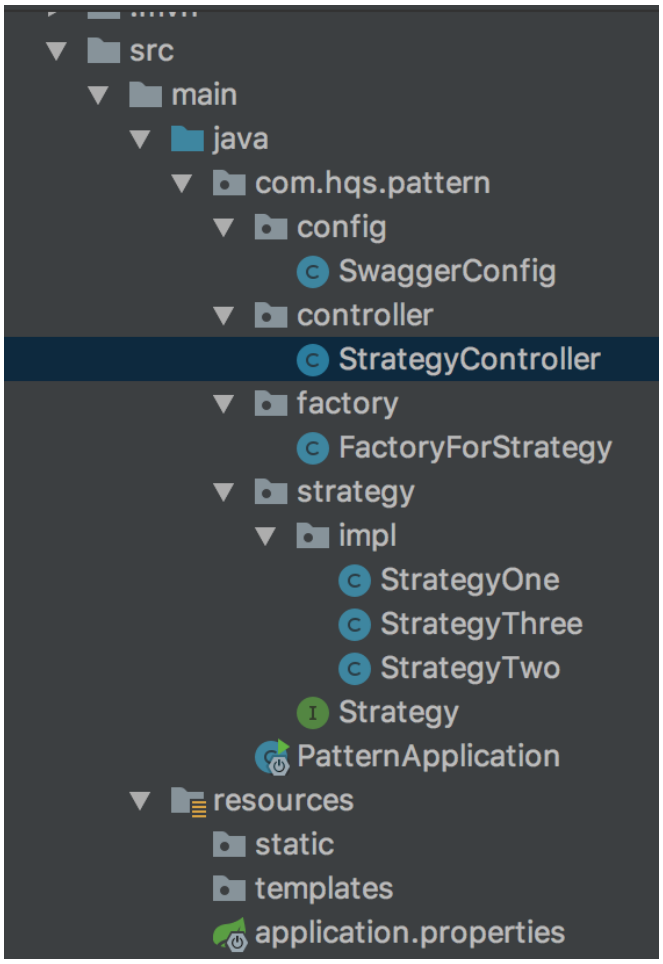
Spring 中实现策略模式+工厂模式

策略模式和工厂模式相信大家都比较熟悉，但是大家有没有在 spring 中实现策略和工厂模式?

具体策略模式和工厂模式的 UML 我就不给出来了，使用这个这两个模式主要是防止程序中出现大量的 IF ELSE IF ELSE....。接下来咱们直接实现，项目结构图：



工厂类 FactoryStrategy 负责创建策略的工厂，代码比较简单，比较关键的一点是 AutoWired 一个 Map<String, Strategy> 这个会在初始化的时候将所有的 Strategy 自动加载到 Map 中，是不是很方便。使用 concurrentHashMap 是防止多线程操作的时候出现问题。同时还要注意@Service 注解。

```
package com.hqs.pattern.factory;

import com.hqs.pattern.strategy.Strategy;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import java.util.Map;
import java.util.concurrent.ConcurrentHashMap;
```

```
/**
 * @author huangqingshi
 * @Date 2019-01-31
 */
@Service
public class FactoryForStrategy {

    @Autowired
    Map<String, Strategy> strategys = new ConcurrentHashMap<>(3);

    public Strategy getStrategy(String component) throws Exception{
        Strategy strategy = strategys.get(component);
        if(strategy == null) {
            throw new RuntimeException("no strategy defined");
        }
        return strategy;
    }

}
```

接下来就是 Strategy 接口，就一个 doOperation 方法。

```
package com.hqs.pattern.strategy;

/**
 * @author huangqingshi
 * @Date 2019-01-31
 */
public interface Strategy {

    String doOperation();

}
```

定义接口的实现，我定义了三个，都类似，这里我就放出一个来吧。Component 里边的 one 是指定其名字，这个会作为 key 放到 Map strategys 里边。

```
package com.hqs.pattern.strategy.impl;

import com.hqs.pattern.strategy.Strategy;
import org.springframework.stereotype.Component;
```

```
/**
 * @author huangqingshi
 * @Date 2019-01-31
 */
@Component("one")
public class StrategyOne implements Strategy {
    @Override
    public String doOperation() {
        return "one";
    }
}
```

好了，写一个 Controller 类，用于进行测试，当然我还是使用 swagger，使用 swagger 的时候有个细节，就是注意生产上一定不能打开，否则是个非常可怕的事情。

```
package com.hqs.pattern.controller;

import com.hqs.pattern.factory.FactoryForStrategy;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.ResponseBody;

/**
 * @author huangqingshi
 * @Date 2019-01-31
 */
@Controller
public class StrategyController {

    @Autowired
    FactoryForStrategy factoryForStrategy;

    @PostMapping("/strategy")
    @ResponseBody
    public String strategy(@RequestParam("key") String key) {
        String result;
        try {
            result = factoryForStrategy.getStrategy(key).doOperation();
        } catch (Exception e) {
            result = e.getMessage();
        }
```

```
        return result;
    }


}
```

打开 swagger 进行测试，输入 one，返回 one。输入 four，返回 no strategy defined。后续如果有新策略的话，直接实现即可。