

<< spring interceptor >>

스프링 환경에서 사용자의 요청에 따라 선행처리 작업 위해 사용

Filter와의 차이

Filter- > DispatcherServlet 요청전에 수행됨
실행되는 시점이 요청전으로 제한됨

spring interceptor

⇒ DispatcherServlet 실행후 동작함
실행되는 시점을 세밀하게 설정가능

- 요청처리전 preHandle
- 요청처리 메소드 진입후 postHandle
- View로 가기전 afterCompletion

인터셉터 만들기

@Component

```
public class MyInterceptor2 extends HandlerInterceptorAdapter {
```

```
    @Override
```

```
    public boolean preHandle(HttpServletRequest request, HttpServletResponse  
response, Object handler)
```

```
        throws Exception {
```

```
        // TODO Auto-generated method stub
```

```
        System.out.println("요청처리 전입니다.");
```

```
        return super.preHandle(request, response, handler);
```

```
    }
```

```
    @Override
```

```
    public void postHandle(HttpServletRequest request, HttpServletResponse response,  
Object handler,
```

```
        ModelAndView modelAndView) throws Exception {
```

```
        // TODO Auto-generated method stub
```

```
        System.out.println("요청메소드 진입입니다.");
```

```
        super.postHandle(request, response, handler, modelAndView);
```

```
    }
```

```
    @Override
```

```
    public void afterCompletion(HttpServletRequest request, HttpServletResponse  
response, Object handler, Exception ex)
```

```
        throws Exception {
```

```

        // TODO Auto-generated method stub
        System.out.println("view가기 전입니다.");
        super.afterCompletion(request, response, handler, ex);
    }
}

```

인터셉터의 설정

@Configuration

public class SistConfig implements WebMvcConfigurer {

 @Override

 public void addInterceptors(InterceptorRegistry registry) {

 // TODO Auto-generated method stub

 // WebMvcConfigurer.super.addInterceptors(registry);

 registry.addInterceptor(new MyInterceptor2())

 .addPathPatterns("/hello.do", "/list.do", "/member/**");

 //.excludePathPatterns("/member/image/**");

 }

}

