

# 高级 Web 技术实验二：基于 SSM 的校级课程管理程序

## 1, 任务目的

掌握SpringMVC的数据绑定和响应，学会使用 JSP、Velocity等页面视图进行交互和结果的页面展示。

## 2, 实验要求

1, 技术选型：Java + Spring + MyBatis + SpringMVC； 2, 实验交互及结果要以Web视图形式进行展示（推荐）； 3, 提交实验报告，项目代码推荐提交至Github并在实验报告中给出项目的仓库地址。

## 3, 实验内容

该实验继续沿用高级Web 技术实验一中的题目背景，结合你设计的数据模型和表设计，在此基础上，结合SpringMVC 框架完成：

1. 新增课程：支持通过表单提交一门新的课程，该表单包含的元素：课程名（文本框），课时（文本框），开课学院（下拉框，options参考实验一设定）。要求：课程名不能与已有的课程重复。
2. 展示课程：显示所有学院开设的所有课程，按照学院类别进行排序。要求：暂时不要求分页展示所有的课程。
3. 修改课程：在课程列表展示中可以选择任一门课程进行修改。要求：新修改课程名不能与已有的课程名重复。
4. 删除课程：选择一门课程进行删除，删除前，弹出对话框，确认是否删除。暂时不考虑批量删除。

页面设计及交互参考下图，可以自己设计css和js脚本，js使用jquery即可。学有余力的同学，可以尝试使用VUE、React等前端框架结合流行的UI组件等来完成展示。

### 页面原型设计及交互参考



新增课程

课程1	课时数1	学院1	编辑	删除
课程2	课时数2	学院1	编辑	删除

1, 点击新增课程按钮，弹出右侧表单对话框，在该对话框中填写新的课程信息，点击 ok 新增课程返回并刷新上述课程列表页面。

2, 列表展示所有的课程，右侧是对应于该列表项的数据操作（编辑/删除）。为了简单，左侧只展示了两门课程。  
编辑：编辑该数据项，对应为更新操作。  
删除：删除该数据项，对应为删除操作。

#### 新增课程表单

Input / Text

Input / Text

Input / Select

Buttons / Primary

Buttons / Secondary

## 4, 实验指导

1. 页面展示时候, 如果选用的是JSP视图, 可以选用 JSP JSTL:

```
<c:forEach>
```

标签对所有的课程进行迭代输出。如果选用的是Velocity视图, 可以选用

```
#foreach($class in $classes)
```

来迭代。记住后端要返回课程的id号; 否则在后续的删除和更新操作的时候没有这个id号后端无法确认用户是针对哪一门课程进行操作。课程列表展示不要求展示课程id号, 因此可以利用css在页面上把id信息隐藏。

2. 新增课程的时候要判断课程名是否和已有的课程名重复, 可以在提交请求后在后端进行判断; 如果对Ajax技术有兴趣的同学, 可以选择在前端用户输入完课程名后发送Ajax请求来判断是否满足条件。

 最后更新: 2023年5月11日