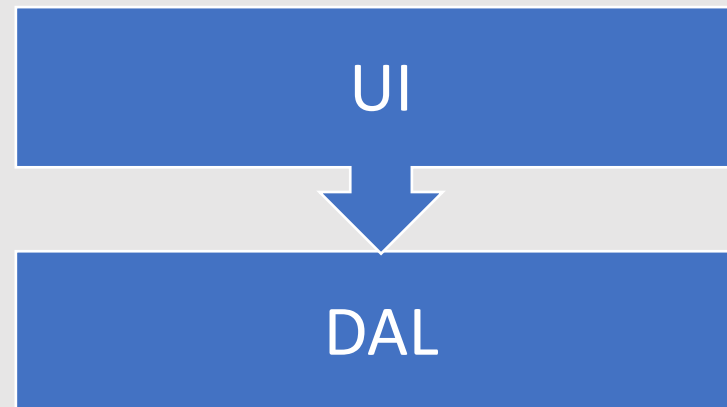


JAVA JDBC

Thực hành 2

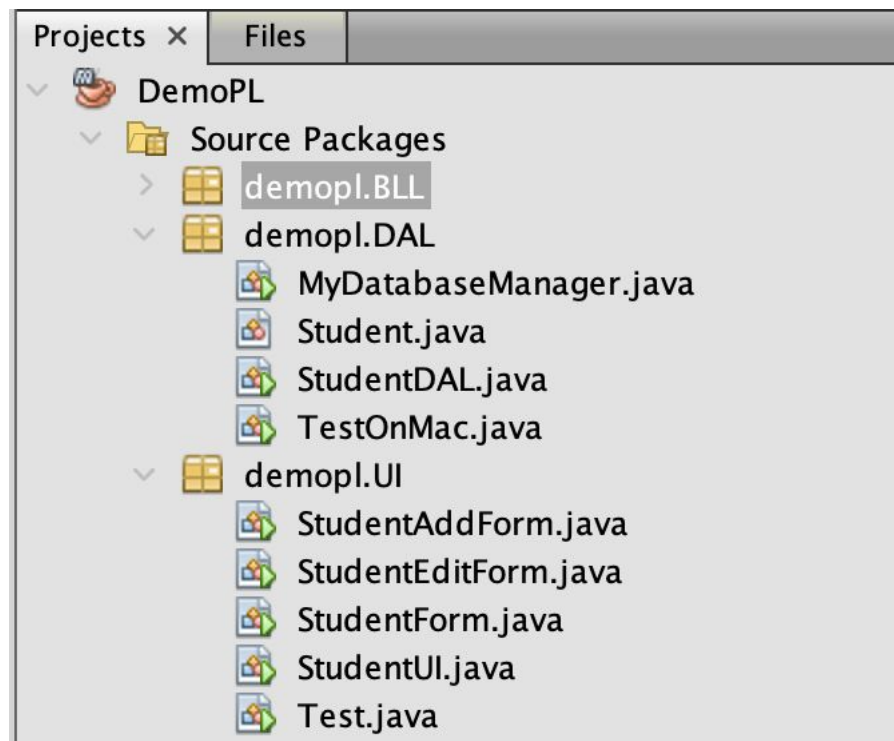
Nội dung chính

- Viết 2 Layer
 - DAL: Truy xuất CSDL
 - UI: hiển thị dữ liệu



Cấu trúc project

- Mỗi layer là 1 package
 - DAL
 - UI



1. Lớp DAL: StudentDAL

- Code lại các hàm của lớp StudentDAL
 - Code đọc CSDL: giữ lại và chuyển dữ liệu thành mảng
 - Hiển thị dữ liệu ☐ chuyển sang lớp UI

```
//1 layer
public void readStudents() throws SQLException {
    String query = "SELECT * FROM Person WHERE EnrollmentDate >0";
    ResultSet rs = StudentDAL.doReadQuery(query);
    if (rs != null) {
        int i = 1;
        System.out.println("TT \t PersonID \t FirstName \t\t LastName");
        while (rs.next()) {

            System.out.print(i + "\t" + rs.getInt("PersonID"));
            System.out.println("\t\t" + rs.getString("FirstName") +
                               "\t\t\t" + rs.getString("LastName"));
            i++;
        }
    }
}
```

```
//2 layer
public ArrayList readStudent() throws SQLException
{
    String query = "SELECT * FROM Person WHERE EnrollmentDate >0";
    ResultSet rs = StudentDAL.doReadQuery(query);
    ArrayList list = new ArrayList();

    if (rs != null) {
        int i = 1;

        while (rs.next()) {
            Student s = new Student();
            s.setPersonId(rs.getInt("PersonID"));
            s.setFirstName(rs.getString("FirstName"));
            s.setLastName(rs.getString("LastName"));
            list.add(s);
        }
    }
    return list;
}
```

1. Lớp DAL: StudentDAL

```
public void findStudent(String fullName) throws SQLException {
    String query = "SELECT * FROM Person WHERE concat(FirstName, ' ', LastName) LIKE ?";
    PreparedStatement p = StudentDAL.getConnection().prepareStatement(query);
    p.setString(1, "%" + fullName + "%");
    ResultSet rs = p.executeQuery();

    if (rs != null) {
        int i = 1;
        while (rs.next()) {
            System.out.print(rs.getInt("PersonID") + " - " //3 layer
                System.out.println(rs.getString("Lastname") + rs.getString("Firstname"));
            i++;
        }
    } else {
        System.out.println("Not found");
    }
}

public List findStudents(String fullName) throws SQLException
{
    String query = "SELECT * FROM Person WHERE concat(FirstName, ' ', LastName) LIKE ?";
    PreparedStatement p = StudentDAL.getConnection().prepareStatement(query);
    p.setString(1, "%" + fullName + "%");
    ResultSet rs = p.executeQuery();
    List list = new ArrayList();

    if (rs != null) {
        int i = 1;

        while (rs.next()) {
            Student s = new Student();
            s.setPersonId(rs.getInt("PersonID"));
            s.setFirstName(rs.getString("FirstName"));
            s.setLastName(rs.getString("LastName"));
            list.add(s);
        }
    }
    return list;
}
```

2. Lớp UI: StudentUI

- Gọi lớp DAL để hiển thị dữ liệu

```
public class StudentUI {  
  
    /**  
     * @param args the command line arguments  
     */  
    public static void main(String[] args) { ...10 lines }  
  
    public static void listStudent() throws SQLException  
    {  
        StudentDAL std = new StudentDAL();  
        ArrayList list = std.readStudent();  
        for(int i=0; i< list.size(); i++)  
        {  
            Student s = (Student)list.get(i);  
            System.out.println(s.getPersonId() + " " + s.getFirstName() + " " + s.getLastName());  
        }  
    }  
}
```

3. Lớp UI: StudentForm

- Dùng Jtable để hiển thị data
- Hàm convert mảng thành model

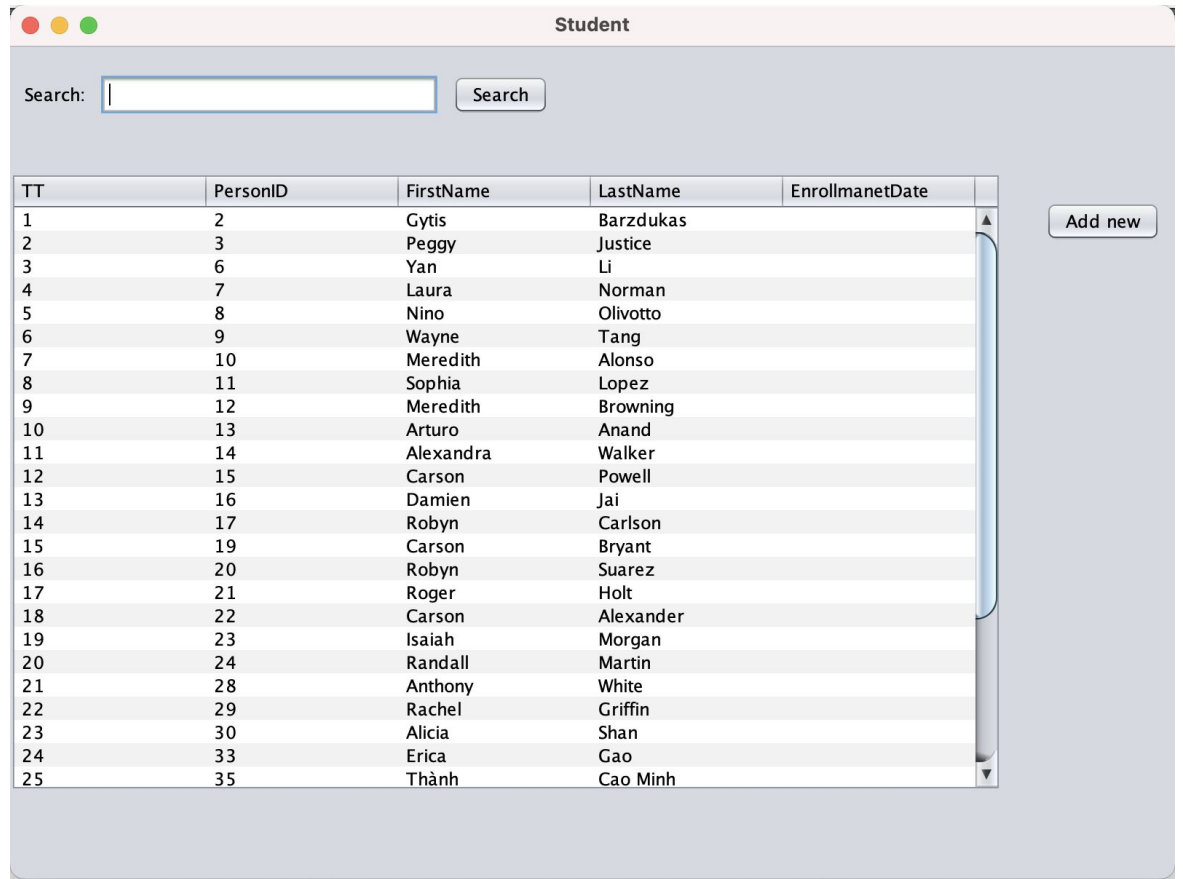
```
private DefaultTableModel convertStudent(List list) {  
    String[] columnNames = {"TT", "PersonID", "FirstName", "LastName", "EnrollmanetDate"};  
    Object[][] data = new Object[list.size()][5];  
    for (int i = 0; i < list.size(); i++) {  
        Student s = (Student) list.get(i);  
        data[i][0] = i + 1;  
        data[i][1] = s.getPersonId();  
        data[i][2] = s.getFirstName();  
        data[i][3] = s.getLastName();  
        data[i][4] = s.getEnrollmentDate();  
    }  
    DefaultTableModel model = new DefaultTableModel(data, columnNames);  
    return model;  
}
```

3. Lớp UI: StudentForm

- Dùng Jtable để hiển thị data

```
//for 2 layer
private void listStudent() throws SQLException {
    StudentDAL std = new StudentDAL();
    List list = std.readStudent();

    DefaultTableModel model = convertStudent(list);
    jTable1.setModel(model);
}
```



TT	PersonID	FirstName	LastName	EnrollmentDate
1	2	Gytis	Barzdukas	
2	3	Peggy	Justice	
3	6	Yan	Li	
4	7	Laura	Norman	
5	8	Nino	Olivotto	
6	9	Wayne	Tang	
7	10	Meredith	Alonso	
8	11	Sophia	Lopez	
9	12	Meredith	Browning	
10	13	Arturo	Anand	
11	14	Alexandra	Walker	
12	15	Carson	Powell	
13	16	Damien	Jai	
14	17	Robyn	Carlson	
15	19	Carson	Bryant	
16	20	Robyn	Suarez	
17	21	Roger	Holt	
18	22	Carson	Alexander	
19	23	Isaiah	Morgan	
20	24	Randall	Martin	
21	28	Anthony	White	
22	29	Rachel	Griffin	
23	30	Alicia	Shan	
24	33	Erica	Gao	
25	35	Thành	Cao Minh	