

# Hello World

前后端分离框如何快速进入开发，请参照下面hello world实现demo

## 一、后台服务代码实现

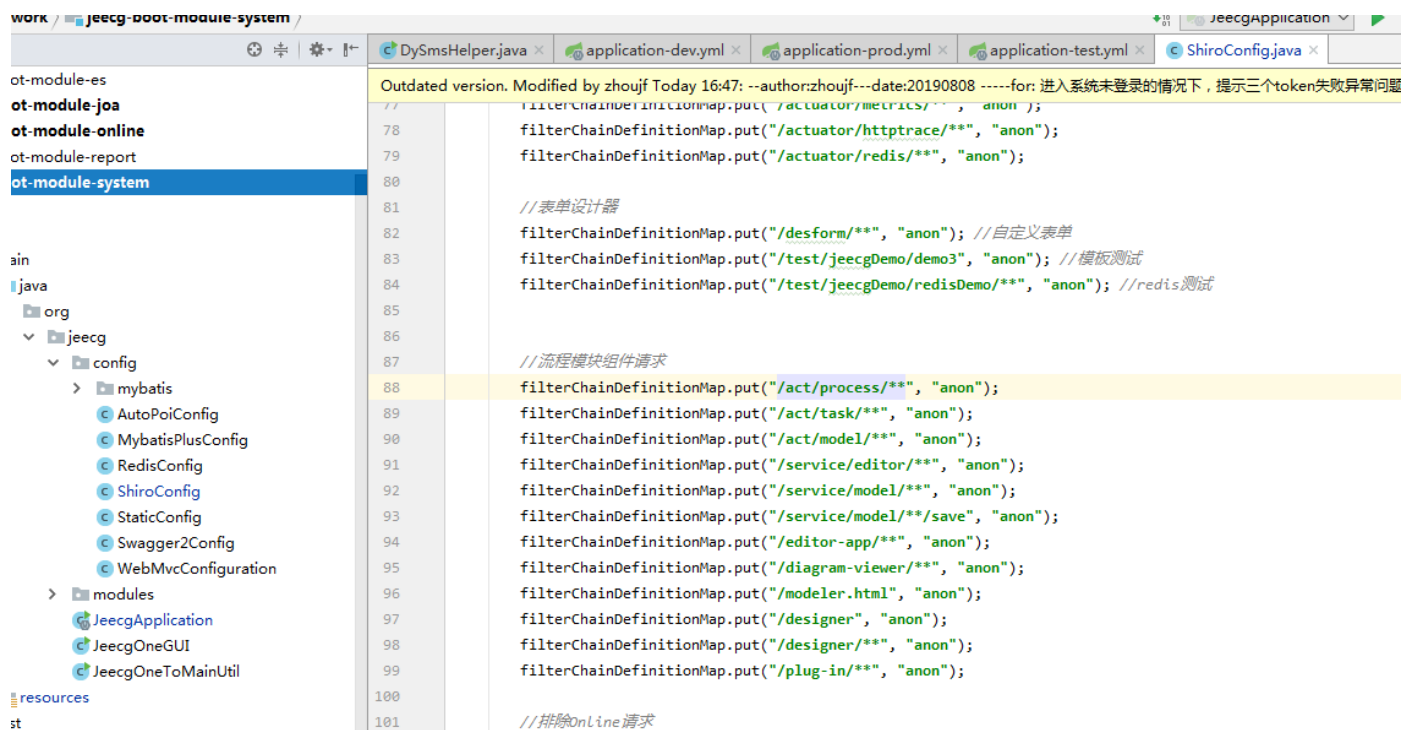
```
@RestController
@RequestMapping("/test/jeecgDemo")
@Slf4j
public class JeecgDemoController {

    /**
     * hello world
     *
     * @param id
     * @return
     */
    @GetMapping(value = "/hello")
    public Result<String> hello() {
        Result<String> result = new Result<String>();
        result.setResult("Hello World!");
        result.setSuccess(true);
        return result;
    }
}
```

访问请求http://localhost:8080/jeecg-boot/test/jeecgDemo/hello 会提示token无效，

所以需要配置下拦截器ShiroConfig排除。

配置文件： jeecg-boot-module-system/src/main/java/org/jeecg/config/ShiroConfig.java  
加入配置： filterChainDefinitionMap.put("/test/jeecgDemo/hello", "anon");



再访问http://localhost:8080/jeecg-boot/test/jeecgDemo/hello，会返回结果如下：

```
{
  "success": true,
  "message": null,
  "code": null,
  "result": "Hello World!",
}
```



"timestamp": 1548833208562

}



## 二、前台vue页面实现

(1) 创建vue页面src/views/jeecg/helloworld.vue

调用后台请求，获取返回的Hello World! 输出到页面，页面代码如下：

```
<template>
  <div>
    {{ msg }}
  </div>
</template>

<script>
  import {getAction} from '@/api/manage'
  export default {
    data () {
      return {
        msg: ""
      }
    },
    methods: {
      hello () {
        var url = "/test/jeecgDemo/hello"
        getAction(url).then((res) => {
          if (res.success) {
            this.msg = res.result;
          }
        })
      }
    },
    created() {
      this.hello();
    }
  }
</script>
```

代码说明：

1、data() 方法中定义数据对象msg

2、数据对象msg输出到页面，表达式如下：

```
{{ msg }}
```

3、定义一个方法，发起请求获取后台响应，后台实现的是get方法，引入getAction方法

```
import {getAction} from '@/api/manage'
```

定义方法调用：

```
hello () {
  var url = "/test/jeecgDemo/hello"
  getAction(url).then((res) => {
    if (res.success) {
      this.msg = res.result;
    }
  })
}
```

4、Vue生命周期 created 中调用方法

```
created() {
  this.hello();
}
```

hello方法中

this.msg = res.result;

把请求返回的Hello World! 赋值给msg数据对象，msg值改变则页面显示也改变。

三、配置菜单

1. 配置helloworld菜单【系统管理】-【菜单管理】

菜单类型：☐ 一级菜单 ☒ 子菜单 ☐ 按钮

\* 菜单名称：

helloworld

✓

上级菜单：

JEECG案例

▼

菜单路径：

/jeecg/helloworld

前端组件：

jeecg/helloworld

菜单图标：

请输入菜单图标

排序：

2

隐藏路由

☐

关闭

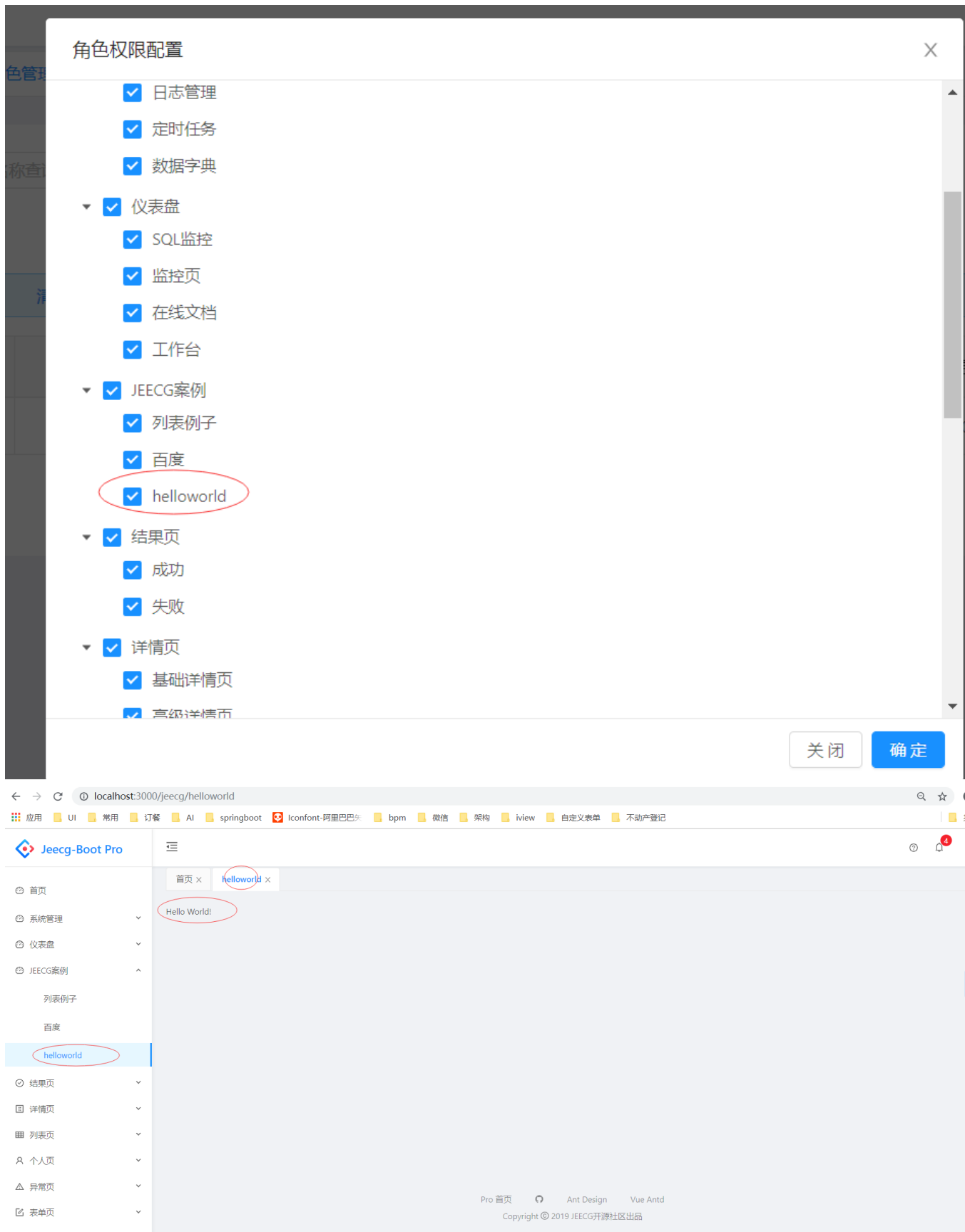
确定

- 其中前端组件配置相对src/views/目录下的 目录名+文件名
- 例如页面src/views/jeecg/helloworld.vue 前端组件配置 jeecg/helloworld

+ 新增

已选择 0项 清空

<input type="checkbox"/>	菜单名称	菜单类型	icon	组件	路径	排序	操作
<input type="checkbox"/>	首页	菜单	dashboard	dashboard/Analysis	/dashboard/analysis	1	编辑 更多 ▾
<input type="checkbox"/>	+ 系统管理	菜单	dashboard	layouts/RouteView	/isystem	8	编辑 更多 ▾
<input type="checkbox"/>	+ 仪表盘	菜单	dashboard	layouts/RouteView	/dashboard2	9	编辑 更多 ▾
<input type="checkbox"/>	- JEECG案例	菜单	dashboard	layouts/RouteView	/jeecg	9	编辑 更多 ▾
<input type="checkbox"/>	列表例子	菜单		jeecg/JeecgDemoList	/jeecg/JeecgDemoList	1	编辑 更多 ▾
<input type="checkbox"/>	百度	菜单		layouts/IframePageView	http://www.baidu.com	1	编辑 更多 ▾
<input type="checkbox"/>	helloworld	菜单		jeecg/helloworld	/jeecg/helloworld	2	编辑 更多 ▾
<input type="checkbox"/>	+ 结果页	菜单	check-circle-o	layouts/PageView	/result	9	编辑 更多 ▾
<input type="checkbox"/>	+ 详情页	菜单	profile	layouts/RouteView	/profile	9	编辑 更多 ▾



点击菜单访问页面展示Hello World!

上一篇: [快速开始](#)

下一篇: [上线发布](#)