

乐字节教育高级架构课程

正所谓“授人以鱼不如授人以渔”，你们想要的 **Java 学习资料** 来啦！

不管你是学生，还是已经步入职场的同行，希望你们都要珍惜眼前的学习机会，奋斗没有终点，知识永不过时。

扫描下方二维码即可领取



乐字节晓啡



乐字节官方交流群



```

/**
 * 获取多级联动的商品分类
 * id:当前选择框的值
 * next: 下级选择框显示的内容
 * select_id:
 */
function getCategory(id, next, select_id) {
    var url = '${ctx}/goods/category/' + id;
    // 用户重新选择顶级分类时, 重置下级分类为: 请选择商品分类, 且清空下级分类信息
    var htmlStr = "<option value='0'>请选择商品分类</option>";
    if(0==id){
        $("#"+next).html(htmlStr);
        return;
    }
    $.ajax({
        type: "GET",
        url: url,
        error: function (request) {
            layer.alert("获取子分类失败! ");
        },
        success: function (result) {
            if (result.length > 0) {
                for (i = 0; i < result.length; i++) {
                    htmlStr += "<option value='" + result[i].id + "'>" + result[i].name + "
                </option>"
                }
                $("#"+next).html(htmlStr);
            } else {
                layer.alert("获取子分类失败! ");
            }
        }
    });
}

```

商品分类-新增分类-保存

编写公共保存状态返回对象

shop-common的BaseResultEnum.java

```

package com.xxxx.common.enums;

/**
 * 为什么使用枚举类?
 * 1.提升代码的阅读性, 避免硬编码。
 * 2.程序解耦
 */
public enum BaseResultEnum {
    // 自定义枚举类
}

```

```

SUCCESS(200, "成功"),
ERROR(400, "失败"),
PASS_ERROR_01(501, "两次输入的密码不一致"),
PASS_ERROR_02(502, "输入的原始密码不正确"),
PASS_ERROR_03(503, "输入的验证码不正确"),
PASS_ERROR_04(504, "用户名或者密码错误");

private Integer code;
private String message;

private BaseResultEnum(Integer code, String message) {
    this.code = code;
    this.message = message;
}

public Integer getCode() {
    return code;
}

public void setCode(Integer code) {
    this.code = code;
}

public String getMessage() {
    return message;
}

public void setMessage(String message) {
    this.message = message;
}
}

```

shop-common的BaseResult.java

```

package com.xxxx.common.result;

import com.github.pagehelper.PageInfo;
import com.xxxx.common.enums.BaseResultEnum;

import java.io.Serializable;

/**
 * 公共保存状态返回对象
 */
public class BaseResult implements Serializable {
    // 状态编码
    private Integer code;
    // 状态描述
    private String message;
    // 分页对象(商品列表需要-pom.xml添加依赖)
    PageInfo<?> pageInfo;
}

```

//成功返回的对象

```
public static BaseResult success() {  
    BaseResult result = new BaseResult();  
    result.setCode(BaseResultEnum.SUCCESS.getCode());  
    result.setMessage(BaseResultEnum.SUCCESS.getMessage());  
    return result;  
}
```

//成功返回的对象-带分页对象

```
public static BaseResult success(PageInfo<?> pageInfo) {  
    BaseResult result = new BaseResult();  
    result.setCode(BaseResultEnum.SUCCESS.getCode());  
    result.setMessage(BaseResultEnum.SUCCESS.getMessage());  
    result.setPageInfo(pageInfo);  
    return result;  
}
```

//失败返回的对象

```
public static BaseResult error() {  
    BaseResult result = new BaseResult();  
    result.setCode(BaseResultEnum.ERROR.getCode());  
    result.setMessage(BaseResultEnum.ERROR.getMessage());  
    return result;  
}
```

```
public BaseResult(Integer code, String message) {  
    this.code = code;  
    this.message = message;  
}
```

```
public BaseResult() {  
}
```

```
public Integer getCode() {  
    return code;  
}
```

```
public void setCode(Integer code) {  
    this.code = code;  
}
```

```
public String getMessage() {  
    return message;  
}
```

```
public void setMessage(String message) {  
    this.message = message;  
}
```

```
public PageInfo<?> getPageInfo() {  
    return pageInfo;  
}
```

```
public void setPageInfo(PageInfo<?> pageInfo) {
    this.pageInfo = pageInfo;
}

@Override
public String toString() {
    return "BaseResult [code=" + code + ", message=" + message + "];"
}
}
```

Service服务层

GoodsCategoryService.java

```
/**
 * 商品分类-新增分类-保存
 * @param goodsCategory
 * @return
 */
int categorySave(GoodsCategory goodsCategory);
```

GoodsCategoryServiceImpl.java

```
/**
 * 商品分类-新增分类-保存
 * @param goodsCategory
 * @return
 */
@Override
public int categorySave(GoodsCategory goodsCategory) {
    return goodsCategoryMapper.insertSelective(goodsCategory);
}
```

Controller控制层

GoodsController.java

```
/**
 * 商品分类-添加-保存
 *
 * @param goodsCategory
 * @return
 */
@RequestMapping("category/save")
@ResponseBody
public BaseResult CategorySave(GoodsCategory goodsCategory) {
    int i = goodsCategoryService.categorySave(goodsCategory);
    //三目运算符
    return i > 0 ? BaseResult.success() : BaseResult.error();
}
```

ftl页面

category-add.ftl

添加两个隐藏域，根据用户操作onchange动态赋值传往后台

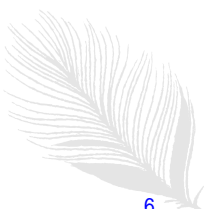
```
<input type="hidden" name="parentId" id="parentId" value="0">
<input type="hidden" name="level" id="level" value="1">
<div class="col-sm-3">
    <select name="parent_id_1" id="parent_id_1"
        onchange="getCategory(this.value,'parent_id_2','0');"
        class="small form-control">
        <option value="0">顶级分类</option>
        <#list gcList as gc>
            <option value="${gc.id}">${gc.name}</option>
        </#list>
    </select>
</div>
```

```
<div class="col-sm-3">
    <select name="parent_id_2" id="parent_id_2" class="small form-control"
        onchange="setParentId(this.value,'3!')">
        <option value="0">请选择商品分类</option>
    </select>
</div>
```

```
<div class="box-footer">
    <input type="hidden" name="id" value="">
    <button type="reset" class="btn btn-primary pull-left"><i class="icon-ok"></i>重填
</button>
    <button type="button"
        onclick="ajaxSubmitForm();"
        class="btn btn-primary pull-right"><i class="icon-ok"></i>提交
    </button>
</div>
```

修改页面form表单中name的域名称与GoodsCategory对象中的属性——对应

```
<input type="text" name="parentId" id="parentId" value="0"/>
<input type="text" name="level" id="level" value="1"/>
<input type="text" name="image" id="image"/>
```



```

/** 商品分类名称 name */
private String name;

/** 手机端显示的商品分类名 mobile_name */
private String mobileName;

/** 父id parent id */
private Short parentId;

/** 家族图谱 parent_id_path */
private String parentIdPath;

/** 等级 level */
private Byte level;

```

js动态赋值，实现保存新增分类

```

/**
 * 获取多级联动的商品分类
 * id:当前选择框的值
 * next: 下级选择框显示的内容
 * select_id:level
 */
function getCategory(id, next, select_id) {
    var url = '${ctx}/goods/category/' + id;
    // 用户重新选择顶级分类时，重置下级分类为：请选择商品分类，且清空下级分类信息
    var htmlStr = "<option value='0'>请选择商品分类</option>";
    //修改parentId
    $("#parentId").val(id);
    if (0 == id) {
        $("#" + next).html(htmlStr);
        //修改level
        $("#level").val(1);
        return;
    }
    $.ajax({
        type: "GET",
        url: url,
        error: function (request) {
            layer.alert("获取子分类失败! ");
        },
        success: function (result) {
            if (result.length > 0) {
                for (i = 0; i < result.length; i++) {
                    htmlStr += "<option value='" + result[i].id + "'>" + result[i].name + "
                </option>"
                }
                $("#" + next).html(htmlStr);
                $("#level").val(2);
            } else {

```

```

        layer.alert("获取子分类失败！");
    }
}
});
}

/**
 *保存分类
 */
function ajaxSubmitForm() {
    $.ajax({
        url: "${ctx}/goods/category/save",
        type: "post",
        data: $("#category_form").serialize(),
        dataType: "JSON",
        success: function (result) {
            if (result.code==200){
                layer.confirm("保存成功",{btn:['继续新增','返回列表']},
                    function () {
                        window.location.href="${ctx}/goods/category/add";
                    },function () {
                        window.location.href="${ctx}/goods/category/list";
                    });
            }else {
                layer.alert("保存失败");
            }
        },
        error: function () {
            layer.alert("保存失败");
        }
    });
}

/**
 * 设置parentId和level
 */
function setParentId(parentId,level) {
    // 修改parentId和level
    if (0==parentId){
        $("#parentId").val($("#parent_id_1").val);
        $("#level").val(2);
        return;
    }
    // 修改parentId和level
    $("#parentId").val(parentId);
    $("#level").val(level);
}

```

