**Suyash Tandon Divya Kharbanda**

standon1@binghamton.edudkharba1@binghamton.edu

**Documentation**

The Student Registration System uses Oracle’s PL/SQL and JDBC connection. It is a menu driven Desktop application which displays the below data.

* Show Students
* Show Courses
* Show prerequisites
* Show logs
* Show classes
* Add Students
* Show all Direct and Indirect Prerequisites
* Enroll students in classes
* Delete Students

**Sequence:**

We have created a sequence to automatically generate the log id starting from 3-digit number i.e. 100 to 999.

**Trigger:**

We have created the following triggers:

* For Logs table:

1. On insertion in enrollments, there should be entry in log table
2. On deletion in enrollments, there should be entry in log table
3. On insertion in student, there should be entry in log table
4. On deletion in students, there should be entry in log table

* Triggers created for other Dependencies:

1. On deletion in students, there should be deletion in enrollments table
2. On insertion in enrollments, there should be increment in class size
3. On deletion in enrollments, there should be decrement in class size

**Package and Procedure:**

**Package created:** Project2\_core\_package

**Show\_students:** This procedure is used to display all the records present in the students table in the database. For this purpose, it executes the select query on students table. It further returns a cursor of type ‘ref cursor’ which is then fetched into java code to display all the records on the UI. **Procedure name:** **Show\_students**

**show\_courses:** This procedure is used to display all the records present in the students table in the database. For this purpose, it executes the select query on students table. It further returns a cursor of type ‘ref cursor’ which is then fetched into java code to display all the records on the UI. **Procedure name: show\_courses**

**show\_prerequisite:** This procedure is used to display all the records present in the prerequisite table in the database. For this purpose, it executes the select query on students table. It further returns a cursor of type ‘ref cursor’ which is then fetched into java code to display all the records on the UI. **Procedure name: show\_prerequisite**

**show\_classes:** This procedure is used to display all the records present in the classes table in the database. For this purpose, it executes the select query on students table. It further returns a cursor of type ‘ref cursor’ which is then fetched into java code to display all the records on the UI. **Procedure name: show\_classes**

**show\_enrollments:** This procedure is used to display all the records present in the enrollments table in the database. For this purpose, it executes the select query on students table. It further returns a cursor of type ‘ref cursor’ which is then fetched into java code to display all the records on the UI. **Procedure name: show\_enrollments**

**show\_logs:** This procedure is used to display all the records present in the log table in the database. For this purpose, it executes the select query on students table. It further returns a cursor of type ‘ref cursor’ which is then fetched into java code to display all the records on the UI. **Procedure name: show\_logs**

**add\_students:** It adds Students into table. It checks the validity on sid, if it is not found not then it should through error message: The sid is invalid. If the sid has not taken any course, then the error message should be: The student has not taken any course. **Procedure name: add\_students**

**Parameters Used:** sid\_t IN students.sid%type, firstname\_t IN students.firstname%type,

lastname\_t IN students.lastname%type, status\_t IN students.status%type,

gpa\_t IN students.gpa%type, email\_t IN students.email%type,

msg OUT varchar

**class\_information:** This procedure is checking the validation on classid, it is not found then it should through error: The cid is invalid. If no student has taken any course then it should through report: No student is enrolled in the class. **Procedure name: class\_information**

**Parameters:** classid\_n IN classes.classid%type,

message OUT varchar2,cl\_cursor OUT data\_cursor

,stu\_cursor OUT data\_cursor

**show\_prereq:** This procedure is used to show all the direct and indirect prerequisites.

**Parameters:** prerequisites\_dept\_code in prerequisites.dept\_code%type,

prerequisites\_course\_number in prerequisites.course\_no%type,

Message out varchar

**Students\_information:** For this we verified the following validations:

* If the class is not in the classes table, report “The cid is invalid.”
* If no student has taken or is taking the class, report “No student is enrolled in the class.”

**Parameters:** student\_id in students.sid%type,

Message OUT varchar,

Student\_cursor OUT data\_cursor

**enroll\_a\_student\_into\_a\_class:**

We have implemented the following checks to this implementation:

* If class is not in the class table then print, " The classid is invalid".
* If the enrollment of the student into a class would cause “class\_size > limit”, reject the enrollment and
* print “The class is closed.”
* If the student is already in the class, print “The student is already in the class.”
* If the student is already enrolled in two other classes in the same semester and the same year, print “You are overloaded.”
* and allow the student to be enrolled.
* If the student is already enrolled in three other classes in the same semester and the same year, report “Students cannot be
* enrolled in more than three classes in the same semester.” and reject the enrollment.
* If the student has not completed the required prerequisite courses with minimum grade “C”, reject the enrollment
* and report “Prerequisite courses have not been completed.”

**Parameters:** classes\_classid in classes.classid%type,

student\_sid in students.sid%type,

message out varchar

**delete\_student:** If the student is not in the students table, report “The sid is invalid.

We have implemented the following checks for this implementation:

* If the student is not in the students table, report “The sid is invalid.”
* If the classid is not in the classes table, report “The classid is invalid.”
* If the student is not enrolled in the class, report “The student is not enrolled in the class.”
* If dropping the student from the class would cause a violation of the prerequisite requirement for another class, then reject the drop attempt and report “The drop is not permitted because another class uses it as a prerequisite.”
* In all the other cases, the student should be dropped from the class. If the class is the last class for the student, report “This student is not enrolled in any classes.”
* If the student is the last student in the class, report “The class now has no students.”

**Parameters:** sidn in students.sid%type,out\_message out varchar

**References:** We have implemented the queries mostly with the use of cursors, and we mostly learned its implementation from the following references mentioned below:

* https://www.tutorialspoint.com/plsql/plsql\_cursors.htm
* http://www.oracle.com/technetwork/issue-archive/2013/13-mar/o23plsql-1906474.html
* <https://docs.oracle.com/cd/B12037_01/appdev.101/b10807/13_elems011.htm>
* Various YouTube Videos