Java Server Page

JSP程序设计(上)

isszym sysu.edu.cn
2020.7.15

目 录

- 概述 …
- Servlet程序 …
- JSP Web服务器 …
- JSP工作原理 …
- 第一个JSP程序 …
- 声明 …
- 如何获取提交的数据 …

  提交给另一个网页 …

  提交给自己 …

  提交给自己并保持输入 …

  取到所有提交的名值对 …

  获得http请求的头部内容 …

- 数据库操作 …

  浏览学生记录 …

  翻页浏览学生记录 …

  新增操作 …

- 附录

  附录1、hello_jsp.java程序

  附录2、JSP声明的源码转换

概述

JSP教程

- 什么是**JSP**

  JSP(Java Server Pages)是由Sun Microsystems公司开发的一种实现普通静态HTML和动态HTML混合编码的技术。

- Servlet和JSP

  Servlet是最早开始使用Java Web编程方式。Servlet编程通过扩展预定义的HttpServlet类响应HTTP的Get和Post请求。JSP建立在Servlet类的基础上，并分离了表现逻辑和业务逻辑。

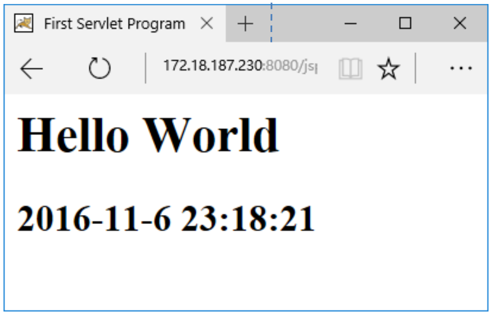
- Web 服务器

  开发Java Web应用所采用的服务器主要是与JSP/Servlet兼容的Web服务器，比较常用的有Tomcat、 JBoss、WebSphere、Resin和 WebLogic 。

Servlet程序

下面的Servlet程序(Hello.java)继承了系统的HttpServlet类.

http://103.26.79.35:8080/jsp/servlet/Hello



Hello World

2016-11-6 23:18:21

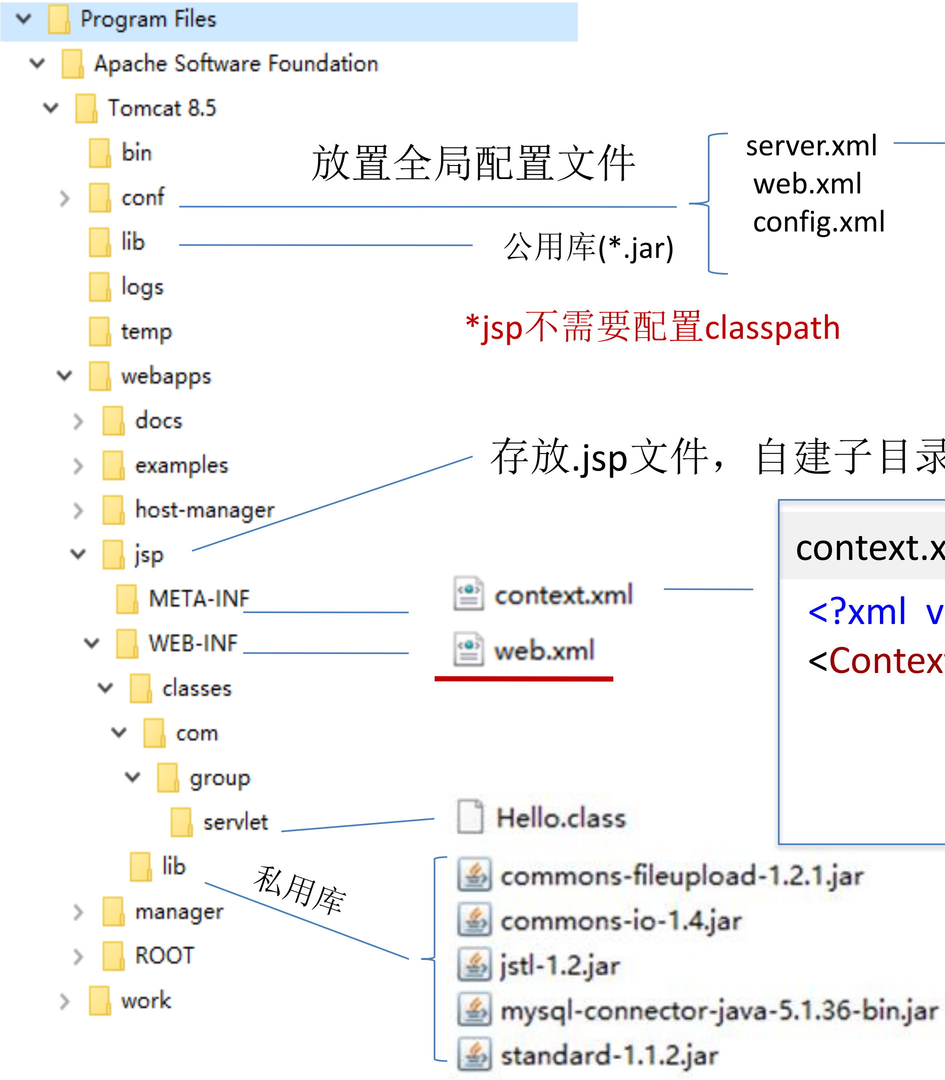
```java
package com.group.servlet;
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Hello extends HttpServlet {
    @SuppressWarnings("deprecation")
    @Override
    public void doGet(HttpServletRequest req, HttpServletResponse resp)
            throws ServletException, IOException  {
        resp.setContentType("text/html");
        PrintWriter out = resp.getWriter();
        out.println("<html><head><title>First Servlet Program</title></head>");
        out.println("<body>");
        out.println("<h1>Hello World<h1>");
        out.println("<h2>" + new Date().toLocaleString()+ "</h2>");
        out.println("</body>");
        out.println("</html>");
        out.flush();
    }
}
```

doPost

```
C:\java>javac  -classpath .\servlet-api.jar  Hello.java
```

* servlet-api.jar在tomcat的lib目录中

http://103.26.79.35:8080/jsp/servlet/Hello



```
Program Files
  Apache Software Foundation
    Tomcat 8.5
      bin
      conf
      lib
      logs
      temp
      webapps
        docs
        examples
        host-manager
        jsp
          META-INF
          WEB-INF
            classes
              com
                group
                  servlet
            lib
        manager
        ROOT
      work
```

放置全局配置文件

server.xml
web.xml
config.xml

公用库(*.jar)

webapps下的子目录都是虚拟目录。要定义其它虚拟目录，需要修改配置文件server.xml。

*jsp不需要配置classpath

存放.jsp文件，自建子目录

```
context.xml
web.xml
```

```
Hello.class
```

私用库

```
commons-fileupload-1.2.1.jar
commons-io-1.4.jar
jstl-1.2.jar
mysql-connector-java-5.1.36-bin.jar
standard-1.1.2.jar
```

也可以放在公用库中

context.xml文件内容：

```
<?xml  version="1.0"?>
<Context  reloadable="true"/>
```

*修改后自动重载，不需要重启Tomcat服务器

http://103.26.79.35:8080/jsp/servlet/Hello

web.xml（在webapps\jsp\WEB-INF目录下）

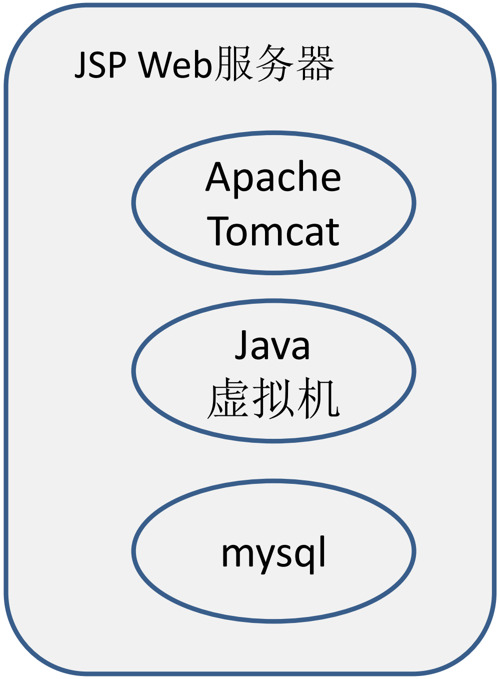
```
<?xml version="1.0" encoding="ISO-8859-1"?>
<web-app>
    <description>Servletand JSP Examples. </description>
    <display-name>Servlet and JSP Examples</display-name>
    <servlet>            （3）
        <servlet-name>HelloWorld</servlet-name>
        <servlet-class>com.group.servlet.Hello</servlet-class>
   </servlet>                 （4）
   <servlet-mapping>       （2）
       <servlet-name>HelloWorld</servlet-name>
       <url-pattern>/servlet/Hello</url-pattern>
   </servlet-mapping>
   <servlet>                  （1） URL
       <servlet-name>Requ</servlet-name>
      <servlet-class>serv.RequestInfo</servlet-class>
   </servlet>
   <servlet-mapping>
      <servlet-name>Requ</servlet-name>
      <url-pattern>/servlet/request</url-pattern>
  </servlet-mapping>
</web-app>
```

一个servlet

另一个servlet

JSP Web服务器

Web服务器是接收HTTP请求并返回HTTP响应的软件。下面是一种JSP web服务器结构，Apache Tomcat为Web服务器，它是基于Java虚拟机的。这个结构采用了MySQL数据库。如果编程使用Eclipse，注意其内部Tomcat与外部的Tomcat可能会冲突（只能启动一个）。

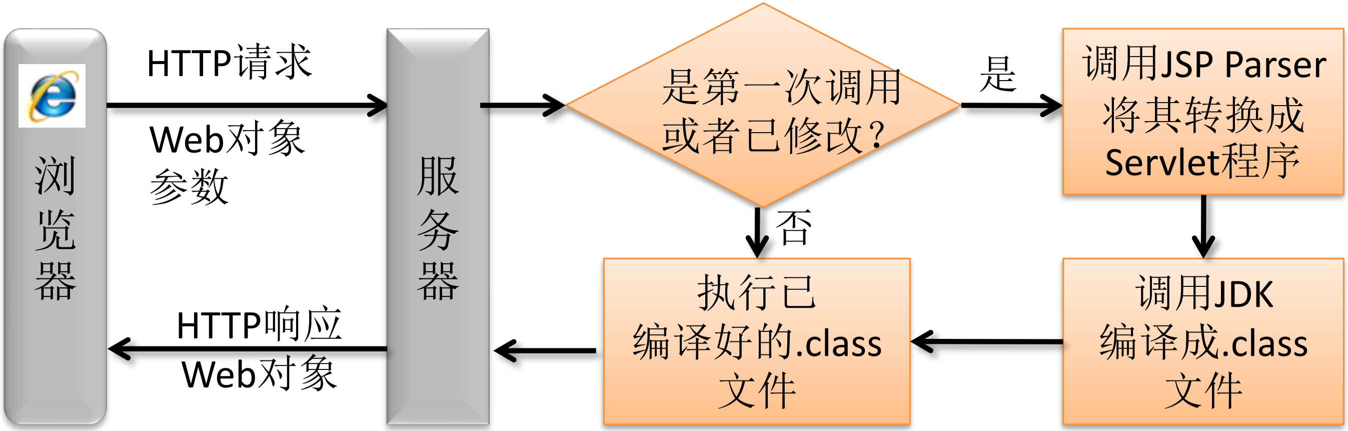


tomcat下载

java下载

mysql下载

JSP工作原理



（1）当Web服务器收到对JSP文件(JSP页面)的HTTP请求，如果所请求的JSP文件是修改后的第一次访问，则调用JSP Parser将其转换成Servlet程序（.java），然后调用JSDK编译成Servlet字节码（.class）。

（2）Java虚拟机执行Servlet字节码发回HTTP响应。

（3）如果有多个HTTP请求发给同一个JSP文件，则Tomcat服务器会为每一个请求启动一个线程执行该文件对应的Servlet字节码。

第一个JSP程序

hello.jsp

生成http响应的指示，例如，要使用的包，http响应的头部content-type.

指令元素

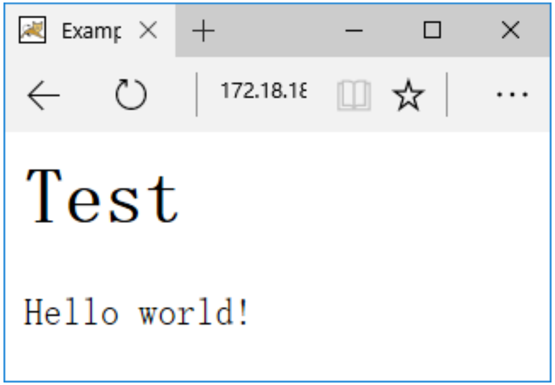
HTML模版

脚本片段(scriptlet)

HTML模版

```
<%@ page contentType="text/html;charset=gb2312"%>
<html>
<head>
<title>Example</title>
</head>
<body>
<h1> Test </h1>
<p>
<%
   out.print("Hello world!");
%>
</p>

</body>
</html>
```

网址



Test

Hello world!

Java程序放在<% %>之间。out为一个JSP的系统对象。

如果只输出一个表达式而不是语句，可以采用**<%=exp%>**输出，例如：<%="Hello world!"%>。

hello.jsp会被**自动转换**为servlet程序hello_jsp.java，放在work子目录下，然后再编译成字节码文件hello_jsp.class，并在Java虚拟机上运行，最后把运行的输出结果用**http响应**传送到客户端。

```
          ...
    public final class hello_jsp extends
              org.apache.jasper.runtime.HttpJspBase{
          ...可以在这里声明<%! %>内定义全局变量、函数和类
        public void _jspService(... request,... response)
          定义了jsp内置对象：out、session、application、config
             ...
          out.write("<html>\r\n");
          out.write("<head>\r\n");
          out.write("<title>Hello world</title>\r\n");
          out.write("</head>\r\n");
          out.write("<body>\r\n");
          out.write("<h1>Test</h1>\r\n");
          out.write("<p>\r\n");
          out.print("Hello world!");
          out.write("</p>\r\n");
          out.write("</body>\r\n");
          out.write("</html>\r\n");
          out.write("\r\n");
             ...
      }
    }
```

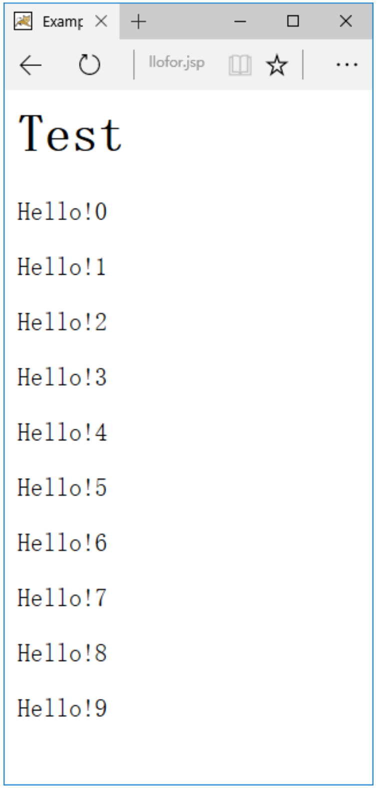
* out.print()只能输出字符，而 out.write还可以输出字节流

* hello_jsp.java的详细内容见附录

为什么for语句里面可以加入html的语句？如何解释？

hellofor.jsp

```
<%@ page contentType="text/html;charset=gb2312"%>
<html>
  <head>
    <title>Example</title>
  </head>
  <body>
    <h1>Test</h1>
    <%for(int i=0;i<10;i++){%>
        <p>Hello!<%=i%></p>
    <%}%>
  </body>
</html>
```



```
Test

Hello!0
Hello!1
Hello!2
Hello!3
Hello!4
Hello!5
Hello!6
Hello!7
Hello!8
Hello!9
```

```
  ...
out.write("\r\n");
out.write("<html>\r\n");
out.write("<head>\r\n");
out.write("<title>Example</title>\r\n");
out.write("</head>\r\n");
out.write("<body>\r\n");
out.write("<h1>Test</h1>\r\n");
for(int i=0;i<10;i++){
    out.write("\r\n");
    out.write("<p>Hello!");
    out.print(i);
    out.write("</p>\r\n");
}
out.write("\r\n");
out.write("</body>\r\n");
out.write("</html>\r\n");
out.write("\r\n");
out.write("\r\n");
out.write("\r\n");
   ...
```

也可以这样编程，先计算出变量值，再把结果代入Web页面，实现Java代码与HTML代码分离：

hellofor2.jsp

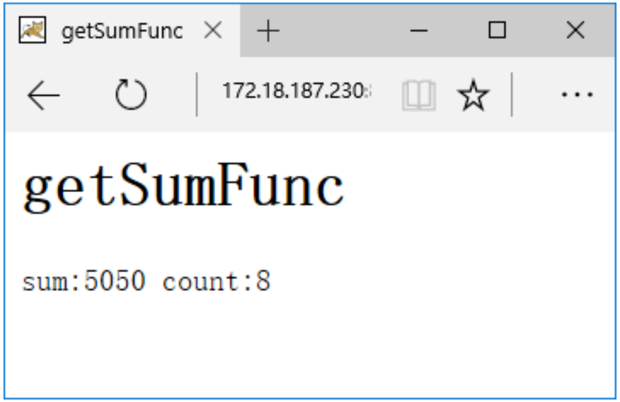
```
<%@ page contentType="text/html;charset=gb2312"%>
<% String s="";
   for(int i=0;i<10;i++){
     s=s+"<p>Hello!"+i+"</p>";
   }
%>
<html>
  <head>
    <title>Example</title>
  </head>
  <body>
    <h1>Test</h1>
    <%=s%>
  \</body>
</html>
```

声明

JSP程序的函数和类只能在**声明**<%!... %>中定义，它们会被分别转换为servlet类的方法和内部类。声明中定义的变量会被所有用户线程所共享。

getSumfunc.jsp

```
<%@ page contentType="text/html;charset=gb2312"%>
<%! int cnt=0;
    int getSum(int n){
      int sum=0;
      for(int i=1;i<=n;i++){
         sum=sum+i;
      }
      return sum;
    }
 %>
<html>
  <head>
    <title>getSumFunc</title>
  </head>
  <body>
     <h1>getSumFunc</h1>
     sum:<%=getSum(100)%>
     count:<%=cnt++%>
  </body>
</html>
```



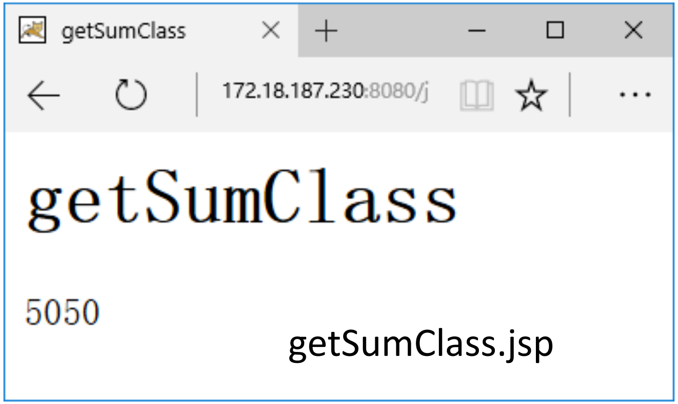
getSumFunc

```
sum:5050 count:8
```

在声明中不能使用jsp内置对象，例如：out、session、request、response等，参见附录的servlet程序。 要在声明中使用它们，需要作为参数传入其中。

getSumClass.jsp

```
<%@ page contentType="text/html;charset=gb2312"%>
<%! class Sum{
      int getSum(int n){
        int sum=0;
        for(int i=1;i<=n;i++){
          sum=sum+i;
        }
        return sum;
      }
    }
 %>
<html>
 <head>
   <title>getSumClass</title>
 </head>
 <body>
    <h1>getSumClass</h1>
    <% Sum sum = new Sum();
       out.print(sum.getSum(100));
    %>
  </body>
</html>
```



getSumClass

5050

getSumClass.jsp

```
// Sum.java
 package com.group.jsp;
 public class Sum {
      public int getSum(int n){
        int sum=0;
        for(int i=1;i<=n;i++){
          sum=sum+i;
        }
        return sum;
      }
 }
```

Sum.class文件放在
tomcat\webapps\jsp\WEB-INF\classes\
com\group\jsp目录下

http://172.18.187.10:8080/jsp/getSumClass2.jsp

```
 // getSumClass2.jsp
<%@ page import="com.group.jsp.Sum"
contentType="text/html;charset=gb2312"%>
<html>
<head>
<title>getSumClass</title>
</head>
<body>
  <h1>getSumClass</h1>
  <%
    Sum sum = new Sum();
    out.print(sum.getSum(100));
  %>
</body>
</html>
```

MySql库的位置见前面
“Servetlet程序” 一节

- JSP的类一定要指定包名，不能用默认包
- 一个网页可以出现多个import：

```
<%@ page import="a.b.*" %>
<%@ page import="x.y.*" %>
```

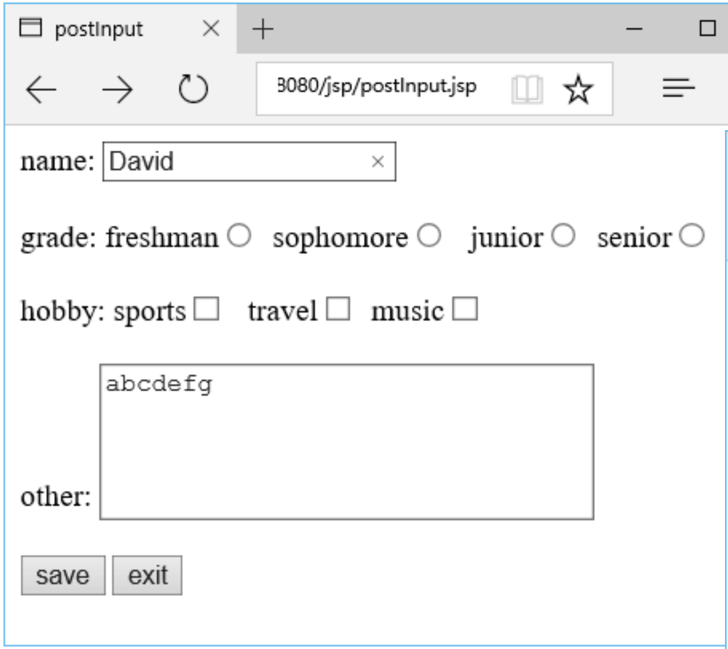
- 可以合并，用逗号隔开：

```
<%@ page import="a.b.*,x.y.*" %>
```

如何获取提交的数据

（1）提交给另一个网页（必须是动态网页）

初始页面



name: David

grade: freshman ○ sophomore ○ junior ○ senior ○

hobby: sports □ travel □ music □

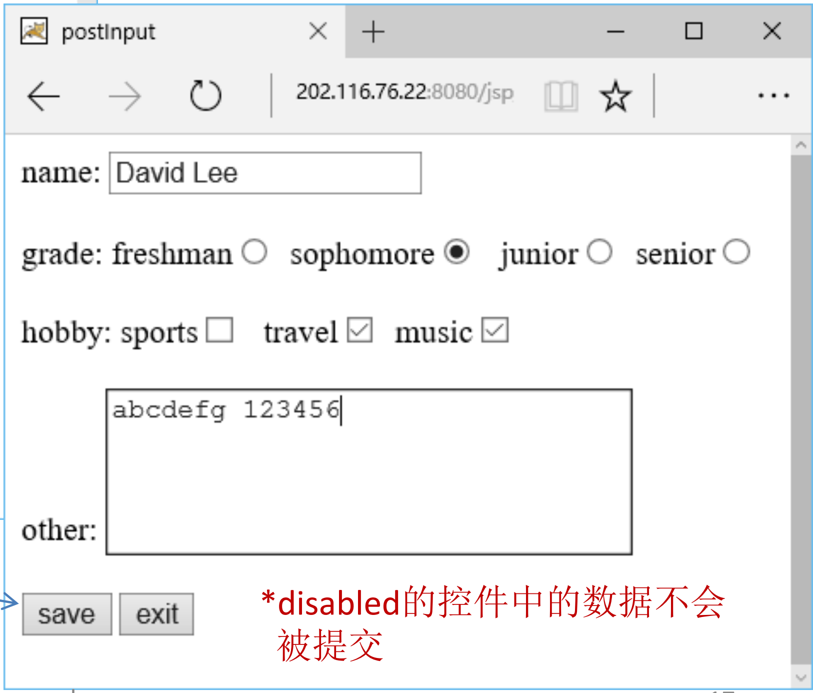
other: abcdefg

save exit

postInput.jsp

postInput.jsp

输入后页面



name: David Lee

grade: freshman ○ sophomore ● junior ○ senior ○

hobby: sports □ travel ☑ music ☑

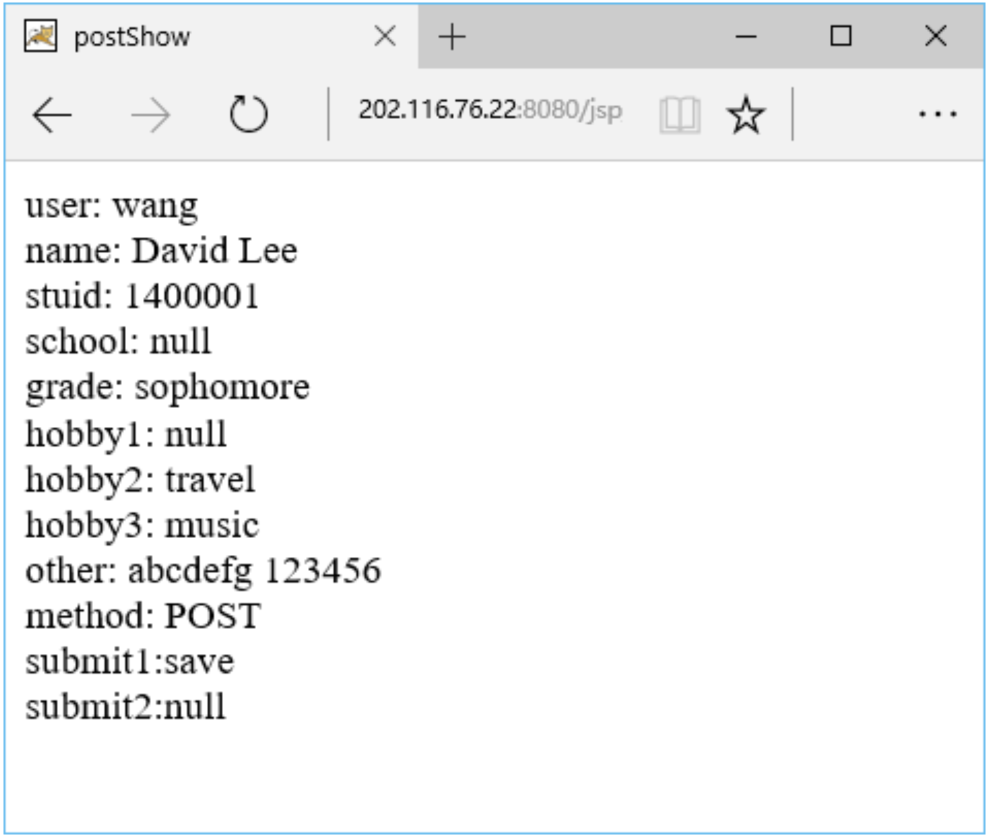
other: abcdefg 123456

save exit

点击

*disabled的控件中的数据不会被提交

点击后进入的页面：　　　　postShow.jsp



```
user: wang
name: David Lee
stuid: 1400001
school: null
grade: sophomore
hobby1: null
hobby2: travel
hobby3: music
other: abcdefg 123456
method: POST
submit1:save
submit2:null
```

postInput.jsp

```
<%@ page language="java" import="java.util.*"
         contentType="text/html; charset=utf-8" pageEncoding="utf-8"%>
<!DOCTYPE HTML>
<html>
  <head><title>postInput</title></head>
  <body>
    <form action="postShow.jsp?user=wang" method="post">
        <input type="hidden" name="stuid" value="1400001" />
        name: <input type="text" name="name" value="David"/>
        grade: freshman<input type="radio" name="grade" value="freshman"/>
               sophomore<input type="radio" name="grade" value="sophomore"/>
               junior<input type="radio" name="grade" value="junior"/>
               senior<input type="radio" name="grade" value="senior"/>
        hobby: sports<input type="checkbox" name="hobby1" value="sports"/>
               travel<input type="checkbox" name="hobby2" value="travel"/>
               music<input type="checkbox" name="hobby3" value="music"/>
        other: <textarea rows="5" cols="30" name="other">abcdefg</textarea>
        <input type="submit"  name="submit1" value="save" />
        <input type="submit"  name="submit2" value="exit" />
    </form>
  </body>
</html>
```

* 提交时，只会提交被点击的submit类型的按钮，没被点击的submit类型的按钮不会被提交。

postShow.jsp

```
<%@ page language="java" import="java.util.*"
         contentType="text/html; charset=utf-8"%>
<%request.setCharacterEncoding("utf-8");%>
<!DOCTYPE HTML>
<html>
  <head> <title>postShow</title></head>
  <body>user:   <%= request.getParameter("user")%>
        name:   <%= request.getParameter("name")    %>
        stuid:  <%= request.getParameter("stuid")   %>
        school: <%= request.getParameter("school")  %>
        grade:  <%= request.getParameter("grade")   %>
        hobby1: <%= request.getParameter("hobby1")  %>
        hobby2: <%= request.getParameter("hobby2")  %>
        hobby3: <%= request.getParameter("hobby3")  %>
        other:  <%= request.getParameter("other")   %>
        method: <%= request.getMethod()             %>
        submit1:<%= request.getParameter("submit1") %>
        submit2:<%= request.getParameter("submit2") %>
  </body>
</html>
```

- url中的参数也是通过getParameter取到的。
- 在所有键值对中找不到键(key)时getParameter取值null。
- setCharacterEncoding()用于指出收到的提交数据所用的编码（也是提交网页的编码）
- 只有被按的submit按钮才会被提交， 其它submit按钮不会被提交。

（2）提交给自己并显示输入值

postInput.jsp

原来提交

```
<form action="postShow.jsp?user=wang" method="post">
   ......
</form>
```

postToSelf.jsp

现在提交

```
<form action="postToSelf.jsp?user=wang" method="post">
   ......
</form>
```

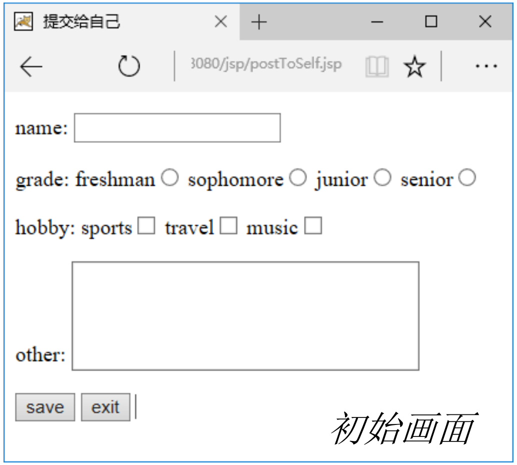
postToSelf.jsp

```
<%@ page language="java" import="java.util.*" contentType="text/html;
charset=utf-8"%>
<%request.setCharacterEncoding("utf-8");%>
<!DOCTYPE HTML> <html> <head> <title>提交给自己</title> </head>
<body>
  <%if (!request.getMethod().equalsIgnoreCase("post")){ %>
    <form action="postToSelf.jsp?user=wang" method="post">
      <input type="hidden" name="stuid">
      name: <input type="text" name="name">
      grade: freshman<input type="radio" name="grade" value="freshman">
             sophomore<input type="radio" name="grade" value="sophomore">
             junior<input type="radio" name="grade" value="junior">
             senior<input type="radio" name="grade" value="senior">
      hobby: sports<input type="checkbox" name="hobby" value="sports">
             travel<input type="checkbox" name="hobby" value="travel">
             music<input type="checkbox" name="hobby" value="music">
      other: <textarea rows="5" cols="30" name="other"></textarea>
      <input type="submit" name="submit1" value="save">
      <input type="submit" name="submit2" value="exit">
    </form>
  <%} %>
```

```
    <%else{ %>
         user:<%= request.getParameter("user")  %>
         name:<%= request.getParameter("name")  %>
         stuid:<%= request.getParameter("stuid") %>
         school:<%= request.getParameter("school") %>
         grade: <%= request.getParameter("grade") %>
         hobby: <%= Arrays.toString(request.getParameterValues("hobby")) %>
         other: <%= request.getParameter("other") %> <br>
         method:  <%= request.getMethod()            %>
         submit1:<%= request.getParameter("submit1") %>
         submit2:<%= request.getParameter("submit2") %>
    <%}%>
</body></html>
```

获得数组

* request.getParameterValues("hobby")可以得到由多个选中值组成的字符串数组。



提交给自己

3080/jsp/postToSelf.jsp

name:

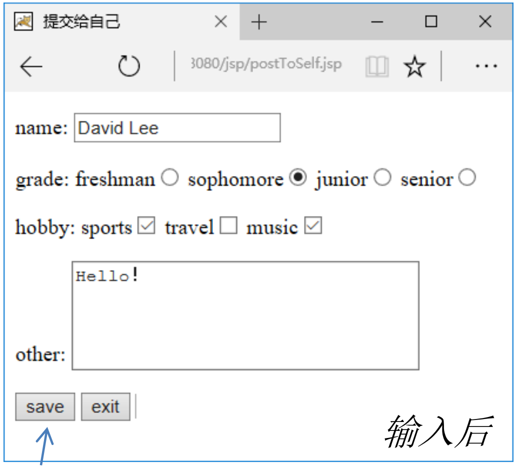
grade: freshman ○ sophomore ○ junior ○ senior ○

hobby: sports □ travel □ music □

other:

save exit

初始画面



提交给自己

3080/jsp/postToSelf.jsp

name: David Lee

grade: freshman ○ sophomore ● junior ○ senior ○

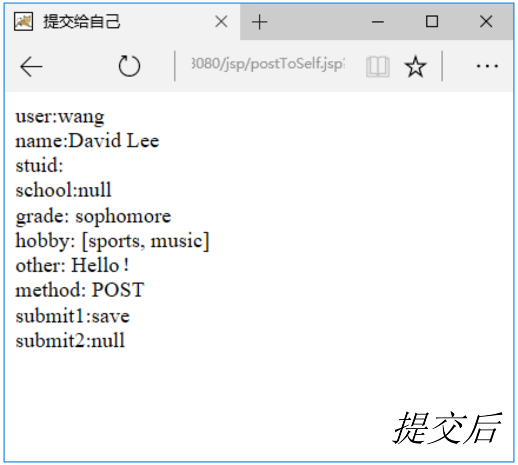
hobby: sports ☑ travel □ music ☑

other: Hello!

save exit

输入后

点击



提交给自己

3080/jsp/postToSelf.jsp

```
user:wang
name:David Lee
stuid:
school:null
grade: sophomore
hobby: [sports, music]
other: Hello!
method: POST
submit1:save
submit2:null
```

提交后

（3）提交给自己（不保持输入值）

postNoKeep.jsp

```
<%@ page language="java" import="java.util.*"
        contentType="text/html; charset=utf-8"%>
<!DOCTYPE HTML><html><head><title>提交给自己2</title></head><body>
 <form action="postNoKeep.jsp?user=wang" method="post">
   <input type="hidden" name="stuid" value="">
   name: <input type="text" name="name" value="">
   grade: freshman<input type="radio" name="grade" value="freshman">
       sophomore<input type="radio" name="grade" value="sophomore">
       junior<input type="radio" name="grade" value="junior">
       senior<input      type="radio" name="grade" value="senior">
   hobby: sports<input type="checkbox" name="hobby" value="sports">
          travel<input type="checkbox" name="hobby" value="travel">
          music<input type="checkbox" name="hobby" value="music">
   other: <textarea rows="5" cols="30" name="other"></textarea>
   <input type="submit" name="submit1" value="save">
   <input type="submit" name="submit2" value="exit">
  </form>
</body></html>
```

一开始得到前面的初始画面，输入之后点击save按钮后会得到什么？

（4）提交给自己并保持输入值

http://103.26.79.35:8080/jsp/postAndKeep.jsp?user=wang

```
<%@ page language="java" import="java.util.*"
         contentType="text/html; charset=utf-8"%>
<%request.setCharacterEncoding("utf-8");
  String user="wang";
  String name="";
  String stuid="";
  String school="";
  String grade="";
  String hobby="";
  String other="";
  if (request.getMethod().equalsIgnoreCase("post")){
        user = request.getParameter("user");
        name = request.getParameter("name");
        stuid = request.getParameter("stuid");
        school = request.getParameter("school");
        grade = request.getParameter("grade");
        hobby = Arrays.toString(request.getParameterValues("hobby"));
        other = request.getParameter("other");
  }
%>
<!DOCTYPE HTML><html><head><title>提交给自己3</title></head><body>
```

```
  <form action="postAndKeep.jsp?user=<%=user%>" method="post">
     <input type="hidden" name="stuid" value="<%=stuid%>">
     name: <input type="text" name="name" value="<%=name%>">
     grade: freshman<input type="radio" name="grade" value="freshman"
                             <%=grade.equals("freshman")?"checked":""%> >
            sophomore<input type="radio" name="grade" value="sophomore"
                             <%=grade.equals("sophomore")?"checked":""%> >
            junior<input type="radio" name="grade" value="junior"
                             <%=grade.equals("junior")?"checked":""%> >
            senior<input type="radio" name="grade" value="senior"
                             <%=grade.equals("senior")?"checked":""%> >
     hobby: sports<input type="checkbox" name="hobby" value="sports"
                             <%=hobby.indexOf("sports")>=0?"checked":""%> >
           travel<input type="checkbox" name="hobby" value="travel"
                             <%=hobby.indexOf("travel")>=0?"checked":""%> >
           music<input type="checkbox" name="hobby" value="music"
                             <%=hobby.indexOf("music")>=0?"checked":""%> >
    other: <textarea rows="5" cols="30" name="other"><%=other%></textarea>
    <input type="submit" name="submit1" value="save">
    <input type="submit" name="submit2" value="exit">
  </form>
</body>
</html>
```

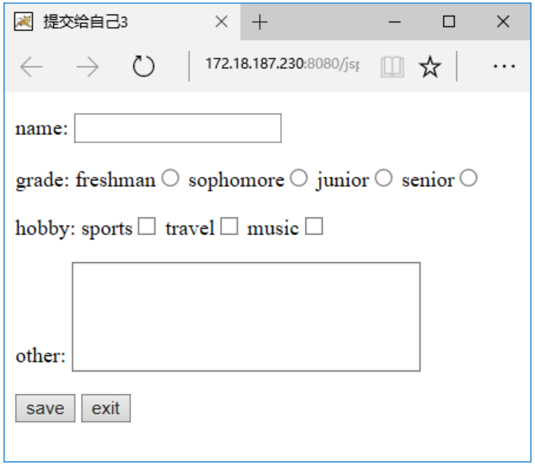
设置checked属性值

三元运算符的格式

[条件表达式] ? [表达式1] : [表达式2]

* [条件表达式]的值为真则返回[表达式1] 的值，否则，返回[表达式2]的值

（a）初始画面


提交给自己3

172.18.187.230:8080/jsp

name:

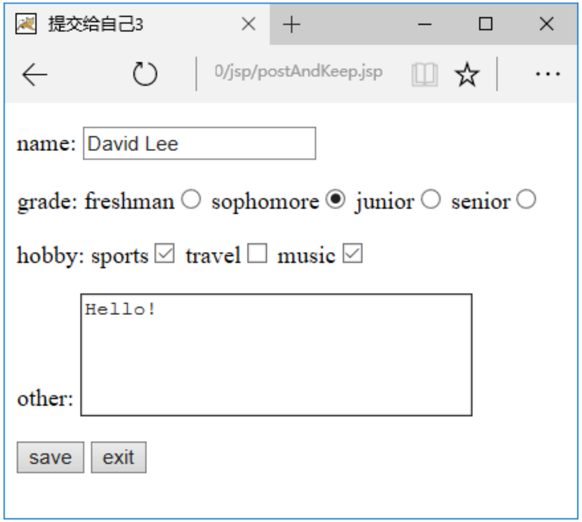
grade: freshman ○ sophomore ○ junior ○ senior ○

hobby: sports □ travel □ music □

other:

save exit

（b）输入后画面


提交给自己3

0/jsp/postAndKeep.jsp

name: David Lee

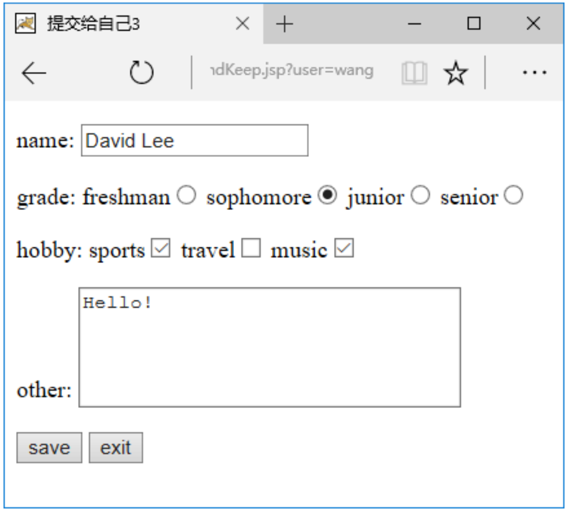
grade: freshman ○ sophomore ● junior ○ senior ○

hobby: sports ☑ travel □ music ☑

other: Hello!

save exit

（c）点击save后的画面


提交给自己3

ıdKeep.jsp?user=wang

name: David Lee

grade: freshman ○ sophomore ● junior ○ senior ○

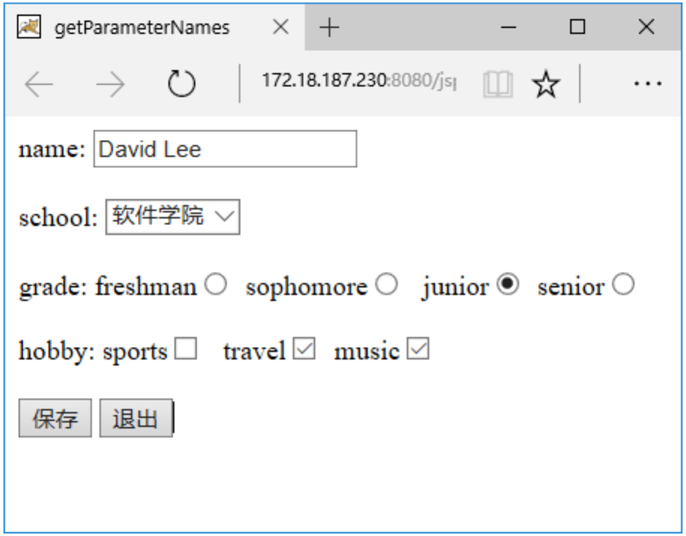
hobby: sports ☑ travel □ music ☑

other: Hello!

save exit

（5）取到所有提交的名值对 getParameterNames.jsp

```
<%@ page language="java" import="java.util.*"
         contentType="text/html; charset=utf-8"%>
<%request.setCharacterEncoding("utf-8");%>
<% String saveButton=request.getParameter("submit1"); //判断是否按了保存按钮
   if(saveButton!=null){
     Enumeration<String> enums=request.getParameterNames();
     while(enums.hasMoreElements()){
        String name=(String)enums.nextElement();
        out.print(name+":"+request.getParameter(name)+"<br />");
     }
   }
%>
```

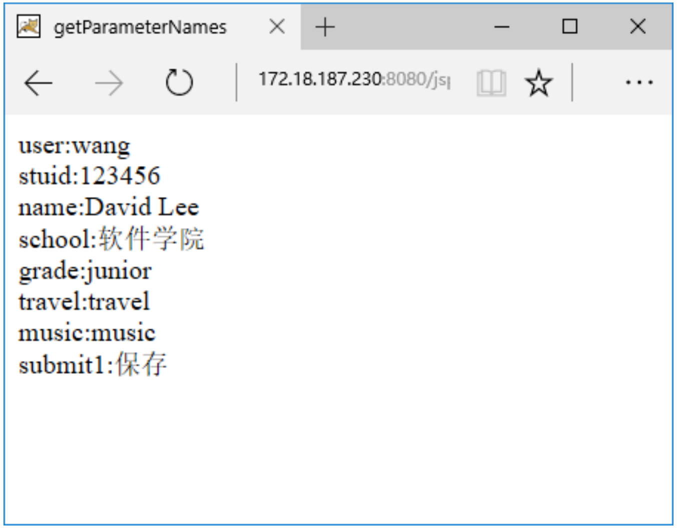


name: David Lee

school: 软件学院

grade: freshman ○ sophomore ○ junior ● senior ○

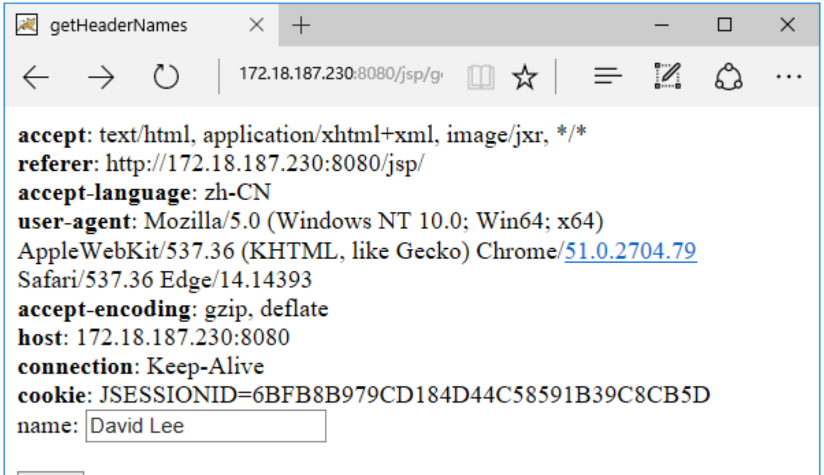
hobby: sports □ travel ☑ music ☑



```
user:wang
stuid:123456
name:David Lee
school:软件学院
grade:junior
travel:travel
music:music
submit1:保存
```

（6）获得http请求的头部内容　getHeaderNames.jsp

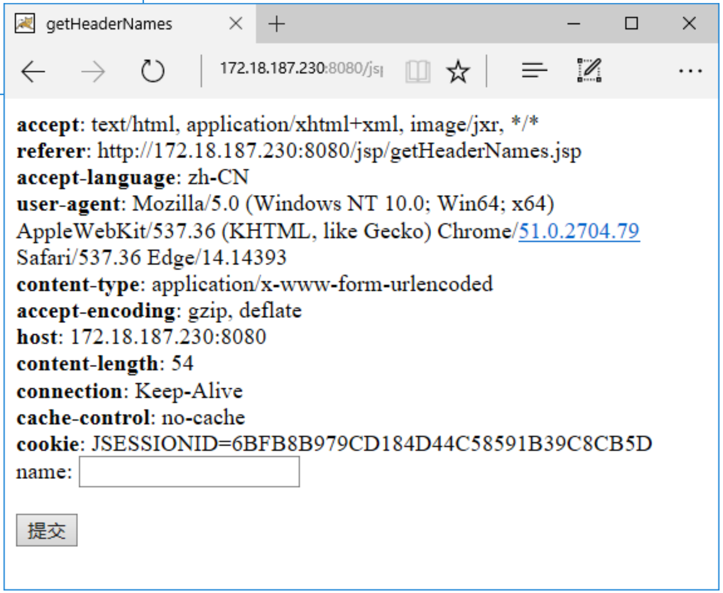
```
<%@ page language="java" import="java.util.*"
         contentType="text/html; charset=utf-8"%>
<%request.setCharacterEncoding("utf-8");%>
<% Enumeration<String> enums=request.getHeaderNames();
    while(enums.hasMoreElements()){
        String name=(String)enums.nextElement();
        out.println("<B>"+name+"</B>: "+request.getHeader(name)+"<br>");
    }
%>
<!DOCTYPE HTML>
<html>
  <head>
    <title>getHeaderNames</title>
  </head>
  <body>
    <form action="getHeaderNames.jsp" method="post">
        <input type="hidden" name="stuid" value="123456" />
        name: <input type="text" name="name" value=""/>
        <input type="submit" name="submit1" value="提交" />
    </form>
  </body>
</html>
```



accept: text/html, application/xhtml+xml, image/jxr, */*
referer: http://172.18.187.230:8080/jsp/
accept-language: zh-CN
user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.79 Safari/537.36 Edge/14.14393
accept-encoding: gzip, deflate
host: 172.18.187.230:8080
connection: Keep-Alive
cookie: JSESSIONID=6BFB8B979CD184D44C58591B39C8CB5D
name: David Lee

提交

点击提交按钮后得到



accept: text/html, application/xhtml+xml, image/jxr, */*
referer: http://172.18.187.230:8080/jsp/getHeaderNames.jsp
accept-language: zh-CN
user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.79 Safari/537.36 Edge/14.14393
content-type: application/x-www-form-urlencoded
accept-encoding: gzip, deflate
host: 172.18.187.230:8080
content-length: 54
connection: Keep-Alive
cache-control: no-cache
cookie: JSESSIONID=6BFB8B979CD184D44C58591B39C8CB5D
name:

提交

数据库操作

- **浏览学生记录(简单)**

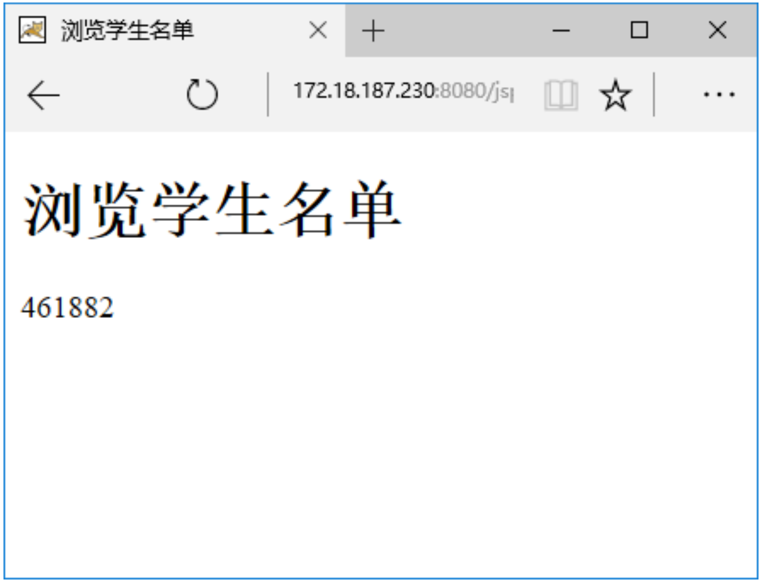
browseStu1.jsp

```
<%@ page language="java" import="java.util.*,java.sql.*"
         contentType="text/html; charset=utf-8"
%><% request.setCharacterEncoding("utf-8");
  String msg="";    String table="";
  String conStr = "jdbc:mysql://localhost:53306/teaching"
       + "?autoReconnect=true&useUnicode=true&characterEncoding=UTF-8";
  try {
    Class.forName("com.mysql.jdbc.Driver");  // 查找数据库驱动类
    Connection con=DriverManager.getConnection(conStr, "user", "123");
    Statement stmt=con.createStatement();  // 创建MySQL语句的对象
    ResultSet rs=stmt.executeQuery("select * from stu");//执行查询，返回结果集
    if(rs.next()) {                   //把游标(cursor)移至第一个或下一个记录
       table=rs.getString("name"); //如果还有记录，则得到数据域name的值
    }
    rs.close();  stmt.close();  con.close();
  }
  catch (Exception e){
     msg = e.getMessage();
  }
%>
```

MySql库的存放位置见前面“Servetlet程序”一节

- 数据库部分查看Java程序设计
- 这里只显示了第一个学生的名字，用一个循环语句就可以显示所有有学生记录。

```
<!DOCTYPE HTML>
<html>
<head>
<title>浏览学生名单</title>
</head>
<body>
  <h1>浏览学生名单</h1>
  <%=table%><br><br>
</body>
</html>
```



浏览学生名单

461882

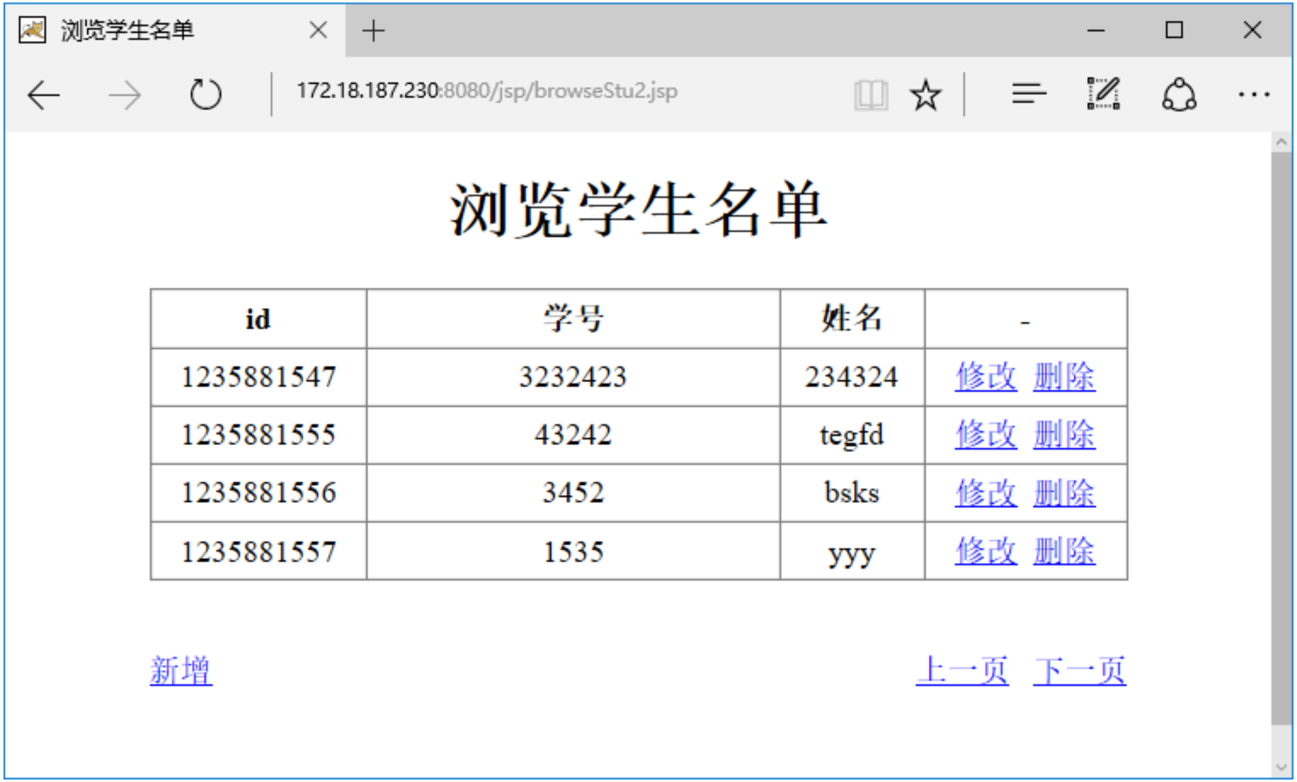
- 翻页浏览学生记录　　　　browseStu2.jsp

```
<%@ page language="java" import="java.util.*,java.sql.*"
         contentType="text/html; charset=utf-8"
%><%  request.setCharacterEncoding("utf-8");
      String msg ="";
      Integer pgno = 0;                   //当前页号
      Integer pgcnt = 4;                  //每页行数
      String param = request.getParameter("pgno");
      if(param != null && !param.isEmpty()){
           pgno = Integer.parseInt(param);
      }
      param = request.getParameter("pgcnt");
      if(param != null && !param.isEmpty()){
           pgcnt = Integer.parseInt(param);
      }
      int pgprev = (pgno>0)?pgno-1:0;
      int pgnext = pgno+1;
      String connectString = "jdbc:mysql://localhost:53306/teaching"
           + "?autoReconnect=true&useUnicode=true&characterEncoding=UTF-8";
      String user="user";
      String pwd="123";
      StringBuilder table = new StringBuilder();
```

```
try{
    Class.forName("com.mysql.jdbc.Driver");
    Connection con=DriverManager.getConnection(connectString, user, pwd);
    Statement stmt=con.createStatement();
    String sql=String.format("select * from stu limit %d,%d",
                              pgno*pgcnt,pgcnt);
    ResultSet rs=stmt.executeQuery(sql);
    table.append("<table><tr><th>id</th><th>学号</th><th>姓名</th>"+
                 "<th>-</th></tr>");
    while(rs.next()) {
       table.append(String.format(
              "<tr><td>%s</td><td>%s</td><td>%s</td><td>%s %s</td></tr>",
              rs.getString("id"),rs.getString("num"),rs.getString("name"),
              "<a href='updateStu.jsp?pid="+rs.getString("id")+"'>修改</a>",
              "<a href='deleteStu.jsp?pid="+rs.getString("id")+"'>删除</a>"
            )
       );
    }
    table.append("</table>");
    rs.close();  stmt.close();  con.close();
  }
  catch (Exception e){
    msg = e.getMessage();
  }
%>
```

```
<!DOCTYPE HTML>
<html>
 <head>
  <title>翻页浏览学生名单</title>
  <style>...  见下页</style>
 </head>
 <body>
  <div class="container">
     <h1>浏览学生名单</h1>
     <%=table%>
     <div style="float:left">
       <a href="addStu.jsp">新增</a>
     </div>
     <div style="float:right">
       <a href="browseStu2.jsp?pgno=<%=pgprev%>&pgcnt=<%=pgcnt%>">上一页</a>
       <a href="browseStu2.jsp?pgno=<%=pgnext%>&pgcnt=<%=pgcnt%>">下一页</a>
     </div>
     <br><br>
     <%=msg%><br><br>
  </div>
 </body>
</html>
```

```
<style>
  table{
          border-collapse: collapse;
          border: none;
          width: 500px;
   }
   td,th{
          border: solid grey 1px;
          margin: 0 0 0 0;
          padding: 5px 5px 5px 5px
   }
   a:link,a:visited {
     color:blue
   }

  .container{
    margin:0 auto;
    width:500px;
    text-align:center;
  }
</style>
```



浏览学生名单

id	学号	姓名	-
1235881547	3232423	234324	修改 删除
1235881555	43242	tegfd	修改 删除
1235881556	3452	bsks	修改 删除
1235881557	1535	yyy	修改 删除

新增 上一页 下一页

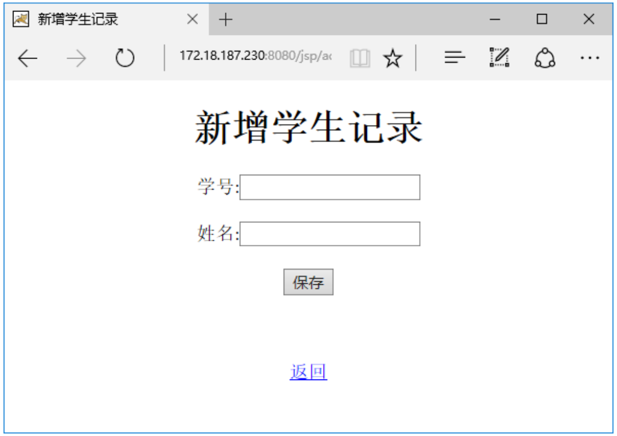
- 新增操作

addStu.jsp

```
<%@ page language="java" import="java.util.*,java.sql.*"
         contentType="text/html; charset=utf-8"%>
<% request.setCharacterEncoding("utf-8");
  String msg = "";
  String connectString = "jdbc:mysql://localhost:53306/teaching"
           + "?autoReconnect=true&useUnicode=true&characterEncoding=UTF-8";
  String user="user"; String pwd="123";
  String num = request.getParameter("num");
  String name = request.getParameter("name");
  if(request.getMethod().equalsIgnoreCase("post")){
     Class.forName("com.mysql.jdbc.Driver");
     Connection con = DriverManager.getConnection(connectString,user, pwd);
     Statement stmt = con.createStatement();
     try{ String fmt="insert into stu(num,name) values('%s', '%s')";
        String sql = String.format(fmt,num,name);
        int cnt = stmt.executeUpdate(sql);
        if(cnt>0)msg = "保存成功!";
        stmt.close();   con.close();
     }catch (Exception e){
        msg = e.getMessage();
     }
  }
%>
```

* num和name在get网页时取值null，因为不用保持输入值，所以null不会出现在输入框。

```
<!DOCTYPE HTML><html><head><title>新增学生记录</title>
  <style> a:link,a:visited {color:blue;}
     .container{
        margin:0 auto;
        width:500px;
        text-align:center;
     }
  </style>
</head><body>
  <div class="container">
        <h1>新增学生记录</h1>
        <form action="addStu.jsp" method="post" name="f">
                 学号:<input id="num" name="num" type="text" >
                 姓名:<input id="name" type="text" name="name" >
                 <input type="submit" name="sub" value="保存">
        </form>
        <%=msg%>
        <a href='browseStu.jsp'>返回</a>
  </div>
 </body>
</html>
```



附录

附录1、hello_jsp.java程序

黑色的粗体字为JSP的基本内置对象

```
package org.apache.jsp;
import javax.servlet.*;
import javax.servlet.http.*;
import javax.servlet.jsp.*;
public final class hello_jsp extends org.apache.jasper.runtime.HttpJspBase
    implements org.apache.jasper.runtime.JspSourceDependent {
    private static final javax.servlet.jsp.JspFactory _jspxFactory =
          javax.servlet.jsp.JspFactory.getDefaultFactory();
    ... //定义了对象：_jspx_dependants; _el_expressionfactory; _jsp_instancemanager;
    ...//定义了方法： getDependants() {...},_jspInit(){...}, _jspDestroy() {...}
    ...
    public void _jspService(
            final javax.servlet.http.HttpServletRequest request,
            final javax.servlet.http.HttpServletResponse response)
        throws java.io.IOException, javax.servlet.ServletException {
      final javax.servlet.jsp.PageContext pageContext;
      javax.servlet.http.HttpSession session = null;
      final javax.servlet.ServletContext application;
      final javax.servlet.ServletConfig config;
      javax.servlet.jsp.JspWriter out = null;
      final java.lang.Object page = this;
      javax.servlet.jsp.JspWriter _jspx_out = null;
      javax.servlet.jsp.PageContext _jspx_page_context = null;
```

```java
try {
      response.setContentType("text/html;charset=gb2312");
      pageContext = _jspxFactory.getPageContext(this, request, response,
                         null, true, 8192, true);
      _jspx_page_context = pageContext;
      application = pageContext.getServletContext();
      config = pageContext.getServletConfig();
      session = pageContext.getSession();
      out = pageContext.getOut();
      _jspx_out = out;
      out.write("<html>\r\n");
      out.write("<head>\r\n");
      out.write("<title>Hello world</title>\r\n");
      out.write("</head>\r\n");
      out.write("<body>\r\n");
      out.print("Hello world!");
      out.write("</body>\r\n");
      out.write("</html>\r\n");
      out.write("\r\n");
      out.write("\r\n");
      out.write("\r\n");
    } catch (java.lang.Throwable t) {
      if (!(t instanceof javax.servlet.jsp.SkipPageException)){
        out = _jspx_out;
        if (out != null && out.getBufferSize() != 0)
          try { out.clearBuffer(); } catch (java.io.IOException e) {}
        if (_jspx_page_context != null)
               _jspx_page_context.handlePageException(t);
      }
    } finally {
      _jspxFactory.releasePageContext(_jspx_page_context);
    }
  }  // _jspService
} // class
```

附录2、JSP声明的源码转换

```
<%@ page contentType="text/html;charset=gb2312"%>
<%! int cnt = 0;
    int getSum(int n){
      int sum=0;
      for(int i=1;i<=n;i++){
         sum=sum+i;
      }
      return sum;
    }
    class Sum{
      int getSum(int n){
        int sum=0;
        for(int i=1;i<=n;i++){
          sum=sum+i;
        }
        return sum;
      }
    }
 %>
<html><head><title>getSumFunc</title></head><body>
  <h1>getSumFunc</h1>
  <p>sum1:<%=getSum(100)%></p>
  <p>sum2:<% Sum sum=new Sum();%><%=sum.getSum(100)%></p>
  count:<%=++cnt%>
</body></html>
```

getSumFuncClass.jsp

getSumFuncClass_jsp.jsp

```
package org.apache.jsp;
import javax.servlet.*;
import javax.servlet.http.*;
import javax.servlet.jsp.*;
public final class getSumfuncClass_jsp extends
                        org.apache.jasper.runtime.HttpJspBase
    implements org.apache.jasper.runtime.JspSourceDependent,
                 org.apache.jasper.runtime.JspSourceImports {
    int cnt = 0;
    int getSum(int n){
      int sum=0;
      for(int i=1;i<=n;i++){
         sum=sum+i;
      }
      return sum;
    }
    class Sum{
      int getSum(int n){
        int sum=0;
        for(int i=1;i<=n;i++){
          sum=sum+i;
        }
        return sum;
      }
    }
```

```
private static final javax.servlet.jsp.JspFactory _jspxFactory =
          javax.servlet.jsp.JspFactory.getDefaultFactory();
  private static java.util.Map<java.lang.String,java.lang.Long>
                  _jspx_dependants;
  private static final java.util.Set<java.lang.String> _jspx_imports_packages;
  private static final java.util.Set<java.lang.String> _jspx_imports_classes;
  static {
    _jspx_imports_packages = new java.util.HashSet<>();
    _jspx_imports_packages.add("javax.servlet");
    _jspx_imports_packages.add("javax.servlet.http");
    _jspx_imports_packages.add("javax.servlet.jsp");
    _jspx_imports_classes = null;
  }
private volatile javax.el.ExpressionFactory _el_expressionfactory;
  private volatile org.apache.tomcat.InstanceManager _jsp_instancemanager;
  public java.util.Map<java.lang.String,java.lang.Long> getDependants() {
    return _jspx_dependants;
  }
  public java.util.Set<java.lang.String> getPackageImports() {
    return _jspx_imports_packages;
  }
  public java.util.Set<java.lang.String> getClassImports() {
    return _jspx_imports_classes;
  }
```

```
 public javax.el.ExpressionFactory _jsp_getExpressionFactory() {
  if (_el_expressionfactory == null) {
    synchronized (this) {
      if (_el_expressionfactory == null) {
        _el_expressionfactory =
               _jspxFactory.getJspApplicationContext(getServletConfig().getS
               ervletContext()).getExpressionFactory();
      }
    }
  }
  return _el_expressionfactory;
}
public org.apache.tomcat.InstanceManager _jsp_getInstanceManager() {
  if (_jsp_instancemanager == null) {
    synchronized (this) {
      if (_jsp_instancemanager == null) {
        _jsp_instancemanager =
                org.apache.jasper.runtime.InstanceManagerFactory.getInstance
                Manager(getServletConfig());
      }
    }
  }
  return _jsp_instancemanager;
}
public void _jspInit() {
}
public void _jspDestroy() {
}
```

```
public void _jspService(final javax.servlet.http.HttpServletRequest request,
                         final javax.servlet.http.HttpServletResponse response)
     throws java.io.IOException, javax.servlet.ServletException {
   final java.lang.String _jspx_method = request.getMethod();
   if (!"GET".equals(_jspx_method) && !"POST".equals(_jspx_method) &&
        !"HEAD".equals(_jspx_method)&&
        !javax.servlet.DispatcherType.ERROR.equals(request.getDispatcherType())
        ) {
     response.sendError(HttpServletResponse.SC_METHOD_NOT_ALLOWED, "JSPs only
      permit GET POST or HEAD");
     return;
   }
   final javax.servlet.jsp.PageContext pageContext;
   javax.servlet.http.HttpSession session = null;
   final javax.servlet.ServletContext application;
   final javax.servlet.ServletConfig config;
   javax.servlet.jsp.JspWriter out = null;
   final java.lang.Object page = this;
   javax.servlet.jsp.JspWriter _jspx_out = null;
   javax.servlet.jsp.PageContext _jspx_page_context = null;
```

```
try { response.setContentType("text/html;charset=gb2312");
    pageContext = _jspxFactory.getPageContext(this, request, response,
                        null, true, 8192, true);
    _jspx_page_context = pageContext;
    application = pageContext.getServletContext();
    config = pageContext.getServletConfig();
    session = pageContext.getSession();
    out = pageContext.getOut();    _jspx_out = out;
    out.write('\r');
    out.write('\n');
    out.write("\r\n");
    out.write("<html>\r\n");
    out.write("<head>\r\n");
    out.write("<title>getSumFunc</title>\r\n");
    out.write("</head>\r\n");
    out.write("<body>\r\n");
    out.write("  <h1>getSumFunc</h1>\r\n");
    out.write("  <p>sum1:");
    out.print(getSum(100));
    out.write("</p>\r\n");
    out.write("  <p>sum2:");
    Sum sum=new Sum();
    out.print(sum.getSum(100));
    out.write("</p>\r\n");
    out.write("  count:");
    out.print(++cnt);
    out.write("\r\n");
    out.write("</body>\r\n");
    out.write("</html>\r\n");
}
```

```
   catch (java.lang.Throwable t) {
      if (!(t instanceof javax.servlet.jsp.SkipPageException)){
        out = _jspx_out;
        if (out != null && out.getBufferSize() != 0)
          try {
            if (response.isCommitted()) {
              out.flush();
            } else {
              out.clearBuffer();
            }
          }
          catch (java.io.IOException e) {}
        if (_jspx_page_context != null)
             _jspx_page_context.handlePageException(t);
        else
            throw new ServletException(t);
      }
    } finally {
      _jspxFactory.releasePageContext(_jspx_page_context);
    }
  }
}
```