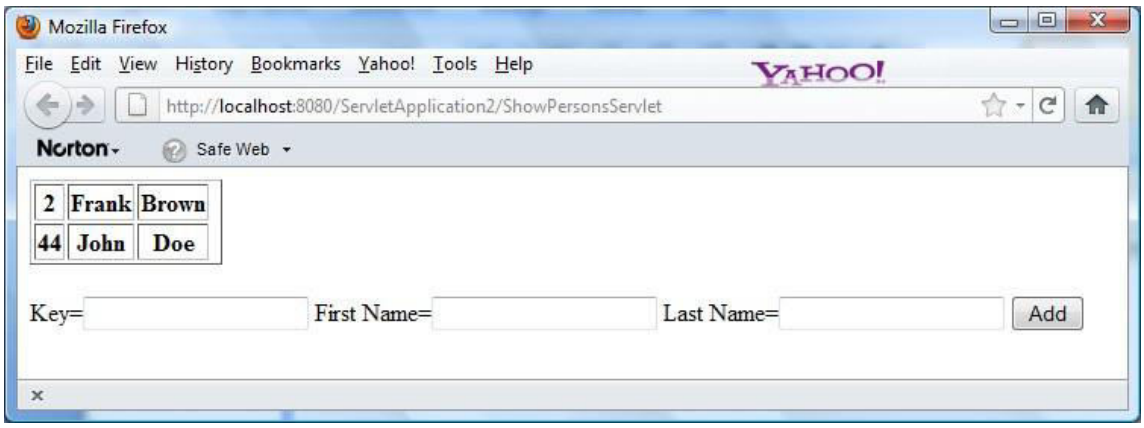
CS472 Final Exam Training April 2016

PRIVATE AND CONFIDENTIAL

Question 1. Servlet

Write a web application using Servlets only (not JSP's) that allows you to add names to a list.



When you fill in the key, firstname and lastname, and click the add button, this person is then added to the list of persons shown on the page.

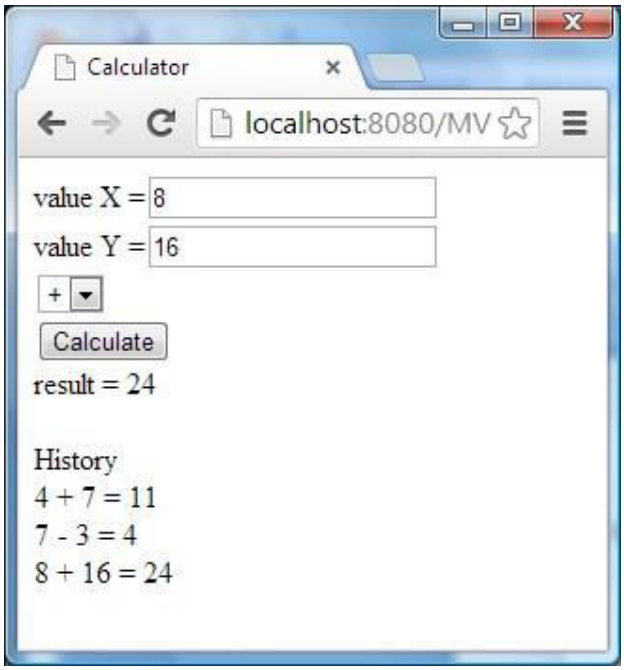
Complete the partial given code. Make sure you add all code that is necessary for the correct working of this application. You are only allowed to use Servlets and Java objects, not JSP's
Complete the partial given code. <u>Do NOT write getter and setter methods!</u>

```
public class Person {

   private String key;
   private String firstname;
   private String lastname;

   public Person(String key, String firstname, String lastname) {
         this.key = key;
         this.firstname = firstname;
         this.lastname = lastname;
  }
}
```

```
public class AddPersonServlet extends HttpServlet {

   protected void doGet(HttpServletRequest request, HttpServletResponse response)
                                        throws ServletException, IOException {
      response.setContentType("text/html");
      PrintWriter out = response.getWriter();
      Collection<Person> personlist = new ArrayList<Person>();

      String key = request.getParameter("key");
      String firstname = request.getParameter("firstname");
      String lastname = request.getParameter("lastname");

      if (key != null && firstname != null && lastname != null) {
        HttpSession session = request.getSession();
        personlist = (Collection<Person>) session.getAttribute("list");
        if (personlist == null) {
          personlist = new ArrayList<Person>();
          session.setAttribute("list", personlist);
        }
        personlist.add(new Person(key, firstname, lastname));
      }
      out.println("<html>");
      out.println("<body>");
      out.println("<table  border='1'>");
      for (Person p : personlist) {
         out.println("<tr><th>" + p.getKey() + "</th><th>" + p.getFirstname() + "</th><th>"
+ p.getLastname() + "</th><th>");
      }
      out.println("</table>");
      out.println("<br>");

      out.println("<form method=GET action=AddPersonServlet>");
      out.println("Key=<input type=text name=key>");
      out.println("First Name=<input type=text name=firstname>");
      out.println("Last Name=<input type=text name=lastname>");
      out.println("<input type=submit value='Add'>");

      out.println("</body>");
      out.println("</html>");

   }
}
```

Question 2. MVC

Write the following calculator application:



When you enter the x and y value, and then select the operator (can be + or -) and click the Calculate button, the page shows the result of the calculation, and also shows the history of all calculations done so far with this application.

Your implementation should follow the following requirements:

1. The calculator can only **add** and **subtract** 2 numbers
2. The application should follow the correct **Model-View-Controller** principles using Servlets, JSP's and Java classes.
4. It is allowed to write JAVA code in the JSP page.
5. You can assume the user only enters integers for the x and y value on the webpage. You do not need to validate the input.
6. You can see only the history of your own calculations and not the history of someone else's calculations.

Complete the partial given code and write all other necessary classes:

calculator.jsp:

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@ page import="java.util.*" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">


<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Calculator</title>
  </head>
  <body>
    <form method=GET action=CalcServlet>
     value X =<input type=text name=x><br/>
     value Y =<input type=text name=y><br/>
     <select name="operator">
      <option label="+" value="+">+</option>
      <option label="-" value="-">-</option>
     </select>
     <br/>
     <input type=submit value='Calculate'><br/>
     </form>


    result = ${result}
    <br/>
    <br/>
    History
    <br/>
    <c:forEach var="calculation" items="${history}">
      ${calculation}<br/>
    </c:forEach>




  </body>
</html>
```

CalcServlet.java

```
public class CalcServlet extends HttpServlet {

   protected void doGet(HttpServletRequest  request, HttpServletResponse  response)
        throws ServletException, IOException {
      response.setContentType("text/html”);
      String strX = request.getParameter("x");
      String strY = request.getParameter("y");
      String operator = request.getParameter("operator");
      int firstnumber = Integer.parseInt(strX);
      int secondnumber = Integer.parseInt(strY);
      int result = 0;
      Calculator calculator = new Calculator();
      result = calculator.calculate(operator, firstnumber, secondnumber);

      request.setAttribute("result", result);

      HttpSession session = request.getSession();
      List<String> historylist = (List<String>) session.getAttribute("history");
      if (historylist == null) {
        historylist = new ArrayList<String>();
        session.setAttribute("history", historylist);
      }
      String calculation = strX +" "+ operator +" "+ strY +" = "+ result;
      historylist.add(calculation);

      RequestDispatcher view = request.getRequestDispatcher("calculator.jsp");
      view.forward(request, response);
      }
}
```

Calculator.java

```
public class Calculator {
  public int add(int x, int y) {
    return x + y;
  }

  public int subtract(int x, int y) {
    return x - y;
  }

  public int calculate(String operator, int firstnumber, int secondnumber) {
    if (operator.equals("+")) {
      return add(firstnumber, secondnumber);
    }
    if (operator.equals("-")) {
      return subtract(firstnumber, secondnumber);
    }
    return 0;
  }
}
```

Question 3. AJAX

Complete the code given below. You need to write the code for imdb.js. You may assume that all 3 files will be located in the same directory on the web server.

Screen shot:

index.jsp:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title></title>
    <script src="http://code.jquery.com/jquery-latest.js"></script>
    <script src="imdb.js" type="text/javascript" ></script>
  </head>
  <body>
    Movie ID:<input type="text" name="id" id="id" />
    <input type="button" name="submit" id="submit" value="Get Details" />

    <div id="info"></div>
  </body>
</html>
```

You may assume that /imdb servlet will return a standard JSON string that have ('movieId', 'movieTitle', 'movieYear' and 'movieRating') properties.

Imdb.js

```
$(function(){
   $("#submit").click(function(){
      $.get("/imdb", { "id": $("#id").val() }) . done(success) . fail(failed));
   });
   function success(data){
         var JSONdata = JSON.parse(data);  $("#info").html(.....); etc
   }
   function failed(xhr, code, msg){ console.log(xhr, code, msg); }
})
```