**Practical: 3.5**

**Aim:**

Write a PL/SQL Function (StoredProcedure) 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**.\*;**

publicclassusingProcedure**{**

// JDBC driver name and database URL

staticfinal String JDBC\_DRIVER **=**"org.postgresql.Driver"**;**

staticfinal String DB\_URL **=**"jdbc:postgresql://127.0.0.1:5432/s2b130050131044"**;**

// Database credentials

staticfinal String USER **=**"postgres"**;**

staticfinal String PASS **=**"12345"**;**

publicstaticvoid main**(**String**[]**args**){**

Connection conn **=null;**

CallableStatementcStmt**=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: 130050131044"**);**

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

cStmt**.**execute**();**

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

System**.**out**.**println**();**

ResultSetrs**=**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**(**"prgram 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:**

