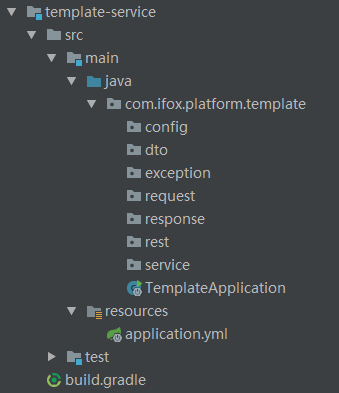
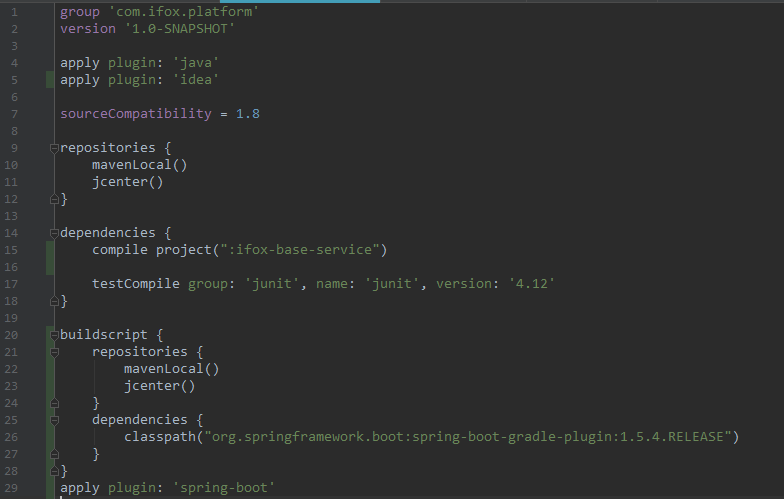
一、项目结构

1、新建一个项目，至少应该包含如下结构：



请参考template-service项目

2、build.gradle规范



请参考template-service/build.gradle

3、包的说明以及约定

config包存放当前service相关的配置类

dto包存放数据传输对象，命名规范xxxDTO，必须覆写toString方法

exception包存放当前service相关的异常类，命名规范xxxException，必须继承ApplicationException并覆写父类构造方法，必须定义异常状态码expStatus在ifox-common 项目的ExceptionStatusConstant类中（异常状态码依次排列编号）

request包定义当前service的所有请求对象，命名规范xxxRequest，必须覆写toString方法

response包定义当前service的所有响应对象，命名规范xxxVO，必须覆写toString方法

service包定义当前service的业务处理对象

rest包定义当前service 的所有接口请求，命名规范xxxController，必须在类层面定义RequestMapping，必须定义headers = "api-version=1.0" ，必须定义swagger2相关注解，包括@Api, @ApiOperation, @ApiParam, @ApiResponses，必须打印request和response日志以及异常日志，请参考AdminUserController

xxxApplication类启动当前service

application.yml定义当前service的所有配置信息

二、约定

1、每一个类，接口，方法上必须有相应的注释

2、controller层必须打印请求和响应的log，service层根据情况打印log

3、controller层必须返回VO对象，service层必须返回DTO对象，并且使用ModelMapper做属性拷贝

4、每个功能页面，必须引入三个js：

<script th:src="@{/dist/js/common/common.js}"></script>

<script th:src="@{/dist/js/common/main-header.js}"></script>

<script th:src="@{/dist/js/common/main-sidebar.js}"></script>

5、Table开发使用自定义的ifox-table.js

<script th:src="@{/dist/js/ifox-table/ifox-table.js}"></script>

6、swagger2注解使用必须保持一致：

Controller类上注解：@Api(tags = "控制器说明")

method上注解：@ApiOperation("接口说明")

request上注解：@ApiParam

response上注解：@ApiResponses({ @ApiResponse(code = 480, message = "用户名或者密码错误"), @ApiResponse(code = 500, message = "服务器错误") })

**PS:response的注解尤其重要，必须写全所有的返回值可能性**