深圳四方精创资讯股份有限公司

Java EE开发平台——树形工具

使用手册



|  |  |  |  |
| --- | --- | --- | --- |
| **文档名称** | Java EE开发平台——树形工具使用手册 | **文档编号** |  |
| **文档类别** | 使用手册 | **版本信息** | V1.0 |
| **内部密级** | 公开 | **外部密级** | 自用 |
| **创 建 人** | 黄钟 | **创建日期** | 2016-6-15 |
| **审 批 人** |  | **审批日期** |  |

修改历史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本号 | 日期 | \*状态 | 修订人 | 摘要 |
| V1.0 | 2016-6-15 | C | 黄钟 | 创建文档 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

\*状态：C – 创建 A – 增加 M – 修改 D – 删除

# 目录

[目录 I](#_Toc20894)

[第1章 树形结构使用说明 1](#_Toc3879)

[§1.1 树形工具使用 1](#_Toc498)

[§1.1.1 使用树形工具类 1](#_Toc8204)

[§1.2 使用树形注解@TreeJsonBody 2](#_Toc14392)

# 树形结构使用说明

## 树形工具使用

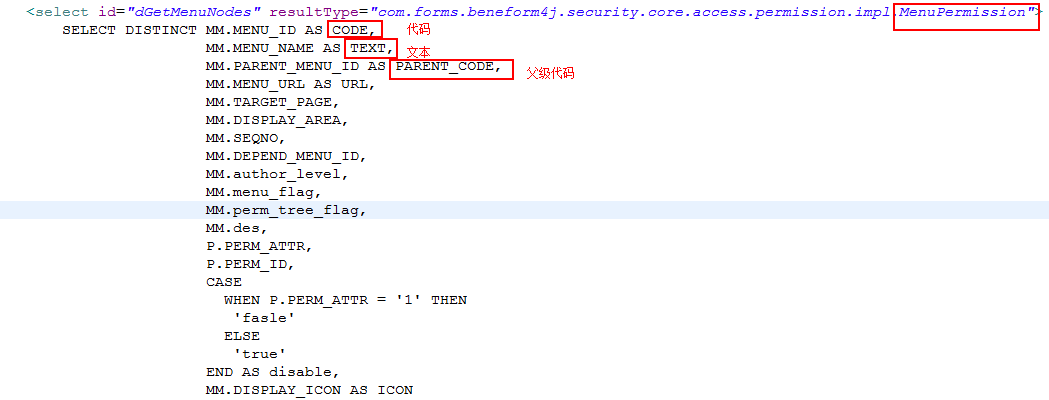
### 使用树形工具类

这里拿平台首页菜单的生成来做示例说明。由于平台的菜单和权限相关，这里仅说明树形数据的创建和树形结构的生成。

* 构建树形结构数据

要能生成树形结构，数据格式必须满足一定的规范：

SQL片段如下：



首先，MenuPermission类是一个实现了ITreeNode的平台默认实现类，应用如果有特殊的数节点属性，可自行实现ITreeNode替换平台的MenuPermission类

其次，SQL脚本中CODE，TEXT和PARENT\_CODE分别对应MenuPermission类的三个属性节点代码，节点文本和父节点代码，这三个属性的名称必须为code,parentCode和text，不允许更改，也就是说，应用实现了ITreeNode接口的类必须包含以上三个属性。

* 生成树

应用通过调用以下API生成完整的树：

List<ITreeNode> nodeList = “上述SQL返回的结果集”;

ITree<ITreeNode> tree = **new** Tree(nodeList);

其中ITreeNode可替换为应用自己的实现类

* API

平台提供以下API方法对树进行操作：

-getNode(code):根据code获取指定节点,返回一个ITreeNode对象

-getRoot()：获取根节点,返回一个ITreeNode对象

平台提供以下API方法对树节点进行操作：

-getChildren()：得到指定节点的全部子节点集合，返回List<ITreeNode>集合

注意：

1.ITree<ITreeNode> tree = **new** Tree(nodeList)；拿到的是一颗完整的树对象，但是在很多时候，比如获取菜单需要的仅仅是某个节点下的所有节点（菜单），这里我们仅仅需要返回树的某个节点的子节点即可，比如：

获取根节点的子节点集合：tree.getRoot().getChildren();(参见平台获取顶部一级菜单)

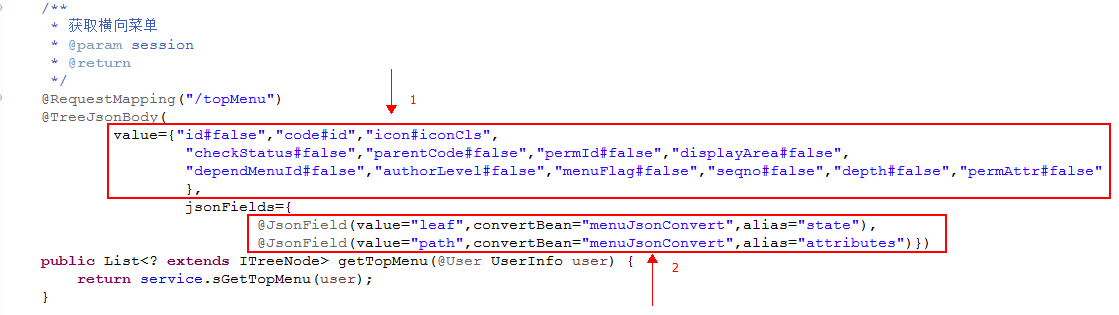
获取指定节点的子节点集合:tree.getNode(code).getChildren();(参见平台根据顶部某个菜单获取左侧菜单)

2.应用需要的树节点对象可实现ITreeNode接口，也可直接继承TreeNode类，建议直接继承TreeNode类，因为树节点的属性基本一致，无需再通过实现接口重复定义。

注:这里仅列出常用方法，更详细的说明请参考平台API文档

## 使用树形注解@TreeJsonBody

控制层通过服务层拿到树节点数据后，可使用TreeJsonBody这个注解返回数据：



箭头1标识区域作用

1. 可设置返回给前端属性的别名
2. 可设置忽略某个属性（不返回给前台）

具体写法如下：

property或property#1或property#true 配置属性名称 property#false或property#0 表示忽略属性

property#alias 表示将属性property使用别名alias输出，隐式表示是需要输出，但不影响属性筛选模式

注：同一属性多次配置以前面的为准

箭头2标识区域作用

可用jsonField注解定义属性转换类，比如上述就对leaf这个属性通过menuJsonConvert这个ID对应的类进行了转换

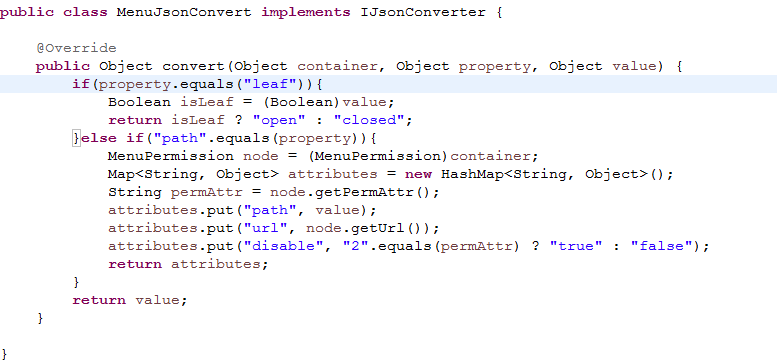
jsonField注解说明：

Value：待转换的属性名称

convertBean：转换类ID（必须由spring容器管理）

Alias：转换后的属性别名

menuJsonConvert转换类代码片段：



Leaf属性经过转换后，返回给前台的别名就变成了state，state的值由：

这段代码确定

