

## [今日课程大纲]

内容分类查询 内容分类重命名 内容分类重命名 内容分类重命名 内容分类删除 内容查询 内容新增 内容删除 内容修改

# [知识点详解]

#### 一. 内容分类查询

1. 在 ego-service 新建 TbContentCategoryDubboService 及实现类

```
public interface TbContentCategoryDubboService {
    /**
    * 根据父id 查询所有子类目
    * @param id
    * @return
    */
    List<TbContentCategory> selByPid(long id);
}
```



```
public class TbContentCategoryDubboServiceImpl
implements TbContentCategoryDubboService {
  @Resource
  private TbContentCategoryMapper
tbContentCategoryMapper;
  @Override
  public List<TbContentCategory> selByPid(long id) {
     TbContentCategoryExample example = new
TbContentCategoryExample();
  example.createCriteria().andParentIdEqualTo(id).and
StatusEqualTo(1);
     return
tbContentCategoryMapper.selectByExample(example);
  }
}
```

2. 在 ego-service-impl 的 applicationContext-dubbo.xml 中注册接口



3. 在 ego-manage 中新建 TbContentCategoryService 及实现类

```
public interface TbContentCategoryServie {
   /**
   * 查询所有类目并转换为 <u>easyui</u> tree 的属性要求
   * @return
   */
  List<EasyUiTree> showCategory(long id);
}
@Service
public class TbContentCategoryServiceImpl implements
TbContentCategoryServie {
  @Reference
  private TbContentCategoryDubboService
tbContentCategoryDubboServiceImpl;
  @Override
  public List<EasyUiTree> showCategory(long id) {
     List<EasyUiTree> listTree = new
```



```
ArrayList<EasyUiTree>();
     List<TbContentCategory> list =
tbContentCategoryDubboServiceImpl.selByPid(id);
     for (TbContentCategory cate : list) {
        EasyUiTree tree = new EasyUiTree();
        tree.setId(cate.getId());
        tree.setText(cate.getName());
  tree.setState(cate.getIsParent()?"closed":"open");
        listTree.add(tree);
     }
     return listTree;
  }
}
```

4. 在 ego-manage 中新建 TbContentCategoryController

```
@Controller
public class TbContentCategoryController {
    @Resource
    private TbContentCategoryServie
```



```
tbContentCategoryServiceImpl;
    @RequestMapping("content/category/list")
    @ResponseBody
    public List<EasyUiTree>
showCategory(@RequestParam(defaultValue="0") long id){
    return

tbContentCategoryServiceImpl.showCategory(id);
    }
}
```

### 二. 内容分类新增功能

1. 在 ego-service 的 TbContentCategoryDubboService 及实现类添加

```
/**

* 新增

* @param cate

* @return

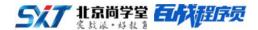
*/

int insTbContentCategory(TbContentCategory cate);

/**

* 修改 isparent 通过 id

* @param id
```



```
* @param isParent
   * @return
   */
  int updIsParentById(TbContentCategory cate);
@Override
  public int insTbContentCategory(TbContentCategory
cate) {
     return
tbContentCategoryMapper.insertSelective(cate);
  }
  @Override
  public int updIsParentById(TbContentCategory cate) {
     return
tbContentCategoryMapper.updateByPrimaryKeySelective(c
ate);
  }
```

2. 在 ego-manage 中 TbContentCategoryService 中添加

```
/**
    * 类目新增
    * @return
    */
```



```
EgoResult create(TbContentCategory cate);
  public EgoResult create(TbContentCategory cate) {
     EgoResult er = new EgoResult();
     //判断当前节点名称是否已经存在
     List<TbContentCategory> children =
tbContentCategoryDubboServiceImpl.selByPid(cate.getPa
rentId());
     for (TbContentCategory child : children) {
        if(child.getName().equals(cate.getName())){
          er.setData("该分类名已存在");
          return er;
        }
     }
     Date date = new Date();
     cate.setCreated(date);
     cate.setUpdated(date);
     cate.setStatus(1);
```



```
cate.setSortOrder(1);
     cate.setIsParent(false);
     long id = IDUtils.genItemId();
     cate.setId(id);
     int index =
tbContentCategoryDubboServiceImpl.insTbContentCategor
y(cate);
     if(index>0){
        TbContentCategory parent = new
TbContentCategory();
        parent.setId(cate.getParentId());
        parent.setIsParent(true);
  tbContentCategoryDubboServiceImpl.updIsParentById(p
arent);
     }
     er.setStatus(200);
     Map<String,Long> map = new HashMap<>();
     map.put("id", id);
     er.setData(map);
     return er;
  }
```



3. 在 ego-manage 的 TbContentCategoryController 中添加

```
/**
 * 新增內容类目
 * @param cate
 * @return
 */
    @RequestMapping("content/category/create")
    @ResponseBody
    public EgoResult create(TbContentCategory cate){
        return tbContentCategoryServiceImpl.create(cate);
    }
```

#### 三内容分类修改功能

1. 在 ego-service 的 TbContentCategoryDubboService 及实现类添加方法

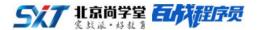
```
/**

* 通过 id 查询内容类目详细信息

*/
TbContentCategory selById(long id);

@Override

public TbContentCategory selById(long id) {
    return
```



```
tbContentCategoryMapper.selectByPrimaryKey(id);
}
```

2. 在 ego-manage 中的 TbContentCategoryService 及实现类添加

```
/**
   * 类目重命名
   * @param cate
   * @return
   */
  EgoResult update(TbContentCategory cate);
  @Override
  public EgoResult update(TbContentCategory cate) {
     EgoResult er = new EgoResult();
     //查询当前节点信息
     TbContentCategory cateSelect =
tbContentCategoryDubboServiceImpl.selById(cate.getId(
));
     //查询当前节点的父节点的所有子节点信息
     List<TbContentCategory> children =
tbContentCategoryDubboServiceImpl.selByPid(cateSelect
.getParentId());
     for (TbContentCategory child : children) {
        if(child.getName().equals(cate.getName())){
```



```
er.setData("该分类名已存在");
return er;
}
int index =
tbContentCategoryDubboServiceImpl.updIsParentById(cat
e);
if(index>0){
    er.setStatus(200);
}
return er;
}
```

3. 在 ego-manage 的 TbContentCategoryController 添加

```
/**

* 重命名

* @param cate

* @return

*/

@RequestMapping("content/category/update")

@ResponseBody

public EgoResult update(TbContentCategory cate){
```



```
return tbContentCategoryServiceImpl.update(cate);
}
```

### 四. 内容分类删除功能

1. 在 ego-manage 的 TbContentCategoryService 及实现类添加

```
/**
   * 删除类目
   * @param id
   * @return
   */
  EgoResult delete(TbContentCategory cate);
@Override
  public EgoResult delete(TbContentCategory cate) {
     EgoResult er = new EgoResult();
     cate.setStatus(0);
     int index =
tbContentCategoryDubboServiceImpl.updIsParentById(cat
e);
     if(index>0){
        TbContentCategory curr =
```



```
tbContentCategoryDubboServiceImpl.selById(cate.getId(
));
        List<TbContentCategory> list =
tbContentCategoryDubboServiceImpl.selByPid(curr.getPa
rentId());
        if(list==null||list.size()==0){
           TbContentCategory parent = new
TbContentCategory();
           parent.setId(curr.getParentId());
           parent.setIsParent(false);
           int result =
tbContentCategoryDubboServiceImpl.updIsParentById(par
ent);
           if(result>0){
              er.setStatus(200);
           }
        }else{
           er.setStatus(200);
        }
     }
     return er;
  }
```



2. 在 ego-manage 的 TbContentCategoryController 中添加

```
/**
 * 删除
 * @param cate
 * @return
 */
  @RequestMapping("content/category/delete")
  @ResponseBody
  public EgoResult delete(TbContentCategory cate){
    return tbContentCategoryServiceImpl.delete(cate);
}
```

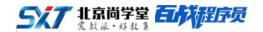
### 五. 内容查询

1. 在 ego-service 新建 TbContentDubboService 及实现类

```
public interface TbContentDubboService {
    /**
    * 分页查询
    * @param categoryId
    * @param page
    * @param rows
    * @return
```



```
*/
  EasyUIDataGrid selContentByPage(long categoryId,int
page,int rows);
}
public class TbContentDubboServiceImpl implements
TbContentDubboService {
  @Resource
  private TbContentMapper tbContentMapper;
  @Override
  public EasyUIDataGrid selContentByPage(long
categoryId, int page, int rows) {
     PageHelper.startPage(page, rows);
     TbContentExample example = new TbContentExample();
     if(categoryId!=0){
  example.createCriteria().andCategoryIdEqualTo(categ
oryId);
     List<TbContent> list =
tbContentMapper.selectByExampleWithBLOBs(example);
     PageInfo<TbContent> pi = new PageInfo<>(list);
```



```
EasyUIDataGrid datagrid = new EasyUIDataGrid();
    datagrid.setRows(pi.getList());
    datagrid.setTotal(pi.getTotal());
    return datagrid;
}
```

2. 在 ego-service-impl 中 applicationContext-dubbo.xml 注册接口

3. 在 ego-manage 中新建 TbContentService 及实现类

```
public interface TbContentService {
    /**
    * 分页显示内容信息
    * @param categoryId
    * @param page
```

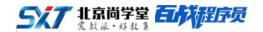


```
* @param rows
   * @return
   */
  EasyUIDataGrid showContent(long categoryId,int
page,int rows);
}
@Service
public class TbContentServiceImpl implements
TbContentService {
  @Reference
  private TbContentDubboService
tbContentDubboServiceImpl;
  @Override
  public EasyUIDataGrid showContent(long categoryId,
int page, int rows) {
     return
tbContentDubboServiceImpl.selContentByPage(categoryId
, page, rows);
  }
}
```

4. 在 ego-manage 中新建 TbContentController



```
@Controller
public class TbContentController {
  @Resource
  private TbContentService tbContentServiceImpl;
  /**
   * 显示内容信息
   * @param categoryId
   * @param page
   * @param rows
   * @return
   */
  @RequestMapping("content/query/list")
  @ResponseBody
  public EasyUIDataGrid showContent(long
categoryId,int page,int rows){
     return
tbContentServiceImpl.showContent(categoryId, page,
rows);
  }
}
```



#### 六. 内容新增

1. 在 ego-service 的 tbcontentdubboService 及实现类添加

```
/**

* 新增

* @param content

* @return

*/

int insContent(TbContent content);

@Override

public int insContent(TbContent content) {

return tbContentMapper.insertSelective(content);
}
```

2. 在 ego-manage 的 TbContentService 及实现类添加

```
/**

* 新增內容

* @param content

* @return

*/

int save(TbContent content);

@Override

public int save(TbContent content) {

Date date = new Date();
```



```
content.setCreated(date);
    content.setUpdated(date);
    return

tbContentDubboServiceImpl.insContent(content);
}
```

3. 在 ego-manage 的 TbContentController 添加

```
/**
   * 新增内容
   * @param content
   * @return
   */
  @RequestMapping("content/save")
  @ResponseBody
  public EgoResult save(TbContent content){
     EgoResult er = new EgoResult();
     int index = tbContentServiceImpl.save(content);
     if(index>0){
        er.setStatus(200);
     }
     return er;
  }
```