Group Members Name:

Name: Prince Jodhani Student ID: 149455206

Name: Juvairiya Shaikh Studen ID: 145185203

Name: Juan Castel Student ID: 147891204

ER Diagram of College registration system

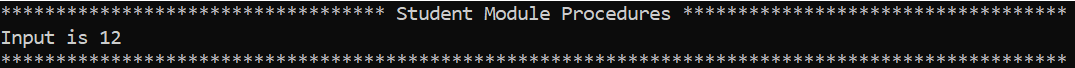
Diagram

Description automatically generated

Procedures created for tables:

|  |  |  |
| --- | --- | --- |
| Prince Jodhani | Juvairiya Shaikh | Juan Castel |
| Students | Professors | Stud\_Fees |
| Registration | Advisors | Courses |

**Students Table Module**



**spStudentsInsert:**

**Description:** Insert stud\_id to students table given by user input and it’s dummy record. It will output with success or not.

Input Parameter:

|  |  |
| --- | --- |
| Input | Type |
| sid | Integer |

Meaningful Error Handling:

|  |  |
| --- | --- |
| Error | Description |
| If row count 0 | Stud: (input) is already exist |
| If row count is not 0 | Stud: (input) inserted successfully |

PLSQL output:

Graphical user interface, application

Description automatically generated

C++ output:

Graphical user interface, text

Description automatically generated

**spStudentsSelect:**

**Description:** Select particular record given by user input from the students Table.

Input Parameter:

|  |  |
| --- | --- |
| Input | Type |
| sid | Integer |

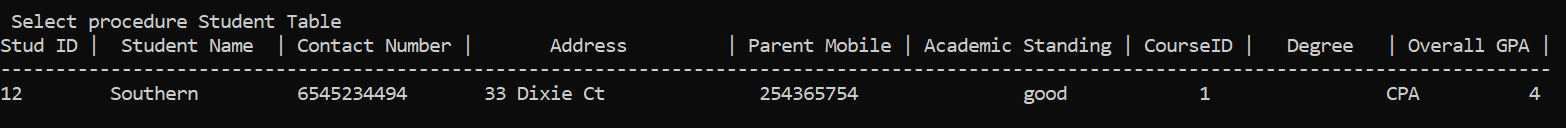
Meaningful Error Handling:

|  |  |
| --- | --- |
| Error | Description |
| If row count 0 | No any Row found for Select |
| If row count is not 0 | Send data to out variables. |

PLSQL output:

Graphical user interface

Description automatically generated

C++ output:

**spStudentsUpdate:**

**Description:** updating the email to ‘new@myseneca.ca’ of stud\_id based on user input. It will give result output as updated or not.

Input Parameter:

|  |  |
| --- | --- |
| Input | Type |
| sid | Integer |

Meaningful Error Handling:

|  |  |
| --- | --- |
| Error | Description |
| If row count 0 | This student id is not exist to update |
| If row count is not 0 | Stud: (input) : email updated |

PLSQL output:

Graphical user interface, application

Description automatically generated

C++ output:

Graphical user interface, text

Description automatically generated

**spStudentsDelete:**

**Description:** Deleting particular record based on stud\_id input given by user. And it will provide weather it is deleted or not.

Input Parameter:

|  |  |
| --- | --- |
| Input | Type |
| sid | Integer |

Meaningful Error Handling:

|  |  |
| --- | --- |
| Error | Description |
| If row count 0 | No any row found for delete |
| If row count is not 0 | stud\_id: (input) deleted successfully |

PLSQL output:

Graphical user interface, text, application

Description automatically generated

C++ output:

Graphical user interface, text

Description automatically generated

**Registration Table Module**

Text

Description automatically generated

**spRegistrationInsert:**

**Description:** Insert stud\_id to registration table giver by user input. It will output with success or not.

Input Parameter:

|  |  |
| --- | --- |
| Input | Type |
| sid | Integer |

Meaningful Error Handling:

|  |  |
| --- | --- |
| Error | Description |
| If row count 0 | Stud: (input) is already exist |
| If row count is not 0 | Reg for: (input) inserted |

PLSQL output:

Graphical user interface, application

Description automatically generated

C++ output:

Graphical user interface, text

Description automatically generated

**spRegistrationSelect:**

**Description:** Select particular record given by user input from the Registration Table.

Input Parameter:

|  |  |
| --- | --- |
| Input | Type |
| sid | Integer |

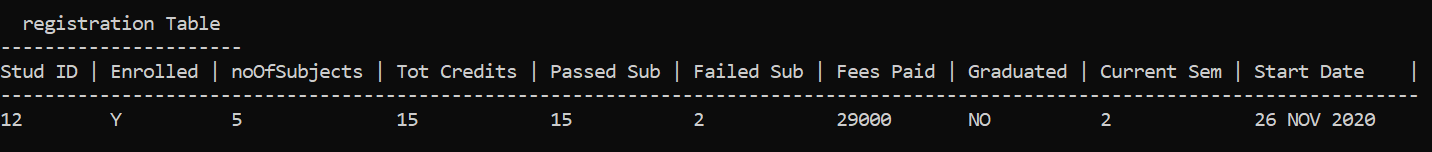
Meaningful Error Handling:

|  |  |
| --- | --- |
| Error | Description |
| If row count 0 | No any Row found for Select |
| If row count is not 0 | Send data to out variables. |

PLSQL output:

Graphical user interface, application

Description automatically generated

C++ output:

**spRegistrationUpdate:**

**Description:** updating the enrolled to ‘N’ of stud\_id based on user input. It will give result output as updated or not.

Input Parameter:

|  |  |
| --- | --- |
| Input | Type |
| sid | Integer |

Meaningful Error Handling:

|  |  |
| --- | --- |
| Error | Description |
| If row count 0 | This student id is not exist to update |
| If row count is not 0 | Stud: (input) : updated! |

PLSQL output:

Graphical user interface, application

Description automatically generated

C++ output:

Graphical user interface

Description automatically generated

**spRegistrationDelete:**

**Description:** Deleting particular record based on stud\_id input given by user. And it will provide weather it is deleted or not.

Input Parameter:

|  |  |
| --- | --- |
| Input | Type |
| sid | Integer |

Meaningful Error Handling:

|  |  |
| --- | --- |
| Error | Description |
| If row count 0 | No any row found for delete |
| If row count is not 0 | Registration stud\_id: (input) deleted |

PLSQL output:

Graphical user interface, application

Description automatically generated

C++ output:

Text

Description automatically generated

A screenshot of a computer screen

Description automatically generated with low confidence**This is the whole c++ terminal screenshot**

**spProfessorsInsert:**

**Description:** Insert profID to Professors table given by user input and its dummy record. It will output with success or not

Input Parameter:

|  |  |
| --- | --- |
| **Input** | **Type** |
| profID | Integer |

**Meaningful Error Handling:**

|  |  |
| --- | --- |
| **Error** | **Description** |
| If row count is 0 | Prof: (input) already exists |
| If row count is not 0 | Prof: (input) inserted successfully |

**spProfessorsUpdate:**

**Description:** Update ProfID to Professors table given by user input and its dummy record. It will output with success or not

Input Parameter:

|  |  |
| --- | --- |
| **Input** | **Type** |
| profID | Integer |

**Meaningful Error Handling:**

|  |  |
| --- | --- |
| **Error** | **Description** |
| If row count is 0 | Prof: (input) already exists |
| If row count is not 0 | Prof: (input) updated successfully |

**spProfessorsDelete:**

**Description:** Delete profID to Professors table given by user input and its dummy record. It will output with success or not

**Input Parameter:**

|  |  |
| --- | --- |
| **Input** | **Type** |
| profID | Integer |

**Meaningful Error Handling:**

|  |  |
| --- | --- |
| **Error** | **Description** |
| If row count is 0 | Prof: (input) already exists |
| If row count is not 0 | Prof: (input) deleted successfully |

**C++ output:**

Graphical user interface, text

Description automatically generated

**spAdvisorsInsert**:

**Description**: Insert AID to Advisors table given by user input and its dummy record. It will output with success or not

**Input Parameter:**

|  |  |
| --- | --- |
| **Input** | **Type** |
| AID | Integer |

**Meaningful Error Handling:**

|  |  |
| --- | --- |
| **Error** | **Description** |
| If row count is 0 | Adv: (input) already exists |
| If row count is not 0 | Adv: (input) inserted successfully. |

**spAdvisorsUpdate**:

**Description**: Update AID to Advisors table given by user input and its dummy record. It will output with success or not

**Input Parameter:**

|  |  |
| --- | --- |
| **Input** | **Type** |
| AID | Integer |

**Meaningful Error Handling:**

|  |  |
| --- | --- |
| **Error** | **Description** |
| If row count is 0 | Adv: (input) already exists |
| If row count is not 0 | Adv: (input) updated successfully. |

**spAdvisorsDelete**:

**Description**: Delete AID to Advisors table given by user input and its dummy record. It will output with success or not

**Input Parameter:**

|  |  |
| --- | --- |
| **Input** | **Type** |
| AID | Integer |

**Meaningful Error Handling:**

|  |  |
| --- | --- |
| **Error** | **Description** |
| If row count is 0 | Adv: (input) already exists |
| If row count is not 0 | Adv: (input) deleted successfully. |

**C++ output:**

Graphical user interface, text

Description automatically generated

Text

Description automatically generated

**First Table: STUD\_FEES TABLE**

Procedure insert\_fees:

Description: Insert stdId to the stud\_fees table given by user in the input to insert a new fee

Input Parameter:

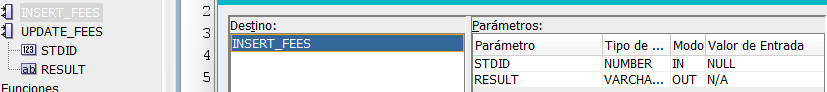
|  |  |
| --- | --- |
| **Input** | **Type** |
| stdId | Number |

Meaningful Error Handling:

If row count 0 -> Insertion was done

If row count is not 0 -> Insertion Succesfuly

PLSQL Output:



**First Table: STUD\_FEES TABLE**

Procedure update\_fees :

Description: Insert stdId to the stud\_fees table given by user in the input to update any category in the table

Input Parameter:

|  |  |
| --- | --- |
| **Input** | **Type** |
| stdId | Number |

Meaningful Error Handling:

If row count 0 -> Update was done

If row count is not 0 -> Update Succesfuly

PLSQL Output:

Graphical user interface, application

Description automatically generated

**First Table: COURSES TABLE**

Procedure update\_fees :

Description: Insert crsId to the courses table given by user in the input to delete the specific course

Input Parameter:

|  |  |
| --- | --- |
| **Input** | **Type** |
| crsId | Number |

Meaningful Error Handling:

If row count 0 -> Delete was done

If row count is not 0 -> Delete Succesfuly

PLSQL Output:

