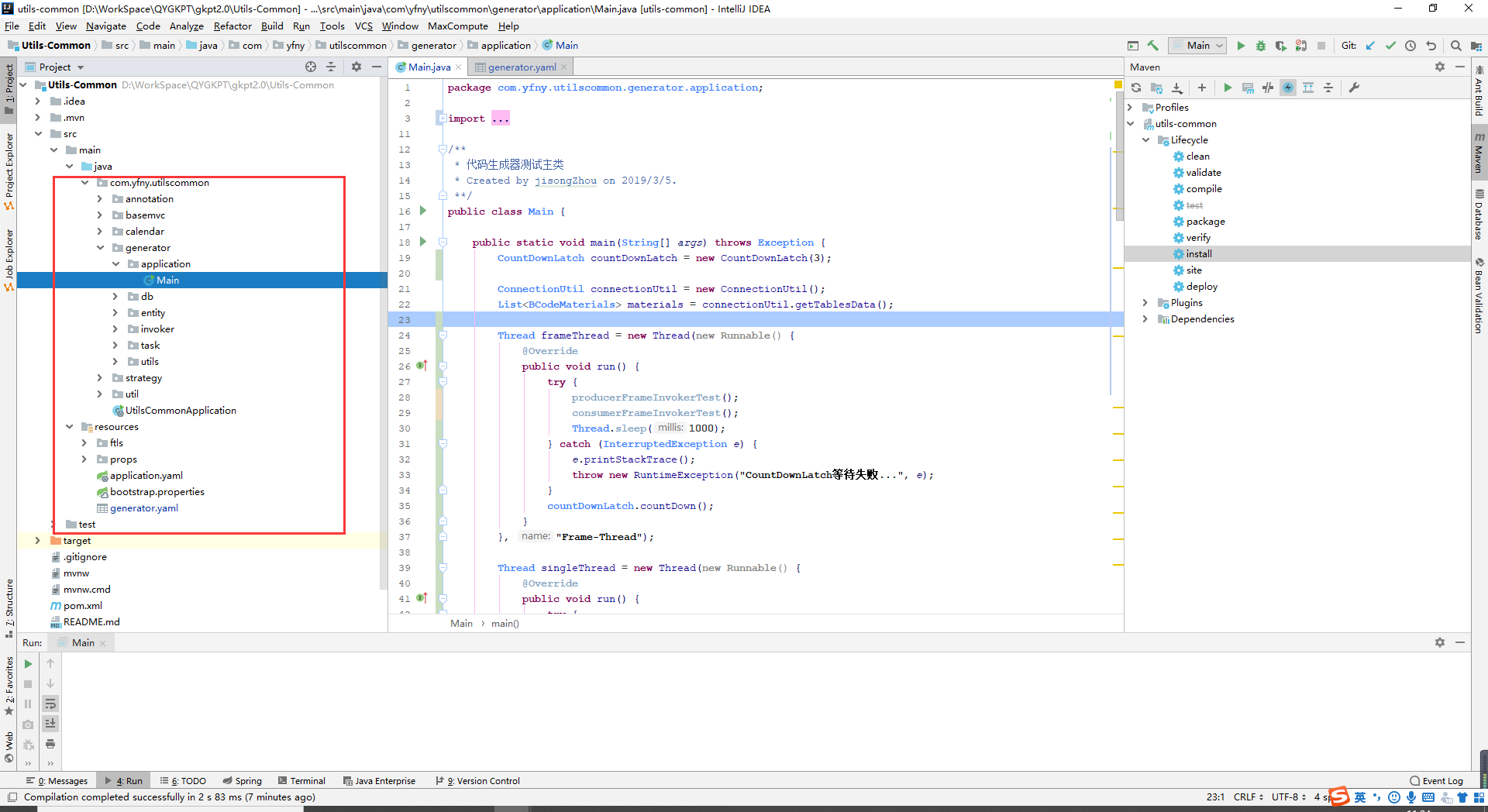
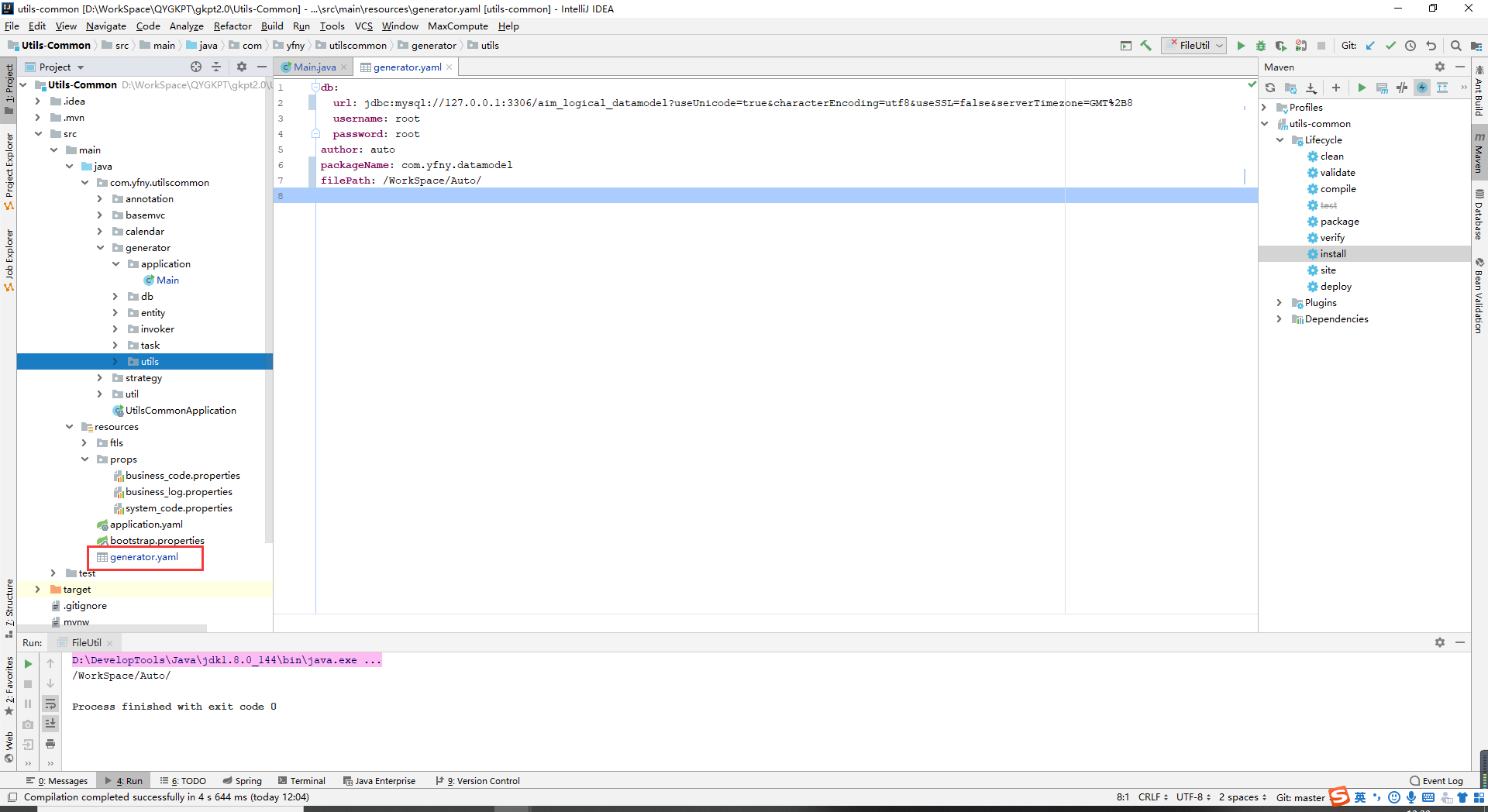
<https://github.com/feiyueyijia/Common-Utils>

# 导入项目加载maven依赖得到如下代码结构

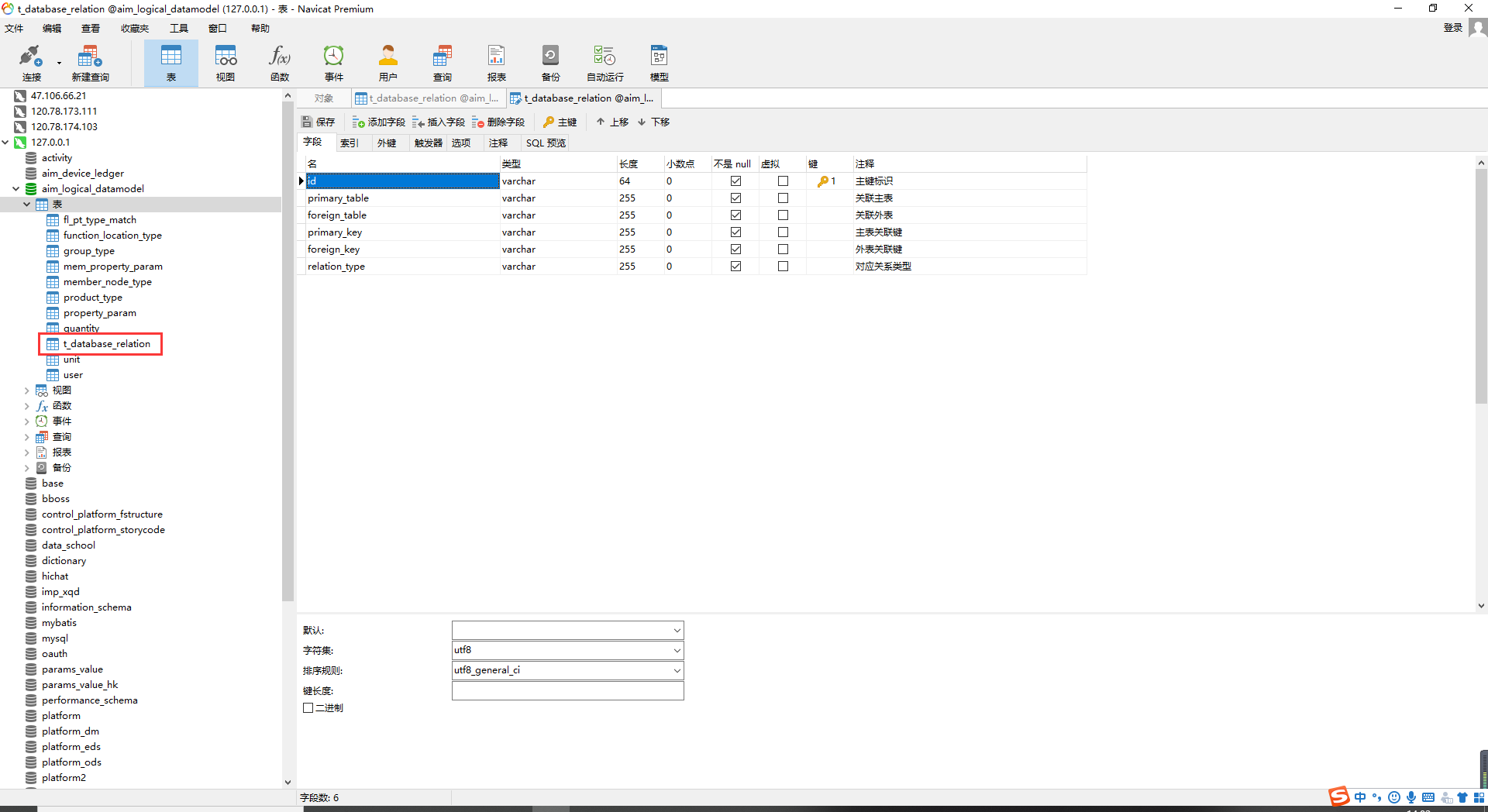


# 设计库表结构，建议按照功能模块内容进行区分不同的库，不建议在数据库表较多的情况下全部放在一个库中，每次生成代码只支持选择一个库

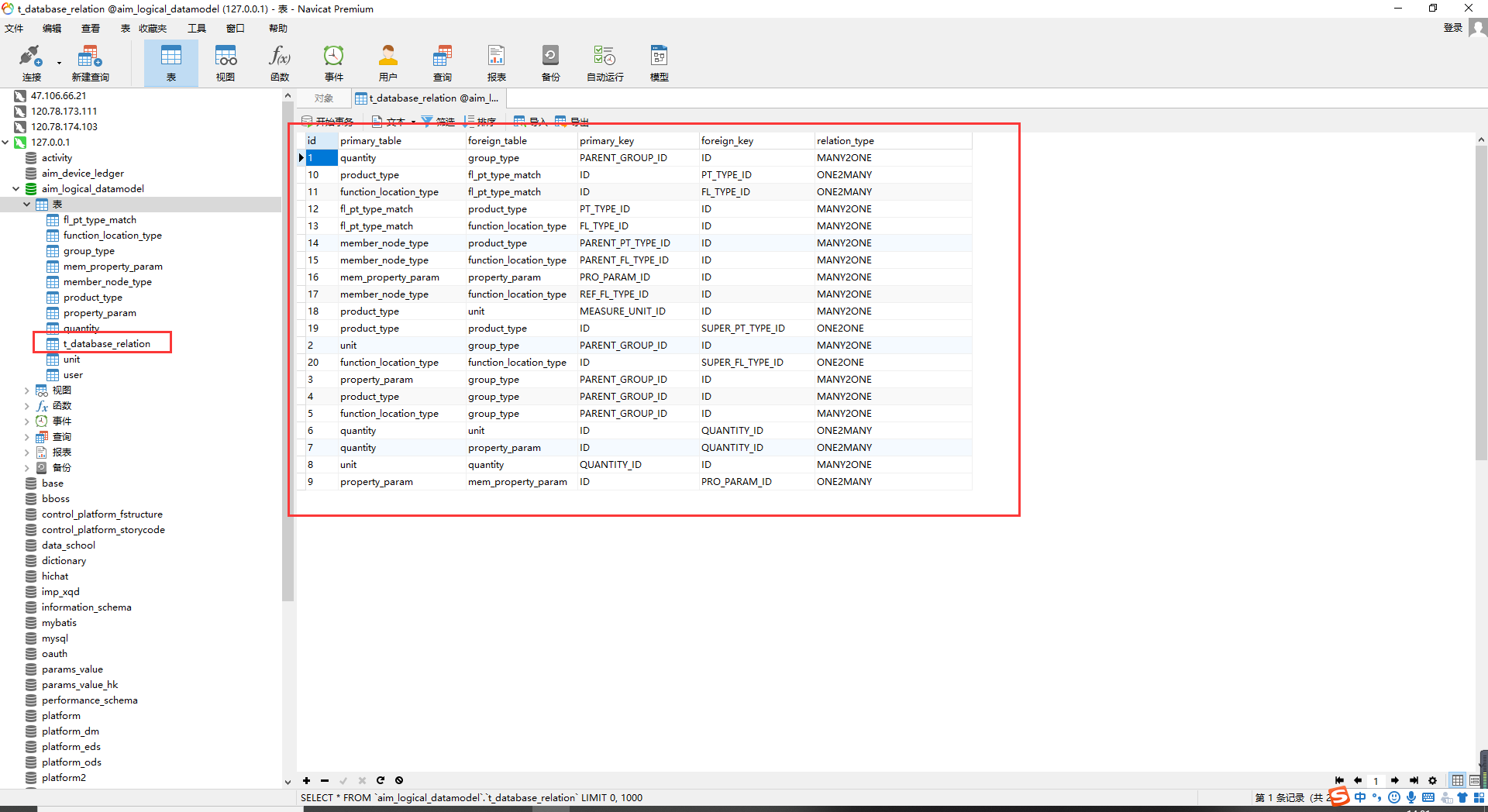
# 找到generator.yaml文件可以定义要连接的数据库信息及代码存储路径



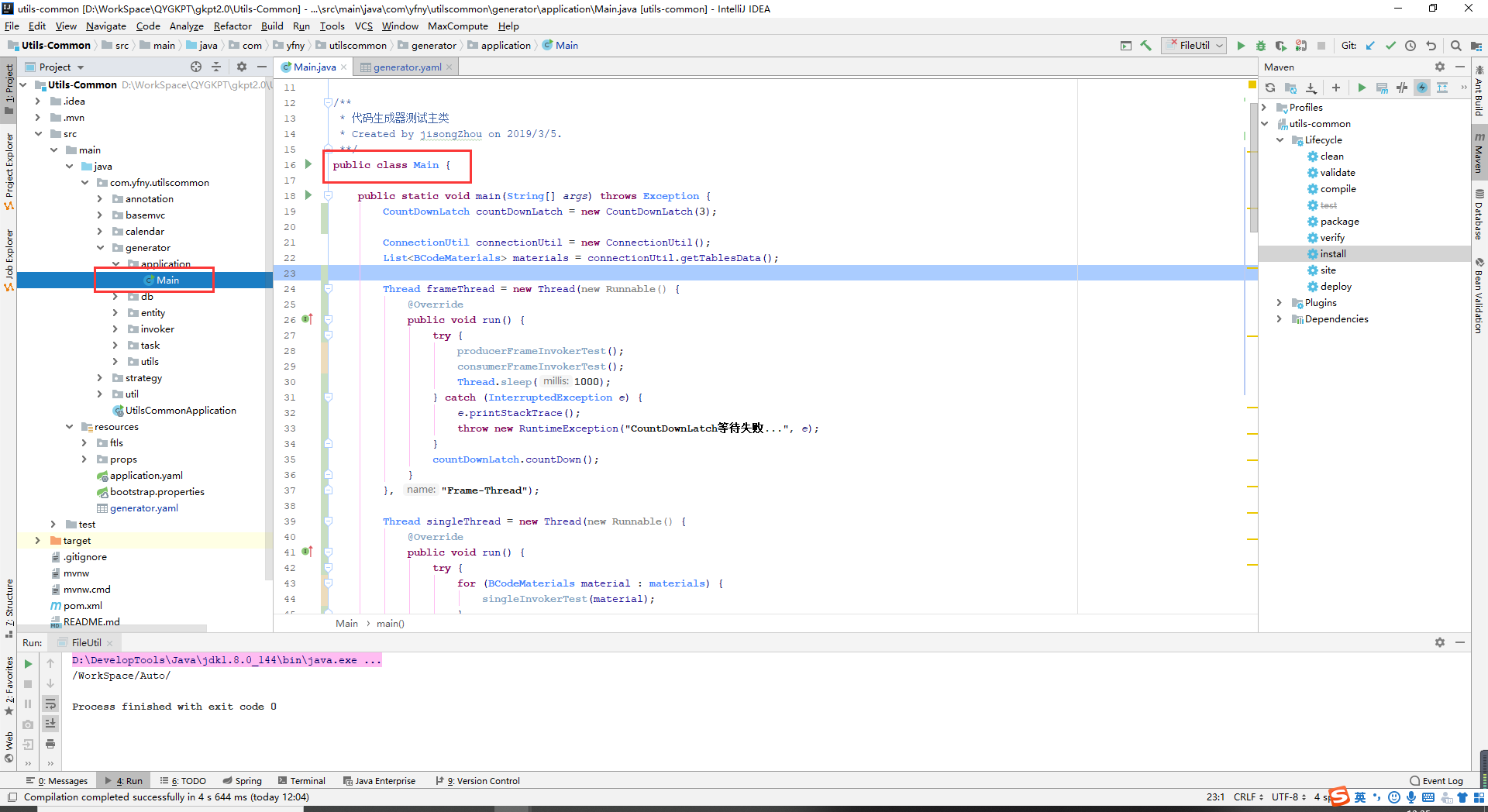
# 如果表间有关联键联系需要在库中添加 t\_database\_relation 表，设定关联键关系



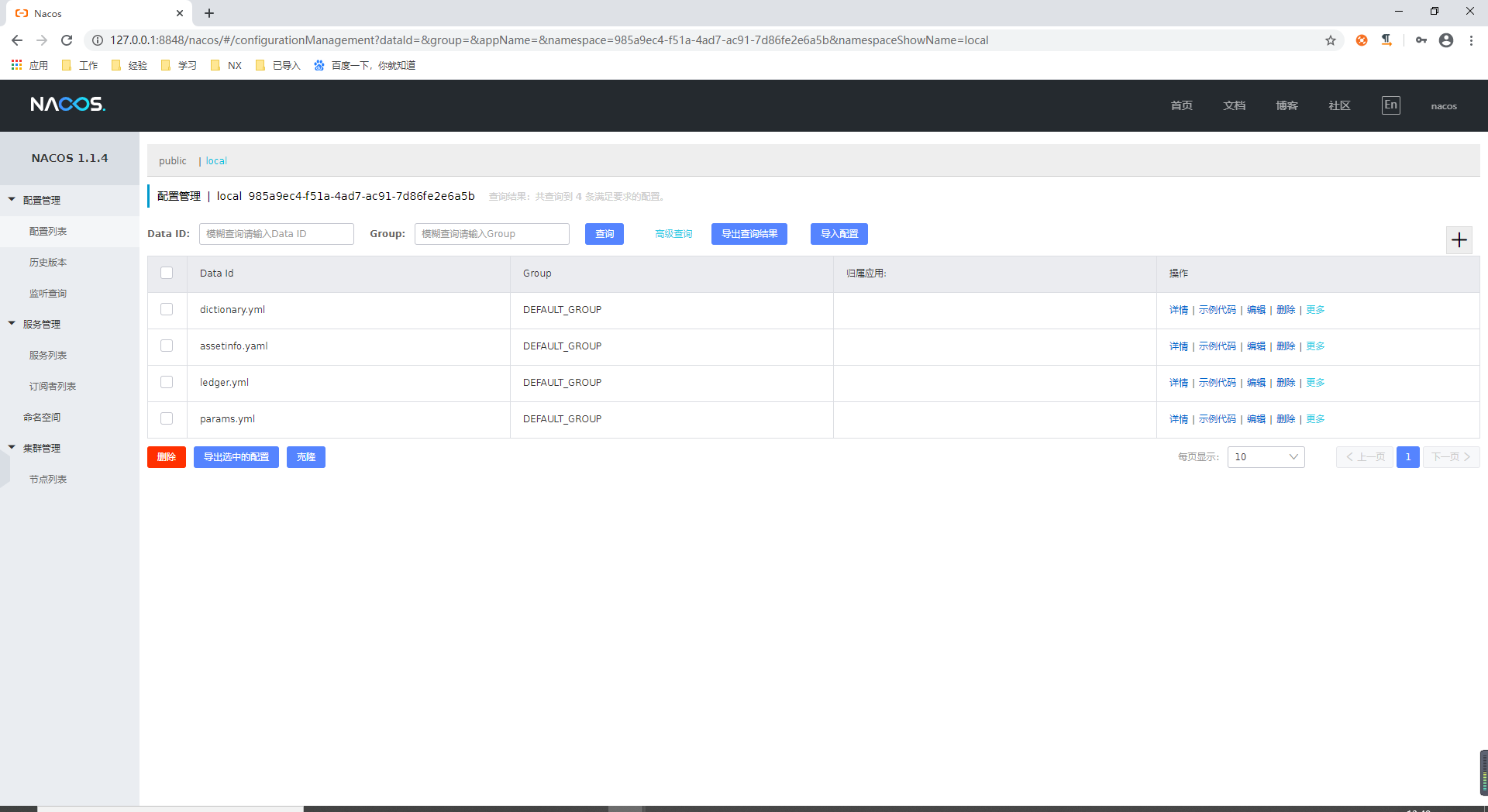
relation\_type 分为三种类型 ONE2ONE、ONE2MANY、MANY2MANY



# 找到Main.java类为代码生成运行主类，执行 main() 方法即可在配置路径下生成代码

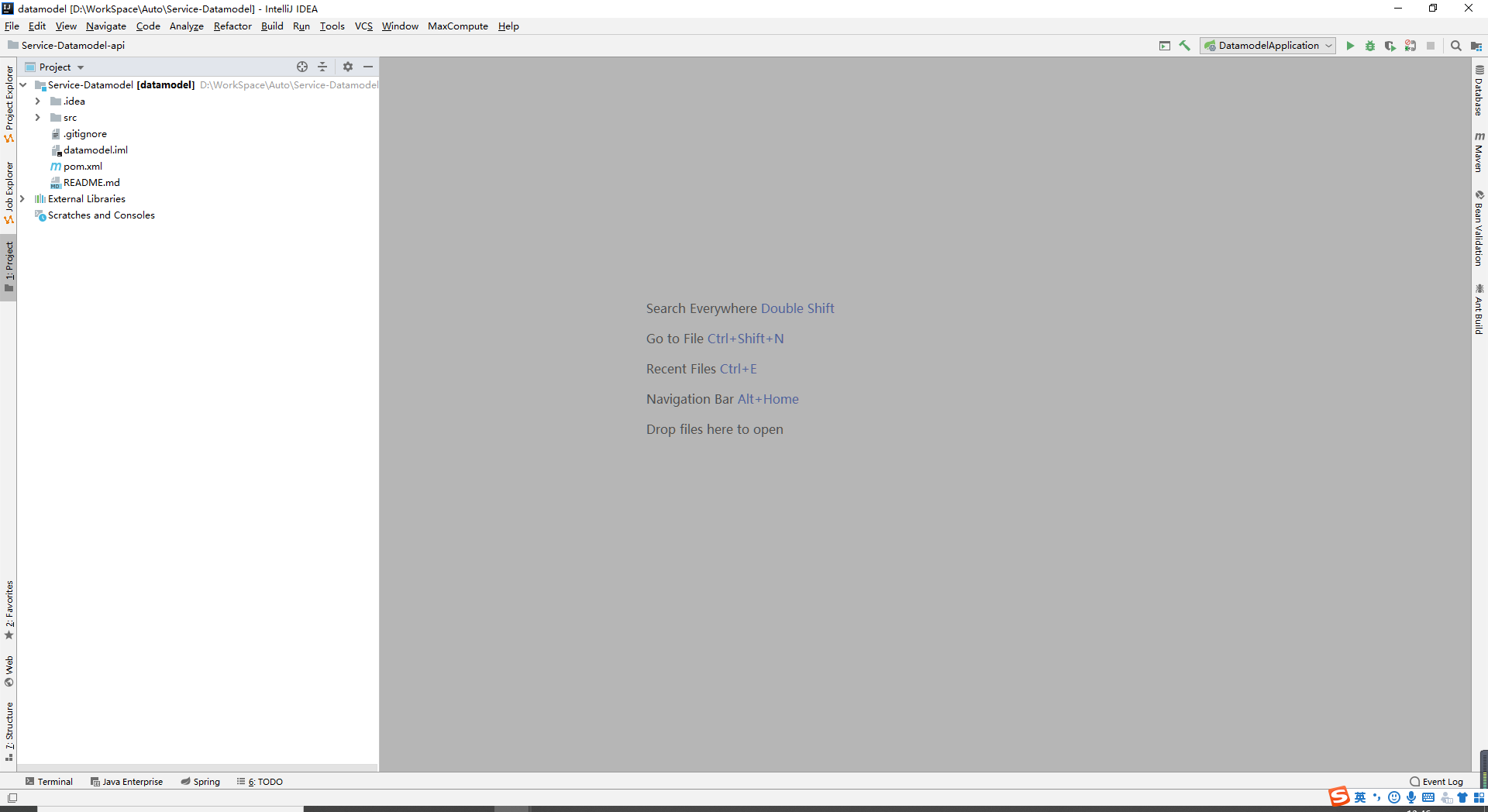


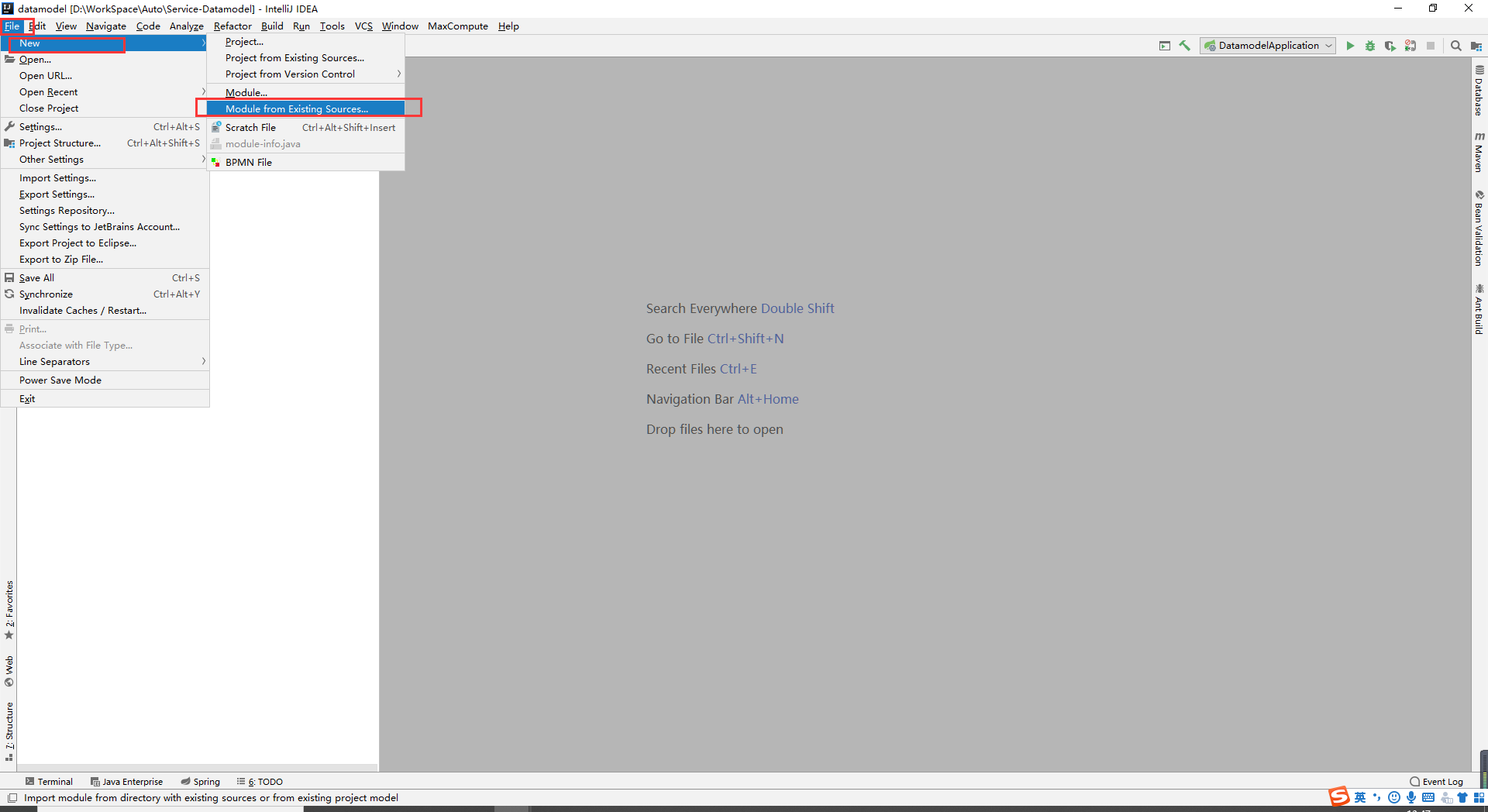
# 准备Nacos 服务管理

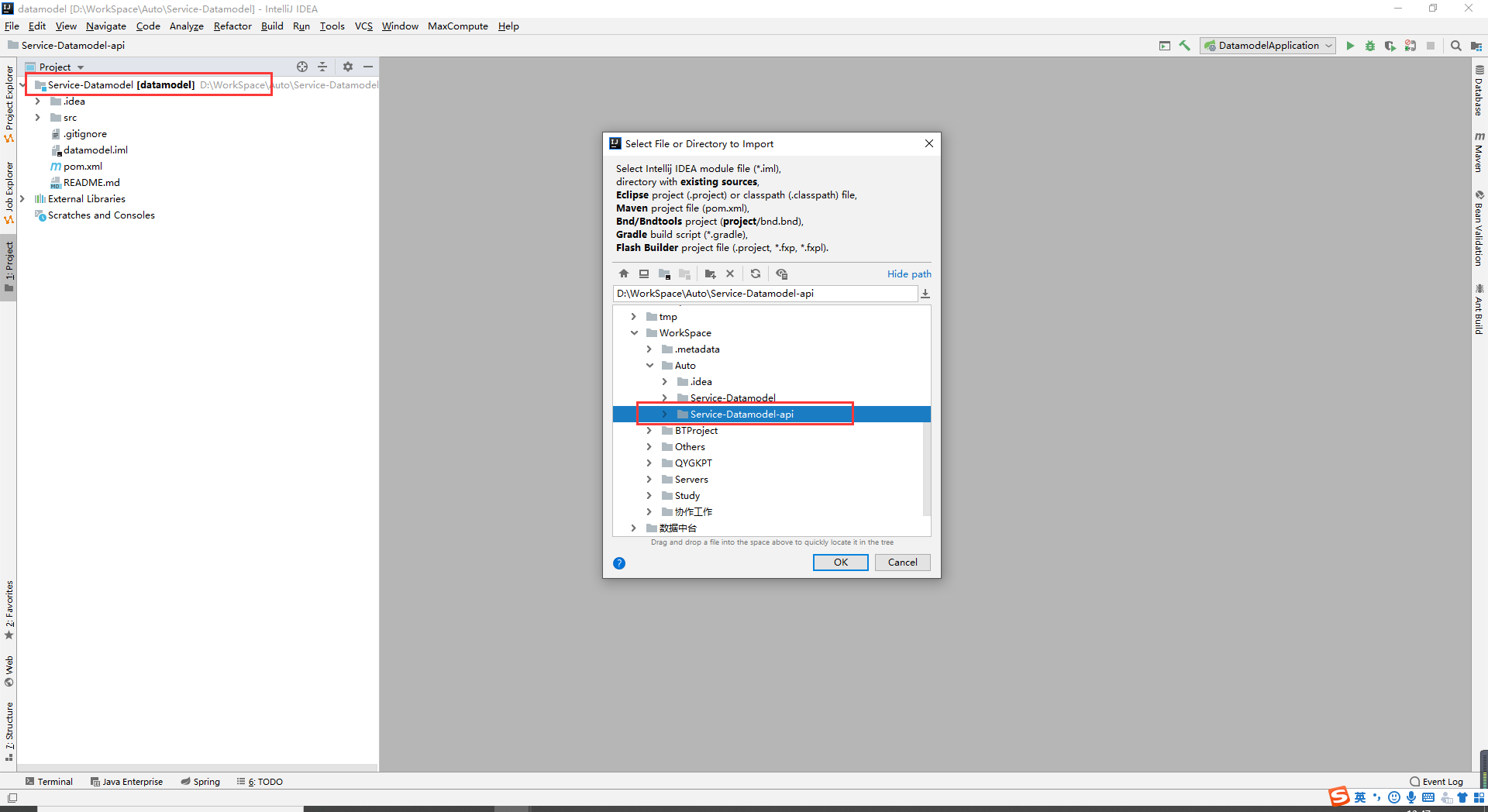


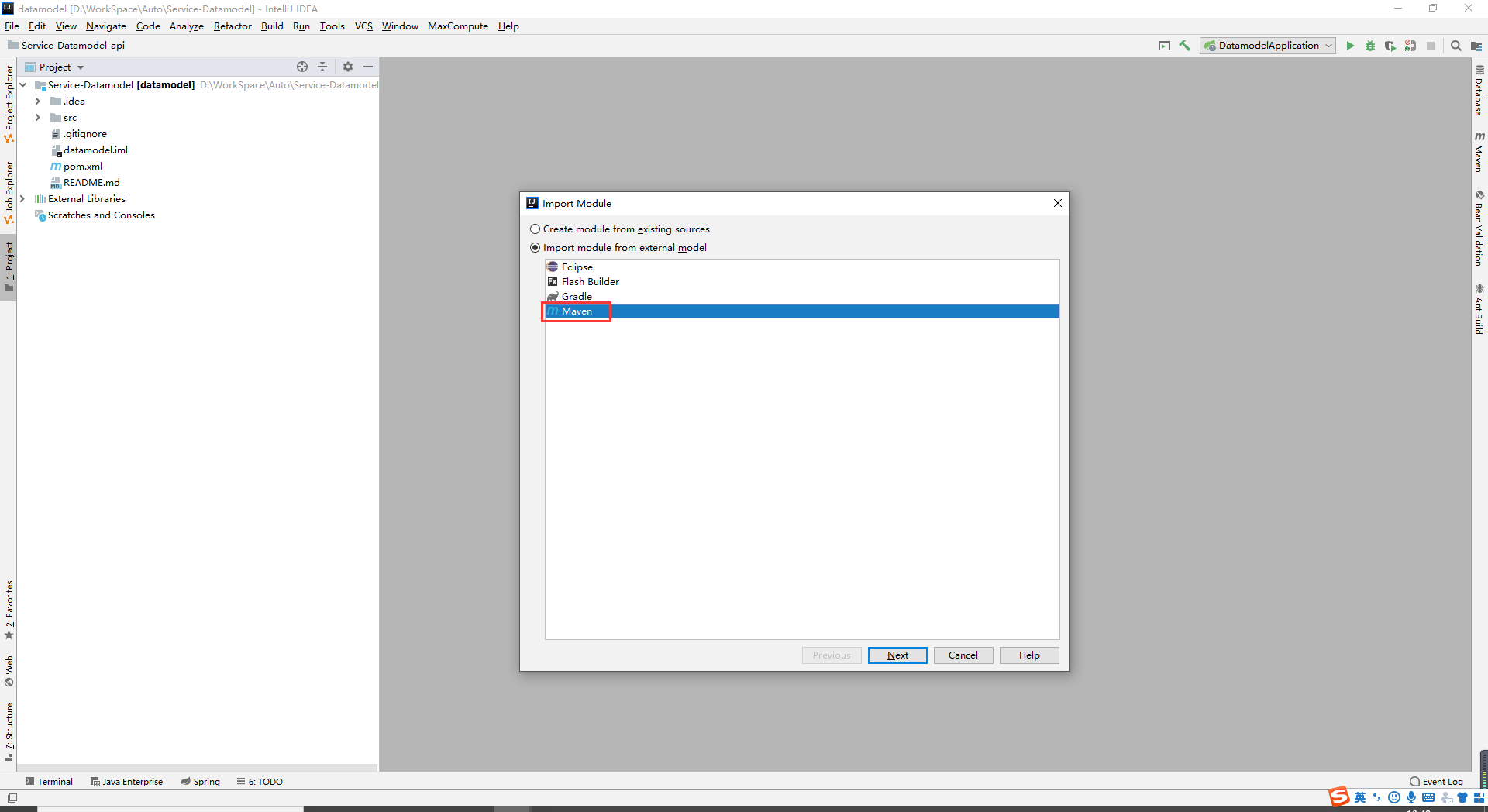
可以通过 <https://nacos.io/zh-cn/docs/quick-start.html> 了解 Nacos 的使用方法

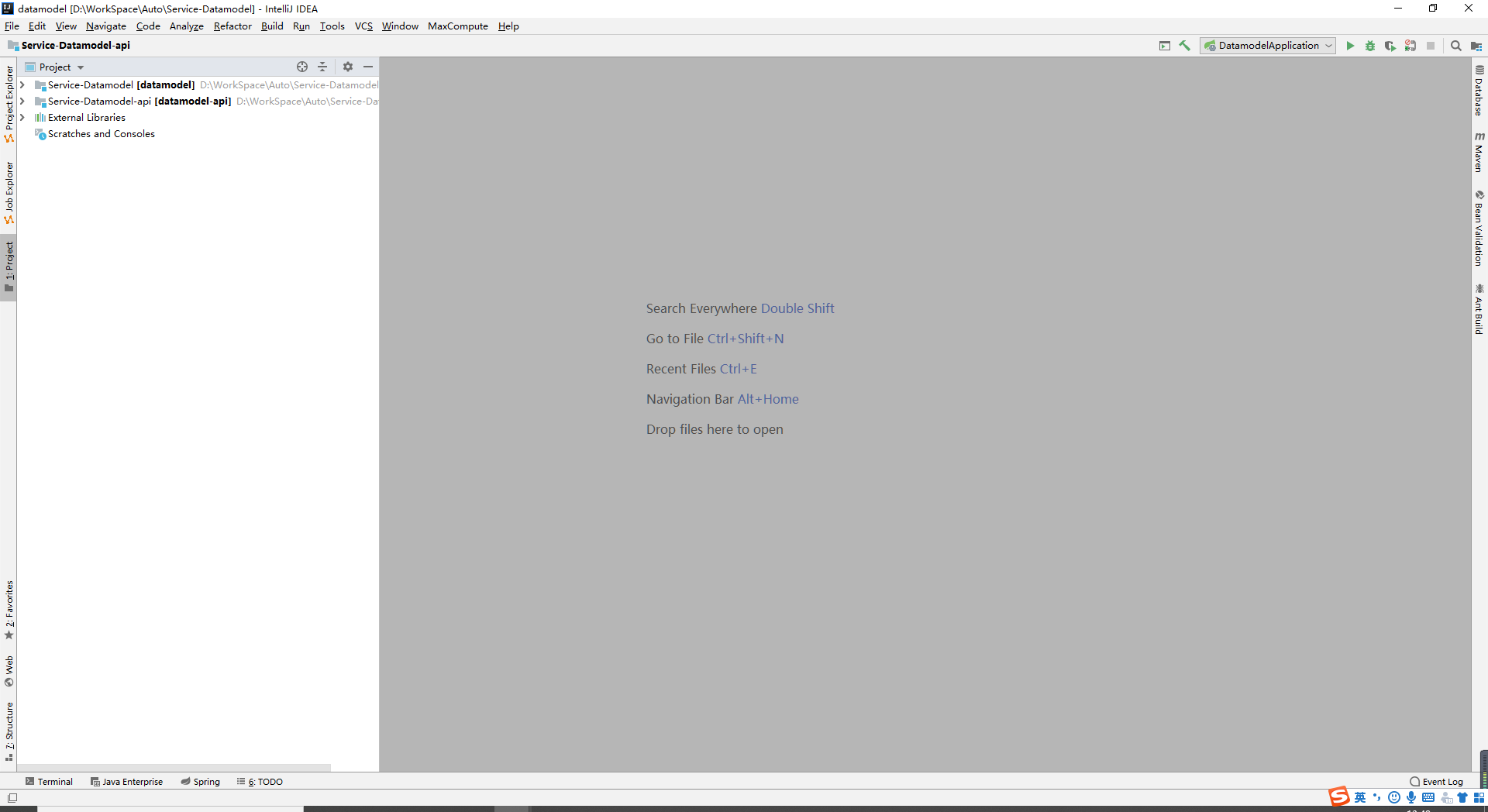
# 选择生成的主Service代码编辑器打开并添加依赖模块



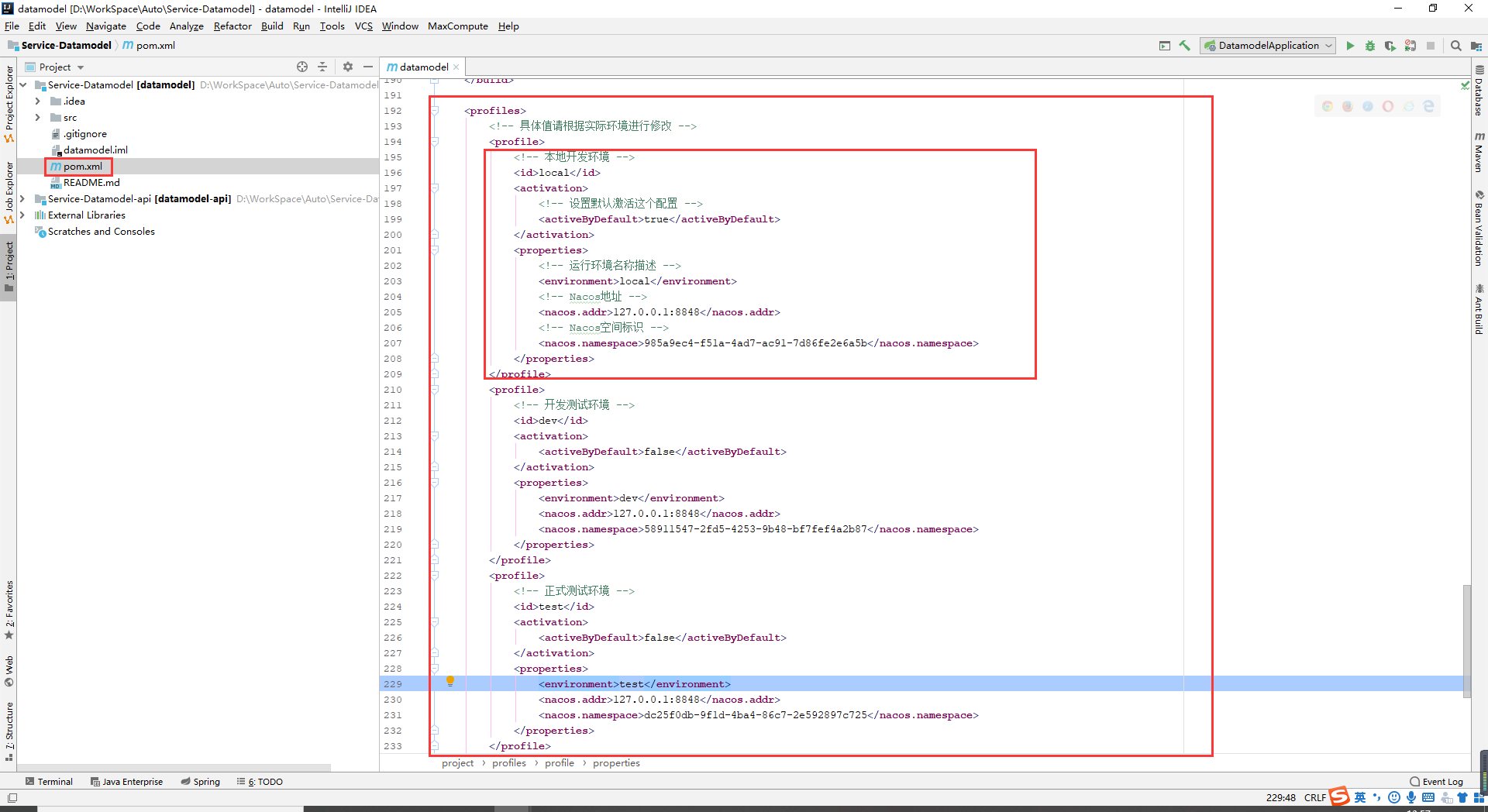




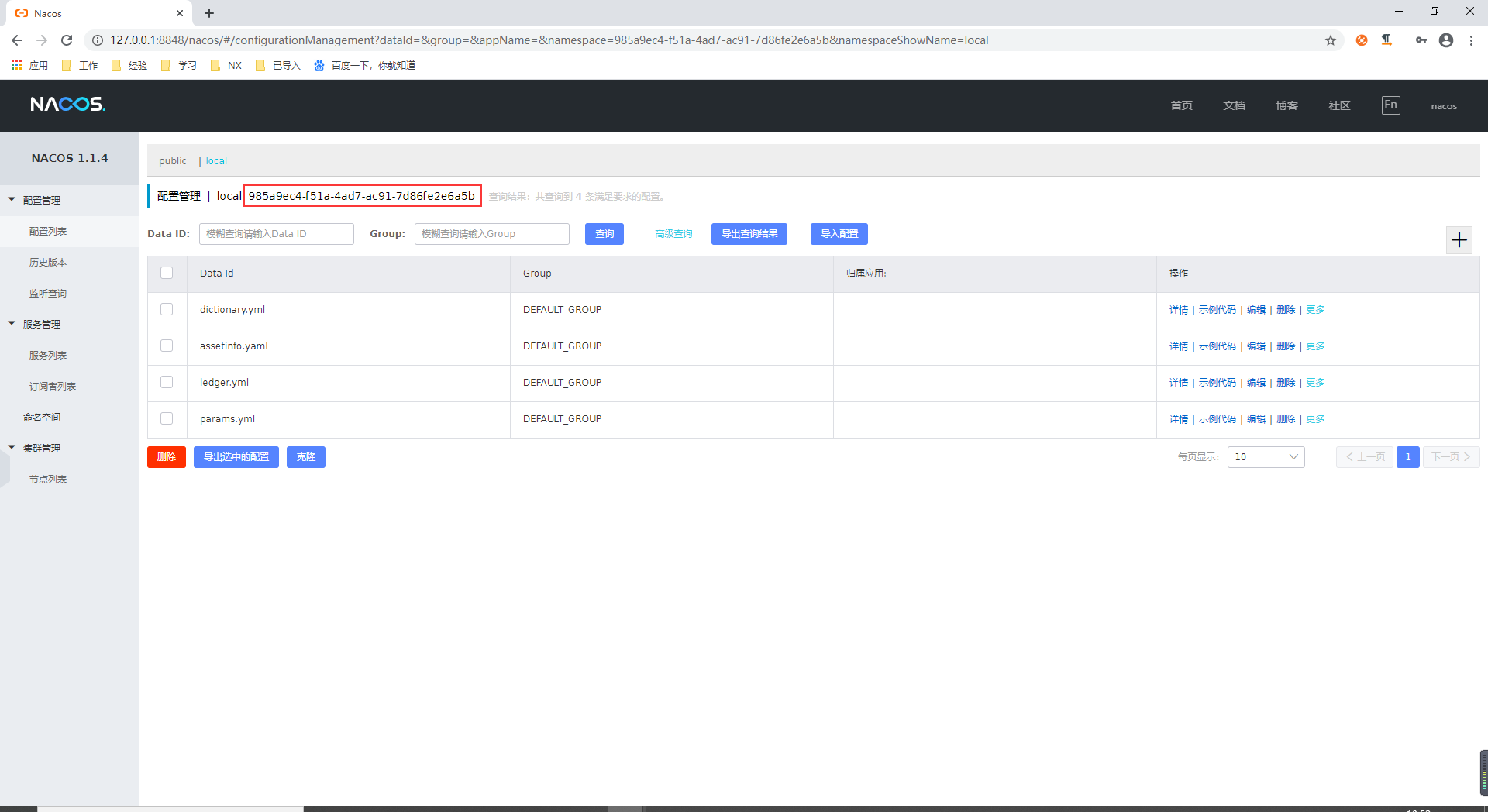




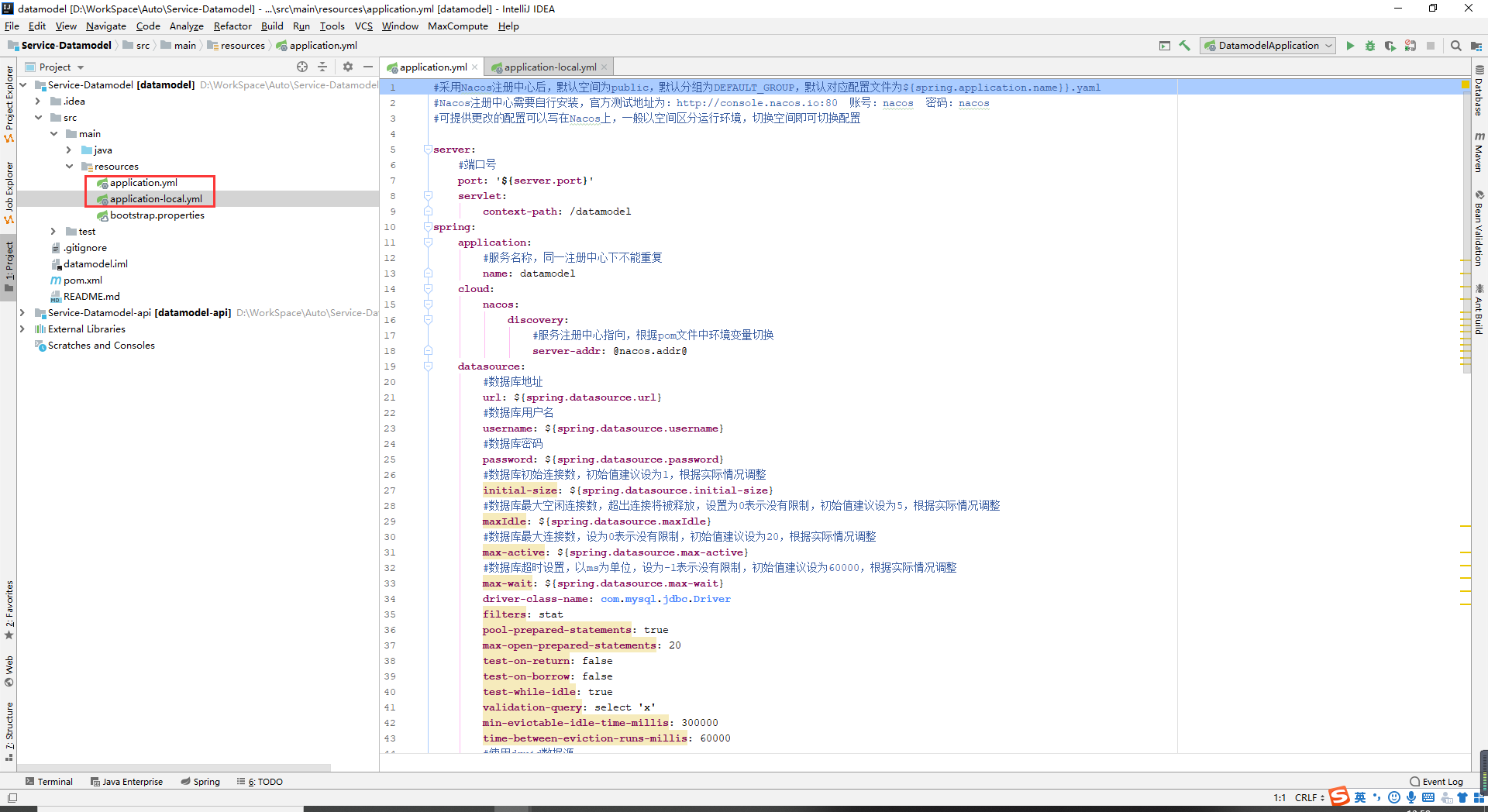
# 根据自身的环境配置修改Nacos 属性

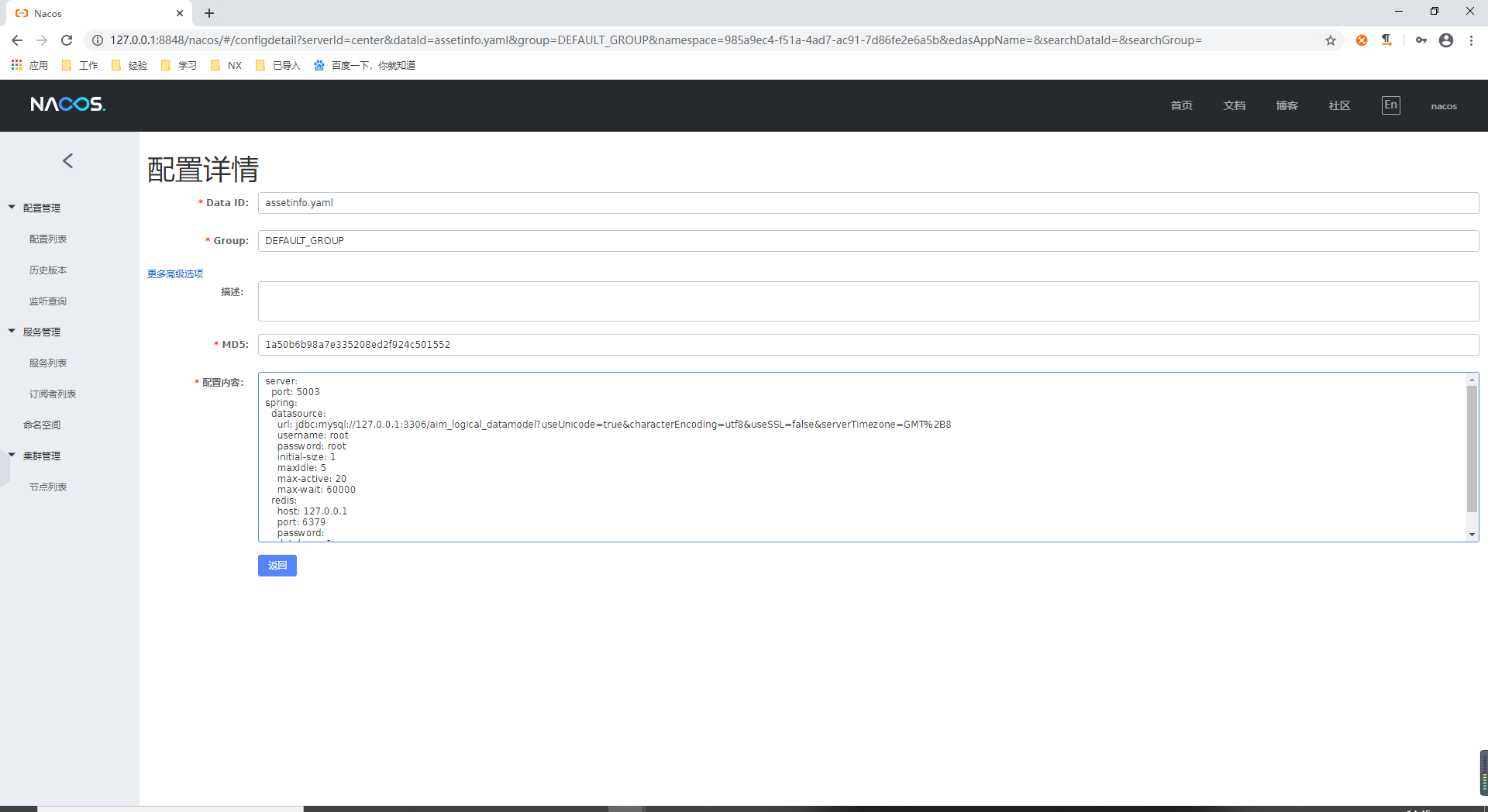


namespace 标识需要从 Nacos 中获取

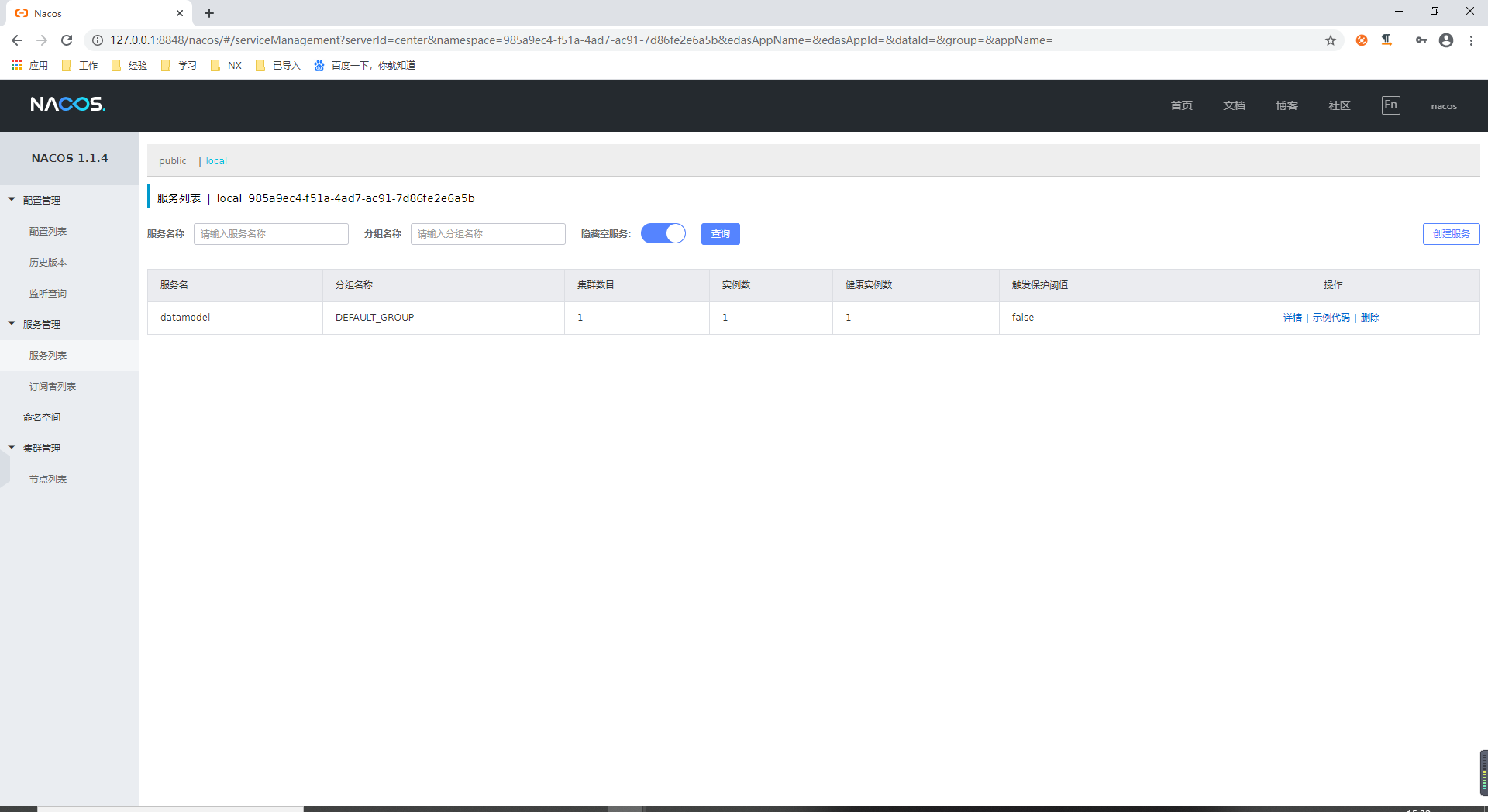


# 最好将配置注册到 Nacos 上，也可以使用 local 配置



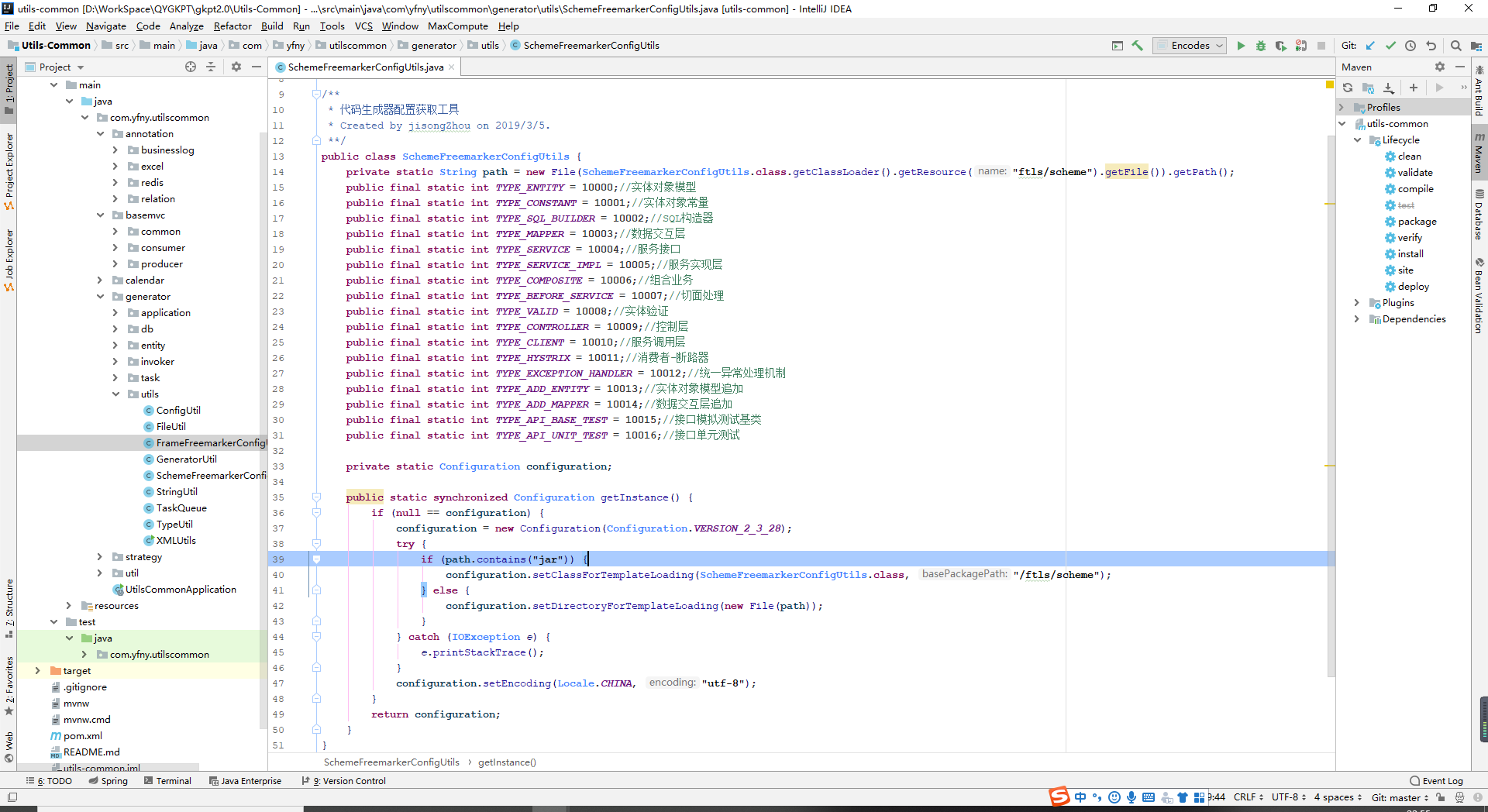


# 启动Service服务



启动成功后会在服务列表中看到启动的服务

# 代码层级说明



# 提供的接口

格式：[http://[ip]:[port]/[service\_name]/[model\_name]/[action](http://[ip]:[port]/%5bservice_name%5d/%5bmodel_name%5d/%5baction)]

可在postman中获取具体数据响应

## 新增接口

|  |  |
| --- | --- |
| 接口名称 | 新增接口 |
| 接口路径 | [http://[ip]:[port]/[service\_name]/[model\_name]/insertSelective](http://[ip]:[port]/%5bservice_name%5d/%5bmodel_name%5d/insertSelective) |
| 接口描述 | 保存一个实体，null的属性不会保存，会使用数据库默认值 |
| 接口方式 | Post |
| 参数说明 | 对象实体 |
| 返回说明 | 返回0为失败，返回1为成功 |
| 接口参数 | {  "属性1":"属性值1",  "属性2":"属性值2",  ……  "属性n":"属性值n"  } |
| 响应数据 | {  "code": "sys.request.success",  "message": "创建成功",  "data": "1"  } |
| 响应数据说明 | 【code】：返回状态码（可通过配置中心配置）  【message】：返回信息提示（可通过配置中心配置）  【data】：返回数据 |

## 更新接口

|  |  |
| --- | --- |
| 接口名称 | 更新接口 |
| 接口路径 | [http://[ip]:[port]/[service\_name]/[model\_name]/updateSelective](http://[ip]:[port]/%5bservice_name%5d/%5bmodel_name%5d/updateSelective) |
| 接口描述 | 根据主键更新属性不为null的值 |
| 接口方式 | Post |
| 参数说明 | 对象实体 |
| 返回说明 | 返回0为失败，返回1为成功 |
| 接口参数 | {  "属性1":"属性值1",  "属性2":"属性值2",  ……  "属性n":"属性值n"  } |
| 响应数据 | {  "code": "sys.request.success",  "message": "修改成功",  "data": "1"  } |
| 响应数据说明 | 【code】：返回状态码（可通过配置中心配置）  【message】：返回信息提示（可通过配置中心配置）  【data】：返回数据 |

## 删除接口

|  |  |
| --- | --- |
| 接口名称 | 删除接口 |
| 接口路径 | [http://[ip]:[port]/[service\_name]/[model\_name]/delete](http://[ip]:[port]/%5bservice_name%5d/%5bmodel_name%5d/delete) |
| 接口描述 | 根据实体属性作为条件进行删除，查询条件使用等号 |
| 接口方式 | Post |
| 参数说明 | 对象实体 |
| 返回说明 | 返回0为失败，返回1为成功 |
| 接口参数 | {  "属性1":"属性值1",  "属性2":"属性值2",  ……  "属性n":"属性值n"  } |
| 响应数据 | {  "code": "sys.request.success",  "message": "删除成功",  "data": "1"  } |
| 响应数据说明 | 【code】：返回状态码（可通过配置中心配置）  【message】：返回信息提示（可通过配置中心配置）  【data】：返回数据 |

## 常用字段说明

|  |  |
| --- | --- |
| 字段名 | 字段含义 |
| 【action】 | 执行动作【1-插入，2-更新，3-删除，4-查询】 |
| 【pageNum】 | 当前页数 |
| 【pageSize】 | 每页数量 |
| 【pageCount】 | 总页数 |
| 【total】 | 总数 |
| 【groupBy】 | 分组字段 |
| 【complexRate】 | 列表条件复杂度指定，默认为NORMAL【NORMAL-正常，COMPLEX-复杂】 |
| 【treeConfig】 | 树形结构指定 |
| 【treeConfig-id】 | 标识字段定义 |
| 【treeConfig-name】 | 名称字段定义 |
| 【treeConfig-parentId】 | 父级标识字段定义 |
| 【treeConfig-level】 | 层级字段定义 |
| 【treeConfig-rootId】 | 根节点 |
| 【treeConfig-leaf】 | 是否叶子节点 |
| 【treeConfig-orderBy】 | 排序依据 |
| 【treeConfig-haveList】 | 是否返回列表 |
| 【treeConfig-extraProps】 | 额外属性名列表 |
| 【queries】 | 查询字段及匹配方式 |
| 【queries-queryBy】 | 查询字段名称 |
| 【queries-queryType】 | 查询方式，默认为EQUAL【EQUAL-等于，UNEQUAL-不等于，LIKE-模糊】 |
| 【queries-logical】 | 列表条件逻辑指定，默认为AND【AND-逻辑与，OR-逻辑或】 |
| 【orders】 | 排序字段及排序方式（升序，降序） |
| 【orders-orderBy】 | 排序字段名称 |
| 【orders-orderSort】 | 排序方式【ASC-升序，DESC-降序】 |
| 【numScopes】 | 数字范围指定 |
| 【numScopes-name】 | 数字字段名称 |
| 【numScopes-start】 | 起始值 |
| 【numScopes-end】 | 结束值 |
| 【timeScopes】 | 时间范围指定 |
| 【timeScopes-name】 | 时间字段名称 |
| 【timeScopes-start】 | 起始时间 |
| 【timeScopes-end】 | 结束时间 |
| 【param1】 | 备用参数1 |
| 【param2】 | 备用参数2 |
| 【param3】 | 备用参数3 |

## 对象查询接口

|  |  |
| --- | --- |
| 接口名称 | 对象查询接口 |
| 接口路径 | [http://[ip]:[port]/[service\_name]/[model\_name]/selectOne](http://[ip]:[port]/%5bservice_name%5d/%5bmodel_name%5d/selectOne) |
| 接口描述 | 根据实体中的属性进行查询，只能有一个返回值，有多个结果是抛出异常，查询条件使用等号 |
| 接口方式 | Post |
| 参数说明 | 对象实体 |
| 返回说明 | 返回null为未查询到结果，返回对象为查询结果，返回多个结果则抛出异常 |
| 接口参数 | {  "属性1":"属性值1",  "属性2":"属性值2",  ……  "属性n":"属性值n"  } |
| 响应数据 | {  "code": "sys.request.success",  "message": "查看成功",  "data": {  "action": 4,  "pageNum": 1,  "pageSize": 20,  "pageCount": null,  "total": null,  "groupBy": null,  "complexRate": "NORMAL",  "treeConfig": {  "id": "id",  "name": "name",  "parentId": "parentId",  "level": null,  "rootId": null,  "leaf": "true",  "orderBy": null,  "haveList": "false",  "extraProps": []  },  "queries": null,  "orders": null,  "numScopes": null,  "timeScopes": null,  "param1": null,  "param2": null,  "param3": null,  "属性1":"属性值1",  "属性2":"属性值2",  ……  "属性n":"属性值n"  }  } |
| 响应数据说明 | 【code】：返回状态码（可通过配置中心配置）  【message】：返回信息提示（可通过配置中心配置）  【data】：返回数据 |

## 复杂对象查询接口

|  |  |
| --- | --- |
| 接口名称 | 复杂对象查询接口 |
| 接口路径 | [http://[ip]:[port]/[service\_name]/[model\_name]/selectComplexById](http://[ip]:[port]/%5bservice_name%5d/%5bmodel_name%5d/selectComplexById) |
| 接口描述 | 根据主键字段进行查询复合嵌套的整体对象（如有），方法参数必须包含完整的主键属性，查询条件使用等号 |
| 接口方式 | Post |
| 参数说明 | 主键 |
| 返回说明 | 返回null为未查询到结果，返回对象为查询结果 |
| 接口参数 | "id": "1" |
| 响应数据 | 失败  {  "code": "sys.custom.error",  "message": "[未检测到该数据的复合实现，请更换接口！]",  "data": true  }  成功  {  "code": "sys.request.success",  "message": "查看成功",  "data": {  "action": 4,  "pageNum": 1,  "pageSize": 20,  "pageCount": null,  "total": null,  "groupBy": null,  "complexRate": "NORMAL",  "treeConfig": {  "id": "id",  "name": "name",  "parentId": "parentId",  "level": null,  "rootId": null,  "leaf": "true",  "orderBy": null,  "haveList": "false",  "extraProps": []  },  "queries": null,  "orders": null,  "numScopes": null,  "timeScopes": null,  "param1": null,  "param2": null,  "param3": null,  "属性1":"属性值1",  "属性2":"属性值2",  ……  "属性n":"属性值n"  }  } |
| 响应数据说明 | 【code】：返回状态码（可通过配置中心配置）  【message】：返回信息提示（可通过配置中心配置）  【data】：返回数据 |

## 结果数量查询接口

|  |  |
| --- | --- |
| 接口名称 | 结果数量查询接口 |
| 接口路径 | [http://[ip]:[port]/[service\_name]/[model\_name]/selectCount](http://[ip]:[port]/%5bservice_name%5d/%5bmodel_name%5d/selectCount) |
| 接口描述 | 根据实体中的属性查询总数，查询条件使用等号 |
| 接口方式 | Post |
| 参数说明 | 对象实体 |
| 返回说明 | 返回查询结果数量 |
| 接口参数 | {  "属性1":"属性值1",  "属性2":"属性值2",  ……  "属性n":"属性值n"  } |
| 响应数据 | {  "code": "sys.request.success",  "message": "请求成功",  "data": 100  } |
| 响应数据说明 | 【code】：返回状态码（可通过配置中心配置）  【message】：返回信息提示（可通过配置中心配置）  【data】：返回数据 |

## 列表查询接口

|  |  |
| --- | --- |
| 接口名称 | 分组维度查询接口 |
| 接口路径 | http://[ip]:[port]/[service\_name]/[model\_name]/findList/[pageNum]/[pageSize] |
| 接口描述 | 根据实体中的属性值进行查询，查询条件使用等号，分页返回 |
| 接口方式 | Post |
| 参数说明 | 对象实体，页数（选填），每页数量（选填） |
| 返回说明 | 返回对象列表为查询结果 |
| 接口参数 | {  "属性1":"属性值1",  "属性2":"属性值2",  ……  "属性n":"属性值n"  } |
| 响应数据 | {  "code": "sys.request.success",  "message": "列表加载成功",  "data": [  {……},  {……},  ……  {……}  ]  } |
| 响应数据说明 | 【code】：返回状态码（可通过配置中心配置）  【message】：返回信息提示（可通过配置中心配置）  【data】：返回数据 |

## 分组维度查询接口

|  |  |
| --- | --- |
| 接口名称 | 列表查询接口 |
| 接口路径 | http://[ip]:[port]/[service\_name]/[model\_name]/findGroupBy |
| 接口描述 | 定义实体中分组维度，并返回分组 |
| 接口方式 | Post |
| 参数说明 | 对象实体 |
| 返回说明 | 返回对象分组为查询结果 |
| 接口参数 | {  "groupBy":"分组字段属性名称"  } |
| 响应数据 | {  "code": "sys.request.success",  "message": "分组加载成功",  "data": [  "分组1",  "分组2",  ……  "分组n"  ]  } |
| 响应数据说明 | 【code】：返回状态码（可通过配置中心配置）  【message】：返回信息提示（可通过配置中心配置）  【data】：返回数据 |

## 分组数据查询接口

|  |  |
| --- | --- |
| 接口名称 | 分组数据查询接口 |
| 接口路径 | [http://[ip]:[port]/[service\_name]/[model\_name]/findMapGroupByCondition](http://[ip]:[port]/%5bservice_name%5d/%5bmodel_name%5d/findMapGroupByCondition) |
| 接口描述 | 根据实体中的属性值进行查询，查询条件使用LIKE，并列查询取交集，分组返回 |
| 接口方式 | Post |
| 参数说明 | 对象实体 |
| 返回说明 | 返回分组对象列表为查询结果 |
| 接口参数 | {  "groupBy":"分组字段属性名称"  } |
| 响应数据 | {  "code": "sys.request.success",  "message": "列表加载成功",  "data": {  "分组1":[  {……},  {……},  ……  {……}  ],  "分组2":[……],  ……  "分组n":[……]  }  } |
| 响应数据说明 | 【code】：返回状态码（可通过配置中心配置）  【message】：返回信息提示（可通过配置中心配置）  【data】：返回数据 |

## 树结构查询接口

|  |  |
| --- | --- |
| 接口名称 | 树结构查询接口 |
| 接口路径 | http://[ip]:[port]/[service\_name]/[model\_name]/getTreeOf |
| 接口描述 | 根据实体中的属性值进行查询，查询条件使用LIKE，并列查询取交集，树形结构返回 |
| 接口方式 | Post |
| 参数说明 | 对象实体 |
| 返回说明 | 返回树形结构对象列表为查询结果 |
| 接口参数 | {  "treeConfig": {  "id":"id",  "name":"flName",  "parentId":"parentId",  "level":"1",  "rootId":"07010050143081",  "leaf":"true",  "orderBy":" flName ",  "haveList":"false"  }  } |
| 响应数据 | {  "code": "sys.request.success",  "message": "数结构加载成功",  "data": {  "tree":"json结构树形数据"  "list":[]  }  } |
| 响应数据说明 | 【code】：返回状态码（可通过配置中心配置）  【message】：返回信息提示（可通过配置中心配置）  【data】：返回数据 |