**类目中心设计**

# 定位

类目中心负责管理电商平台里的行业、类目、类目对应属性的管理等，是电商平台最为核心的模块之一。

# 数据库设计

## 行业表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **行业表（industry）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| Id | int | 自增主键，行业ID | 是 | 主键索引 |
| name | varchar(200) | 行业名字 | 是 |  |
| business\_type | int | 业务类型；  1：普通 | 是，默认1 |  |
| description | varchar(500) | 行业描述 | 否 |  |
| sort\_num | int | 排序号 | 是，默认0 |  |
| status | int | 状态，1:可用，0: 不可用 | 是，默认1 |  |
| features | varchar(4096) | 预留行业扩展字段 |  |  |
| creator | varchar(100) | 创建者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

## 类目表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类目表（industry）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| id | int | 自增主键，行业ID | 是 | 主键索引 |
| name | varchar(200) | 类目名字 | 是 |  |
| pid | int | 父级类目 | 是，默认0为一级类目 |  |
| cids | varchar(500) | 子类目ID，逗号隔开，方便查询该类型下所有子类目包含的商品 |  |  |
| description | varchar(500) | 类目描述 | 否 |  |
| sort\_num | int | 排序号 | 是，默认0 |  |
| status | int | 状态，1:可用，0: 不可用 | 是，默认1 |  |
| features | varchar(4096) | 预留类目扩展字段（JSON格式） |  |  |
| creator | varchar(100) | 创建者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

## 行业-类目关系表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **行业类目关系表（industry\_category）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| id | int | 自增主键，行业ID | 是 | 主键索引 |
| industry\_id | int | 行业ID | 是 |  |
| category\_id | int | 一级类目ID，行业只跟一级类目产生关联，子类目同属于本行业 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

# 接口设计

## 新增类目

### 入参

类目名称

父级类目ID

类目描述

类目状态

类目扩展属性（JSON）

创建者

### 出参

成功/失败状态（参考API中Result标准）

### 处理逻辑注意事项

注意创建子类目是，更新父级的cids，一级类目不需要更新

## 更新类目

### 入参

类目ID

类目名称

父级类目ID

类目描述

类目状态

类目扩展属性（JSON）

### 出参

成功/失败状态（参考API中Result标准）

### 处理逻辑注意事项

注意创建子类目是，更新父级的cids，一级类目不需要更新

## 修改类目状态

### 入参

类目ID

类目新状态

### 出参

成功/失败状态（参考API中Result标准）

### 处理逻辑注意事项

暂无

## 分配行业

### 入参

类目ID

行业ID集合

### 出参

成功/失败状态（参考API中Result标准）

### 处理逻辑注意事项

参考用户-权限管理处理方式，这里不需要写逻辑，利用JPA的多对多自动处理就好。

## 删除类目

### 入参

类目ID

### 出参

成功/失败状态（参考API中Result标准）

### 处理逻辑注意事项

包含子类目的提示失败，不能删除

## 批量获取类目详情

### 入参

类目ID集合

### 出参

成功/失败状态（参考API中Result标准）

类目详情集合（排序）

### 处理逻辑注意事项

根据sort\_num进行排序

只展示可用状态的类目

## 获取类目树

### 入参

父级类目ID（0为根类目，代表从根类目开始）

是否及联包含子类目（true：递归包含子类目，false：只查询当前PID的类目集合）

### 出参

成功/失败状态（参考API中Result标准）

类目树

### 处理逻辑注意事项

根据传入的父级类目ID，递归往下查询，形成类目树

父级类目传0，表示获取所有

只展示可用状态的类目

## 获取类目行业

### 入参

类目ID

### 出参

成功/失败状态（参考API中Result标准）

行业详情集合

### 处理逻辑注意事项

可利用JPA多对多关系，查询类目时就可以把行业详情带出来。不用写很复杂的逻辑。但要注意状态的过滤

## 行业接口

### 接口列表

1. 新增
2. 修改
3. 删除
4. 改变行业状态
5. 获取所有行业列表
6. 根据ID集合获取行业列表

### 注意事项

注意删除时及联删除行业-分类关系表，这里可利用JPA特性自动删除

注意获取行业时候的状态过滤