Lab 03

Thực hành JSPServlet Session03

1. *Mục tiêu*

- Tạo CSDL cho ứng dụng
- Tạo ứng dụng Web với JSP Servlet và thực hiện các chức năng:
  - Load tất cả dữ liệu
  - Xem thông tin chi tiết
  - Thêm mới
  - Cập nhật
  - Xóa
  - Tìm kiếm

2. *Bài tập Step by Step*

Bài toán 1: Tạo database để lưu trữ thông tin các sinh viên. Tạo ứng dụng JSPServlet và sử dụng JDBC để thực hiện các chức năng CRUD với CSDL.

Phần I: Thực hiện chức năng load dữ liệu

Bước 1: Tạo database

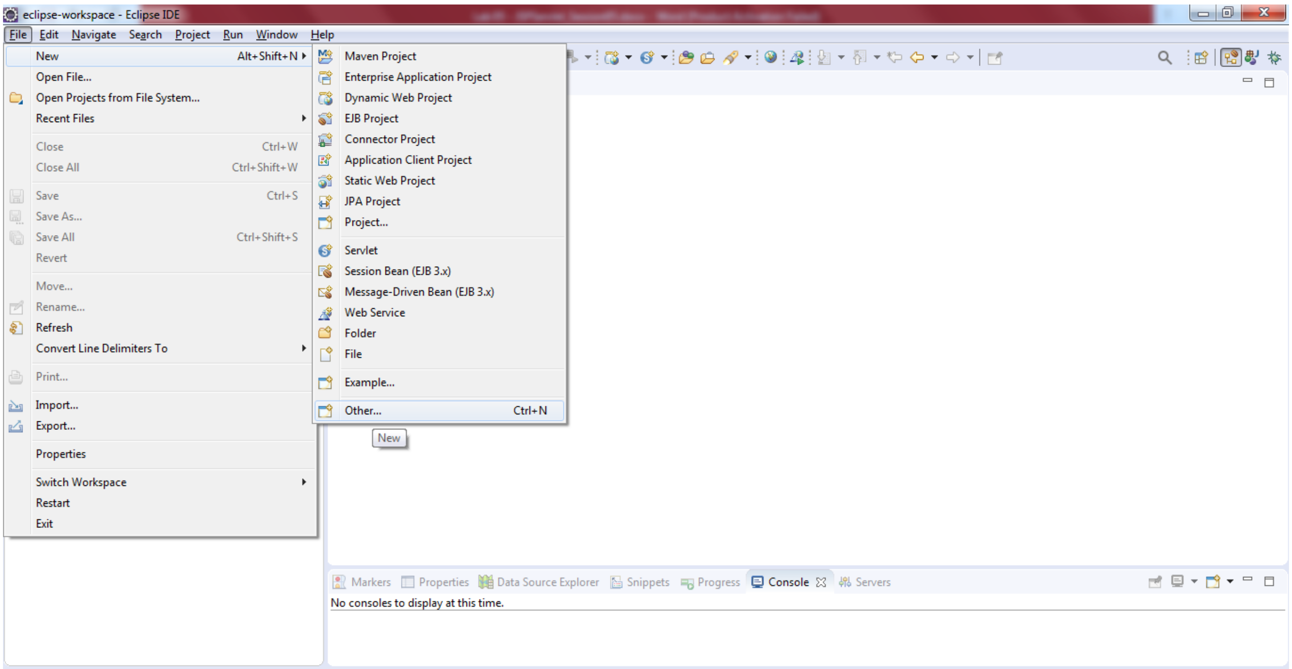
Tạo database như sau:

```
create database DBStudent
go
use DBStudent
go
create table Student(
       StuId int identity primary key,
       FullName nvarchar(70),
       Gender bit,
       Birthday datetime,
       Address nvarchar(200),
       ClassName nvarchar(100),
       ScholarShip float)

insert into Student values (N'Nguyễn Minh Đức',1,'2002-12-21',N'Ý Yên - Nam
Định','JW2008LM',5000000)
insert into Student values (N'Nguyễn Lan Anh',0,'2002-04-11',N'Nam Trực - Nam
Định','JW2008LM',5000000)
```

```
insert into Student values (N'Nguyễn Thành Nam',1,'2002-11-05',N'Vụ Bản - Nam
Định','JW2008LM',3000000)
insert into Student values (N'Nguyễn Quang Minh',1,'2002-07-08',N'Sóc Sơn - Hà
Nội','JW2008LM',3000000)

select * from Student
```

Bước 2: Tạo ứng dụng Dynamic web project

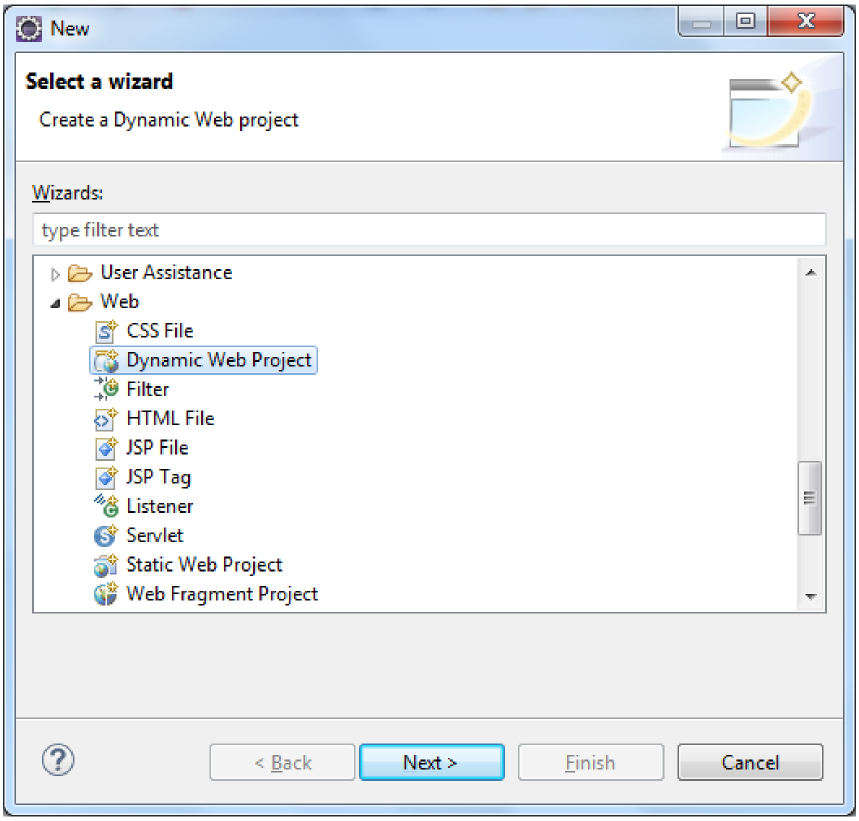
File / New / Other



No consoles to display at this time.

Chọn Web / Dynamic Web Project

0968 27 6996 www.bachkhoa-aptech.edu.vn tuyensinh@bachkhoa-aptech.edu.vn


New

Select a wizard

Create a Dynamic Web project

Wizards:

type filter text

Đặt tên project **JSPServlet_Database**

Thêm các thư viện sau vào thư mục lib:

- **jstl-1.2.jar**
- **sqljdbc4-4.0.jar**

Sau đó click phải vào thư viện đó và chọn: **Build Path/ Add to Build Path.**

Bước 2: Tạo class mở kết nối với CSDL

src / new / class. Đặt tên class DBUtility, package db.

```
package db;

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

public class DBUtility {
    public static Connection openConnection() {
        Connection con = null;
        try {
            Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
            con = DriverManager.getConnection("jdbc:sqlserver://localhost:143
3;databaseName=DBStudent","sa","1234$");
        } catch (ClassNotFoundException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        return con;
    }

    public static void main(String[] args) {
        System.out.println(openConnection());
    }
}
```

Bước 3: Tạo class Student ánh xạ với bảng Student trong database

src / new / class. Đặt tên class Student, package entity.

```
package entity;

import java.util.Date;

public class Student {
    private int stuId;
    private String fullName;
    private boolean gender;
    private Date birthday;
    private String address;
    private String className;
    private Double scholarShip;

    public Student() {
        super();
    }

    public Student(int stuId, String fullName, boolean gender, Date birthday, String addr
            Double scholarShip) {
        super();
        this.stuId = stuId;
        this.fullName = fullName;
        this.gender = gender;
        this.birthday = birthday;
        this.address = address;
        this.className = className;
        this.scholarShip = scholarShip;
    }

    public int getStuId() {
        return stuId;
    }

    public void setStuId(int stuId) {
        this.stuId = stuId;
    }

    public String getFullName() {
        return fullName;
    }

    public void setFullName(String fullName) {
        this.fullName = fullName;
    }

    public boolean isGender() {
        return gender;
    }

    public void setGender(boolean gender) {
        this.gender = gender;
    }

    public Date getBirthday() {
        return birthday;
    }

    public void setBirthday(Date birthday) {
        this.birthday = birthday;
    }

    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }
    public String getClassName() {
        return className;
    }

    public void setClassName(String className) {
        this.className = className;
    }

    public Double getScholarShip() {
        return scholarShip;
    }

    public void setScholarShip(Double scholarShip) {
        this.scholarShip = scholarShip;
    }
}
```

Bước 4: Tạo giao diện DAO trong package dao

```
package dao;

import java.util.List;

import entity.Student;

public interface StudentDAO { // Database Access Object
    public List<Student> getAllStudents();
}
```

Bước 5: Tạo lớp thực thi DAO trong package dao

```
package 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 db.DBUtility;
import entity.Student;

public class StudentDAOImpl implements StudentDAO{// Impl = implements
```

```
    @Override
    public List<Student> getAllStudents() {
        List<Student> list = new ArrayList<Student>();

        Connection con;
        PreparedStatement pstmt;
        ResultSet rs;

        con = DBUtility.openConnection();
        try {
            pstmt = con.prepareStatement("select * from Student");

            rs = pstmt.executeQuery();

            while(rs.next()) {
                Student s = new Student();
                s.setStuId(rs.getInt("StuId"));
                s.setFullName(rs.getString("FullName"));
                s.setGender(rs.getBoolean("Gender"));
                s.setBirthday(rs.getDate("Birthday"));
                s.setAddress(rs.getString("Address"));
                s.setClassName(rs.getString("ClassName"));
                s.setScholarShip(rs.getDouble("ScholarShip"));
                list.add(s);
            }
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }

        return list;
    }
}
```

Bước 6: Tạo servlet để xử lý

src / new / servlet. Đặt tên servlet LoadStudents, package controller

```
package controller;

import java.io.IOException;
import java.util.List;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import dao.StudentDAOImpl;
import entity.Student;

/**
 * Servlet implementation class LoadStudents
 */
@WebServlet("/LoadStudents")
public class LoadStudents extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public LoadStudents() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
 response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse resp
onse) throws ServletException, IOException {
        // TODO Auto-generated method stub

        List<Student> allStudents = new StudentDAOImpl().getAllStudents();
        request.setAttribute("list", allStudents);

        request.getRequestDispatcher("listStudents.jsp").forward(request, res
ponse);
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletRespons
e response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse res
ponse) throws ServletException, IOException {
        // TODO Auto-generated method stub
        doGet(request, response);
    }
}
```

Bước 7: Tạo trang index.jsp trong mục WebContent

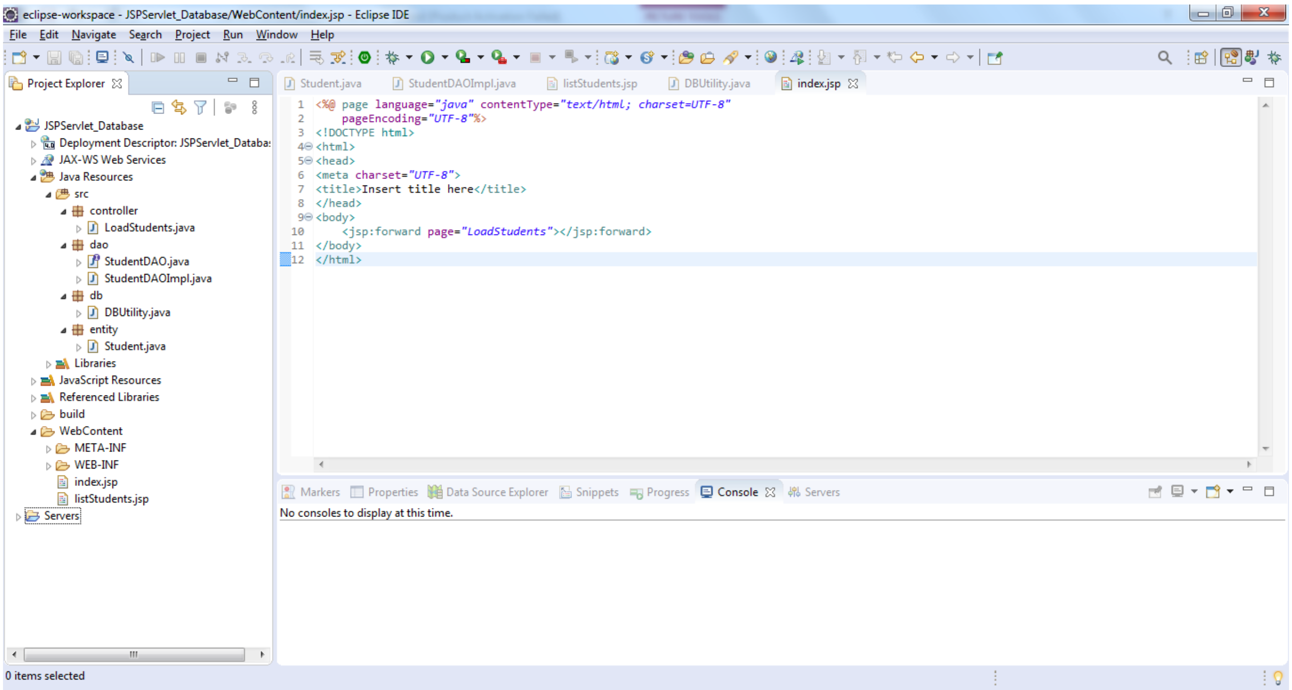
```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
        <jsp:forward page="LoadStudents"></jsp:forward>
</body>
</html>
```

Bước 8: Tạo trang listStudents.jsp trong mục WebContent

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@taglib prefix="c"  uri="http://java.sun.com/jsp/jstl/core" %>
<%@taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>List Students</title>
<style>
        .center{
                display: flex;
                flex-direction: column;
                justify-content:center;
                align-items: center;
        }
</style>
</head>
<body>
        <div class="center">
                <h1>List Students</h1>
                <table border="1">
                        <tr>
                                <th>Student Id</th>
                                <th>Full Name</th>
                                <th>Gender</th>
                                <th>Birthday</th>
                                <th>Address</th>
                                <th>Class Name</th>
                                <th>Scholar Ship</th>
                        </tr>

                        <c:forEach items="${list }" var="s">
```

```
                          <tr>
                                <td>${s.stuId }</td>
                                <td>${s.fullName }</td>
                                <td>${s.gender?"Male":"Female" }</td>
                                <td><fmt:formatDate value="${s.birthday }"
pattern="dd/MM/yyyy"/></td>
                                <td>${s.address }</td>
                                <td>${s.className }</td>
                                <td><fmt:formatNumber value="${s.scholarShip
}"/></td>
                          </tr>
                    </c:forEach>
              </table>
        </div>
</body>
</html>
```

Cấu trúc project như sau:

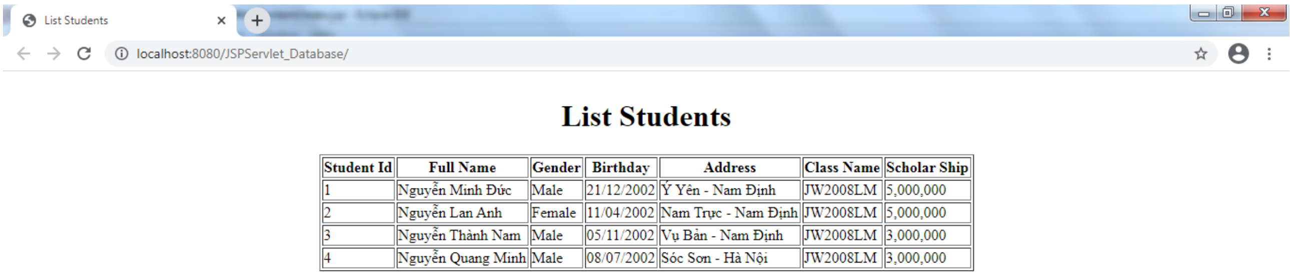


```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
    <jsp:forward page="LoadStudents"></jsp:forward>
</body>
</html>
```

No consoles to display at this time.

Chạy thử ứng dụng:

Click phải vào project / Run As / Run on Server



List Students

Student Id	Full Name	Gender	Birthday	Address	Class Name	Scholar Ship
1	Nguyễn Minh Đức	Male	21/12/2002	Ý Yên - Nam Định	JW2008LM	5,000,000
2	Nguyễn Lan Anh	Female	11/04/2002	Nam Trực - Nam Định	JW2008LM	5,000,000
3	Nguyễn Thành Nam	Male	05/11/2002	Vụ Bản - Nam Định	JW2008LM	3,000,000
4	Nguyễn Quang Minh	Male	08/07/2002	Sóc Sơn - Hà Nội	JW2008LM	3,000,000

Phần II: Thực hiện chức năng insert để thêm mới dữ liệu cho ứng dụng.

Bước 1: Tạo liên kết tới trang insertStudent.jsp

Từ trang listStudents.jsp tạo liên kết sau:

```
<a href="insertStudent.jsp">Add new student</a>
```

Bước 2: Tạo trang insertStudent.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@taglib prefix="c"  uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>List Students</title>
<style>
    .center{
        display: flex;
        flex-direction: column;
        justify-content:center;
        align-items: center;
    }
</style>
</head>
<body>
    <div class="center">
        <h1>Insert Student</h1>
        <h3 style="color:red">${err }</h3>

        <form action="InsertStudent" method="post">
```

```
            <table>
                <tr>
                    <td>Full name:</td>
                    <td>
                        <input type="text" name="fullName" required="required"/>
                    </td>
                </tr>
                <tr>
                    <td>Gender:</td>
                    <td>
                        <input type="radio" name="gender" value="true"/>Male
                        <input type="radio" name="gender" value="false"/>Female
                    </td>
                </tr>
                <tr>
                    <td>Birthday:</td>
                    <td>
                        <input type="date" name="birthday" required="required"/>
                    </td>
                </tr>
                <tr>
                    <td>Address:</td>
                    <td>
                        <textarea name="address" required="required"></textarea>
                    </td>
                </tr>
                <tr>
                    <td>Class name:</td>
                    <td>
                        <select name="className">
                            <option value="JW2008LM">JW2008LM</option>
                            <option value="JW2010LM">JW2010LM</option>
                            <option value="JW2002LM">JW2002LM</option>
                            <option value="JW2004LM">JW2004LM</option>
                            <option value="JW2012LM">JW2012LM</option>
                        </select>
                    </td>
                </tr>
                <tr>
                    <td>ScholarShip:</td>
                    <td>
                        <input type="text" name="scholarShip" required="required"/>
                    </td>
                </tr>
                <tr>
                    <td></td>
                    <td>
                        <input type="submit" value="Insert"/>
                        <input type="reset" value="Clear"/>
                    </td>
                </tr>
            </table>
        </form>

        <a href="index.jsp">Back</a>
    </div>
</body>
</html>
```

Bước 3: Tạo servlet InsertStudent

```
package controller;

import java.io.IOException;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Date;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import dao.StudentDAOImpl;
import entity.Student;

/**
 * Servlet implementation class InsertStudent
 */
@WebServlet("/InsertStudent")
public class InsertStudent extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public InsertStudent() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
 response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse resp
onse) throws ServletException, IOException {
        // TODO Auto-generated method stub
        request.setCharacterEncoding("UTF-8");

        String fullName = request.getParameter("fullName");
        boolean gender = Boolean.parseBoolean(request.getParameter("gender"))
;
        SimpleDateFormat s = new SimpleDateFormat("yyyy-MM-dd");
        Date birthday = null;
        try {
            birthday = s.parse(request.getParameter("birthday"));
        } catch (ParseException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        String address = request.getParameter("address");
        String className = request.getParameter("className");
        Double scholarShip = Double.parseDouble(request.getParameter("scholar
Ship"));

        Student stu = new Student();
        stu.setFullName(fullName);
        stu.setGender(gender);
        stu.setBirthday(birthday);
        stu.setAddress(address);
        stu.setClassName(className);
        stu.setScholarShip(scholarShip);

        boolean bl = new StudentDAOImpl().insertStudent(stu);
        if(bl) {
            request.getRequestDispatcher("index.jsp").forward(request, respon
se);
        }else {
            request.setAttribute("err", "Insert failed!");
            request.getRequestDispatcher("insertStudent.jsp").forward(request
, response);
        }
    }
    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletRespons
e response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse res
ponse) throws ServletException, IOException {
        // TODO Auto-generated method stub
        doGet(request, response);
    }
}
```

Bước 4: Khai báo hàm trong giao diện DAO

```
package dao;

import java.util.List;

import entity.Student;

public interface StudentDAO { // Database Access Object
    public List<Student> getAllStudents();
    public boolean insertStudent(Student s);
}
```

Bước 5: Cài đặt hàm insertStudent trong class thực thi DAO

```
package dao;

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

import javax.crypto.spec.PSource;

import db.DBUtility;
import entity.Student;

public class StudentDAOImpl implements StudentDAO{// Impl = implements

    @Override
    public boolean insertStudent(Student s) {
        boolean bl = false;

        Connection con;
        PreparedStatement pstmt;
        ResultSet rs;

        con = DBUtility.openConnection();
        try {
            pstmt = con.prepareStatement("insert into Student values (?,?,?,?
,?,?)");

            pstmt.setString(1, s.getFullName());
            pstmt.setBoolean(2, s.isGender());
            pstmt.setDate(3, new Date(s.getBirthday().getTime()));  //chuyen
doi ngay thang cua Java sang ngay thang cua sql
            pstmt.setString(4, s.getAddress());
            pstmt.setString(5, s.getClassName());
            pstmt.setDouble(6, s.getScholarShip());

            int i = pstmt.executeUpdate();
            if(i>0) {
                bl = true;
            }
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }

        return bl;
    }

}
```

Phần III: Thực hiện chức năng xem chi tiết thông tin Student

Bước 1: Tạo liên kết trong trang listStudents.jsp

```
<td>
     <a href="DetailStudent?stuId=${s.stuId}">detail</a>
</td>
```

Bước 2: Tạo servlet DetailStudent

```java
package controller;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import dao.StudentDAOImpl;
import entity.Student;

/**
 * Servlet implementation class DetailStudent
 */
@WebServlet("/DetailStudent")
public class DetailStudent extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public DetailStudent() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub

        int id = Integer.parseInt(request.getParameter("stuId"));

        Student student = new StudentDAOImpl().getStudentById(id);
        request.setAttribute("s", student);

        request.getRequestDispatcher("detailStudent.jsp").forward(request, response);
    }
    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse respo
nse) throws ServletException, IOException {
        // TODO Auto-generated method stub
        doGet(request, response);
    }

}
```

Bước 3: Khai báo hàm getStudentById trong giao diện DAO

```
package dao;

import java.util.List;

import entity.Student;

public interface StudentDAO { // Database Access Object
    public List<Student> getAllStudents();
    public boolean insertStudent(Student s);
    public Student getStudentById(int id);
 }
```

Bước 4: Cài đặt hàm getStudentById trong lớp thực thi DAO

```
package dao;

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

import javax.crypto.spec.PSource;

import db.DBUtility;
import entity.Student;
```

```
public class StudentDAOImpl implements StudentDAO{// Impl = implements

    @Override
    public Student getStudentById(int id) {
        Student s = null;

        Connection con;
        PreparedStatement pstmt;
        ResultSet rs;

        con = DBUtility.openConnection();
        try {
            pstmt = con.prepareStatement("select * from Student where StuId = ?
");
            pstmt.setInt(1, id);

            rs = pstmt.executeQuery();

            if(rs.next()) {
                s = new Student();
                s.setStuId(rs.getInt("StuId"));
                s.setFullName(rs.getString("FullName"));
                s.setGender(rs.getBoolean("Gender"));
                s.setBirthday(rs.getDate("Birthday"));
                s.setAddress(rs.getString("Address"));
                s.setClassName(rs.getString("ClassName"));
                s.setScholarShip(rs.getDouble("ScholarShip"));
            }
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }

        return s;
    }

}
```

Bước 5: Tạo trang detailStudent.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@taglib prefix="c"  uri="http://java.sun.com/jsp/jstl/core" %>
<%@taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<!DOCTYPE html>
<html>
```

```
<head>
<meta charset="UTF-8">
<title>List Students</title>
<style>
    .center{
        display: flex;
        flex-direction: column;
        justify-content:center;
        align-items: center;
    }
</style>
</head>
<body>
    <div class="center">
        <h1>Detail Student</h1>
        <table border="1">
            <tr>
                <td><b>Student Id:</b></td>
                <td>${s.stuId }</td>
            </tr>
            <tr>
                <td><b>Full Name:</b></td>
                <td>${s.fullName }</td>
            </tr>
            <tr>
                <td><b>Gender:</b></td>
                <td>${s.gender?"Male":"Female" }</td>
            </tr>
            <tr>
                <td><b>Birthday:</b></td>
                <td><fmt:formatDate value="${s.birthday }" pattern="dd/MM/yyyy"
/></td>
            </tr>
            <tr>
                <td><b>Address:</b></td>
                <td>${s.address }</td>
            </tr>
            <tr>
                <td><b>Class Name:</b></td>
                <td>${s.className }</td>
            </tr>
            <tr>
                <td><b>ScholarShip:</b></td>
                <td><fmt:formatNumber value="${s.scholarShip }"/></td>
            </tr>
            <tr>
        <td colspan="2">
          <a href="PreUpdateStudent?stuId=${s.stuId }">update</a>
        </td>
      </tr>
    </table>
    <a href="index.jsp">Back</a>
  </div>
</body>
</html>
```

Phần IV: Thực hiện chức năng UPDATE thông tin cho Student

Bước 1: Tạo liên kết tới trang insertStudent.jsp

Từ trang detailStudent.jsp tạo liên kết sau:

```
<tr>
   <td colspan="2">
      <a href="PreUpdateStudent?stuId=${s.stuId }">update</a>
   </td>
</tr>
```

Bước 2: Tạo servlet PreUpdateStudent

```
package controller;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import dao.StudentDAOImpl;
import entity.Student;

/**
 * Servlet implementation class PreUpdateStudent
 */
@WebServlet("/PreUpdateStudent")
public class PreUpdateStudent extends HttpServlet {
  private static final long serialVersionUID = 1L;
```

```
    /**
     * @see HttpServlet#HttpServlet()
     */
    public PreUpdateStudent() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub

        int id = Integer.parseInt(request.getParameter("stuId"));

        Student stu = new StudentDAOImpl().getStudentById(id);

        request.setAttribute("s", stu);

        request.getRequestDispatcher("updateStudent.jsp").forward(request, response);
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        doGet(request, response);
    }
}
```

Bước 3: Tạo trang updateStudent.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@taglib prefix="c"  uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
```

```
<head>
<meta charset="UTF-8">
<title>List Students</title>
<style>
    .center{
        display: flex;
        flex-direction: column;
        justify-content:center;
        align-items: center;
    }
</style>
</head>
<body>
    <div class="center">
        <h1>Update Student</h1>
        <h3 style="color:red">${err }</h3>

        <form action="UpdateStudent" method="post">
            <table>
                <tr>
                    <td>Student Id:</td>
                    <td>
                        <input type="text" name="stuId" value="${s.stuId}" re
adonly="readonly"/>
                    </td>
                </tr>
                <tr>
                    <td>Full name:</td>
                    <td>
                        <input type="text" name="fullName" required="required
" value="${s.fullName }"/>
                    </td>
                </tr>
                <tr>
                    <td>Gender:</td>
                    <td>
                        <input type="radio" name="gender" value="true" ${s.ge
nder?"checked":"" }/>Male
                        <input type="radio" name="gender" value="false" ${!s.
gender?"checked":"" }/>Female
                    </td>
                </tr>
                <tr>
                    <td>Birthday:</td>
                    <td>
```

```
                        <input type="date" name="birthday" required="required" value="${s.birthday }"/>
                    </td>
                </tr>
                <tr>
                    <td>Address:</td>
                    <td>
                        <textarea name="address" required="required">${s.address }</textarea>
                    </td>
                </tr>
                <tr>
                    <td>Class name:</td>
                    <td>
                        <select name="className">
                            <option value="JW2008LM" ${s.className.equals("JW2008LM")?"selected":"" }>JW2008LM</option>
                            <option value="JW2010LM" ${s.className.equals("JW2010LM")?"selected":"" }>JW2010LM</option>
                            <option value="JW2002LM" ${s.className.equals("JW2002LM")?"selected":"" }>JW2002LM</option>
                            <option value="JW2004LM" ${s.className.equals("JW2004LM")?"selected":"" }>JW2004LM</option>
                            <option value="JW2012LM" ${s.className.equals("JW2012LM")?"selected":"" }>JW2012LM</option>
                        </select>
                    </td>
                </tr>
                <tr>
                    <td>ScholarShip:</td>
                    <td>
                        <input type="text" name="scholarShip" required="required"/>
                    </td>
                </tr>
                <tr>
                    <td></td>
                    <td>
                        <input type="submit" value="Update"/>
                        <input type="reset" value="Clear"/>
                    </td>
                </tr>
            </table>
        </form>

        <a href="DetailStudent?stuId=${s.stuId}">Back</a>
```

```
    </div>
</body>
</html>
```

Bước 4: Tạo servlet UpdateStudent

```
package controller;

import java.io.IOException;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Date;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import dao.StudentDAOImpl;
import entity.Student;

/**
 * Servlet implementation class UpdateStudent
 */
@WebServlet("/UpdateStudent")
public class UpdateStudent extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public UpdateStudent() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
 response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse resp
onse) throws ServletException, IOException {
        // TODO Auto-generated method stub
        request.setCharacterEncoding("UTF-8");
        int stuId = Integer.parseInt(request.getParameter("stuId"));
        String fullName = request.getParameter("fullName");
        boolean gender = Boolean.parseBoolean(request.getParameter("gender"))
;
        SimpleDateFormat s = new SimpleDateFormat("yyyy-MM-dd");
        Date birthday = null;
        try {
            birthday = s.parse(request.getParameter("birthday"));
        } catch (ParseException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        String address = request.getParameter("address");
        String className = request.getParameter("className");
        Double scholarShip = Double.parseDouble(request.getParameter("scholar
Ship"));

        Student stu = new Student();
        stu.setStuId(stuId);
        stu.setFullName(fullName);
        stu.setGender(gender);
        stu.setBirthday(birthday);
        stu.setAddress(address);
        stu.setClassName(className);
        stu.setScholarShip(scholarShip);

        boolean bl = new StudentDAOImpl().updateStudent(stu);
        if(bl) {
            request.getRequestDispatcher("DetailStudent?stuId="+stuId).forwar
d(request, response);
        }else {
            request.setAttribute("err", "Update failed!");
            request.setAttribute("s", stu);
            request.getRequestDispatcher("updateStudent.jsp").forward(request
, response);
        }
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletRespons
e response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse res
ponse) throws ServletException, IOException {
        // TODO Auto-generated method stub
        doGet(request, response);
    }

}
```

Bước 5: Khai báo hàm updateStudent trong giao diện DAO

```
package dao;

import java.util.List;

import entity.Student;

public interface StudentDAO { // Database Access Object
    public List<Student> getAllStudents();
    public boolean insertStudent(Student s);
    public boolean updateStudent(Student s);
    public Student getStudentById(int id);
}
```

Bước 6: Cài đặt hàm updateStudent trong class thực thi DAO

```
package dao;

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

import javax.crypto.spec.PSource;

import db.DBUtility;
import entity.Student;

public class StudentDAOImpl implements StudentDAO{// Impl = implements


    @Override
    public boolean updateStudent(Student s) {
```

0968 27 6996 www.bachkhoa-aptech.edu.vn tuyensinh@bachkhoa-aptech.edu.vn

```
        boolean bl = false;

        Connection con;
        PreparedStatement pstmt;
        ResultSet rs;

        con = DBUtility.openConnection();
        try {
            pstmt = con.prepareStatement("update Student set FullName=?, Gend
er=?, Birthday=?, Address=?, ClassName=?, ScholarShip=? where StuId=?");

            pstmt.setString(1, s.getFullName());
            pstmt.setBoolean(2, s.isGender());
            pstmt.setDate(3, new Date(s.getBirthday().getTime()));  //chuyen
doi ngay thang cua Java sang ngay thang cua sql
            pstmt.setString(4, s.getAddress());
            pstmt.setString(5, s.getClassName());
            pstmt.setDouble(6, s.getScholarShip());
            pstmt.setInt(7, s.getStuId());

            int i = pstmt.executeUpdate();
            if(i>0) {
                bl = true;
            }
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }

        return bl;
    }
}
```

Phần V: Thực hiện chức năng xóa thông tin Student

Bước 1: Tạo liên kết trong trang listStudents.jsp

```
<td>
     <a href="DeleteStudent?stuId=${s.stuId}" onclick="return confirm('Sure?')
">delete</a>
</td>
```

Bước 2: Tạo servlet DeleteStudent

```
package controller;

import java.io.IOException;
import java.util.List;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import dao.StudentDAOImpl;
import entity.Student;

/**
 * Servlet implementation class DeleteStudent
 */
@WebServlet("/DeleteStudent")
public class DeleteStudent extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public DeleteStudent() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse r
esponse)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse respon
se) throws ServletException, IOException {
        // TODO Auto-generated method stub
        int id = Integer.parseInt(request.getParameter("stuId"));

        boolean bl = new StudentDAOImpl().deleteStudent(id);
        if(bl) {
            request.setAttribute("success", "Delete successfully!");
        }else {
            request.setAttribute("err", "Delete failed!");
        }

        List<Student> allStudents = new StudentDAOImpl().getAllStudents();
        request.setAttribute("list", allStudents);

        request.getRequestDispatcher("listStudents.jsp").forward(request, respo
nse);
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse respo
nse) throws ServletException, IOException {
        // TODO Auto-generated method stub
        doGet(request, response);
    }
}
```

Bước 3: Khai báo hàm deleteStudent trong giao diện DAO

```
package dao;

import java.util.List;

import entity.Student;

public interface StudentDAO { // Database Access Object
    public List<Student> getAllStudents();
    public boolean insertStudent(Student s);
    public boolean updateStudent(Student s);
    public Student getStudentById(int id);
    public boolean deleteStudent(int id);
}
```

Bước 4: Cài đặt hàm deleteStudent trong lớp thực thi DAO

```
package dao;

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

import javax.crypto.spec.PSource;

import db.DBUtility;
import entity.Student;

public class StudentDAOImpl implements StudentDAO{// Impl = implements

    @Override
    public boolean deleteStudent(int id) {
        boolean bl = false;

        Connection con;
        PreparedStatement pstmt;
        ResultSet rs;

        con = DBUtility.openConnection();
        try {
            pstmt = con.prepareStatement("delete from Student where StuId=?");

            pstmt.setInt(1, id);

            int i = pstmt.executeUpdate();
            if(i>0) {
                bl = true;
            }
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }

        return bl;
    }
}
```

3. *Bài tập tự làm*

Bài 1.

Tạo database chứa thông tin về Sản phẩm:

- Mã sản phẩm
- Tên sản phẩm
- Nhà sản xuất
- Năm sản xuất
- Ngày hết hạn
- Đơn vị tính
- Giá bán

Tạo ứng dụng JSPServlet thực hiện các chức năng sau:

- Load dữ liệu lên trang đầu tiên khi ứng dụng được chạy.
- Insert thêm sinh viên mới
- Cập nhật thông tin sinh viên
- Xem chi tiết thông tin sinh viên
- Xóa thông tin sinh viên

Bài 2.

Tạo database chứa thông tin về Khách hàng

- Mã khách hang
- Họ tên
- Giới tính
- Ngày sinh
- Quê quán
- Email
- Số điện thoại
- Số tài khoản
- Ngân hàng

Tạo ứng dụng JSPServlet thực hiện các chức năng sau:

- Load dữ liệu lên trang đầu tiên khi ứng dụng được chạy.
- Insert thêm sinh viên mới
- Cập nhật thông tin sinh viên
- Xem chi tiết thông tin sinh viên
- Xóa thông tin sinh viên

0968 27 6996 www.bachkhoa-aptech.edu.vn tuyensinh@bachkhoa-aptech.edu.vn