**<https://www.cnblogs.com/mr-yang-localhost/p/8971327.html>**

**一、问题**

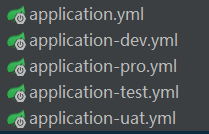
　　软件开发中经常有开发环境、测试环境、预发布环境、生产环境，而且一般这些环境配置会各不相同，手动改配置麻烦且容易出错，如何管理不同环境的配置参数呢？spring-boot + maven可以解决不同环境独立配置不同参数的问题。

**二、多环境配置**

不同环境的配置yml文件名不一样：

* application-dev.yml（开发环境）
* application-test.yml（测试环境）
* application-uat.yml（预发布）
* application-pro.yml（生产环境）

eg:



application-dev.yml配置示例：

[复制代码](javascript:void(0);)

info:

build:

name: ${project.artifactId}

groupId: ${project.groupId}

artifactId: ${project.artifactId}

version: ${project.version}

server:

port: 8081

endpoints:

enabled: true

sensitive: false

data:

test:

envName: dev

envconfig: 127.0.0.1:8081

[复制代码](javascript:void(0);)

application.yml

spring:

profiles:

active: dev

如果要切换不同环境，只需要修改spring.profiles.active即可。

读取配置参数：

[复制代码](javascript:void(0);)

@Setter

@Getter

@NoArgsConstructor

@AllArgsConstructor

@Component

@ConfigurationProperties(prefix = "data.test")

public class DataConfig {

private String envName;

private String envconfig;

}

[复制代码](javascript:void(0);)

 验证环境参数：

[复制代码](javascript:void(0);)

@Api("home controller")

@RestController

public class HomeController {

@Autowired

private DataConfig dataConfig;

@RequestMapping("/env")

@ApiOperation("env")

public Object testEnv() {

return dataConfig;

}

}

[复制代码](javascript:void(0);)

[回到顶部](https://www.cnblogs.com/mr-yang-localhost/p/8971327.html#_labelTop)

**三、设置环境**

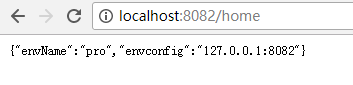
1、启动jar包时设置spring.profiles.active

java -jar muti-env-config.jar --spring.profiles.active=test

 2、maven打包时候设置环境(设置 pro 环境)

clean package -DskipTests -Ppro

pro环境示例：



demo参考地址：[muti-env-demo](https://github.com/yangyp8110/wasd-xxx/tree/master/springboot-demo-repository/muti-env-config)