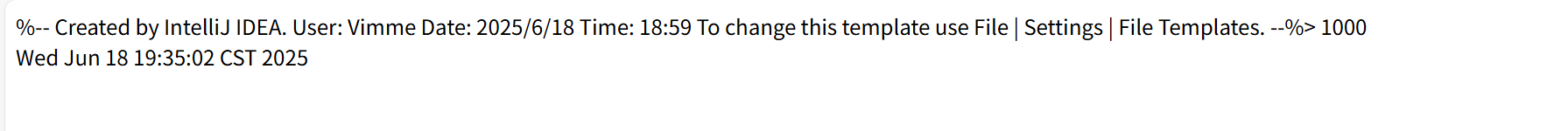
<%@ **page** import="java.util.Date" %>  
%--  
 Created by IntelliJ IDEA.  
 User: Vimme  
 Date: 2025/6/18  
 Time: 18:59  
 To change this template use File | Settings | File Templates.  
--%>  
<%@ **page** contentType="text/html;charset=UTF-8" language="java" %>  
<html>  
 <head>  
 <title>Insert title here</title>  
 </head>  
 <body>  
 <%  
 int myNumber =1000;  
 Date myDate =new Date();  
 out.print(myNumber);  
 out.print("<br>");  
 out.print(myDate);  
 %>  
 </body>  
</html>



JSP页面显示数字和日期代码分析与实验心得

这段代码是一个简单的JSP页面，展示了如何在JSP中使用Java代码输出数字和日期。下面我将分析其核心内容和实验心得。

核心内容分析

1.JSP指令：

<%@ page import="java.util.Date" %> - 导入Java的Date类

<%@ page contentType="text/html;charset=UTF-8" language="java" %> - 设置页面内容类型和编码

2.HTML结构：

标准的HTML文档结构

标题设置为"Insert title here"（可改进为更有意义的标题）

3.Java代码片段：

声明并初始化一个整型变量myNumber = 1000

创建Date对象myDate，获取当前系统时间

使用out.print()输出：

数字变量myNumber的值

HTML换行标签<br>

当前日期时间对象myDate

实际运行效果

页面将显示：

1000

Wed Jun 18 18:59:00 CST 2025

(实际日期时间会根据服务器当前时间显示)

实验心得

1.技术学习：

掌握了JSP中导入Java类的方法

学习了如何在JSP中声明和使用Java变量

理解了Date类的基本用法

熟悉了out.print()方法输出多种类型数据

2.优点：

代码结构清晰，简单易懂

演示了JSP中Java代码与HTML的结合

展示了基本的数据类型输出

3.改进建议：

可以添加格式化日期输出，使日期显示更友好：

jsp

<%@ page import="java.text.SimpleDateFormat" %>

<%

SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");

out.print(sdf.format(myDate));

%>

为页面设置更有意义的标题

可以添加CSS样式美化输出

考虑将业务逻辑与表现分离（虽然对简单示例不是必须）

4.学习价值：

展示了JSP页面处理Java对象的基本能力

演示了服务器端动态内容生成

帮助理解JSP作为服务器端技术的优势

为进一步学习更复杂的JSP功能打下基础

这个简单的JSP页面很好地展示了如何在Web页面中嵌入动态内容，特别是服务器端数据的处理和显示。通过这个例子，可以理解JSP作为Java Web开发技术的基本工作模式。