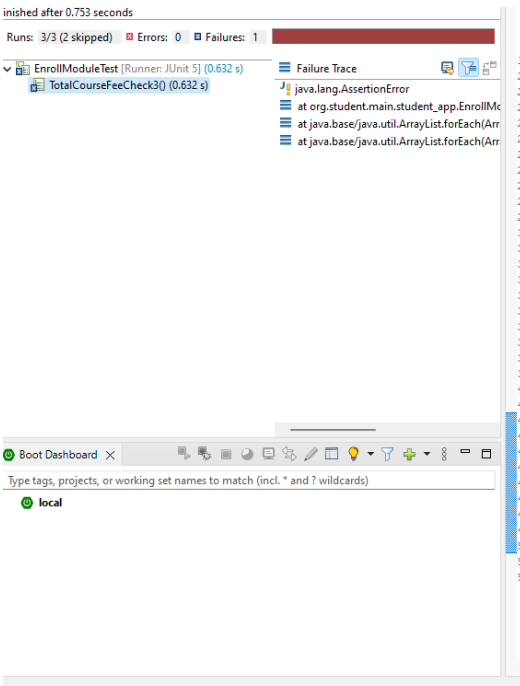
```
=============================================================================================
============================ STUDENT OPERATIONS ============================
=============================================================================================
                    Enter 1-> Generate ID of the student
                    Enter 2-> Enroll
                    Enter 3-> TutionBalance
                    Enter 4-> payTutionfee.
                    Enter 5-> status
                    Enter 6-> logout
=============================================================================================
                    Enter a valid input between 1 to 6:
6
Successfully logged out Bye!!
*********************************
```

Check status module test:



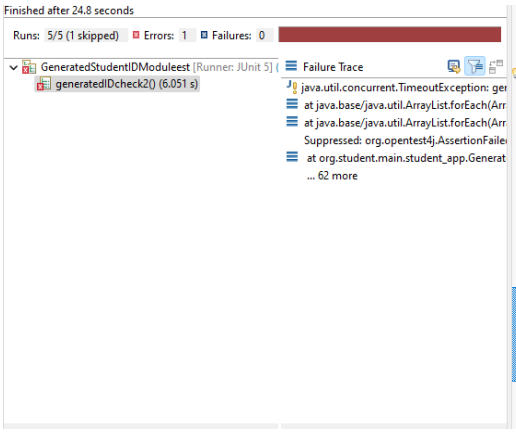

```java
package org.student.main.student_app;


import static org.junit.jupiter.api.Assertions.assertAll;

public class CheckStudentStatusModuleTest {
String string = null;


    @Test
    public void failTest() {
    int studentID = 1005;
    int courseID = 110;
     if((studentID == 1005)&&(courseID == 110)) {
         fail("StudentID and CourseID not found");
     }

    }
}

```

Enroll Module Test:



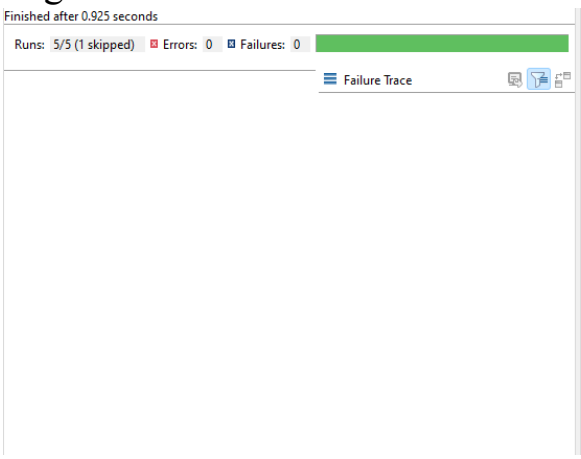

```java
package org.student.main.student_app;

import static org.junit.Assert.assertFalse;

@TestMethodOrder(MethodOrderer.OrderAnnotation.class)
public class EnrollModuleTest {
    final Logger log=Logger.getLogger(org.student.main.dao.StudentOperations.class.getName());

    @Order(1)
    @Test
    @EnabledOnOs(value = OS.LINUX)
public void  TotalCourseFeecheck1() throws ClassNotFoundException, SQLException, RemainingBalanc

        log.info("TotalCourseFeecheck1 invoked");
         assertTrue(StudentOperations.enroll(100, 900, 26600, 23300));
                    }
    @Order(2)
     @Test
     @EnabledOnOs(value = OS.WINDOWS)
     @EnabledOnJre(value =  JRE.JAVA_12)
     public void TotalCourseFeeCheck2() throws ClassNotFoundException, SQLException, RemainingBa
     {
         log.info("TotalCourseFeecheck2 invoked");
         assertFalse(StudentOperations.enroll(110, 9876, 9800, 21000));

     }
    @Order(3)
     @Test
     @EnabledOnOs(value = {OS.LINUX, OS.WINDOWS})
     public void TotalCourseFeeCheck3() throws ClassNotFoundException, SQLException, RemainingBa
     {
         log.info("TotalCourseFeecheck3 invoked");
         assertFalse(StudentOperations.enroll(103, 1700, 8400, 17600));

     }
}

```

Generated StudentID Module Test:



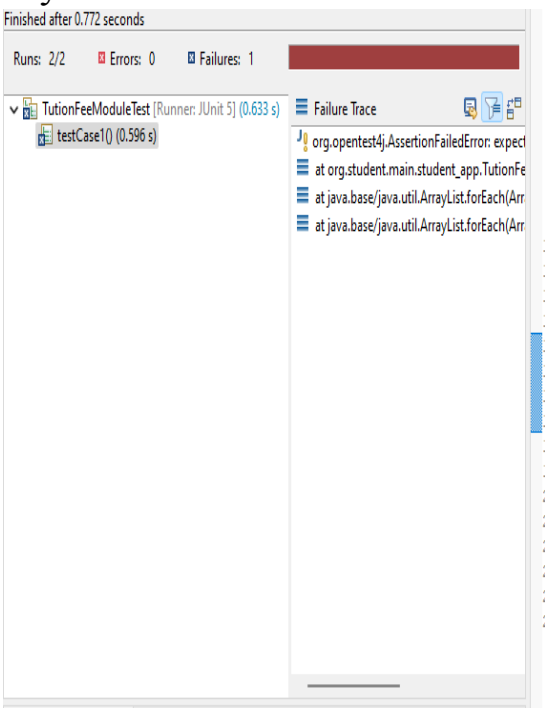

```java
package org.student.main.student_app;


import static org.junit.jupiter.api.Assertions.assertEquals;

public class GeneratedStudentIDModuleest {
    @RepeatedTest(3)
     public void generatedIDcheck1() throws ClassNotFoundException, SQLException, InterruptedExc
         assertEquals("91000",StudentOperations.generateStudentID(1000, 9));

     }
    @Test
    @Timeout(value=6000,unit = TimeUnit.MILLISECONDS)
    public void generatedIDcheck2() throws ClassNotFoundException, SQLException{
     assertTimeout(Duration.ofSeconds(6),()->StudentOperations.generateStudentID(1000, 9));

    }
    @Test
    @Disabled
    public void generatedIDcheck3() throws ClassNotFoundException, SQLException, InterruptedExce
         assertEquals("91005",StudentOperations.generateStudentID(1000, 9));

       }
}

```

Login Module Test:



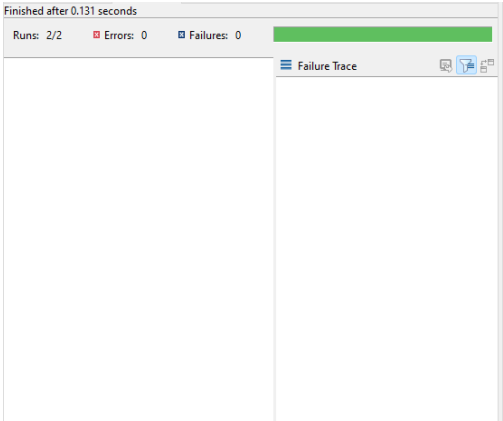

```java
package org.student.main.student_app;

import static org.junit.jupiter.api.Assertions.assertNotSame;

public class LoginModuleTest {
    @RepeatedTest(3)

     public void logintest1() {
           assertSame(true,StudentOperations.login("Shantanu108", "12345"));

     }

    @Test
    public void logintest2() {
        assertNotSame(true,StudentOperations.login("Sushma", "567890"));
    }

    @Test
    @Disabled
    public void logintest3() {
        assertNotSame(false,StudentOperations.login("Soundarya", "99099"));
    }
}
```

Pay Tuition Fees Module Test:




```java
package org.student.main.student_app;

import static org.junit.jupiter.api.Assertions.assertEquals;
import static org.junit.jupiter.api.Assertions.assertNotEquals;

import java.sql.SQLException;

import org.junit.jupiter.api.Test;
import org.student.main.dao.StudentOperations;
import org.student.main.exception.InvalidPaymentException;

public class TutionFeeModuleTest {

    @Test
     public void testCase1() throws ClassNotFoundException, SQLException, InvalidPaymentException
        assertEquals(9800, StudentOperations.payTutionFee(1004, 100));
    }

    @Test
     public void testCase2() throws ClassNotFoundException, SQLException, InvalidPaymentException
        assertNotEquals(9800, StudentOperations.payTutionFee(1004, 100));
    }

}
```

View Balance Module Test:




```java
package org.student.main.student_app;

import static org.junit.jupiter.api.Assertions.assertFalse;

public class ViewBalanceModuleTest {
    @Test(timeout = 1000)
   public void Balancecheck1() throws ClassNotFoundException, SQLException, CourseIDNotFoundExcep
        assertTrue( true ,"Invalid courseID");


   }
     @Test
     public void Balancecheck2() throws ClassNotFoundException, SQLException, CourseIDNotFoundExc
           assertFalse( false,"Incorrect");


       }

}
```

THANK YOU