## 工作日志

学号：1830090115

姓名：李响

* 学习内容：

|  |  |
| --- | --- |
| 序号 | 内容 |
| 1 | 首页的内容展示：轮播图展示。 |
| 2 | Cms系统的实现。 |
| 3 | Taotao-rest发布服务。 |
| 4 | 表现层调用服务，展示轮播图 |

* 心得体会：

初始化easyUI的tree控件url：/content/category/list

内容分类是个Tree，每一类的内容在tb\_content115表里。

图形用户界面, 文本

描述已自动生成

package top.lixiang007.taotao.controller**;**import org.springframework.beans.factory.annotation.Autowired**;**import org.springframework.stereotype.Controller**;**import org.springframework.web.bind.annotation.RequestMapping**;**import org.springframework.web.bind.annotation.RequestParam**;**import org.springframework.web.bind.annotation.ResponseBody**;**import top.lixiang007.taotao.common.pojo.EasyUITreeNode115**;**import top.lixiang007.taotao.common.pojo.TaotaoResult**;**import top.lixiang007.taotao.service.ContentCatService115**;**import java.util.List**;***/\*\*  
 \* 内容分类管理Controller  
 \*/  
/\*\*  
 \** ***@Author*** *Xiang Li  
 \** ***@Date*** *2021/8/31 上午9:05  
 \** ***@Since*** *version-1.0  
 \** ***@青软实训*** *\*/*@Controller  
@RequestMapping("/content/category")  
public class ContentCatController115 {  
  
 @Autowired(required = false)  
 private ContentCatService115 contentCatService115**;** @ResponseBody  
 @RequestMapping("/list")  
 public List<EasyUITreeNode115> getContentCatList(@RequestParam(value = "id"**,** defaultValue = "0") Long parentId) {  
 return contentCatService115.getContentCatList(parentId)**;** }  
  
 @ResponseBody  
 @RequestMapping("/create")  
 public TaotaoResult insertCategory(Long parentId**,** String name) {  
 return contentCatService115.insertCategory(parentId**,** name)**;** }  
}

新增节点:

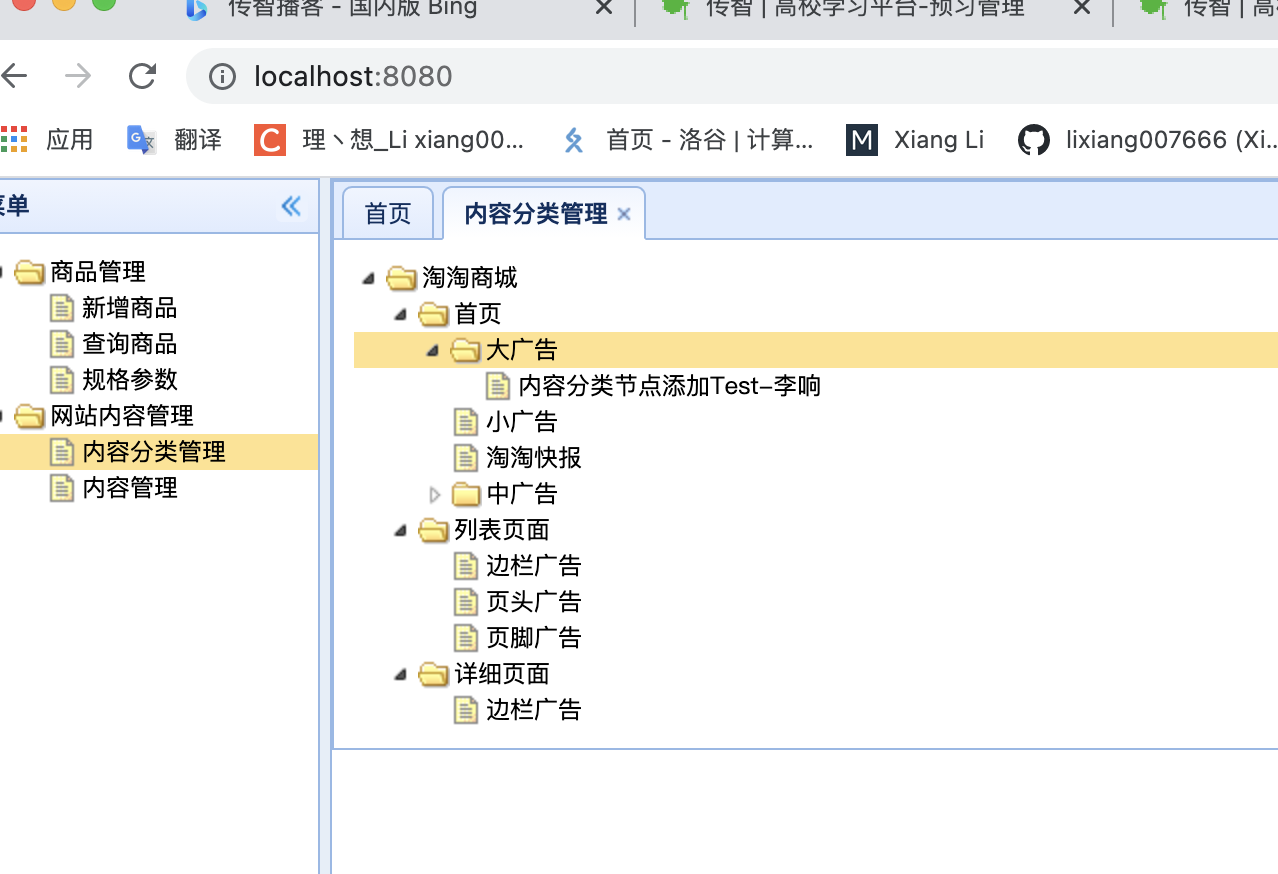
图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

请求的url：/content/category/create

请求的参数：parentId、name

返回结果：返回TaotaoResult，包含新添加记录的id。



删除和重命名同理，重命名貌似没有涉及到Controller的逻辑，和添加create使用一个url。

系统架构的分析很重要。

点击内容管理出现内容表的内容，如存放的图像。

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

新增内容：  
请求url：/content-add

初始化页面：

图形用户界面, 文本

描述已自动生成

表单提交处理：

图形用户界面, 文本, 应用程序

描述已自动生成

请求的url：/content/save

提交的内容：表单中的内容。使用TbContent接收。

返回值：TaotaoResult

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

编写点击提交插入内容信息代码即可。

内容添加测试：

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

内容查询服务发布：

图示

描述已自动生成

门户调用：

可以使用jsonp获得数据。做网站不推荐大面积使用ajax。

**不使用ajax如何获得数据？**

只需要在portal的Service层调用taotao-rest的服务，获得数据，把数据传递给jsp即可。

要调用服务需要使用java代码模拟浏览器调用服务请求数据。

**可以使用HttpClient。**

第一步：把HttpClient使用的jar包添加到工程中。

第二步：创建一个HttpClient的测试类

第三步：创建测试方法。

第四步：创建一个HttpClient对象

第五步：创建一个HttpGet对象，需要制定一个请求的url

第六步：执行请求。

第七步：接收返回结果。HttpEntity对象。

第八步：取响应的内容。

package top.lixiang007.taotao.portal.service.impl**;**import java.util.List**;**import org.springframework.beans.factory.annotation.Value**;**import org.springframework.stereotype.Service**;**import top.lixiang007.taotao.common.pojo.TaotaoResult**;**import top.lixiang007.taotao.common.utils.HttpClientUtil**;**import top.lixiang007.taotao.pojo.TbContent**;**import top.lixiang007.taotao.portal.service.ContentService**;***/\*\*  
 \** ***@Author*** *Xiang Li  
 \** ***@Date*** *2021/8/30 上午10:15  
 \** ***@Since*** *version-1.0  
 \** ***@青软实训*** *\*/*@Service  
public class ContentServiceImpl implements ContentService {  
   
 @Value("${SERVICE\_BASE\_URL}")  
 private String SERVICE\_BASE\_URL **;** @Value("${INDEX\_AD1\_URL}")  
 private String INDEX\_AD1\_URL **;** @Override  
 public List<TbContent> getContentList(long categoryId) {  
 //调用服务层的服务  
 String resStr = HttpClientUtil.*doGet*(SERVICE\_BASE\_URL + INDEX\_AD1\_URL + categoryId)**;** //把字符串转换成java对象  
 TaotaoResult result = TaotaoResult.*formatToList*(resStr**,** TbContent.class)**;** if (result.getStatus() == **200**) {  
 List<TbContent> listContent = (List<TbContent>) result.getData()**;** return listContent**;** }  
 return null**;** }  
  
}

* 遇到问题：

|  |  |
| --- | --- |
| 序号 | 问题 |
| 1 | 严重: Servlet.service() for servlet [taotao-portal] in context with path [] threw exception [Request processing failed; nested exception is java.lang.NullPointerException] with root cause |

* 解决方案：

|  |  |
| --- | --- |
| 序号 | 方案 |
| 1 | 查询大广告位的id为89，没有设置。 |

图形用户界面, 网站

描述已自动生成