

Spring boot使用拦截器是通过@Configuration标志的配置类来注册实现

首先先写一个拦截器

```
public class LoginInterceptor implements HandlerInterceptor{

    @Override
    public void afterCompletion(HttpServletRequest arg0, HttpServletResponse arg1,
Object arg2, Exception arg3)
        throws Exception {
        // TODO Auto-generated method stub

    }

    @Override
    public void postHandle(HttpServletRequest arg0, HttpServletResponse arg1, Object
arg2, ModelAndView arg3)
        throws Exception {
        // TODO Auto-generated method stub

    }

    @Override
    public boolean preHandle(HttpServletRequest arg0, HttpServletResponse arg1, Object
arg2) throws Exception {
        System.out.println("已经进入了登录拦截器.....");

        //逻辑代码

        return true;
    }

}
```

然后写一个配置类用来注册拦截器

```
@Configuration
public class WebConfig extends WebMvcConfigurerAdapter {

    @Override
    public void addInterceptors(InterceptorRegistry registry) {

        // 需要拦截的路径
        String[] addPathPatterns = { "/boot/**" };

        // 不拦截的路径
        String[] excludePathPatterns = { "/boot/getUser" };

    }

}
```

```

        //注册登录拦截器
        registry.addInterceptor(new
LoginInterceptor()).addPathPatterns(addPathPatterns)
                .excludePathPatterns(excludePathPatterns);
    }
}

```

该类继承WebMvcConfigurerAdapter 并手动重写addInterceptors，如代码中一般将我们的拦截器注册进去并添加拦截和不拦截的路径。

访问/boot/demo的结果

已经进入了登录拦截器.....

小叶子

2018-10-11 22:22:37 5500 50470 5 1 2000 11 2 5 1