**Practical:** 3.5

**Aim:**

Write a PL/SQL Function (Stored Procedure) to retrieve all the student information whose branch is CSE and using java program display all selected records on console.

**Program code:**

import java.sql.\*;

public class usingProcedure {

// JDBC driver name and database URL

static final String JDBC\_DRIVER = "org.postgresql.Driver";

static final String DB\_URL = "jdbc:postgresql://127.0.0.1:5432/s2d130050131042";

// Database credentials

static final String USER = "postgres";

static final String PASS = "12345";

public static void main(String[] args) {

Connection conn = null;

CallableStatement cStmt = null;

try{

//STEP 2: Register JDBC driver

Class.forName(JDBC\_DRIVER);

//STEP 3: Open a connection

System.out.println("Connecting to a selected database...");

conn = DriverManager.getConnection(DB\_URL, USER, PASS);

System.out.println("Connected database successfully...");

//STEP 4: Execute a query

System.out.println("Enrollno: 130050131042");

cStmt = conn.prepareCall("{call student1()}");

cStmt.execute();

System.out.println("Selected Data is:");

System.out.println();

ResultSet rs = cStmt.getResultSet();

System.out.print("id:");

System.out.print("Name:");

System.out.print("Branch:");

while (rs.next())

{

System.out.println();

System.out.print(rs.getInt(1)+"\t");

System.out.print(rs.getString(2)+"\t");

System.out.print(rs.getString(3)+"\t");

System.out.println();

}

cStmt.close();

}catch(SQLException se){

//Handle errors for JDBC

se.printStackTrace();

}catch(Exception e){

//Handle errors for Class.forName

e.printStackTrace();

}finally{

//finally block used to close resources

try{

if(cStmt!=null)

conn.close();

cStmt.close();

}catch(SQLException se){

}// do nothing

try{

if(conn!=null)

conn.close();

}catch(SQLException se){

se.printStackTrace();

}//end finally try

}//end try

System.out.println("program completed");

}//end main

}//end

**Procedure:**

-- Function: student1()

-- DROP FUNCTION student1();

CREATE OR REPLACE FUNCTION student1()

RETURNS TABLE(id integer, name character varying, branch character varying) AS

$BODY$

DECLARE

BEGIN

RETURN QUERY

SELECT \* FROM student

WHERE student.branch='CSE';

RETURN;

END

$BODY$

LANGUAGE plpgsql VOLATILE

COST 100

ROWS 1000;

ALTER FUNCTION student1()

OWNER TO postgres;

**Input Output:**

