DAO PATTERN (STUDENT)

DTO CLASS OR POJO CLASS

```
package com.student.dto;
//DTO class(POJO class)
public class Student {

    //instance variables
    private int id;
    private String name;
    private long phone;
    private String mail;
    private String branch;
    private String location;
    private String pass;
    private String date;


    //GETTERS AND SETTERS
    public int getId() {
```

```
    return id;
}
public void setId(int id) {
    this.id = id;
}
public String getName() {
    return name;
}
public void setName(String name) {
    this.name = name;
}
public long getPhone() {
    return phone;
}
public void setPhone(long phone) {
    this.phone = phone;
}
public String getMail() {
    return mail;
}
public void setMail(String mail) {
    this.mail = mail;
```

```
}
public String getBranch() {
    return branch;
}
public void setBranch(String branch) {
    this.branch = branch;
}
public String getLocation() {
    return location;
}
public void setLocation(String location) {
    this.location = location;
}
public String getPass() {
    return pass;
}
public void setPass(String pass) {
    this.pass = pass;
}
public String getDate() {
    return date;
}
```

```
    public void setDate(String date) {
        this.date = date;
    }
    @Override
    public String toString() {
        return "Student [id=" + id + ", name=" + name + ", phone="
+ phone + ", mail=" + mail + ", branch=" + branch
               + ", location=" + location + "]";
    }
}
```

DAO INTERFACE

```
package com.student.dao;

import java.util.List;
import com.student.dto.Student;

public interface StudentDAO {
    public boolean insertStudent(Student s);
    public boolean updateStudent(Student s);
    public boolean deleteStudent(Student s);
    public Student getStudent(String mail,String pass);
    public List<Student> getStudent();
}
```

DAO IMPLIMENTATION CLASS

```
package com.student.dao;


import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;

import com.student.connectors.ConnectionFactory;
import com.student.dto.Student;

public class StudentDAOImpl implements StudentDAO {
    private Connection con;


    public StudentDAOImpl() {
        this.con = ConnectionFactory.requestConnection();
    }

    @Override
    public boolean insertStudent(Student s) {
        //JDBC logics for insert operation
        String query="INSERT INTO STUDENT2 VALUES
(0,?,?,?,?,?,?,SYSDATE())";
        PreparedStatement ps=null;
        int res=0;
        try {
            ps=con.prepareStatement(query);
            ps.setString(1,s.getName());
            ps.setLong(2,s.getPhone());
            ps.setString(3, s.getMail());
            ps.setString(4, s.getBranch());
            ps.setString(5,s.getLocation());
            ps.setString(6, s.getPass());
            res=ps.executeUpdate();//res stores the value whether 
the data inserted or not
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        if(res>0)
        {
            return true;
        }
        else
        {
            return false;
        }
    }

    @Override
    public boolean updateStudent(Student s) {
        String query="UPDATE STUDENT1 SET PHONE=? WHERE ID=?";
        int res=0;
        PreparedStatement ps=null;
        try {
```

```
        ps=con.prepareStatement(query);
        //ps.setString(1,s.getName());
        ps.setLong(1,s.getPhone());
        ps.setInt(2, s.getId());
        //ps.setString(4, s.getBranch());
        //ps.setInt(5, s.getLid());
        //ps.setString(6, s.getPass());
        res=ps.executeUpdate();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    if(res>0)
    {
        return true;
    }
    else
    {
        return false;
    }
}
```

```
@Override
public boolean deleteStudent(Student s) {
    String query="DELETE FROM STUDENT WHERE ID=?";
    int res=0;
    PreparedStatement ps=null;
    try {
        ps=con.prepareStatement(query);
        ps.setInt(1,s.getId());
        //ps.setLong(2,s.getPhone());
        //ps.setString(3, s.getMail());
        //ps.setString(4, s.getBranch());
        //ps.setInt(5, s.getLid());
        //ps.setString(6, s.getPass());
        res=ps.executeUpdate();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    if(res>0)
    {
        return true;
    }
```

```
        else
        {
            return false;
        }
    }

    @Override
    public Student getStudent(String mail, String pass) {
        String query="SELECT * FROM STUDENT2 WHERE MAILID=? AND 
PASSWORD=?";
        Student s=null;
        PreparedStatement ps=null;
        ResultSet rs=null;
        try {
            ps=con.prepareStatement(query);
            ps.setString(1, mail);
            ps.setString(2, pass);
            rs=ps.executeQuery();
            while(rs.next())
            {
                s=new Student();
                //String name=rs.getString("name");
            //s.setName(name);
            s.setId(rs.getInt("id"));
            s.setName(rs.getString("name"));
            s.setPhone(rs.getLong("phone"));
            s.setMail(rs.getString("mailid"));
            s.setBranch(rs.getString("branch"));
            s.setLocation(rs.getString("location"));
            s.setPass(rs.getString("password"));
            s.setDate(rs.getString("date"));
        }
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    return s;
}

@Override
public List<Student> getStudent() {
    ArrayList<Student> students=new ArrayList<Student>();
    Student s=null;
    String query="SELECT * FROM STUDENT2";
```

```
PreparedStatement ps=null;
ResultSet rs=null;
try {
    ps=con.prepareStatement(query);
    rs=ps.executeQuery();
    while(rs.next())
    {
        s=new Student();
        //String name=rs.getString("name");
        //s.setName(name);
        s.setId(rs.getInt("id"));
        s.setName(rs.getString("name"));
        s.setPhone(rs.getLong("phone"));
        s.setMail(rs.getString("mailid"));
        s.setBranch(rs.getString("branch"));
        s.setLocation(rs.getString("location"));
        s.setPass(rs.getString("password"));
        s.setDate(rs.getString("date"));
        students.add(s);
    }
} catch (SQLException e) {
    // TODO Auto-generated catch block
```

```
            e.printStackTrace();
        }

        return students;
    }
}
```

CONNECTOR FACTORY CLASS

```
package com.student.connectors;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class ConnectionFactory {
    public static Connection requestConnection()
    {
        Connection con=null;
        String url="jdbc:mysql://localhost:3306/company_1";
        String user="root";
        String password="tiger";
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            con=DriverManager.getConnection(url, user, password);

        } catch (ClassNotFoundException | SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        return con;
    }
}
```

TEST(MAIN) CLASS

```
package com.student.main;

import java.util.Iterator;
import java.util.List;
import java.util.Scanner;

import com.student.dao.StudentDAO;
import com.student.dao.StudentDAOImpl;
import com.student.dto.Student;
```

```
public class Test {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        Student s=new Student();
        StudentDAO sdao=new StudentDAOImpl();

        //get all the data
        List<Student> students=sdao.getStudent();
        Iterator<Student> it=students.iterator();
        while(it.hasNext())
        {
            s=it.next();
            System.out.println(s);
        }

        //inserting the student data
        System.out.println("Enter the name:");
        String name=sc.next();
        System.out.println("Enter the Phone number:");
        long phone=sc.nextLong();
        System.out.println("Enter the mail ID:");
```

```
String mail=sc.next();
System.out.println("Enter the branch:");
String branch=sc.next();
System.out.println("Enter your location ID:");
int lid=sc.nextInt();
System.out.println("Enter your password:");
String pass1=sc.next();
System.out.println("Confirm your password:");
String pass2=sc.next();
if(pass1.equals(pass2))
{
    s.setName(name);
    s.setPhone(phone);
    s.setMail(mail);
    s.setBranch(branch);
    s.setLid(lid);
    s.setPass(pass1);

    boolean res=sdao.insertStudent(s);
    if(res)
    {
        System.out.println("Signup successful");
```

```
            }
            else
            {
                System.out.println("failed to signup");
            }
        }

        //updating student data
        System.out.println("Enter the phone number to be updated");
        s1.setPhone(sc.nextLong());
        boolean res=sdao.updateStudent(s1);
        if(res)
        {
            System.out.println("phone number updated to 
"+s1.getPhone()+"successfully");
        }
        else
        {
            System.out.println("failed to update");
        }
    }
}
```