Department of Information Technology

Academic Year: 2022-23 | **Name of Student: Harsh Joshi**
Semester: III | **Student ID: 22204012**
Class / Branch: SE (IT) | **Subject: SQL Lab**

Experiment No:8

Aim: To demonstrate of database connectivity using JDBC.

Software used:- Eclipse IDE, PostgreSQL, JDBC Driver

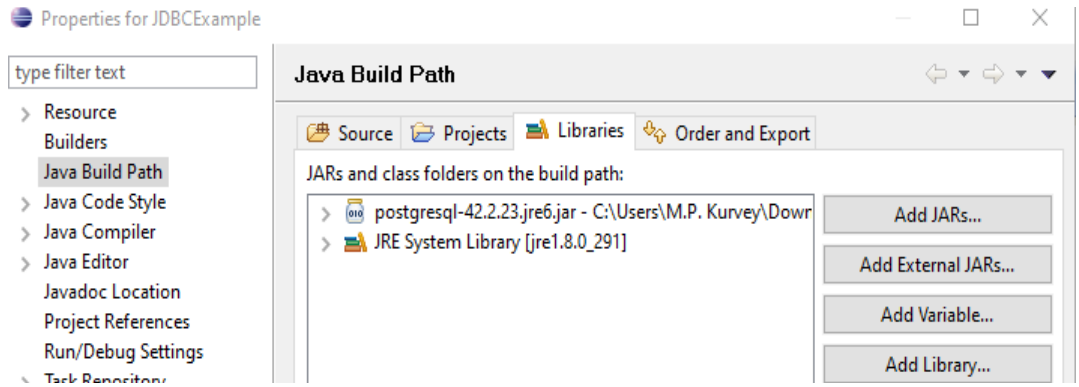
Downloading JDBC Driver for PostgreSQL

Binary Jar file downloads of the JDBC driver is available.

https://jdbc.postgresql.org/download.html

3.1 Load the driver in Eclipse IDE and Register Driver

3.2 Add the driver which is external JAR file to Libraries section of the project

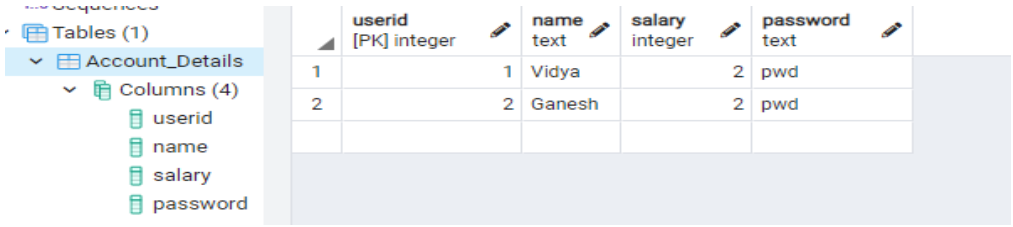


Step 2 : Create Accounts Database

In PostgreSQL design database having name Accounts.

Create table Account_Details in this database. And add columns in this table. Insert rows in this table.

Following is the Structure of Account_Details table to be created:



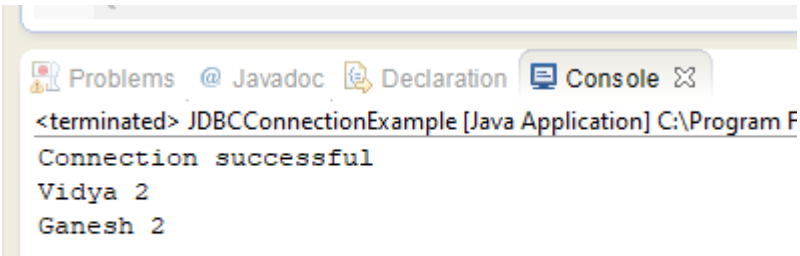
	userid [PK] integer	name text	salary integer	password text
1	1	Vidya	2	pwd
2	2	Ganesh	2	pwd

Step 3 : Write Java code to Establish the connection with Database and select multiple columns

```
import java.sql.*;
public class JDBCConnectionExample {

    private final String url = "jdbc:postgresql://localhost/Accounts";
    private final String username = "postgres";
    private final String pwd = "user";
    Connection connect;

    //Step4:Establish the connection
    //connect method to connect to database
    private void connect()
    {
            try
        {
            //getConnection: static: Return a connection object
            connect = DriverManager.getConnection(url, username, pwd);
        }
        catch(SQLException e)
        {
            System.out.println("Connection issues");
            e.printStackTrace();
        }

        if(connect!=null)
            System.out.println("Connection successful");
        else
            System.out.println("Connection issues");

    }
```

```
//execute
private void execute()
{
    try
    {
        //Create a statement
        Statement stmt = connect.createStatement(); //created a Statement object
        ResultSet result = stmt.executeQuery("Select name, salary "
                + "from public.\"Account_Details\"    " );


        while(result.next())
        {
            System.out.println(result.getString(1)+ " " + result.getInt(2));
        }
    }
    catch(SQLException e)
    {
        System.out.println("excution issues");
        e.printStackTrace();
    }


}




   public static void main(String[] args) {
       // TODO Auto-generated method stub
       JDBCConnectionExample jdbc = new JDBCConnectionExample();
       jdbc.connect();
       jdbc.execute();

   }

}
```

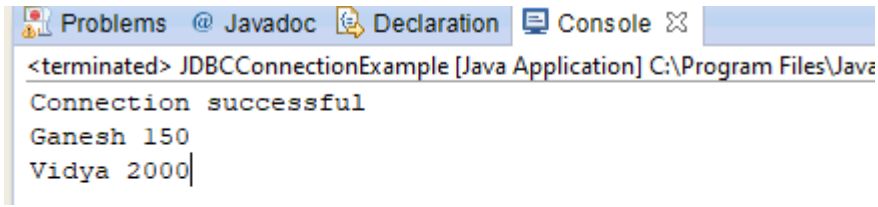
Output:



Step 4: Write Java code to update table

```
//execute
private void execute()
{
    try
    {
        //Create a statement
        PreparedStatement stmt = connect.prepareStatement("update "
                + "public.\"Account_Details\" SET salary = 2000 where userid=1 ")
        stmt.execute();
    }
    catch(SQLException e)
    {
        System.out.println("excution issues");
        e.printStackTrace();
    }


}
```

Output:



Conclusion : Thus we have implemented database connectivity using JDBC.