





一 模块查询

1 需求分析

针对模块部分进行查询

2 页面原型

 添加
  修改
  删除

模块名称:
 操作码值:
 父模块名称:
 查询

	<input type="checkbox"/>	模块名称	父模块名称	模块样式	操作权限值	url	层级	创建时间	更新时间
1	<input type="checkbox"/>	营销管理		yxgl	10	#	根级	2017-09-28	2017-09-28
2	<input type="checkbox"/>	营销机会管理	营销管理	icon-yxjhgl	1010	sale_chance/index	第一级	2017-09-28	2017-09-28
3	<input type="checkbox"/>	营销机会管理添加	营销机会管理	1111	101001	#	第二级	2017-09-28	2017-09-28
4	<input type="checkbox"/>	修改	营销机会管理		101002	#	第二级	2017-09-28	2017-09-28
5	<input type="checkbox"/>	删除	营销机会管理		101003	#	第二级	2017-09-28	2017-09-28
6	<input type="checkbox"/>	客户开发计划	营销管理	icon-khkfh	1020	sale_chance/index?sate=1	第一级	2017-09-28	2017-09-28
7	<input type="checkbox"/>	查看详情	客户开发计划		102001	#	第二级	2017-09-28	2017-09-28
8	<input type="checkbox"/>	客户管理		icon-khgl	20	customer_serive/index	根级	2017-07-01	2017-07-01
9	<input type="checkbox"/>	客户信息管理	客户管理	icon-khxxgl	2010	customer/index	第一级	2017-09-06	2017-09-06
10	<input type="checkbox"/>	创建	客户信息管理		201001	#	第二级	2017-07-01	2017-07-01

3 编写Controller

```

@RequestMapping("queryModulesByParams")
@ResponseBody
public Map<String, Object> queryModulesByParams(ModuleQuery moduleQuery){
    return moduleService.queryModulesByParams(moduleQuery);
}
    
```

4 编写Service

```

public Map<String, Object> queryModulesByParams(ModuleQuery moduleQuery) {
    // 初始化分页环境
    PageHelper.startPage(moduleQuery.getPage(), moduleQuery.getRows());
    List<Module> modules = moduleDao.queryModulesByParams(moduleQuery);
    PageInfo<Module> pageInfo = new PageInfo<>(modules);
    Map<String, Object> map = new HashMap<String, Object>();
    map.put("total", pageInfo.getTotal());
    map.put("rows", pageInfo.getList());
    return map;
}
    
```

5 编写Dao

```
public List<Module> queryModulesByParams(ModuleQuery moduleQuery);
```

6 编写Mapper

```
<select id="queryModulesByParams" parameterType="moduleQuery" resultType="module">
    SELECT
    m1.id,
    m1.parent_id as parentId,
    m1.module_name AS moduleName,
    m2.module_name AS parentModuleName,
    m1.module_style as moduleStyle,
    m1.url,
    m1.grade,
    m1.opt_value AS optValue,
    m1.orders,
    m1.create_date AS createDate,
    m1.update_date AS updateDate
    FROM
    t_module m1
    LEFT JOIN t_module m2 ON m1.parent_id = m2.id
    <where>
        m1.is_valid=1
        <if test="moduleName !=null and moduleName!=''">
            AND m1.module_name LIKE concat('%',{moduleName},'%')
        </if>
        <if test="optValue !=null and optValue!=''">
            AND m1.opt_value LIKE concat('%',{optValue },'%')
        </if>
        <if test="parentModuleName !=null and parentModuleName!=''">
            AND m2.module_name LIKE concat('%',{parentModuleName},'%')
        </if>
    </where>
</select>
```

7 编写Query

```
public class ModuleQuery extends BaseQuery {

    private String moduleName;
    private String optValue;
    private String parentModuleName;
    public String getModuleName() {
        return moduleName;
    }
    public void setModuleName(String moduleName) {
        this.moduleName = moduleName;
    }
}
```



```
public String getOptValue() {  
    return optValue;  
}  
public void setOptValue(String optValue) {  
    this.optValue = optValue;  
}  
public String getParentModuleName() {  
    return parentModuleName;  
}  
public void setParentModuleName(String parentModuleName) {  
    this.parentModuleName = parentModuleName;  
}  
}
```

8 前台js

```
function searchModules(){  
    $("#dg").datagrid("load",{  
        moduleName:$("#moduleName").val(),  
        optValue:$("#optValue").val(),  
        parentModuleName:$("#parentModuleName").val()  
    });  
}
```

二 模块插入

1 需求分析

添加模块内容

2 页面原型

!



2017-09-20 2017-09-20

20 添加模块

20 模块名称:

20 url:

20 模块样式:

权限值:

排序:

层级:

保存 取消

3 编写Controller

```
@RequestMapping("insert")
@ResponseBody
public MessageModel insert(Module module){
    moduleService.insert(module);
    return success("模块记录添加成功!");
}

@RequestMapping("queryModulesByGrade")
@ResponseBody
public List<Module> queryModulesByGrade(Integer grade){
    return moduleService.queryModulesByGrade(grade);
}
```

4 编写Service

```
public void insert(Module module) {
    /**
     * 1.参数校验
     *     模块名称非空
     *     层级 非空
     *     模块权限值 非空
     * 2.权限值不能重复
     * 3.每一层 模块名称不能重复
     * 4.非根级菜单 上级菜单必须存在
     * 5.设置额外字段
```

```
*      isValid
*      createDate
*      updateDate
* 6.执行添加
*/

checkModuleParams(module.getModuleName(), module.getGrade(), module.getOptValue());
AssertUtil.isTrue(null != moduleDao.queryModuleByOptValue(module.getOptValue()), "权限值
不能重复!");
AssertUtil.isTrue(null != moduleDao.queryModuleByGradeAndModuleName(module.getGrade(),
module.getModuleName()),
    "该层级下模块名已存在!");
if (module.getGrade() != 0) {
    AssertUtil.isTrue(null == moduleDao.queryModuleByPid(module.getParentId()), "父级菜
单不存在!");
}
module.setIsValid(1);
module.setCreateDate(new Date());
module.setUpdateDate(new Date());
AssertUtil.isTrue(moduleDao.insert(module) < 1, CrmConstant.OPS_FAILED_MSG);
}

private void checkModuleParams(String moduleName, Integer grade,
    String optValue) {
    AssertUtil.isTrue(StringUtils.isBlank(moduleName), "模块名非空!");
    AssertUtil.isTrue(null == grade, "层级值非法!");
    Boolean flag = (grade != 0 && grade != 1 && grade != 2);
    AssertUtil.isTrue(flag, "层级值非法!");
    AssertUtil.isTrue(StringUtils.isBlank(optValue), "权限值非空!");
}

public List<Module> queryModulesByGrade(Integer grade) {
    return moduleDao.queryModulesByGrade(grade);
}
```

5 编写Dao

```
@Insert("insert into t_module(module_name,module_style,url,opt_value,parent_id,grade,"
+ "orders,is_valid,create_date,update_date) values("
+ "#{moduleName},#{moduleStyle},#{url},#{optValue},#{parentId},#{grade},"
+ "#{orders},#{isValid},#{createDate},#{updateDate})")
public Integer insert(Module module);
@Select("select id,module_name as moduleName from t_module where is_valid=1 and grade=#
{grade}")
public List<Module> queryModulesByGrade(@Param("grade")Integer grade);
```

6 前台js

```
$(function(){
    $("#dlg").dialog({
        onClose:function(){
```



```
        initFormData();
    }
});
$("#parentMenu").hide();
$("#grade").combobox({
    onChange:function(grade){
        if(grade==1||grade==2){
            $("#parentMenu").show();
        }
        if(grade==0){
            $("#parentMenu").hide();
        }
        loadParentModules(grade-1);
    }
});
});
function loadParentModules(grade){
    $("#parentId").combobox("clear");
    $("#parentId").combobox({
        url:ctx+"/module/queryModulesByGrade?grade="+grade,
        valueField:'id',
        textField:'moduleName'
    });
}
function formatterGrade(val){
    if(val==0){
        return "根级";
    }
    if(val==1){
        return "第一级";
    }
    if(val==2){
        return "第二级";
    }
}
function initFormData(){
    $("#fm").form("clear");
}
function saveOrUpdateModule(){
    var id=$("#id").val();
    var url=ctx+"/module/insert";
    if(!isEmpty(id)){
        url=ctx+"/module/update";
    }
    $("#fm").form("submit", {
        url:url,
        onSubmit:function () {
            return $("#fm").form("validate");
        },
        success:function (data) {
            data = JSON.parse(data);
            if(data.code==200){
                $.messager.alert("来自crm系统",data.msg,"info")
            }
        }
    });
}
```

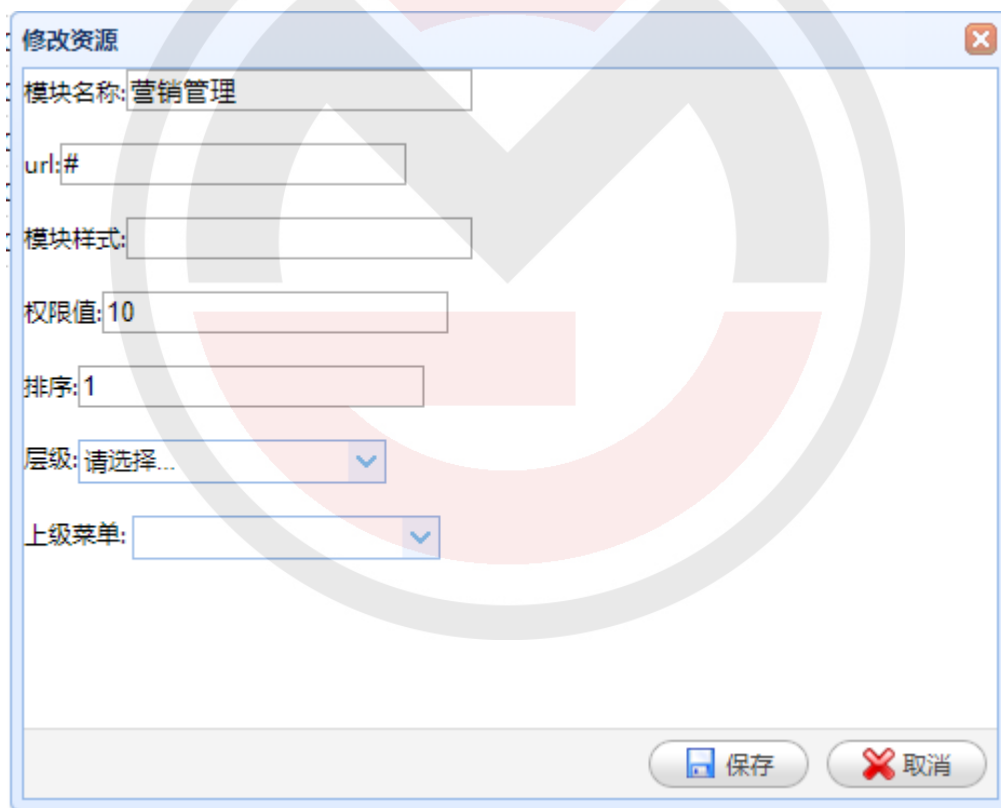
```
        $("#fm").form("clear");  
        closeDialog();  
        searchModules();  
    }else{  
        $.messager.alert("来自crm系统",data.msg,"info")  
    }  
}  
});  
}
```

三 模块更新

1 需求分析

更新模块内容

2 页面原型



3 编写Controller

```
@RequestMapping("update")  
@ResponseBody  
public MessageModel update(Module module){  
    moduleService.update(module);  
    return success("模块记录更新成功!");  
}
```

4 编写Service

```
public void update(Module module) {  
    /**  
     * 1.参数校验  
     *    模块名称非空  
     *    层级 非空  
     *    模块权限值 非空  
     *    id 记录必须存在  
     * 2.权限值不能重复  
     * 3.每一层 模块名称不能重复  
     * 4.非根级菜单 上级菜单必须存在  
     * 5.设置额外字段  
     *    updateDate  
     * 6.执行更新  
     */  
    checkModuleParams(module.getModuleName(), module.getGrade(), module.getOptValue(),  
module.getId());  
  
    Module temp = moduleDao.queryModuleByOptValue(module.getOptValue());  
    AssertUtil.isTrue(null != temp && !temp.getId().equals(module.getId()), "权限值不能重复!");  
    temp = moduleDao.queryModuleByGradeAndModuleName(module.getGrade(),  
module.getModuleName());  
    AssertUtil.isTrue(null != temp && !temp.getId().equals(module.getId()), "该层级下模块名不能  
重复!");  
    if (module.getGrade() != 0) {  
        AssertUtil.isTrue(null == moduleDao.queryModuleByPid(module.getParentId()), "父级菜单  
不存在!");  
    }  
    module.setUpdateDate(new Date());  
    AssertUtil.isTrue(moduleDao.update(module) < 1, CrmConstant.OPS_FAILED_MSG);  
}
```

5 编写Dao

```
@Update("update t_module set module_name=#{moduleName},module_style=#{modulestyle},"  
+ "url=#{url},opt_value=#{optValue},parent_id=#{parentId},"  
+ "grade=#{grade},orders=#{orders},update_date=#{updateDate}"  
+ "where id=#{id} and is_valid=1")  
public Integer update(Module module);  
@Select("select id,module_name as moduleName,url,opt_value as optValue"  
+ " from t_module where opt_value=#{optValue} and is_valid=1")  
public Module queryModuleByOptValue(@Param("optValue")String optValue);  
@Select("select id,module_name as moduleName,url,opt_value as optValue"  
+ " from t_module where grade=#{grade} and module_name=#{mname} and is_valid=1")  
public Module queryModuleByGradeAndModuleName(@Param("grade")Integer  
grade,@Param("mname")String mname);  
@Select("select id,module_name as moduleName,url,opt_value as optValue"  
+ " from t_module where id=#{pid} and is_valid=1")  
public Module queryModuleByPid(@Param("pid")Integer pid);
```


6 前台js

```
function openModifyModuleDialog() {  
    var rows=$("#dg").datagrid("getSelections");  
    if(rows.length==0){  
        $.messager.alert("来自crm","请选择一条记录进行更新!");  
        return ;  
    }  
    if(rows.length>1){  
        $.messager.alert("来自crm","只能选择一条记录进行更新!");  
        return ;  
    }  
    /**  
    * 更新操作  
    */  
    $("#fm").form("load",rows[0]); //填充表单数据 数据回显  
    var grade=rows[0].grade;  
    if(grade!=0){  
        loadParentModules(grade-1);  
        $("#parentMenu").show();  
        $("#parentId").combobox("setValue",rows[0].parentId);  
    }else{  
        $("#grade").combobox("setValue",grade);  
    }  
    $("#dlg").dialog("open").dialog("setTitle","修改资源");  
}
```

四 模块更新

1 需求分析

更新模块内容

2 页面原型

添加 修改 删除

模块名称: 操作码值: 父模块名称: 查询

	<input type="checkbox"/>	模块名称	父模块名称	模块样式	操作权限值	url	层级	创建时间	更新时间
1	<input checked="" type="checkbox"/>	营销管理		yxgl	10	#	根级	2017-09-28	2017-09-28
2	<input type="checkbox"/>	营销机会管理	营销管理	icon-yxjghl	1010	sale_chance/index	第一级	2017-09-28	2017-09-28
3	<input type="checkbox"/>	营销机会管理添加	营销机会管理	1111	101001	#	第二级	2017-09-28	2017-09-28
4	<input type="checkbox"/>	修改	营销机会管理		101002	#	第二级	2017-09-28	2017-09-28
5	<input type="checkbox"/>	删除	营销机会管理		101003	#	第二级	2017-09-28	2017-09-28
6	<input type="checkbox"/>	客户开发计划	营销管理	icon-khkfhj	1020	sale_chance/index?sate=1	第一级	2017-09-28	2017-09-28
7	<input type="checkbox"/>	查看详情	客户开发计划		102001	#	第二级	2017-09-28	2017-09-28
8	<input type="checkbox"/>	客户管理		icon-khgl	20	customer_serve/index	根级	2017-07-01	2017-07-01
9	<input type="checkbox"/>	客户信息管理	客户管理	icon-khxxgl	2010	customer/index	第一级	2017-09-06	2017-09-06
10	<input type="checkbox"/>	创建	客户信息管理		201001	#	第二级	2017-07-01	2017-07-01

来自crm

? 确定删除选中的1条记录?

确定 取消

3 编写Controller

```
@RequestMapping("delete")
@ResponseBody
public MessageModel delete(Integer[] ids){
    moduleService.delete(ids[0]);
    return success("模块记录删除成功!");
}
```

4 编写Service

```
public void delete(Integer id) {
    /**
     * 删除当前节点
     * 级联删除该节点下所有后代节点
     * 后代节点id 值获取
     * 3层 级联查询
     * 查询当前节点 查询后代节点根据当前节点id
     * 层级无限制 递归查询
     * 100 1 99
     */
    AssertUtil.isTrue(null == id || null == moduleDao.queryModuleById(id), "待删除记录不存在!");
    List<Integer> mids = new ArrayList<Integer>();
    mids = getSubModuleIds(id, mids);
    AssertUtil.isTrue(moduleDao.delete(mids) < mids.size(), CrmConstant.OPS_FAILED_MSG);
}
```

5 编写Dao

```
public Integer delete(List<Integer> ids);
```

6 Mapper

```
<delete id="delete" parameterType="list">
    update t_module set is_valid=0
    where id in
    <foreach collection="list" item="item" open="(" separator="," close=")">
        #{item}
    </foreach>
</delete>
```

7 前台js

```
function deleteModule(){
    var rows=$("#dg").datagrid("getSelections");
    if(rows.length==0){
        $.messager.alert("来自crm","请选中待删除记录!","info");
        return;
    }
    // ids=1&ids=2&ids=3
    var ids="ids=";
    for(var i=0;i<rows.length;i++){
        if(i<=rows.length-2){
            ids=ids+rows[i].id+"&ids=";
            continue;
        }
        ids=ids+rows[i].id;
    }
    $.messager.confirm("来自crm","确定删除选中的"+rows.length+"条记录?",function(r){
        if(r){
            $.ajax({
                type:"post",
                url:ctx+"/module/delete",
                data:ids,
                dataType:"json",
                success:function(data){
                    $.messager.alert("来自crm",data.msg,"info");
                    if(data.code==200){
                        closeDialog();
                        searchModules();
                    }
                }
            });
        }
    });
}
```

