

知识点列表

编号	名称	描述	级别
1	改进 ListEmpServlet	通过本案例理解添加员工和查看员工列表流程	*
2	案例 1：重定向	掌握并熟练应用重定向，理解其流程	***
3	案例 2：雇员的增减改查	Servlet 中进行数据库操作的综合应用	**
4	什么是 DAO	了解 DAO 的含义	*
5	如何写一个 DAO	掌握 DAO 的编程步骤	**
6	dao 工厂（工厂模式）	理解并掌握	**

注： "*"理解级别 "***"掌握级别 "****"应用级别

1. 案例演示 1：改进 ListEmpServlet *

演示

步骤 1

在 ListEmpServlet.java 中添加代码“链接”



```

63      }
64      out.println("</table>");
65      out.println(
66          "<a href='addEmp.html'>" +
67          "增加新雇员</a>");
68
69      out.close();
70      } catch (Exception e) {
71          e.printStackTrace();
72          throw new ServletException(e);
73      } finally {
74          if (conn != null) {
75              try {
76                  conn.close();

```

步骤 2

在 AddEmpServlet.java 中添加代码 “链接”

```

AddEmpServlet.java
50     response.setContentType(
51         "text/html;charset=utf-8");
52     PrintWriter out = response.getWriter();
53     out.println("添加雇员成功");
54     out.println(
55         "<a href='list'>" +
56         "查看雇员</a>");
57     out.close();
58
59     } catch (Exception e) {

```

结果演示

1) 添加新用户



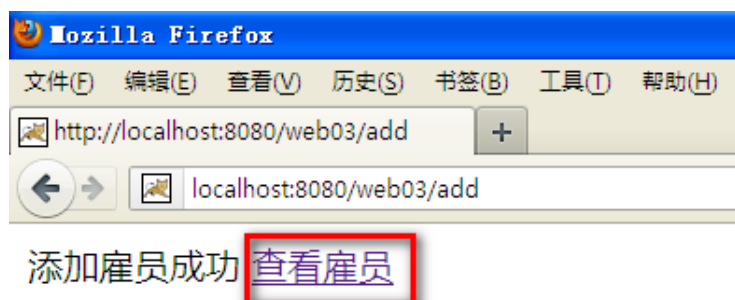
添加雇员

姓名:

薪水:

年龄:

2) 添加成功后显示 “添加雇员成功”，点击 “查看雇员” 链接

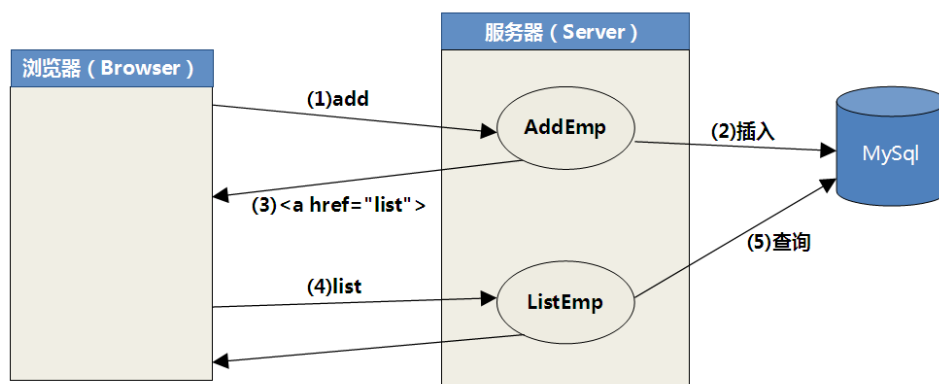


3) 返回“查看雇员”页面，可以重新添加新雇员



图示演示

添加员工和查看员工列表流程图

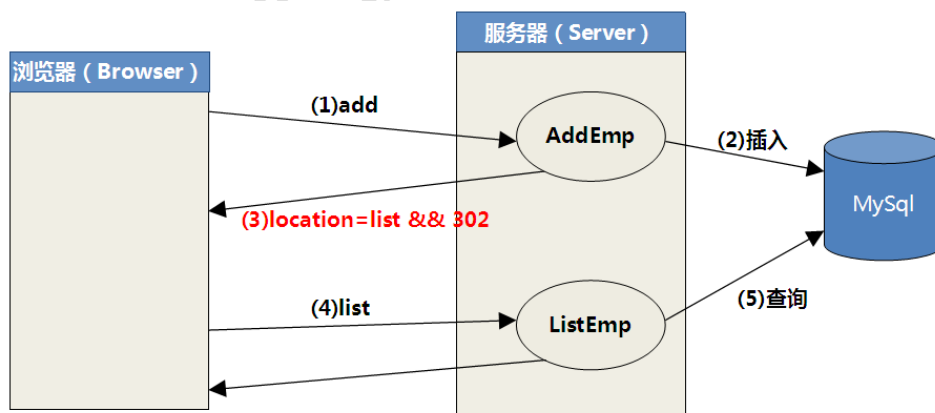


- 1) 用户在浏览器提交 form 表单，通过 add 到服务器找到 AddEmp
- 2) AddEmp 向数据库中插入一条数据
- 3) AddEmp 返回给用户一条信息“数据插入成功”，并且提供一个查看员工信息的链接
- 4) 用户点击链接到服务器找到 ListEmp
- 5) ListEmp 到数据库中查询数据并返回给浏览器

2. 重定向 ***

图示演示

重定向



- a. 用户在浏览器提交 form 表单，通过 add 到服务器找到 AddEmp
- b. AddEmp 向数据库中插入一条数据
- c. **AddEmp 返回给用户一个消息头 location= list (ListEmp 的地址) 和 302 状态码**
当浏览器碰到状态码 302 后，浏览器会自动向新地址发请求

- d. 浏览器自动通过 list 找到 ListEmp
- e. ListEmp 到数据库中查询数据并返回给浏览器

1) 什么是重定向?

服务器向浏览器发送一个状态码 302 及一个消息头 location，浏览器收到后，会立即向 location 所指向的地址发送请求。

2) 如何重定向

response.sendRedirect(String url); 其中,url 是重定向的地址。

3) 编程要注意的问题

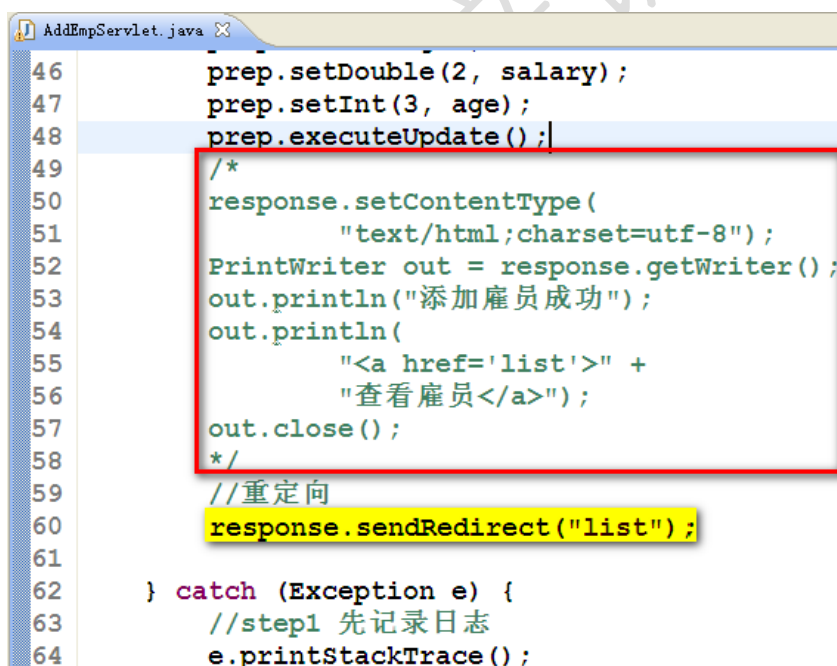
- ✓ 重定向之前，不能够有 out.close()或者 out.flush()操作。
- ✓ 重定向之前，如果 response 对象缓存有数据，则这些数据会被清空。

4) 重定向的特点

- ✓ 重定向的地址是任意的
- ✓ 重定向之后，浏览器地址栏的地址变成了 location 所指定的地址。

演示

重定向



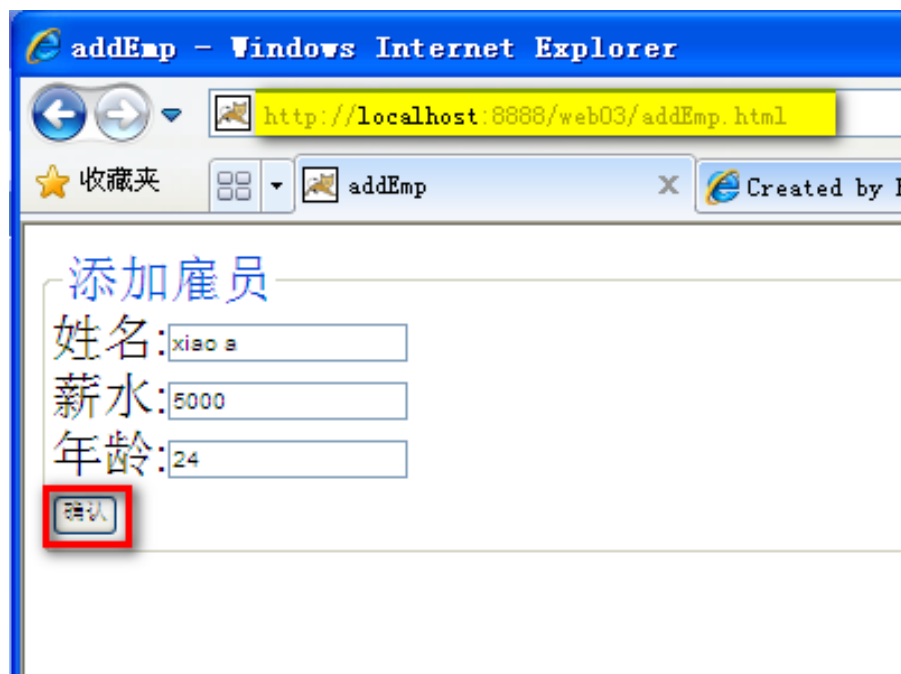
```

46      prep.setDouble(2, salary);
47      prep.setInt(3, age);
48      prep.executeUpdate();
49      /*
50      response.setContentType(
51          "text/html;charset=utf-8");
52      PrintWriter out = response.getWriter();
53      out.println("添加雇员成功");
54      out.println(
55          "<a href='list'>" +
56          "查看雇员</a>");
57      out.close();
58      */
59      //重定向
60      response.sendRedirect("list");
61
62  } catch (Exception e) {
63      //step1 先记录日志
64      e.printStackTrace();
    
```

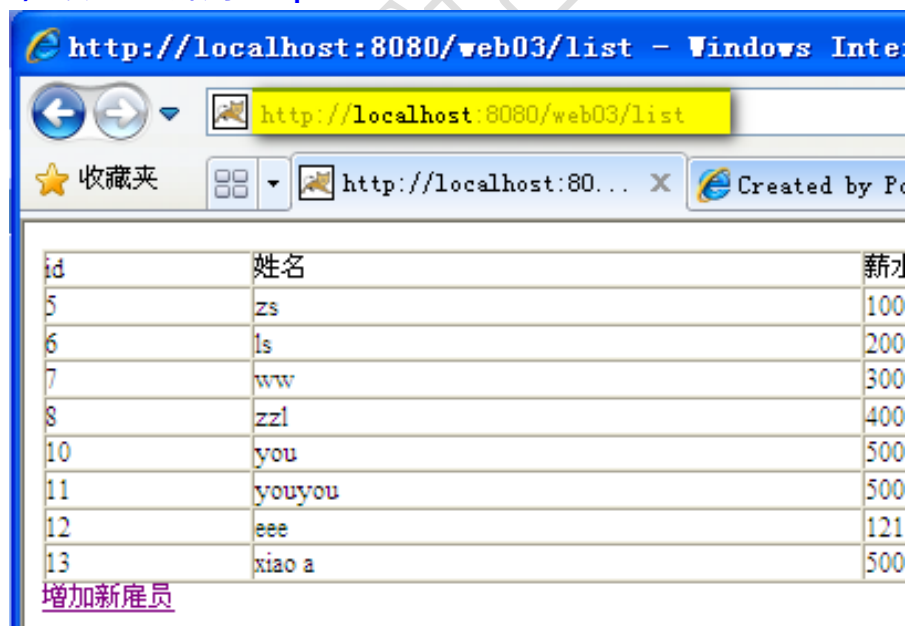
执行结果

- 1) 开启 TCP/IP Monitor，访问 “http://localhost:8080/web03/addEmp.html”

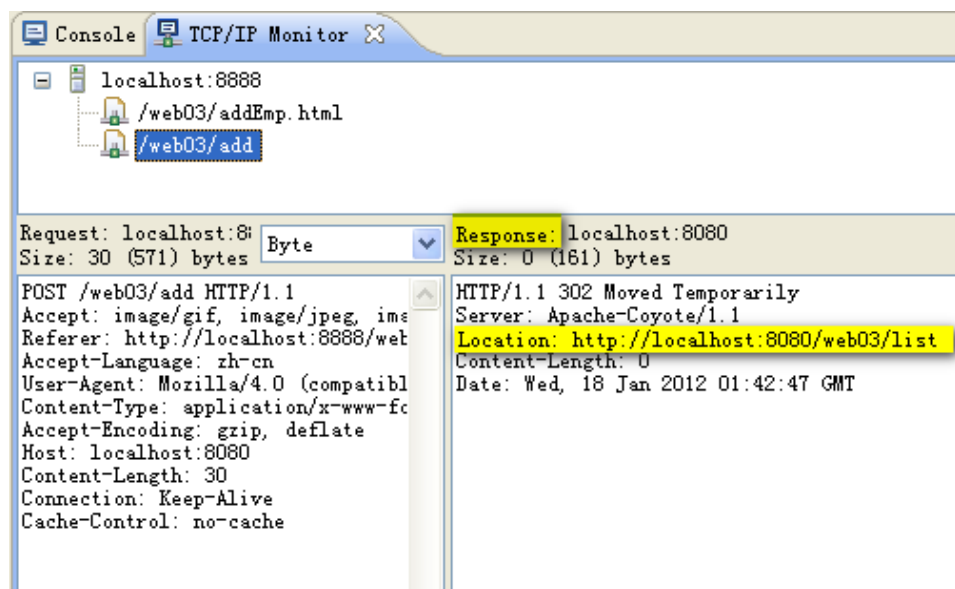
点击“确定”提交数据



2) 页面直接跳转到 “http://localhost:8080/web03/list”

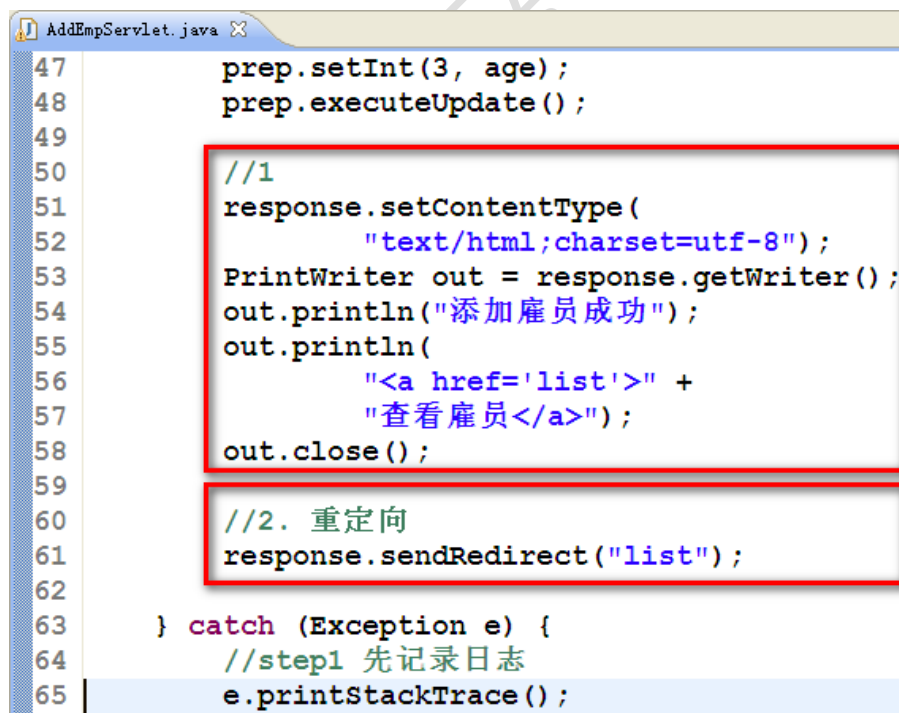


3) TCP/IP Monitor 显示



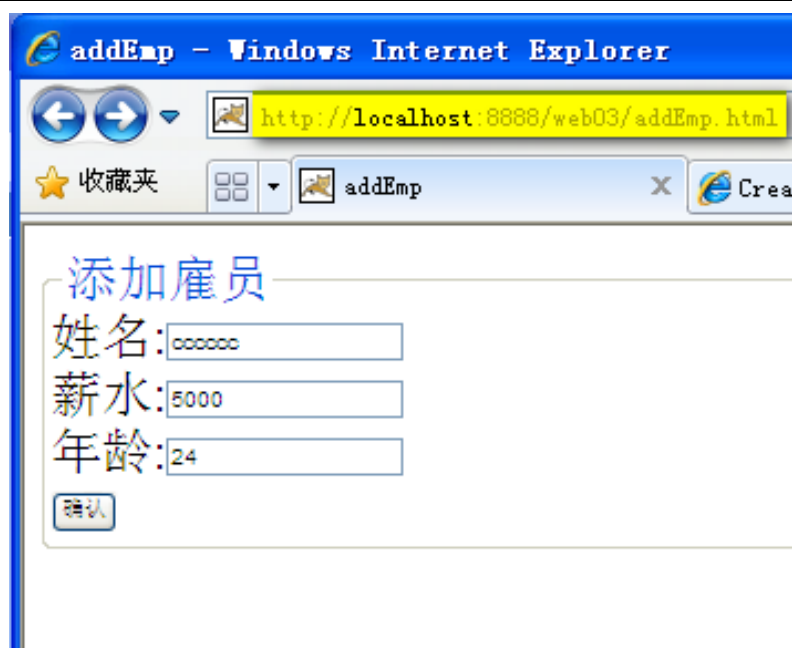
需要注意的问题 01

重定向之前，不能够有 out.close()或者 out.flush()操作，演示如下所示：

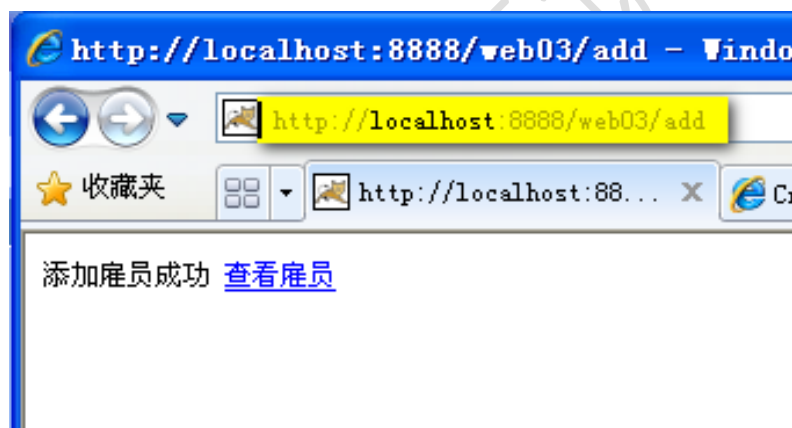


执行结果

- 1) 访问 addEmp.html



2) 重定向无效



3) 并且出现异常


```

Console X TCP/IP Monitor
tomcat5Server [Remote Java Application] C:\Program Files\Java\jdk1.6.0_06\bin\javaw.
name:cccccc
salary:5000.0
age:24
java.lang.IllegalStateException
    at org.apache.catalina.connector.Resp
    at web.AddEmpServlet.service (AddEmpSer
    at javax.servlet.http.HttpServlet.serv
    at org.apache.catalina.core.Applicatio
    at org.apache.catalina.core.Applicatio
    at org.apache.catalina.core.StandardWr
    at org.apache.catalina.core.StandardCc
    at org.apache.catalina.core.StandardHc

```

需要注意的问题 02

重定向之前，如果 response 对象缓存有数据，则这些数据会被清空

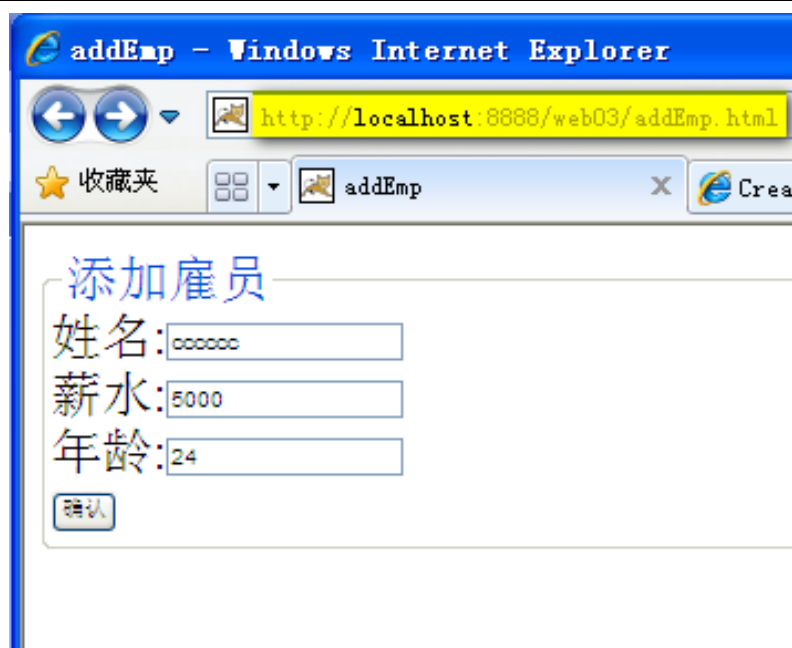
```

AddEmpServlet.java X
47      prep.setInt(3, age);
48      prep.executeUpdate();
49
50      //1
51      response.setContentType(
52          "text/html;charset=utf-8");
53      PrintWriter out = response.getWriter();
54      out.println("添加雇员成功");
55      out.println(
56          "<a href='list'>" +
57          "查看雇员</a>");
58      //out.close();
59
60      //2. 重定向
61      response.sendRedirect("list");
62
63  } catch (Exception e) {
64      //step1 先记录日志
65      e.printStackTrace();
66      //step2 抛出
67      throw new ServletException(e);

```

执行结果

- 1) 访问 addEmp.html



2) 可以重定向

可以重定向, 因为 response 中的数据在重定向前被清空了, out.println(...)没有起作用



【案例 1】重定向

1) AddEmpServlet.java

```
package web;

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

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

public class AddEmpServlet extends HttpServlet{

    public void service(HttpServletRequest request,
                        HttpServletResponse response)
        throws ServletException,IOException{

        //这行代码要放在 getParameter()执行之前。
        request.setCharacterEncoding("utf-8");

        String name = request.getParameter("name");
        double salary = Double.parseDouble(
            request.getParameter("salary"));
        int age = Integer.parseInt(
            request.getParameter("age"));
        System.out.println("name:" + name);
        System.out.println("salary:" + salary);
        System.out.println("age:" + age);

        //访问数据库
        Connection conn = null;
        try {
```

```

Class.forName("com.mysql.jdbc.Driver");
conn = DriverManager
    .getConnection(
        "jdbc:mysql://localhost:3306/jd1109db2",
        "root", "root");

PreparedStatement prep =
    conn.prepareStatement(
        "insert into t_emp(name,salary,age) " +
        "values(?,?,?)");
prep.setString(1, name);
prep.setDouble(2, salary);
prep.setInt(3, age);
prep.executeUpdate();
/*
response.setContentType(
    "text/html;charset=utf-8");
PrintWriter out = response.getWriter();
out.println("添加雇员成功");
out.println(
    "<a href='list'> " +
    "查看雇员</a> ");
out.close();
*/
//重定向
response.sendRedirect("list");

} catch (Exception e) {
    //step1 先记录日志
    e.printStackTrace();
    //step2 抛出
    throw new ServletException(e);

}finally{
    if(conn!=null){
        try {
            conn.close();
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}

```

```

    }
    }
    }
    }
}

```

2) ListEmpServlet.java

```

package web;

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

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

public class ListEmpServlet extends HttpServlet{
    public void service(HttpServletRequest request,
                        HttpServletResponse response)
        throws ServletException,IOException{
        //访问数据库
        Connection conn = null;
        try {
            Class.forName(
                "com.mysql.jdbc.Driver");
            conn =
                DriverManager.getConnection(
                    "jdbc:mysql://localhost:3306/jd1109db2",
                    "root","root");
            Statement stat =
                conn.createStatement();

```

```

ResultSet rst = stat.executeQuery(
    "select * from t_emp");

//使用查询得到的结果，生成一个表格
response.setContentType(
    "text/html;charset=utf-8");
PrintWriter out =
    response.getWriter();
out.println(
    "<table border='1' " +
    "width='60%' " +
    "cellpadding='0' " +
    "cellspacing='0'>");
out.println("<tr> " +
    "<td>id</td> " +
    "<td>姓名</td> " +
    "<td>薪水</td> " +
    "<td>年龄</td></tr>");
while(rst.next()){
    long id =
        rst.getLong("id");
    String name =
        rst.getString("name");
    double salary =
        rst.getDouble("salary");
    int age =
        rst.getInt("age");
    out.println("<tr><td> "
        + id + "</td><td> "
        + name + "</td><td> "
        + salary + "</td><td> "
        + age + "</td></tr>");
}
out.println("</table>");
out.println(
    "<a href='addEmp.html'> " +
    "增加新雇员</a>");

```

```

        out.close();
    } catch (Exception e) {
        e.printStackTrace();
        throw new ServletException(e);
    } finally {
        if(conn!=null){
            try {
                conn.close();
            } catch (SQLException e) {
                e.printStackTrace();
            }
        }
    }
}
}
}
}

```

3) addEmp.html

```

<html>
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=utf-8">
<title>addEmp</title>
</head>
<body style="font-size:30px;">
    <form action="add" method="post">
        <fieldset>
            <legend>添加雇员</legend>
            姓名:<input name="name"/><br/>
            薪水:<input name="salary"/><br/>
            年龄:<input name="age"/><br/>
            <input type="submit" value="确认"/>
        </fieldset>
    </form>
</body>
</html>

```

4) err.html

```
<html>
<head>
<meta http-equiv="Content-Type"
      content="text/html; charset=UTF-8">
<title>Insert title here</title>
</head>
<body style="font-size:30px;color:red;">
    发生了系统错误，请稍后
    <a href="addEmp.html">重试</a>
</body>
</html>
```

5) web.xml

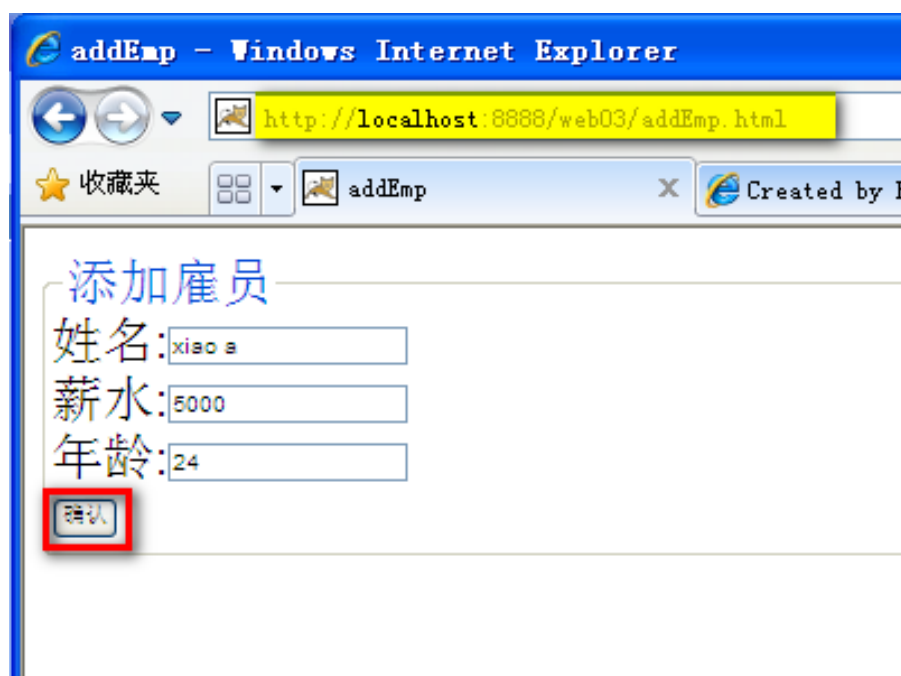
```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.4"
  xmlns=
    "http://java.sun.com/xml/ns/j2ee"
  xmlns:xsi=
    "http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation=
    "http://java.sun.com/xml/ns/j2ee
    http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd">
  <servlet>
    <servlet-name>
      addEmpServlet</servlet-name>
    <servlet-class>
      web.AddEmpServlet</servlet-class>
  </servlet>
  <servlet>
    <servlet-name>
      listEmpServlet</servlet-name>
    <servlet-class>
      web.ListEmpServlet</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>addEmpServlet</servlet-name>
    <url-pattern>/add</url-pattern>
```



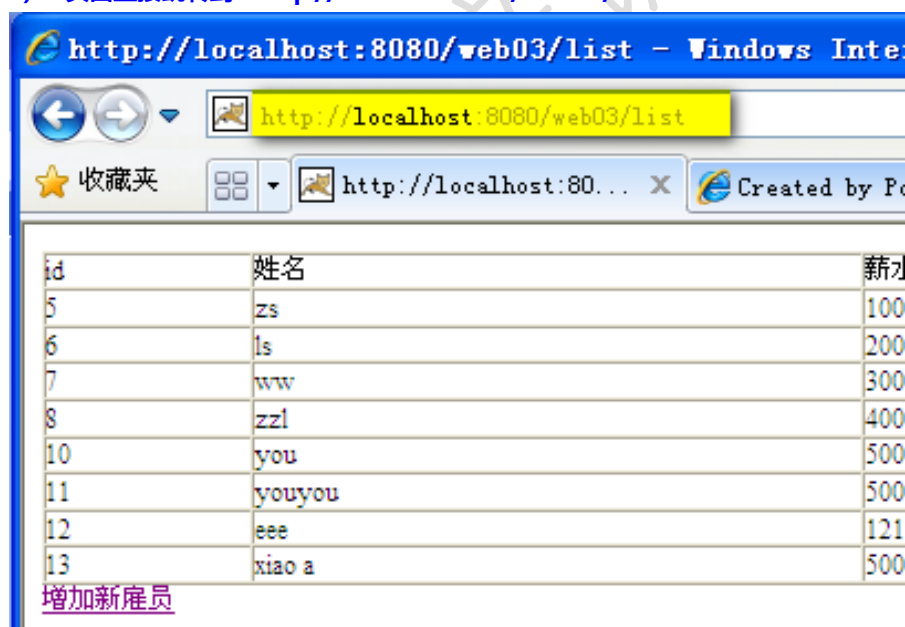
```
</servlet-mapping>
<servlet-mapping>
  <servlet-name>
    listEmpServlet</servlet-name>
  <url-pattern>
    /list</url-pattern>
</servlet-mapping>
<error-page>
  <!-- 方式1
  <error-code>500</error-code> -->
  <!-- 方式2 -->
  <exception-type>
    javax.servlet.ServletException
  </exception-type>
  <location>/error.html</location>
</error-page>
</web-app>
```

执行结果

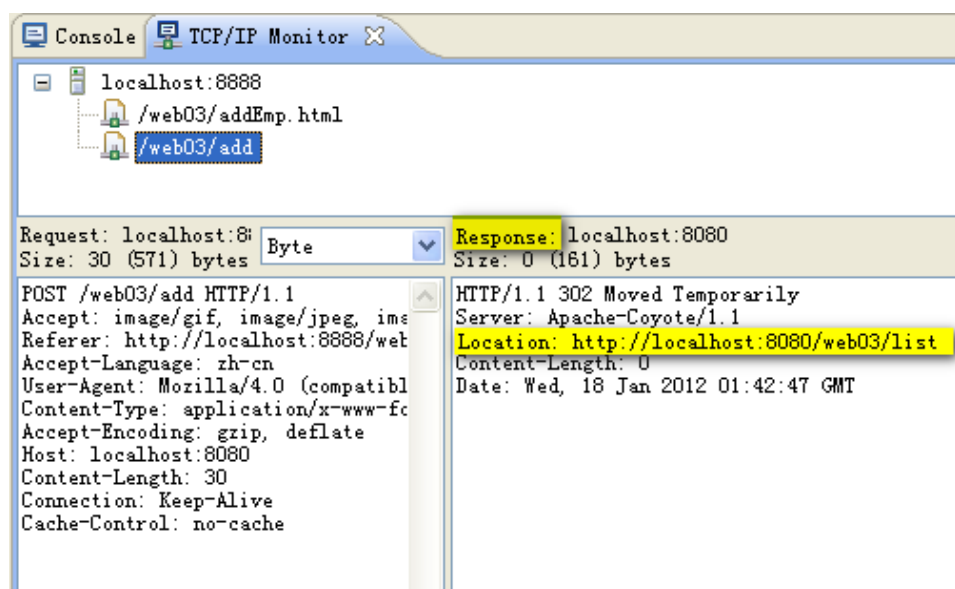
- 1) 开启 TCP/IP Monitor , 访问 “<http://localhost:8888/web03/addEmp.html>”
点击 “确定” 提交数据



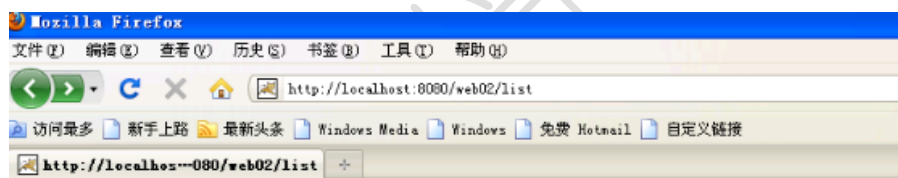
2) 页面直接跳转到 “http://localhost:8080/web03/list”



3) TCP/IP Monitor 显示



3. 案例演示 2：雇员的删改 **



id	姓名	薪水	年龄	操作
1	lg	2000.0	22	删除 修改
2	lyg	2000.0	21	删除 修改
3	csb	2000.0	22	删除 修改
4	sdd	3000.0	22	删除 修改
5	ww	2100.0	22	删除 修改
6	abc	2222.0	22	删除 修改
7	abc2	2222.0	22	删除 修改
8	abc3	2222.0	22	删除 修改
9	abc4	3000.0	22	删除 修改

[增加新雇员](#)

3.1. 步骤演示

演示

1) 修改 ListEmpServlet.java 添加“删除”链接

```

ListEmpServlet.java
48         "<td>年龄</td></tr>");
49         while(rst.next()){
50             long id =
51                 rst.getLong("id");
52             String name =
53                 rst.getString("name");
54             double salary =
55                 rst.getDouble("salary");
56             int age =
57                 rst.getInt("age");
58             out.println("<tr><td>"
59                 + id + "</td><td> "
60                 + name + "</td><td>"
61                 + salary + "</td><td> "
62                 + age + "</td>" +
63                 "<td><a href='del?id=" + id + "'>" +
64                 "删除</td></tr>");
65         }
66         out.println("</table>");
67         out.println(
68             "<a href='addEmp.html'>" +
69             "增加新雇员</a>");
70
71         out.close();
    
```

2) 测试

重新部署，并访问 “http://localhost:8080/web03/list”

id	姓名	薪水	年龄	
5	zs	1000.0	20	删除
6	ls	2000.0	21	删除
7	ww	3000.0	22	删除
8	zzl	4000.0	25	删除
10	you	5000.0	23	删除
11	youyou	5000.0	23	删除
12	eee	1212.0	21	删除
13	xiao a	5000.0	24	删除
14	cccccc	5000.0	24	删除
15	xxxx	12121.0	33	删除

增加新雇员

将鼠标放在“删除”按钮上时，状态栏会显示地址

http://localhost:8080/web03/del?id=5

3) 新建 DelEmpServlet.java

```

DelEmpServlet.java X
1 package web;
2
3 import java.io.IOException;
13
14 public class DelEmpServlet
15         extends HttpServlet{
16
17     public void service(HttpServletRequest request,
18                         HttpServletResponse response)
19         throws ServletException, IOException{
20         long id =
21             Long.parseLong(
22                 request.getParameter("id"));

```

```

23 //访问数据库
24 Connection conn = null;
25 try {
26     Class.forName("com.mysql.jdbc.Driver");
27     conn = DriverManager
28         .getConnection(
29             "jdbc:mysql://localhost:3306/jd1109db2",
30             "root", "root");
31     PreparedStatement prep =
32         conn.prepareStatement(
33             "delete from t_emp where id = ?");
34     prep.setLong(1, id);
35     prep.executeUpdate();
36     //重定向
37     response.sendRedirect("list");
38
39 } catch (Exception e) {
40     //step1 先记录日志
41     e.printStackTrace();
42     //step2 抛出
43     throw new ServletException(e);
44
45 }finally{
46     if(conn!=null){
47         try {
48             conn.close();
49         } catch (SQLException e) {
50             e.printStackTrace();
51         }
52     }
53 }
54 }
55 }

```

4) 配置 web.xml

```

16<
17    <servlet-name>
18        delEmpServlet</servlet-name>
19    <servlet-class>
20        web.DelEmpServlet</servlet-class>
21</servlet>
22<servlet-mapping>
23    <servlet-name>addEmpServlet</servlet-name>
24    <url-pattern>/add</url-pattern>
25</servlet-mapping>
26<servlet-mapping>
27    <servlet-name>listEmpServlet</servlet-name>
28    <url-pattern>/list</url-pattern>
29</servlet-mapping>
30<servlet-mapping>
31    <servlet-name>
32        delEmpServlet</servlet-name>
33    <url-pattern>
34        /del</url-pattern>
35</servlet-mapping>
36</error-page>

```

5) 测试

点击“删除”即可在数据库中删除该条数据

http://localhost:8080/web03/list - Windows Internet Explorer

http://localhost:8080/web03/list

收藏夹 http://localhost:80... Created by PowerCreator

id	姓名	薪水	年龄	操作
5	zs	1000.0	20	删除
6	ls	2000.0	21	删除
7	ww	3000.0	22	删除
8	zzl	4000.0	25	删除
10	you	5000.0	23	删除
11	youyou	5000.0	23	删除

[增加新雇员](#)

6) 为 ListEmpServlet.java 添加“修改”链接

```

ListEmpServlet.java
57         int age =
58             rst.getInt("age");
59         out.println("<tr><td>"
60             + id + "</td><td>"
61             + name + "</td><td>"
62             + salary + "</td><td>"
63             + age + "</td><td>" +
64             "<a href='del?id=" + id + "'>" +
65             "删除</a>" +
66             "&nbsp;" +
67             "<a href='load?id=" + id + "'>" +
68             "修改</a></td></tr>");
69     }
70     out.println("</table>");
71     out.println(
72         "<a href='addEmp.html'>" +
73         "增加新雇员</a>");
74

```

7) 测试

http://localhost:8080/web03/list - Windows Internet Explorer

http://localhost:8080/web03/list

收藏夹 http://localhost:80... Created by PowerCreator

id	姓名	薪水	年龄	操作
6	ls	2000.0	21	删除 修改
7	ww	3000.0	22	删除 修改
10	you	5000.0	23	删除 修改
11	youyou	5000.0	23	删除 修改

[增加新雇员](#)

8) 新建 LoadEmpServlet.java


```
LoadEmpServlet.java X
1 package web;
2
3 import java.io.IOException;
16
17 public class LoadEmpServlet extends HttpServlet{
18     public void service(HttpServletRequest request,
19         HttpServletResponse response)
20         throws ServletException, IOException{
21         long id = Long.parseLong(
22             request.getParameter("id"));
23         //访问数据库
24         Connection conn = null;
25         try {
26             Class.forName(
27                 "com.mysql.jdbc.Driver");
28             conn =
29                 DriverManager.getConnection(
30                     "jdbc:mysql://localhost:3306/jd1109db2"
31                     "root","root");
32             PreparedStatement prep =
33                 conn.prepareStatement(
34                     "select * from t_emp where id = ?");
35             prep.setLong(1, id);
36             ResultSet rst = prep.executeQuery();
37             response.setContentType(
```

```

38         "text/html;charset=utf-8");
39     PrintWriter out = response.getWriter();
40     if(rst.next()){
41         String name =
42             rst.getString("name");
43         double salary =
44             rst.getDouble("salary");
45         int age =
46             rst.getInt("age");
47         out.println(
48             "<form action='modify?id=" +
49             id + "' method='post'>");
50         out.println(
51             "id:" + id + "<br/>");
52         out.println(
53             "姓名:<input name='name' value='" +
54             name + "'/><br/>");
55         out.println(
56             "薪水:<input name='salary' value='" +
57             salary + "'/><br/>");
58         out.println(
59             "年龄:<input name='age' value='" +
60             age + "'/><br/>");
61         out.println(
62             "<input type='submit' " +
63             "value='确认' />");
64         out.println(
65             "</form>");
66         out.close();
67     }
68 } catch (Exception e) {
69     e.printStackTrace();
70     throw new ServletException(e);
71 } finally {
72     if(conn!=null){
73         try {
74             conn.close();
75         } catch (SQLException e) {
76             e.printStackTrace();
77         }
78     }
79 }
80 }
81 }

```

9) 修改 web.xml

```

15  </servlet>
16      <servlet-name>delEmpServlet</servlet-name>
17      <servlet-class>web.DelEmpServlet</servlet-c
18  </servlet>
19  <servlet>
20      <servlet-name>loadEmpServlet</servlet-name>
21      <servlet-class>web.LoadEmpServlet</servlet-
22  </servlet>
23
24  <servlet-mapping>
25      <servlet-name>loadEmpServlet</servlet-name>
26      <url-pattern>/load</url-pattern>
27  </servlet-mapping>
28  <servlet-mapping>
29      <servlet-name>addEmpServlet</servlet-name>
30      <url-pattern>/add</url-pattern>
31  </servlet-mapping>
32  <servlet-mapping>

```

10) 测试

点击修改

http://localhost:8080/web03/list - Windows Internet Explorer

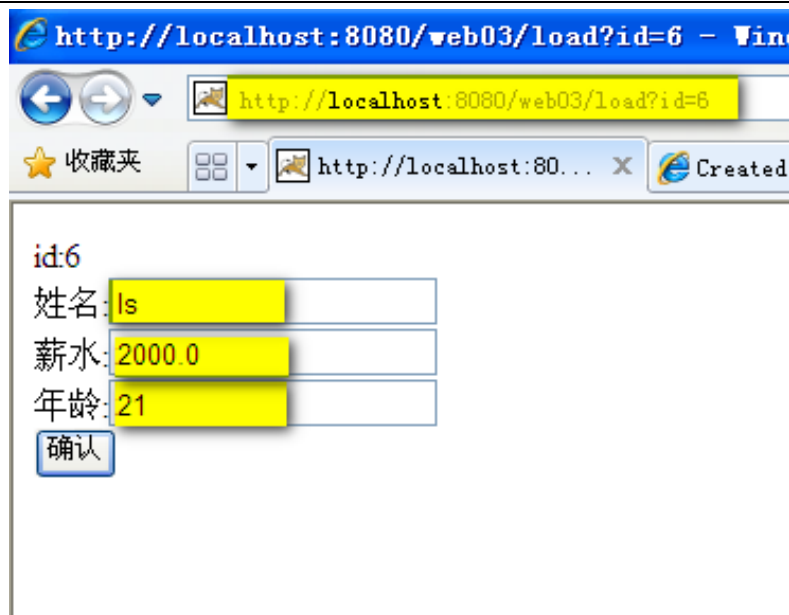
http://localhost:8080/web03/list

收藏夹 http://localhost:80... X Created by PowerCreator

id	姓名	薪水	年龄	操作
6	ls	2000.0	21	删除 修改
7	ww	3000.0	22	删除 修改
10	you	5000.0	23	删除 修改
11	youyou	5000.0	23	删除 修改

[增加新雇员](#)

跳转到 LoadEmpServlet.java



11) 新建 ModifyEmpServlet.java

```

1 package web;
2
3 import java.io.IOException;
4
5
6
7
8
9
10
11
12
13
14
15 public class ModifyEmpServlet extends HttpServlet{
16     public void service(HttpServletRequest request,
17         HttpServletResponse response)
18         throws ServletException, IOException{
19         request.setCharacterEncoding("utf-8");
20         long id =
21             Long.parseLong(
22                 request.getParameter("id"));
23         String name =
24             request.getParameter("name");
25         double salary =
26             Double.parseDouble(
27                 request.getParameter("salary"));
28         int age = Integer.parseInt(
29             request.getParameter("age"));
30

```

```

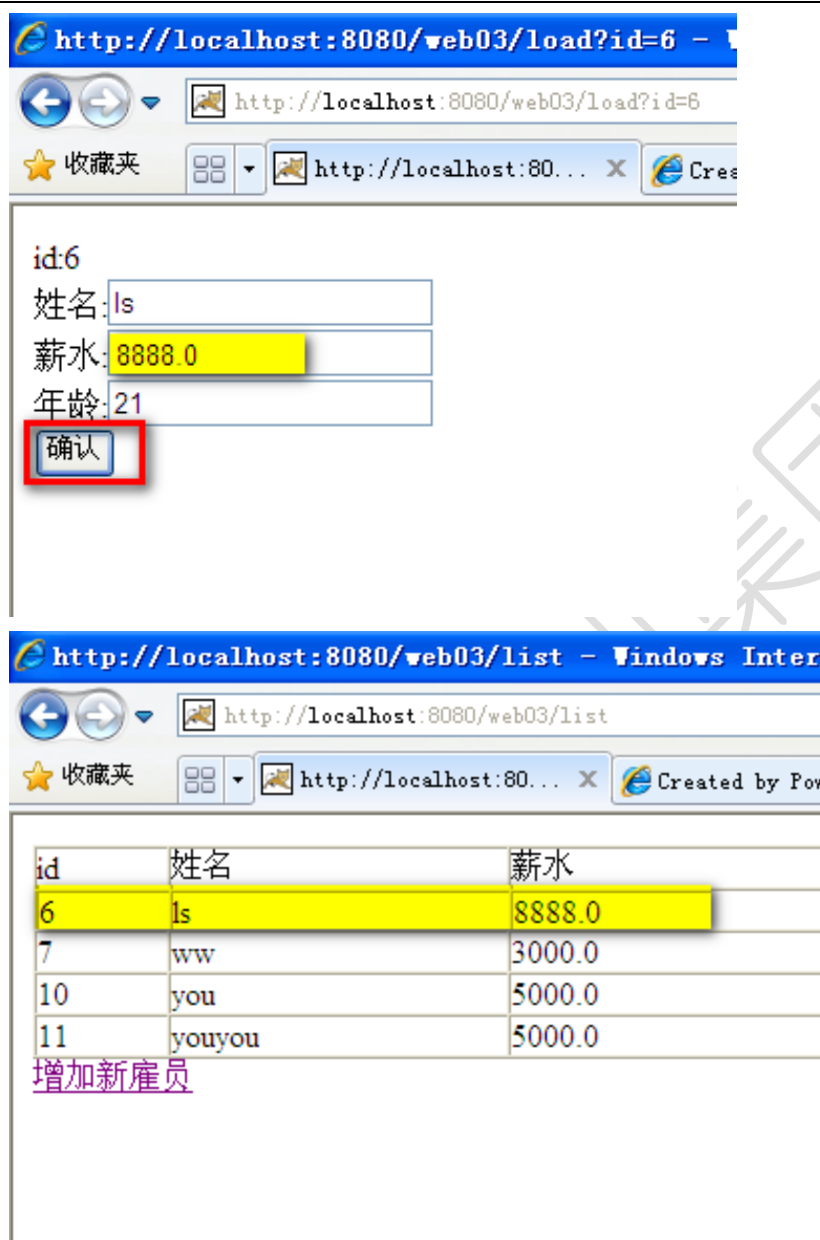
31 //访问数据库|
32 Connection conn = null;
33 try {
34     Class.forName(
35         "com.mysql.jdbc.Driver");
36     conn =
37         DriverManager.getConnection(
38             "jdbc:mysql://localhost:3306/jd110!
39             "root","root");
40     PreparedStatement prep =
41         conn.prepareStatement(
42             "update t_emp " +
43             "set name=?,salary=?,age=? " +
44             "where id=?");
45     prep.setString(1, name);
46     prep.setDouble(2, salary);
47     prep.setLong(3, age);
48     prep.setLong(4, id);
49     prep.executeUpdate();
50
51     response.sendRedirect("list");
52 } catch (Exception e) {
53     e.printStackTrace();
54     throw new ServletException(e);
55 } finally {
56     if (conn != null) {
57         try {
58             conn.close();
59         } catch (SQLException e) {
60             e.printStackTrace();
61         }
62     }
63 }
64 }
65 }

```

12) 修改 web.xml (略)

13) 测试

将 “ls” 的薪水由 2000 修改为 8888



OK, 演示结束。

【案例 2】雇员的增删改查

1) ListEmpServlet.java

```
package web;

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

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

public class ListEmpServlet extends HttpServlet{
    public void service(HttpServletRequest request,
                        HttpServletResponse response)
        throws ServletException,IOException{
        //访问数据库
        Connection conn = null;
        try {
            Class.forName(
                "com.mysql.jdbc.Driver");
            conn =
                DriverManager.getConnection(
                    "jdbc:mysql://localhost:3306/jd1109db2",
                    "root","root");
            Statement stat =
                conn.createStatement();
            ResultSet rst = stat.executeQuery(
                "select * from t_emp");

            //使用查询得到的结果，生成一个表格
            response.setContentType(
                "text/html;charset=utf-8");
            PrintWriter out =
```

```

        response.getWriter();
    out.println(
        "<table border='1' " +
        "width='60%' " +
        "cellpadding='0' " +
        "cellspacing='0'>");
    out.println("<tr> " +
        "<td>id</td> " +
        "<td>姓名</td> " +
        "<td>薪水</td> " +
        "<td>年龄</td> " +
        "<td>操作</td></tr>");
    while(rst.next()){
        long id =
            rst.getLong("id");
        String name =
            rst.getString("name");
        double salary =
            rst.getDouble("salary");
        int age =
            rst.getInt("age");
        out.println("<tr><td> "
            + id + "</td><td> "
            + name + "</td><td> "
            + salary + "</td><td> "
            + age + "</td><td> " +
            "<a href='del?id=" + id + "'> " +
            "删除</a> " +
            "&nbsp;" +
            "<a href='load?id=" + id + "'> " +
            "修改</a></td></tr>");
    }
    out.println("</table>");
    out.println(
        "<a href='addEmp.html'> " +
        "增加新雇员</a>");

    out.close();

```



```

        } catch (Exception e) {
            e.printStackTrace();
            throw new ServletException(e);
        } finally {
            if (conn != null) {
                try {
                    conn.close();
                } catch (SQLException e) {
                    e.printStackTrace();
                }
            }
        }
    }
}

```

2) DelEmpServlet.java

```

package web;

import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

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

public class DelEmpServlet
    extends HttpServlet {

    public void service(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException {
        long id =
            Long.parseLong(

```

```
        request.getParameter("id"));

//访问数据库
Connection conn = null;
try {
    Class.forName("com.mysql.jdbc.Driver");
    conn = DriverManager
        .getConnection(
            "jdbc:mysql://localhost:3306/jd1109db2",
            "root","root");
    PreparedStatement prep =
        conn.prepareStatement(
            "delete from t_emp where id = ?");
    prep.setLong(1, id);
    prep.executeUpdate();
    //重定向
    response.sendRedirect("list");

} catch (Exception e) {
    //step1 先记录日志
    e.printStackTrace();
    //step2 抛出
    throw new ServletException(e);

}finally{
    if(conn!=null){
        try {
            conn.close();
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
}
```

3) LoadEmpServlet.java

```
package web;
```

```
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

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

public class LoadEmpServlet extends HttpServlet{
    public void service(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException,IOException{
        long id = Long.parseLong(
            request.getParameter("id"));
        //访问数据库
        Connection conn = null;
        try {
            Class.forName(
                "com.mysql.jdbc.Driver");
            conn =
                DriverManager.getConnection(
                    "jdbc:mysql://localhost:3306/jd1109db2",
                    "root","root");
            PreparedStatement prep =
                conn.prepareStatement(
                    "select * from t_emp where id = ?");
            prep.setLong(1, id);
            ResultSet rst = prep.executeQuery();
            response.setContentType(
                "text/html;charset=utf-8");
            PrintWriter out = response.getWriter();
```

```
        if(rst.next()){
            String name =
                rst.getString("name");
            double salary =
                rst.getDouble("salary");
            int age =
                rst.getInt("age");
            out.println(
                "<form action='modify?id=" +
                id + "' method='post'>");
            out.println(
                "id:" + id + "<br/>");
            out.println(
                "姓名:<input name='name' value='" +
                name + "'/><br/>");
            out.println(
                "薪水:<input name='salary' value='" +
                salary + "'/><br/>");
            out.println(
                "年龄:<input name='age' value='" +
                age + "'/><br/>");
            out.println(
                "<input type='submit' " +
                "value='确认'/>");
            out.println(
                "</form>");
            out.close();
        }
    }catch(Exception e){
        e.printStackTrace();
        throw new ServletException(e);
    }finally{
        if(conn!=null){
            try {
                conn.close();
            } catch (SQLException e) {
                e.printStackTrace();
            }
        }
    }
```

```
}  
  
}  
  
}  
  
}
```

4) ModifyEmpServlet.java

```
package web;  
  
import java.io.IOException;  
import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.PreparedStatement;  
import java.sql.SQLException;  
  
import javax.servlet.ServletException;  
import javax.servlet.http.HttpServlet;  
import javax.servlet.http.HttpServletRequest;  
import javax.servlet.http.HttpServletResponse;  
  
public class ModifyEmpServlet extends HttpServlet{  
    public void service(HttpServletRequest request,  
        HttpServletResponse response)  
        throws ServletException,IOException{  
        request.setCharacterEncoding("utf-8");  
        long id =  
            Long.parseLong(  
                request.getParameter("id"));  
        String name =  
            request.getParameter("name");  
        double salary =  
            Double.parseDouble(  
                request.getParameter("salary"));  
        int age = Integer.parseInt(  
            request.getParameter("age"));  
  
        //访问数据库
```

```
Connection conn = null;
try {
    Class.forName(
        "com.mysql.jdbc.Driver");
    conn =
        DriverManager.getConnection(
            "jdbc:mysql://localhost:3306/jd1109db2",
            "root","root");
    PreparedStatement prep =
        conn.prepareStatement(
            "update t_emp " +
            "set name=?,salary=?,age=? " +
            "where id=?");
    prep.setString(1, name);
    prep.setDouble(2, salary);
    prep.setLong(3, age);
    prep.setLong(4, id);
    prep.executeUpdate();

    response.sendRedirect("list");
}catch(Exception e){
    e.printStackTrace();
    throw new ServletException(e);
}finally{
    if(conn!=null){
        try {
            conn.close();
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
}
```

5) AddEmpServlet.java

```
package web;
```

```
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

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

public class AddEmpServlet extends HttpServlet{

    public void service(HttpServletRequest request,
                        HttpServletResponse response)
        throws ServletException,IOException{

        //这行代码要放在 getParameter()执行之前。
        request.setCharacterEncoding("utf-8");

        String name = request.getParameter("name");
        double salary = Double.parseDouble(
            request.getParameter("salary"));
        int age = Integer.parseInt(
            request.getParameter("age"));
        System.out.println("name:" + name);
        System.out.println("salary:" + salary);
        System.out.println("age:" + age);

        //访问数据库
        Connection conn = null;
        try {
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager
                .getConnection(
                    "jdbc:mysql://localhost:3306/jd1109db2",
```

```
        "root","root");

        PreparedStatement prep =
            conn.prepareStatement(
                "insert into t_emp(name,salary,age) " +
                "values(?,?,?)");
        prep.setString(1, name);
        prep.setDouble(2, salary);
        prep.setInt(3, age);
        prep.executeUpdate();

        //1
        response.setContentType(
            "text/html;charset=utf-8");
        PrintWriter out = response.getWriter();
        out.println("添加雇员成功");
        out.println(
            "<a href='list'> " +
            "查看雇员</a> ");
        //out.close();

        //2. 重定向
        response.sendRedirect("list");

    } catch (Exception e) {
        //step1 先记录日志
        e.printStackTrace();
        //step2 抛出
        throw new ServletException(e);
    }

    finally{
        if(conn!=null){
            try {
                conn.close();
            } catch (SQLException e) {
                e.printStackTrace();
            }
        }
    }
}
```



```
}  
}
```

6) addEmp.html

```
<html>  
<head>  
<meta http-equiv="Content-Type"  
content="text/html; charset=utf-8">  
<title>addEmp</title>  
</head>  
<body style="font-size:30px;">  
  <form action="add" method="post">  
    <fieldset>  
      <legend>添加雇员</legend>  
      姓名:<input name="name"/><br/>  
      薪水:<input name="salary"/><br/>  
      年龄:<input name="age"/><br/>  
      <input type="submit" value="确认"/>  
    </fieldset>  
  </form>  
</body>  
</html>
```

7) error.html

```
<html>  
<head>  
<meta http-equiv="Content-Type"  
  content="text/html; charset=UTF-8">  
<title>Insert title here</title>  
</head>  
<body style="font-size:30px;color:red;">  
  发生了系统错误，请稍后  
  <a href="addEmp.html">重试</a>  
</body>  
</html>
```

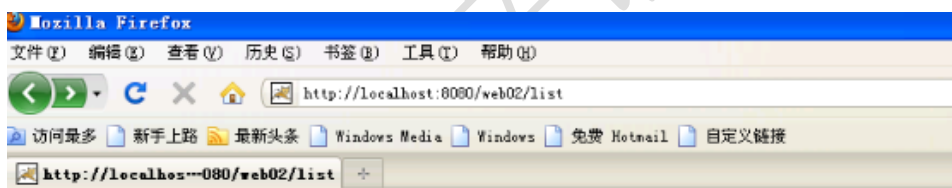
8) web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.4"
  xmlns="http://java.sun.com/xml/ns/j2ee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd">
  <servlet>
    <servlet-name>addEmpServlet</servlet-name>
    <servlet-class>web.AddEmpServlet</servlet-class>
  </servlet>
  <servlet>
    <servlet-name>listEmpServlet</servlet-name>
    <servlet-class>web.ListEmpServlet</servlet-class>
  </servlet>
  <servlet>
    <servlet-name>delEmpServlet</servlet-name>
    <servlet-class>web.DelEmpServlet</servlet-class>
  </servlet>
  <servlet>
    <servlet-name>loadEmpServlet</servlet-name>
    <servlet-class>web.LoadEmpServlet</servlet-class>
  </servlet>
  <servlet>
    <servlet-name>modifyEmpServlet</servlet-name>
    <servlet-class>web.ModifyEmpServlet</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>addEmpServlet</servlet-name>
    <url-pattern>/add</url-pattern>
  </servlet-mapping>
  <servlet-mapping>
    <servlet-name>listEmpServlet</servlet-name>
    <url-pattern>/list</url-pattern>
  </servlet-mapping>
  <servlet-mapping>
    <servlet-name>delEmpServlet</servlet-name>
    <url-pattern>/del</url-pattern>
```

```
</servlet-mapping>
<servlet-mapping>
    <servlet-name>loadEmpServlet</servlet-name>
    <url-pattern>/load</url-pattern>
</servlet-mapping>
<servlet-mapping>
    <servlet-name>modifyEmpServlet</servlet-name>
    <url-pattern>/modify</url-pattern>
</servlet-mapping>
<error-page>
    <error-code>500</error-code>
    <location>/error.html</location>
</error-page>
</web-app>
```

结果演示

访问 <http://localhost:8080/web03/list>



id	姓名	薪水	年龄	操作
1	lg	2000.0	22	删除 修改
2	lyg	2000.0	21	删除 修改
3	csb	2000.0	22	删除 修改
4	sdd	3000.0	22	删除 修改
5	ww	2100.0	22	删除 修改
6	abc	2222.0	22	删除 修改
7	abc2	2222.0	22	删除 修改
8	abc3	2222.0	22	删除 修改
9	abc4	3000.0	22	删除 修改

[增加新雇员](#)

4. DAO **

1) 什么是 DAO?

data access object (DAO) , 封装了**数据访问逻辑**, 调用者(一般是业务逻辑模块)不需要了解底层的数据访问细节就可以实现数据访问。这样一来, 当底层的数据访问细节发生改变, 不会影响到调用者。

2) 如何写一个 dao

a. 实体类

为了方便处理数据库中的记录, 可以定义与记录对应的实体类, 即将数据库中的某条记录转换成一个实体类的实例。

比如 Employee 类, 提供 id, name, salary, age 属性及相关的 get/set 方法, 我们可以将 t_emp 表中的记录转化成一个 Employee 对象。

b. dao 接口

声明一些数据访问方法, 这些方法不要涉及具体的技术。

比如:

EmployeeDAO 接口, 声明了:

```
public void save(Employee e) throws SQLException;
```

这种方式不好, 因为出现了 jdbc 的异常类, 这就要求实现该方法只能使用 jdbc。

换一种方式来声明, 更好

```
public void save(Employee e) throws Exception;
```

```
public List<Employee> findAll() throws Exception;
```

c. dao 实现类

采用具体的技术实现 dao 接口。

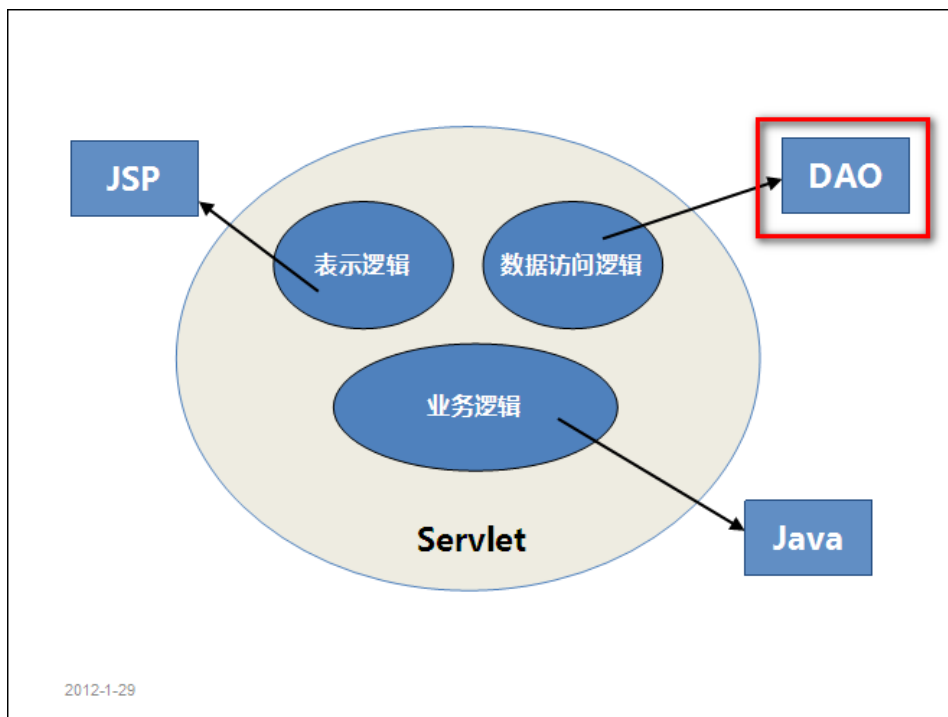
比如:

使用 jdbc 实现 EmployeeDAO 接口, 写一个类 EmployeeDAOJdbcImpl。

d. dao 工厂(工厂模式)

工厂类: 封装了对象的创建细节, 为调用者提供符合要求的对象。

4.1. 什么是 DAO *



- ✓ 数据访问逻辑：DAO (Data Access Object)
- ✓ 表示逻辑：和界面相关的，比如 out.print()

4.2. 如何写一个 DAO **

步骤 1

实体类 Employee.java

```
Employee.java X
1 package entity;
2 /**
3  * 实体类
4  * @author teacher
5  */
6 public class Employee {
7     private long id;
8     private String name;
9     private double salary;
10    private int age;
11    public int getAge() {return age;}
12    public void setAge(int age) {
13        this.age = age;}
14    public long getId() {return id;}
15    public void setId(long id) {this.id = id;}
16    public String getName() {return name;}
17    public void setName(String name) {
18        this.name = name;}
19    public double getSalary() {return salary;}
20    public void setSalary(double salary) {
21        this.salary = salary;}
22 }
23
```

步骤 2

dao 接口 EmployeeDAO.java

```
EmployeeDAO.java
1 package dao;
2 import java.util.List;
3 import entity.Employee;
4 /**
5  * DAO接口
6  * @author teacher
7  */
8 public interface EmployeeDAO {
9     /*
10      * public void save(Employee e)
11      * throws SQLException;
12      * 这种方式不好,因为出现了jdbc的异常类,
13      * 这就要求实现该方法只能使用jdbc */
14     public List<Employee> findAll()
15         throws Exception;
16     public void delete(long id)
17         throws Exception;
18     public void save(Employee e)
19         throws Exception;
20     public Employee findById(long id)
21         throws Exception;
22     public void update(Employee e)
23         throws Exception;
24 }
25
```

步骤 3

工具类 DBUtil.java

```

1 package util;
2
3 import java.sql.Connection;
4
5 /**
6  * jdbc工具类，提供获得连接、关闭连接的方法
7  * @author teacher
8  */
9
10 public class DBUtil {
11     public static Connection getConnection()
12         throws Exception{
13         Connection conn = null;
14         try {
15             Class.forName("com.mysql.jdbc.Driver");
16             conn = DriverManager.getConnection(
17                 "jdbc:mysql://localhost:3306/jd1109db2",
18                 "root","root");
19         } catch (Exception e) {
20             e.printStackTrace();
21             throw e;
22         }
23         return conn;
24     }
25
26     public static void close(Connection conn){
27         if(conn!=null){
28             try {
29                 conn.close();
30             } catch (SQLException e) {
31                 e.printStackTrace();
32             }
33         }
34     }
35
36     //用于测试DBUtil
37     public static void main(String[] args)
38         throws Exception {
39         System.out.println(getConnection());
40     }
41 }
42

```

步骤 4

测试 DBUtil


```

DBUtil.java
34 }
35
36 //用于测试DBUtil
37 public static void main(String[] args)
38     throws Exception {
39     System.out.println(getConnection());
40 }
41 }
42
Console
TCP/IP Monitor
<terminated> DBUtil [Java Application] C:\Program Files\Java\jdk1.6.0_06\bin\javaw.exe (Jan 29, 201
com.mysql.jdbc.Connection@7fdcde

```

步骤 5

dao 实现类

```

EmployeeDAOJdbcImpl.java
1 package dao.impl;
2
3 import java.sql.Connection;
13
14 public class EmployeeDAOJdbcImpl
15     implements EmployeeDAO{
16
17 public List<Employee> findAll()
18     throws Exception {
19     Connection conn = DBUtil.getConnection();
20     Statement stat =
21         conn.createStatement();
22     ResultSet rst = stat.executeQuery(
23         "select * from t_emp");
24     List<Employee> employees =
25         new ArrayList<Employee>();
26     while(rst.next()){
27         Employee e = new Employee();
28         e.setId(rst.getLong("id"));
29         e.setName(rst.getString("name"));
30         e.setSalary(rst.getDouble("salary"));
31         e.setAge(rst.getInt("age"));
32         employees.add(e);
33     }
34     DBUtil.close(conn);

```

```

35     return employees;
36 }
37
38 public void delete(long id)
39     throws Exception {
40     Connection conn = DBUtil.getConnection();
41     PreparedStatement prep =
42         conn.prepareStatement(
43         "delete from t_emp where id=?");
44     prep.setLong(1, id);
45     prep.executeUpdate();
46     DBUtil.close(conn);
47 }
48
49 public void save(Employee e)
50     throws Exception {
51     Connection conn = DBUtil.getConnection();
52     PreparedStatement prep =
53         conn.prepareStatement(
54         "insert into " +
55         "t_emp(name,salary,age) " +
56         "values(?,?,?)");
57     prep.setString(1, e.getName());
58     prep.setDouble(2, e.getSalary());

```

```

59     prep.setInt(3, e.getAge());
60     prep.executeUpdate();
61     DBUtil.close(conn);
62 }
63
64 public Employee findById(long id)
65     throws Exception {
66     Connection conn = DBUtil.getConnection();
67     PreparedStatement prep =
68         conn.prepareStatement(
69             "select * " +
70             "from t_emp where id=?");
71     prep.setLong(1, id);
72     ResultSet rst = prep.executeQuery();
73     Employee e = null;
74     if(rst.next()){
75         e = new Employee();
76         e.setId(id);
77         e.setName(rst.getString("name"));
78         e.setSalary(rst.getDouble("salary"));
79         e.setAge(rst.getInt("age"));
80     }
81     DBUtil.close(conn);
82     return e;
83 }
84
85 public void update(Employee e)
86     throws Exception {
87     Connection conn = DBUtil.getConnection();
88     PreparedStatement prep =
89         conn.prepareStatement(
90             "update t_emp " +
91             "set name=?, salary=?, age=? " +
92             "where id=?");
93     prep.setString(1, e.getName());
94     prep.setDouble(2, e.getSalary());
95     prep.setInt(3, e.getAge());
96     prep.setLong(4, e.getId());
97     prep.executeUpdate();
98     DBUtil.close(conn);
99 }
100 }

```

步骤 6

测试 dao 实现类

```

1 TestDAO.java X
7 public class TestDAO {
8     public static void main(String[] args)
9     throws Exception {
10         EmployeeDAO dao = new EmployeeDAOJdbcImpl();
11
12         /* 查找全部 */
13         //List<Employee> employees = dao.findAll();
14         //System.out.println(employees.size());
15
16         /* 删除 */
17         //dao.delete(1);
18
19         /* 插入数据 */
20         //Employee e = new Employee();
21         //e.setName("abc9");
22         //e.setSalary(1000);
23         //e.setAge(22);
24         //dao.save(e);
25
26         /* 按id查找 */
27         Employee e = dao.findById(7);
28         System.out.println(e.getName());
29     }
30 }
31

```

步骤 7.1

修改 ListEmpServlet.java

```
TestDAO.java ListEmpServlet.java AddEmpServlet.java DelEmpServlet.java
1 package web;
2
3 import java.io.IOException;
22
23 public class ListEmpServlet extends HttpServlet{
24     public void service(HttpServletRequest request,
25         HttpServletResponse response)
26         throws ServletException, IOException{
27         //使用dao访问数据库
28         try {
29             EmployeeDAO dao = new EmployeeDAOJdbcImpl(
30                 List<Employee> employees =
31                 dao.findAll();
32                 //使用查询得到的结果，生成一个表格
33                 response.setContentType(
34                     "text/html;charset=utf-8");
35                 PrintWriter out =
36                     response.getWriter();
37                 out.println(
38                     "<table border='1' width='60%' " +
39                     out.println(
40                     "<table border='1' width='60%' " +
41                     " cellpadding='0' cellspacing='0'>");
42                 out.println(
43                     "<tr><td>id</td>" +
44                     "<td>姓名</td>" +
45                     "<td>薪水</td>" +
46                     "<td>年龄</td>" +
47                     "<td>操作</td></tr>");
48                 for(int i=0;i<employees.size();i++){
49                     Employee e = employees.get(i);
50                     out.println("<tr><td>"
51                         + e.getId() + "</td><td>"
52                         + e.getName() + "</td><td>"
53                         + e.getSalary() + "</td><td>"
54                         + e.getAge() + "</td><td>" +
55                         "<a href='del?id="
56                         + e.getId() + "'>删除</a>" +
57                         "&nbsp;<a href='load?id="
58                         + e.getId() + "'>修改" +
59                         "</a></td></tr>");
60                 }
```

```

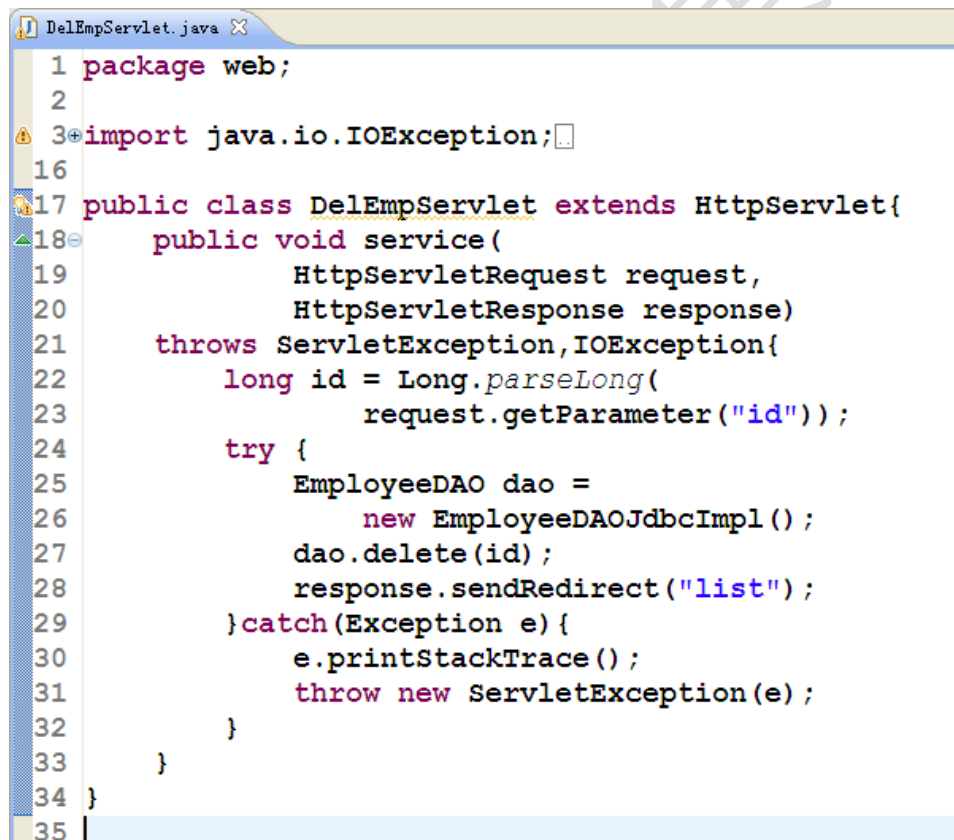
61         out.println("</table>");
62         out.println(
63             "<a href='addEmp.html'>" +
64             "增加新雇员</a>");
65         out.close();
66     } catch (Exception e) {
67         e.printStackTrace();
68         throw new ServletException(e);
69     }
70 }
71 }

```

测试（略）

步骤 7.2

修改 DelEmpServlet.java



```

DelEmpServlet.java
1 package web;
2
3 import java.io.IOException;
16
17 public class DelEmpServlet extends HttpServlet{
18     public void service(
19         HttpServletRequest request,
20         HttpServletResponse response)
21         throws ServletException, IOException{
22         long id = Long.parseLong(
23             request.getParameter("id"));
24         try {
25             EmployeeDAO dao =
26                 new EmployeeDAOJdbcImpl();
27             dao.delete(id);
28             response.sendRedirect("list");
29         } catch (Exception e) {
30             e.printStackTrace();
31             throw new ServletException(e);
32         }
33     }
34 }
35

```

测试

步骤 7.3

修改 AddEmpServlet.java

```

AddEmpServlet.java
1 package web;
2
3 import java.io.IOException;
18
19 public class AddEmpServlet extends HttpServlet{
20     public void service(
21         HttpServletRequest request,
22         HttpServletResponse response)
23         throws ServletException, IOException{
24         //这行代码要放在getParameter()执行之前。
25         request.setCharacterEncoding("utf-8");
26         String name = request.getParameter("name");
27         double salary = Double.parseDouble(
28             request.getParameter("salary"));
29         int age = Integer.parseInt(
30             request.getParameter("age"));
31         System.out.println("name:" + name);
32         System.out.println("salary:" + salary);
33         System.out.println("age:" + age);
34         //访问数据库
35         try {
36             EmployeeDAO dao =
37                 new EmployeeDAOJdbcImpl();
38             Employee e = new Employee();
39             e.setName(name);
40             e.setSalary(salary);
41             e.setAge(age);
42             dao.save(e);
43             response.sendRedirect("list");
44         } catch (Exception e) {
45             //step1 先记录日志
46             e.printStackTrace();
47             //step2 抛出
48             throw new ServletException(e);
49         }
50     }
51 }

```

测试

步骤 7.4

修改 LoadEmpServlet.java


```
LoadEmpServlet.java X
1 package web;
2
3 import java.io.IOException;
19
20 public class LoadEmpServlet extends HttpServlet{
21     public void service(
22         HttpServletRequest request,
23         HttpServletResponse response)
24         throws ServletException, IOException{
25         long id = Long.parseLong(
26             request.getParameter("id"));
27         try {
28             EmployeeDAO dao =
29                 new EmployeeDAOJdbcImpl();
30             Employee e = dao.findById(id);
31             response.setContentType(
32                 "text/html;charset=utf-8");
33             PrintWriter out = response.getWriter();
34             out.println(
35                 "<form action='modify?id="
36                 + id + "' method='post'>");
37             out.println(
38                 "id:" + id + "<br/>");
39             out.println(
40                 "姓名:<input name='name' value='"
41                 + e.getName() + "'/><br/>");
42             out.println(
43                 "薪水:<input name='salary' value='"
44                 + e.getSalary() + "'/><br/>");
45             out.println(
46                 "年龄:<input name='age' value='"
47                 + e.getAge() + "'/><br/>");
48             out.println(
49                 "<input type='submit' " +
50                 "value='确认' />");
51             out.println("</form>");
52             out.close();
53         } catch (Exception e) {
54             e.printStackTrace();
55             throw new ServletException(e);
56         }
57     }
58 }
```

测试

步骤 7.5

修改 ModifyEmpServlet.java

```

1 package web;
2
3 import java.io.IOException;
13
14 public class ModifyEmpServlet extends HttpServlet{
15     public void service(
16         HttpServletRequest request,
17         HttpServletResponse response)
18         throws ServletException, IOException{
19         request.setCharacterEncoding("utf-8");
20         long id = Long.parseLong(
21             request.getParameter("id"));
22         String name = request.getParameter("name");
23         double salary =
24             Double.parseDouble(
25                 request.getParameter("salary"));
26         int age = Integer.parseInt(
27             request.getParameter("age"));
28         try {
29             EmployeeDAO dao =
30                 new EmployeeDAOJdbcImpl();
31             Employee e = new Employee();
32             e.setId(id);
33             e.setName(name);
34             e.setSalary(salary);
35             e.setAge(age);
36             dao.update(e);
37             response.sendRedirect("list");
38         } catch (Exception e) {
39             e.printStackTrace();
40             throw new ServletException(e);
41         }
42     }
43 }

```

测试

完成。

4.3. dao 工厂(工厂模式) **

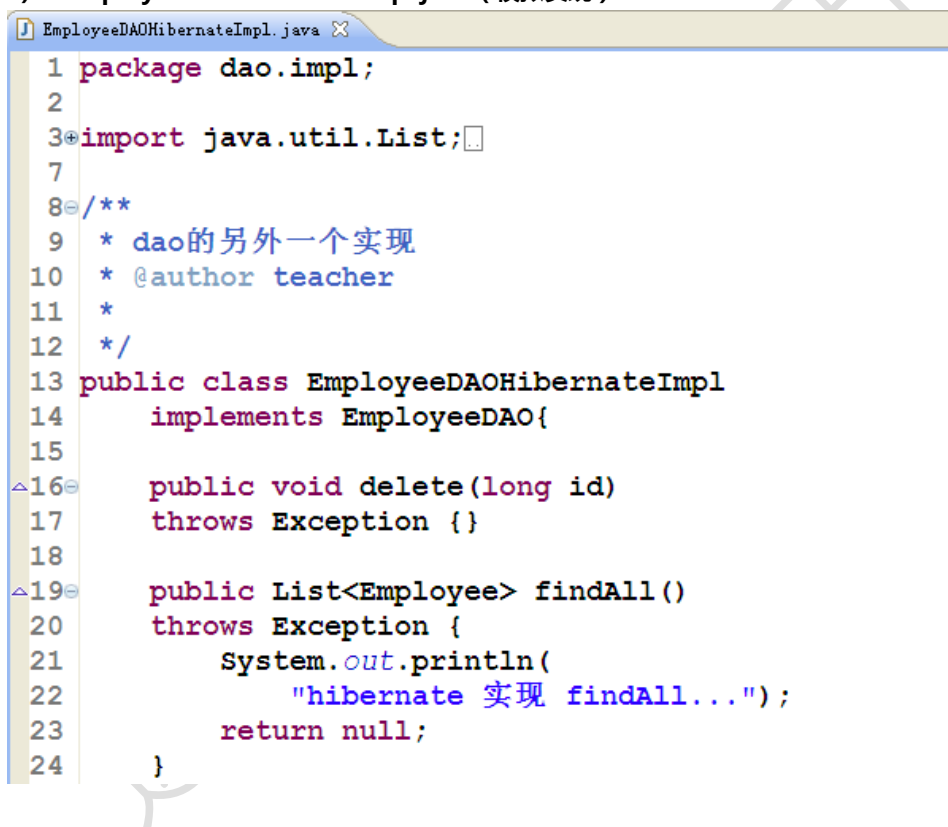
工厂类:封装了对象的创建细节, 为调用者提供符合要求的对象。

现在做这样一个假设, 假如我们的底层数据访问不再使用 JDBC, 而采用 Hibernate (框架技术, 后续会学习, 此处做了解)。

那么我们需要为 **EmployeeDAO** 增加一个实现 **EmployeeDAOHibernateImpl.java**

并且修改之前的代码, DelEmpServlet、AddEmpServlet、ListEmpServlet、LoadEmpServlet、ModifyEmpServlet 都要做相应修改, 如下所示:

1) EmployeeDAOHibernateImpl.java (模拟实现)



```

1 package dao.impl;
2
3 import java.util.List;
4
5
6
7
8 /**
9  * dao的另外一个实现
10  * @author teacher
11  *
12  */
13 public class EmployeeDAOHibernateImpl
14     implements EmployeeDAO{
15
16     public void delete(long id)
17         throws Exception {}
18
19     public List<Employee> findAll()
20         throws Exception {
21         System.out.println(
22             "hibernate 实现 findAll...");
23         return null;
24     }

```

```

25
26 public Employee findById(long id)
27     throws Exception {
28     return null;
29 }
30
31 public void save(Employee e)
32     throws Exception {}
33
34 public void update(Employee e)
35     throws Exception {}
36 }
37

```

2) DelEmpServlet.java

```

DelEmpServlet.java LoadEmpServlet.java ModifyEmpServlet.java DAOFactory.
3 import java.io.IOException;
17
18 public class DelEmpServlet extends HttpServlet{
19     public void service(
20         HttpServletRequest request,
21         HttpServletResponse response)
22         throws ServletException, IOException{
23         long id = Long.parseLong(
24             request.getParameter("id"));
25         try {
26             EmployeeDAO dao =
27             // new EmployeeDAOJdbcImpl();
28             new EmployeeDAOHibernateImpl();
29             dao.delete(id);
30             response.sendRedirect("list");
31         } catch (Exception e) {
32             e.printStackTrace();
33             throw new ServletException(e);
34         }
35     }
36 }
37

```

3) AddEmpServlet.java (略)

4) ListEmpServlet (略)

5) LoadEmpServlet (略)

6) ModifyEmpServlet (略)

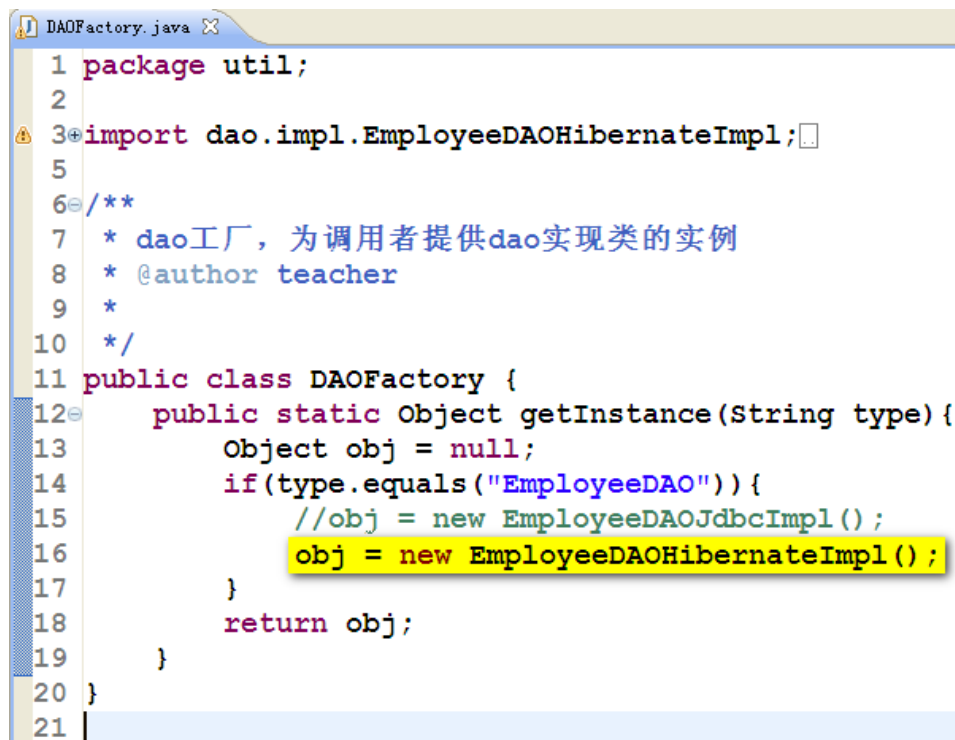
然而，我们希望**尽量不修改写好的代码（Servlet 代码）**

此时，我们可以使用**工厂模式**

演示：工厂模式

步骤 1

新建 DAOFactory.java，在 DAOFactory 中确定 EmployeeDAO 接口的具体实现类



```

1 package util;
2
3 import dao.impl.EmployeeDAOHibernateImpl;
4
5
6 /**
7  * dao工厂，为调用者提供dao实现类的实例
8  * @author teacher
9  *
10 */
11 public class DAOFactory {
12     public static Object getInstance(String type) {
13         Object obj = null;
14         if (type.equals("EmployeeDAO")) {
15             //obj = new EmployeeDAOJdbcImpl();
16             obj = new EmployeeDAOHibernateImpl();
17         }
18         return obj;
19     }
20 }
21

```

步骤 2

Servlet 中增删改查操作都使用 DAOFactory 生成具体实例

```

ListEmpServlet.java X
1 package web;
2
3 import java.io.IOException;
22
23 public class ListEmpServlet extends HttpServlet{
24     public void service(HttpServletRequest request,
25         HttpServletResponse response)
26     throws ServletException, IOException{
27         //使用dao访问数据库
28         try {
29             EmployeeDAO dao =
30                 (EmployeeDAO) DAOFactory
31                     .getInstance("EmployeeDAO");
32             List<Employee> employees =
33                 dao.findAll();
34             //使用查询得到的结果，生成一个表格
35             response.setContentType(
36                 "text/html;charset=utf-8");

```

OK

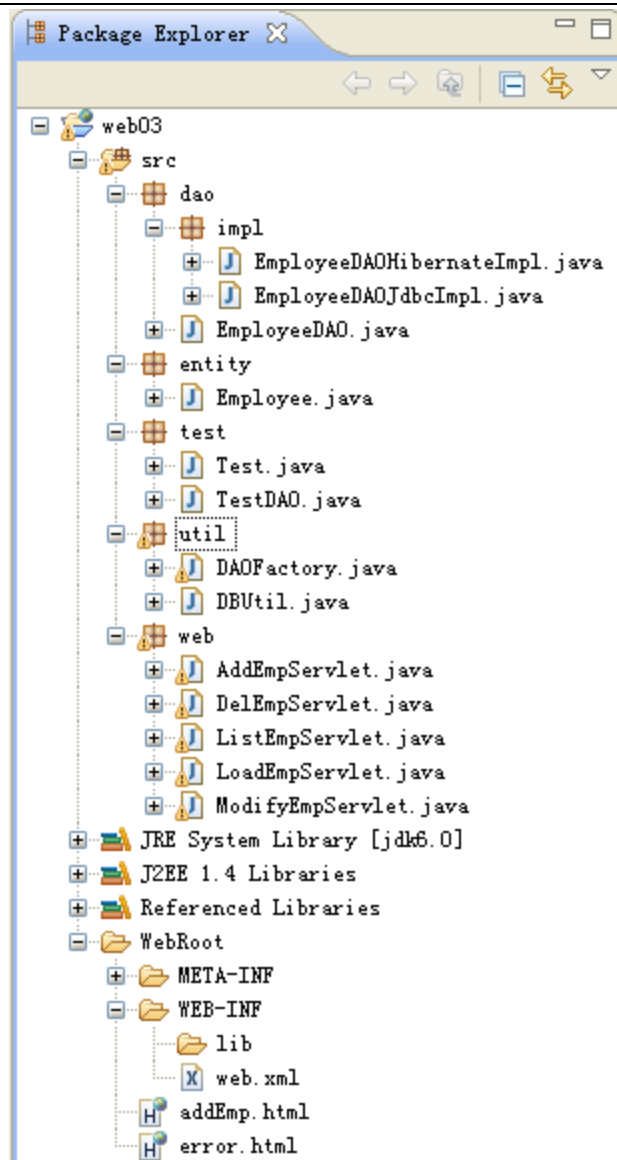
这样我们就不用修改每个 Servlet 了，查询数据库需要不同实现时，只要修改工厂类 DAOFactory 即可。

提问：

这样的方式已经使代码的可扩展性提高很多，但是我们还需要修改 DAOFactory，如果我们不想修改 DAOFactory 该如何实现？

【案例 3】DAO 及工厂模式

项目结构



参考代码

1) Employee.java

```
package entity;

/**
 * 实体类
 * @author teacher
 */
public class Employee {
    private long id;
```

```

private String name;
private double salary;
private int age;
public int getAge() {return age;}
public void setAge(int age) {
    this.age = age;}
public long getId() {return id;}
public void setId(long id) {this.id = id;}
public String getName() {return name;}
public void setName(String name) {
    this.name = name;}
public double getSalary() {return salary;}
public void setSalary(double salary) {
    this.salary = salary;}
}

```

2) EmployeeDAO.java

```

package dao;
import java.util.List;
import entity.Employee;
/**
 * DAO 接口
 * @author teacher
 */
public interface EmployeeDAO {
    /**
     * public void save(Employee e)
     * throws SQLException;
     * 这种方式不好,因为出现了 jdbc 的异常类,
     * 这就要求实现该方法只能使用 jdbc */
    public List<Employee> findAll()
        throws Exception;
    public void delete(long id)
        throws Exception;
    public void save(Employee e)
        throws Exception;
    public Employee findById(long id)

```

```
        throws Exception;  
        public void update(Employee e)  
            throws Exception;  
    }
```

3) EmployeeDAOJdbcImpl.java

```
package dao.impl;  
  
import java.sql.Connection;  
import java.sql.PreparedStatement;  
import java.sql.ResultSet;  
import java.sql.Statement;  
import java.util.ArrayList;  
import java.util.List;  
  
import util.DBUtil;  
import dao.EmployeeDAO;  
import entity.Employee;  
  
public class EmployeeDAOJdbcImpl  
    implements EmployeeDAO{  
  
    public List<Employee> findAll()  
        throws Exception {  
        Connection conn = DBUtil.getConnection();  
        Statement stat =  
            conn.createStatement();  
        ResultSet rst = stat.executeQuery(  
            "select * from t_emp");  
        List<Employee> employees =  
            new ArrayList<Employee> ();  
        while(rst.next()){  
            Employee e = new Employee();  
            e.setId(rst.getLong("id"));  
            e.setName(rst.getString("name"));  
            e.setSalary(rst.getDouble("salary"));  
            e.setAge(rst.getInt("age"));  
        }  
    }  
}
```



```
        employees.add(e);
    }
    DBUtil.close(conn);
    return employees;
}

public void delete(long id)
    throws Exception {
    Connection conn = DBUtil.getConnection();
    PreparedStatement prep =
        conn.prepareStatement(
            "delete from t_emp where id=?");
    prep.setLong(1, id);
    prep.executeUpdate();
    DBUtil.close(conn);
}

public void save(Employee e)
    throws Exception {
    Connection conn = DBUtil.getConnection();
    PreparedStatement prep =
        conn.prepareStatement(
            "insert into " +
            "t_emp(name,salary,age) " +
            "values(?,?,?)");
    prep.setString(1, e.getName());
    prep.setDouble(2, e.getSalary());
    prep.setInt(3, e.getAge());
    prep.executeUpdate();
    DBUtil.close(conn);
}

public Employee findById(long id)
    throws Exception {
    Connection conn = DBUtil.getConnection();
    PreparedStatement prep =
        conn.prepareStatement(
            "select * " +
```

```

        "from t_emp where id=?");

    prep.setLong(1, id);
    ResultSet rst = prep.executeQuery();
    Employee e = null;
    if(rst.next()){
        e = new Employee();
        e.setId(id);
        e.setName(rst.getString("name"));
        e.setSalary(rst.getDouble("salary"));
        e.setAge(rst.getInt("age"));
    }
    DBUtil.close(conn);
    return e;
}

public void update(Employee e)
    throws Exception {
    Connection conn = DBUtil.getConnection();
    PreparedStatement prep =
        conn.prepareStatement(
            "update t_emp " +
            "set name=?,salary=?,age=? " +
            "where id=?");

    prep.setString(1, e.getName());
    prep.setDouble(2, e.getSalary());
    prep.setInt(3, e.getAge());
    prep.setLong(4, e.getId());
    prep.executeUpdate();
    DBUtil.close(conn);
}
}

```

4) EmployeeDAOHibernateImpl.java

```

package dao.impl;

import java.util.List;

```

```
import dao.EmployeeDAO;
import entity.Employee;

/**
 * dao 的另外一个实现
 * @author teacher
 *
 */
public class EmployeeDAOHibernateImpl
    implements EmployeeDAO{

    public void delete(long id)
        throws Exception {}

    public List<Employee> findAll()
        throws Exception {
        System.out.println(
            "hibernate 实现 findAll...");
        return null;
    }

    public Employee findById(long id)
        throws Exception {
        return null;
    }

    public void save(Employee e)
        throws Exception {}

    public void update(Employee e)
        throws Exception {}
}
```

5) TestDAO.java

```
package test;

import java.util.List;
```

```
import dao.EmployeeDAO;
import dao.impl.EmployeeDAOJdbcImpl;
import entity.Employee;

public class TestDAO {
    public static void main(String[] args)
        throws Exception {
        EmployeeDAO dao = new EmployeeDAOJdbcImpl();

        /* 查找全部 */
        List<Employee> employees = dao.findAll();
        System.out.println(employees.size());

        /* 删除 */
        //dao.delete(1);

        /* 插入数据 */
        //Employee e = new Employee();
        //e.setName("abc9");
        //e.setSalary(1000);
        //e.setAge(22);
        //dao.save(e);

        /* 按 id 查找 */
        //Employee e = dao.findById(7);
        //System.out.println(e.getName());
    }
}
```

6) DAOFactory.java

```
package util;

import dao.impl.EmployeeDAOHibernateImpl;
import dao.impl.EmployeeDAOJdbcImpl;

/**
```

```

* dao 工厂，为调用者提供 dao 实现类的实例
* @author teacher
*
*/
public class DAOFactory {
    public static Object getInstance(String type){
        Object obj = null;
        if(type.equals("EmployeeDAO")){
            //obj = new EmployeeDAOJdbcImpl();
            obj = new EmployeeDAOHibernateImpl();
        }
        return obj;
    }
}

```

7) DBUtil.java

```

package util;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
/**
 * jdbc 工具类，提供获得连接、关闭连接的方法
 * @author teacher
 */
public class DBUtil {
    public static Connection getConnection()
        throws Exception{
        Connection conn = null;
        try {
            Class.forName("com.mysql.jdbc.Driver");
            conn = DriverManager.getConnection(
                "jdbc:mysql://localhost:3306/jd1109db2",
                "root","root");
        } catch (Exception e) {
            e.printStackTrace();
            throw e;
        }
    }
}

```

```
    }  
    return conn;  
}  
  
public static void close(Connection conn){  
    if(conn!=null){  
        try {  
            conn.close();  
        } catch (SQLException e) {  
            e.printStackTrace();  
        }  
    }  
}  
  
//用于测试 DBUtil  
public static void main(String[] args)  
    throws Exception {  
    System.out.println(getConnection());  
}  
}
```

8) AddEmpServlet.java

```
package web;  
  
import java.io.IOException;  
import java.io.PrintWriter;  
import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.PreparedStatement;  
import java.sql.SQLException;  
  
import javax.servlet.ServletException;  
import javax.servlet.http.HttpServlet;  
import javax.servlet.http.HttpServletRequest;  
import javax.servlet.http.HttpServletResponse;  
  
import dao.EmployeeDAO;
```

```
import dao.impl.EmployeeDAOJdbcImpl;
import entity.Employee;

public class AddEmpServlet extends HttpServlet{
    public void service(
        HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException,IOException{
        //这行代码要放在 getParameter()执行之前。
        request.setCharacterEncoding("utf-8");
        String name = request.getParameter("name");
        double salary = Double.parseDouble(
            request.getParameter("salary"));
        int age = Integer.parseInt(
            request.getParameter("age"));
        System.out.println("name:" + name);
        System.out.println("salary:" + salary);
        System.out.println("age:" + age);
        //访问数据库
        try {
            EmployeeDAO dao =
                new EmployeeDAOJdbcImpl();
            Employee e = new Employee();
            e.setName(name);
            e.setSalary(salary);
            e.setAge(age);
            dao.save(e);
            response.sendRedirect("list");
        } catch (Exception e) {
            //step1 先记录日志
            e.printStackTrace();
            //step2 抛出
            throw new ServletException(e);
        }
    }
}
```

9) DelEmpServlet.java

```
package web;

import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

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

import dao.EmployeeDAO;
import dao.impl.EmployeeDAOHibernateImpl;
import dao.impl.EmployeeDAOJdbcImpl;

public class DelEmpServlet extends HttpServlet{
    public void service(
        HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException,IOException{
        long id = Long.parseLong(
            request.getParameter("id"));
        try {
            EmployeeDAO dao =
//            new EmployeeDAOJdbcImpl();
            new EmployeeDAOHibernateImpl();
            dao.delete(id);
            response.sendRedirect("list");
        }catch(Exception e){
            e.printStackTrace();
            throw new ServletException(e);
        }
    }
}
```


10) ListEmpServlet.java

```
package web;

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.List;

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

import util.DAOFactory;

import dao.EmployeeDAO;
import dao.impl.EmployeeDAOJdbcImpl;
import entity.Employee;

public class ListEmpServlet extends HttpServlet{
    public void service(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException,IOException{
        //使用 dao 访问数据库
        try {
            EmployeeDAO dao =
                (EmployeeDAO) DAOFactory
                    .getInstance("EmployeeDAO");
            List<Employee> employees =
                dao.findAll();
            //使用查询得到的结果，生成一个表格
            response.setContentType(
                "text/html;charset=utf-8");
```

```

        PrintWriter out =
            response.getWriter();
        out.println(
            "<table border='1' width='60%' " +
            "cellpadding='0' cellspacing='0'>");
        out.println(
            "<tr><td>id</td>" +
            "<td>姓名</td>" +
            "<td>薪水</td>" +
            "<td>年龄</td>" +
            "<td>操作</td></tr>");
        for(int i=0;i<employees.size();i++){
            Employee e = employees.get(i);
            out.println("<tr><td>"
                + e.getId() + "</td><td>"
                + e.getName() + "</td><td>"
                + e.getSalary() + "</td><td>"
                + e.getAge() + "</td><td>" +
                "<a href='del?id="
                    + e.getId() + "'>删除</a>" +
                "&nbsp;<a href='load?id="
                    + e.getId() + "'>修改" +
                "</a></td></tr>");
        }
        out.println("</table>");
        out.println(
            "<a href='addEmp.html'>" +
            "增加新雇员</a>");
        out.close();
    } catch (Exception e) {
        e.printStackTrace();
        throw new ServletException(e);
    }
}
}

```

11) LoadEmpServlet.java

```
package web;

import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

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

import dao.EmployeeDAO;
import dao.impl.EmployeeDAOJdbcImpl;
import entity.Employee;

public class LoadEmpServlet extends HttpServlet{
    public void service(
        HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException,IOException{
        long id = Long.parseLong(
            request.getParameter("id"));
        try {
            EmployeeDAO dao =
                new EmployeeDAOJdbcImpl();
            Employee e = dao.findById(id);
            response.setContentType(
                "text/html;charset=utf-8");
            PrintWriter out = response.getWriter();
            out.println(
                "<form action='modify?id="
                + id + "' method='post'>");
            out.println(
                "id:" + id + "<br/>");
```

```

        out.println(
            "姓名:<input name='name' value='"
            + e.getName() + "'/><br/>");
        out.println(
            "薪水:<input name='salary' value='"
            + e.getSalary() + "'/><br/>");
        out.println(
            "年龄:<input name='age' value='"
            + e.getAge() + "'/><br/>");
        out.println(
            "<input type='submit' " +
            "value='确认'/>");
        out.println("</form>");
        out.close();
    }catch(Exception e){
        e.printStackTrace();
        throw new ServletException(e);
    }
}
}

```

12) ModifyEmpServlet.java

```

package web;

import java.io.IOException;

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

import dao.EmployeeDAO;
import dao.impl.EmployeeDAOJdbcImpl;
import entity.Employee;

public class ModifyEmpServlet extends HttpServlet{
    public void service(

```

```

        HttpServletRequest request,
        HttpServletResponse response)
throws ServletException,IOException{
    request.setCharacterEncoding("utf-8");
    long id = Long.parseLong(
        request.getParameter("id"));
    String name = request.getParameter("name");
    double salary =
        Double.parseDouble(
            request.getParameter("salary"));
    int age = Integer.parseInt(
        request.getParameter("age"));
    try {
        EmployeeDAO dao =
            new EmployeeDAOJdbcImpl();
        Employee e = new Employee();
        e.setId(id);
        e.setName(name);
        e.setSalary(salary);
        e.setAge(age);
        dao.update(e);
        response.sendRedirect("list");
    }catch(Exception e){
        e.printStackTrace();
        throw new ServletException(e);
    }
}
}

```

13) web.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.4"
    xmlns="http://java.sun.com/xml/ns/j2ee"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
    http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd">
    <servlet>

```

```
<servlet-name>addEmpServlet</servlet-name>
<servlet-class>web.AddEmpServlet</servlet-class>
</servlet>
<servlet>
  <servlet-name>listEmpServlet</servlet-name>
  <servlet-class>web.ListEmpServlet</servlet-class>
</servlet>
<servlet>
  <servlet-name>delEmpServlet</servlet-name>
  <servlet-class>web.DelEmpServlet</servlet-class>
</servlet>
<servlet>
  <servlet-name>loadEmpServlet</servlet-name>
  <servlet-class>web.LoadEmpServlet</servlet-class>
</servlet>
<servlet>
  <servlet-name>modifyEmpServlet</servlet-name>
  <servlet-class>web.ModifyEmpServlet</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>addEmpServlet</servlet-name>
  <url-pattern>/add</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>listEmpServlet</servlet-name>
  <url-pattern>/list</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>delEmpServlet</servlet-name>
  <url-pattern>/del</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>loadEmpServlet</servlet-name>
  <url-pattern>/load</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>modifyEmpServlet</servlet-name>
  <url-pattern>/modify</url-pattern>
```

```
</servlet-mapping>
<error-page>
    <error-code>500</error-code>
    <location>/error.html</location>
</error-page>
</web-app>
```

14) addEmp.html

```
<html>
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=utf-8">
<title>addEmp</title>
</head>
<body style="font-size:30px;">
    <form action="add" method="post">
        <fieldset>
            <legend>添加雇员</legend>
            姓名:<input name="name"/><br/>
            薪水:<input name="salary"/><br/>
            年龄:<input name="age"/><br/>
            <input type="submit" value="确认"/>
        </fieldset>
    </form>
</body>
</html>
```

15) error.html

```
<html>
<head>
<meta http-equiv="Content-Type"
    content="text/html; charset=UTF-8">
<title>Insert title here</title>
</head>
<body style="font-size:30px;color:red;">
    发生了系统错误，请稍后
```

```
<a href="addEmp.html">重试</a>  
</body>  
</html>
```

达内IT培训集团