

第 6 章 MVC 设计模式

6.1 引入 MVC 设计模式原因

1. 回顾之前的 JSP+JavaBean

简单计算器代码(见 ComputerBean.java 文件), 业务逻辑和数据存储功能不清晰。

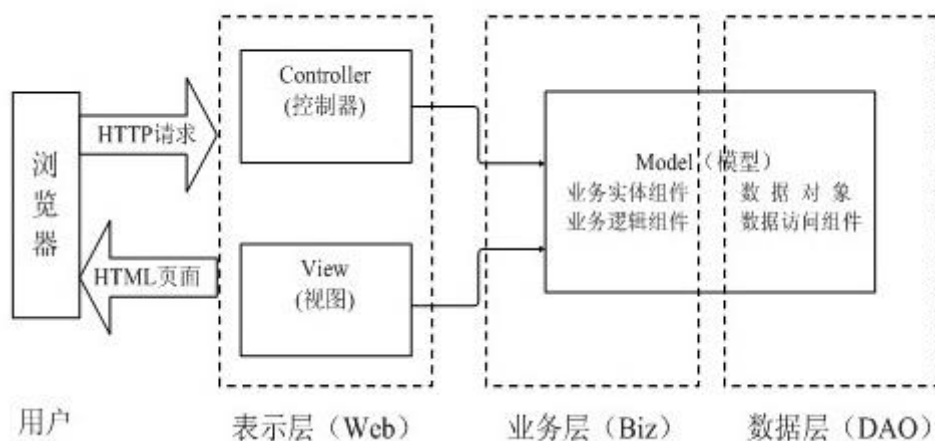
问题代码如下:

```
public double getResult() {  
    if (op.equals("+")) {  
        result = one + two;  
    } else if (op.equals("-")) {  
        result = one - two;  
    } else if (op.equals("*")) {  
        result = one * two;  
    } else if (op.equals("/")) {  
        result = one / two;  
    }  
    return result;  
}
```

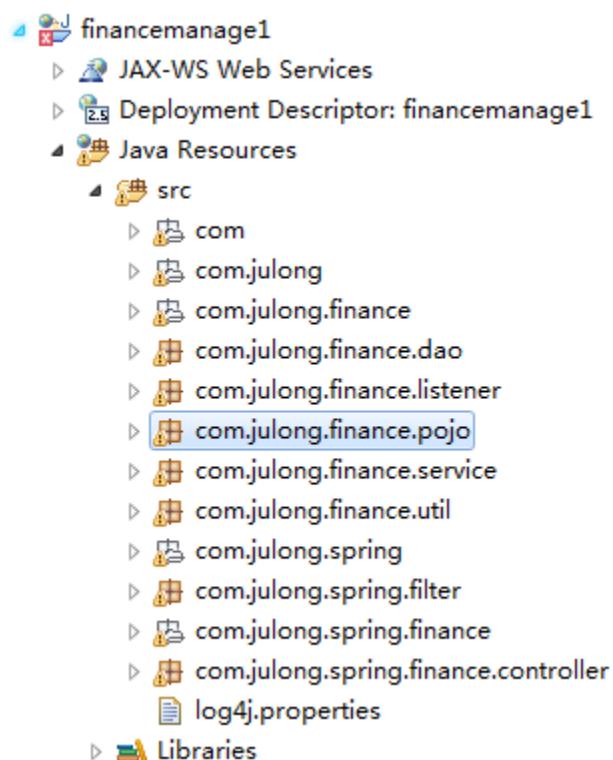
本方法中的业务逻辑和数据存储不清晰(分工不明确)。

2. 软件开发的分层思想:

表示层 + 模型层 + 业务逻辑(程序流程控制)层



3.演示聚龙项目 Spring MVC 分层



6.2 MVC 设计模式及其优点

1.MVC 全称

Model（模型）+View（视图）+Controller（控制器）

2. MVC 优点

松耦合，便于系统维护和扩展

6.3 MVC 组件表现形式及作用

组件名称	作用	表现形式
模型（Model）	定义并存取数据	JavaBean
视图（View）	提交及显示数据	JSP 页面
控制器（Controller）	控制程序流程（业务逻辑）	Servlet

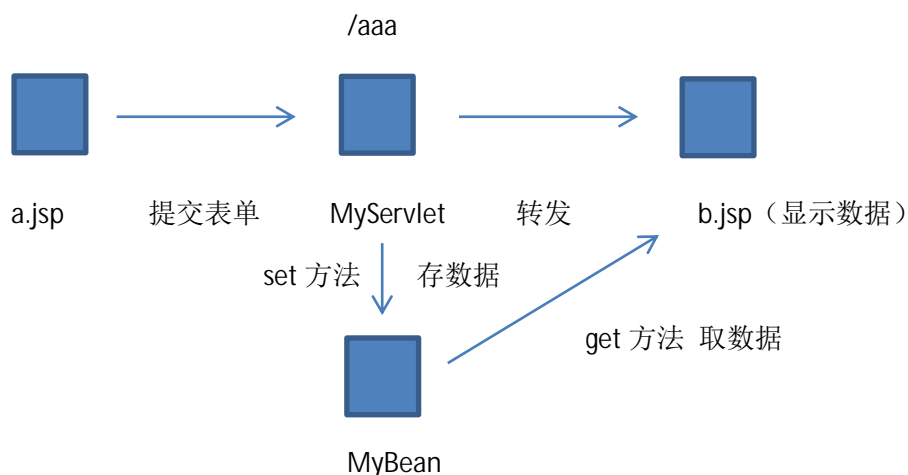
=====

【复习】

【板书】

6.4 实例 – 简单 MVC 设计模式流程

1. 流程示意图（各组件内容及其之间的联系（通信））



2. 各组件内代码流程

(1) login.jsp 页面：定义表单

```
<form action="aaa">【学生】
```

1) 文本框：name="var" 2) 提交按钮【学生】

```
</form>
```

(2) com.yip.bean.MyBean (JavaBean)：定义存取数据方法

```
String var = " ";
```

getVar()方法

setVar()方法

(3) com.yip.servlet.MyServlet (虚拟路径"/aaa")：控制流程

获取表单数据：String var = request.getParameter("var"); 【学生】

处理数据：var = "Welcome," + var; 【学生】

创建 JavaBean 对象：MyBean myBean = new MyBean(); 【学生】

存储数据：myBean.setVar(var); 【学生】

存储 JavaBean 对象至 request 键值表中：

request.setAttribute("myBean", myBean); 【学生】

转发数据至 show.jsp 页面

(4) show.jsp 页面：

定义 Bean: <jsp:useBean id=" myBean" scope=" request" >【学生】

显示数据: <jsp:getProperty property=" var" name=" myBean" >【学生】

6.5 调试程序

1. JSP+JavaBean+Servlet (MVC) 简单工作流程示例

学生敲代码, 给出平时成绩

2. MVC 结构——四则运算示例

(1) a.jsp: 提交数据和显示数据

```
<% @ page language="java" pageEncoding="gb2312"%>
<jsp:useBean id="myBean" class="com.jp.bean.MyBean" scope="session"/>
<html>
<body>
    <form action="aaa">
        <input type="text" name="numberOne" value="0.0" />
        <select name="operator">
            <option value="+">+
            <option value="-">-
            <option value="*">*
            <option value="/">/
        </select>
        <input type="text" name="numberTwo" value="0.0" />
        <input type="submit" value="提交" />
    </form>
    <jsp:getProperty property="numberOne" name="myBean"/>
    <jsp:getProperty property="operator" name="myBean"/>
    <jsp:getProperty property="numberTwo" name="myBean"/> =
    <jsp:getProperty property="result" name="myBean"/>
</body></html>
```

(2) MyBean.java: 模型-存储数据

```
package com.yip.bean;

public class MyBean {

    double numberOne, numberTwo, result;

    String operator = "+";

    public double getNumberOne() {

        return numberOne;

    }

    public void setNumberOne(double numberOne) {

        this.numberOne = numberOne;

    }

    public double getNumberTwo() {

        return numberTwo;

    }

    public void setNumberTwo(double numberTwo) {

        this.numberTwo = numberTwo;

    }

    public double getResult() {

        return result;

    }

    public void setResult(double result) {

        this.result = result;

    }

    public String getOperator() {

        return operator;

    }

    public void setOperator(String operator) {

        this.operator = operator;

    }

}
```

```
}
```

(3) MyServlet.java: 四则运算处理数据及程序流程

```
protected void doGet(HttpServletRequest request,
    HttpServletResponse response) throws ServletException, IOException {
    String numberOne = request.getParameter("numberOne");
    String numberTwo = request.getParameter("numberTwo");
    String operator = request.getParameter("operator");
    double one = Double.parseDouble(numberOne);
    double two = Double.parseDouble(numberTwo);
    double result = 0;
    if (operator.equals("+")) {
        result = one + two;
    } else if (operator.equals("-")) {
        result = one - two;
    } else if (operator.equals("*")) {
        result = one * two;
    } else if (operator.equals("/")) {
        result = one / two;
    }
    MyBean mybean = new MyBean();
    mybean.setNumberOne(one);
    mybean.setNumberTwo(two);
    mybean.setOperator(operator);
    mybean.setResult(result);
    HttpSession session = request.getSession(true);
    session.setAttribute("myBean", mybean);
    RequestDispatcher dis = request.getRequestDispatcher("a.jsp");
    dis.forward(request, response);
}
```

