

# 프로퍼티 설정 및 읽기

프로젝트명 : B05ReadProperites

## 준비사항

- 의존설정 : Spring Web
- JSP 사용을 위한 설정을 한다.
- Refresh Gradle Project 를 눌러 적용한다.

## application.properties 파일

Spring Boot 애플리케이션의 설정 정보 및 프로그래밍에 필요한 여러가지 값을 저장할 수 있다.

### 주요역할

- 설정구성 : 데이터베이스 연결 정보, 포트 번호, 보안 설정, 로깅 구성 등
- 환경별 구성 : 프로파일을 사용하여 개발, 테스트, 운영 등의 다양한 환경에 서로 다른 설정을 제공
- 외부 구성 : 배포시 Jar 파일과 함께 패키징되지 않고 외부에 배포 가능

## 기본사용법

@Value 어노테이션을 통해 속성값을 읽음

	application.properties	Controller
	member.id=nakja member.pass=1234	@Value("\${member.id}") private String 멤버변수1; @Value("\${member.pass}") private String 멤버변수2;

## 사용자정의 properties 파일

프로퍼티 파일 생성 후 빈으로 등록 후 사용가능

<pre>@Configuration public class 등록을위한클래스 {     @Bean(name="별칭")     public PropertiesFactoryBean propertiesFactoryBean() {         PropertiesFactoryBean인스턴스.setLocation(프로퍼티 파일 경로);         return propertiesFactoryBean;     } }</pre>
--

## 1.application.properties에서 가져오기

home : webapp/WEB-INF/views/home.jsp

```
9<body>
10<h2>properties파일에 정의된 값 읽어오기</h2>
11<ul>
12    <li><a href="/">루트</a></li>
13    <li><a href="/link1">application.properties에서 가져오기</a></li>
14    <li><a href="/link2">사용자 정의 properties파일에서 가져오기</a></li>
15</ul>
16</body>
17</html>
```

프로퍼티파일 : resources/application.properties

```
13#프로퍼티 값
14test.key1=morning
15test.key2=evening
16test.key3=Spring,Summer,Autumn,Winter
17|
```

Controller : com.edu.springboot.MainController.java

```
26@Value("${test.key1}")
27private String testKey1;
28@Value("${test.key2}")
29private String testKey2;
30
31@RequestMapping("/link1")
32public String link1(Model model, @Value("${test.key3}")
33    List<String> testKey3) {
34
35    System.out.println("testKey1="+ testKey1);
36    System.out.println("testKey2="+ testKey2);
37    System.out.println("testKey3="+ testKey3);
38
39    model.addAttribute("testKey1", testKey1);
40    model.addAttribute("testKey2", testKey2);
41    model.addAttribute("testKey3", testKey3);
42
43    return "link1";
44}
```

View : webapp/WEB-INF/views/link1.jsp

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <%@ taglib prefix="c" uri="jakarta.tags.core"%>
4 <!DOCTYPE html>
5 <html>
6 <head>
7 <meta charset="UTF-8">
8 <title>Insert title here</title>
9 </head>
10 <body>
11   <h2>application.properties에서 가져오기</h2>
12   <ul>
13     <li>testKey1 : ${testKey1 }</li>
14     <li>testKey2 : ${testKey2 }</li>
15   </ul>
16   <ol>
17     <c:forEach items="${testKey3 }" var="item">
18       <li>testKey3 : ${item }</li>
19     </c:forEach>
20   </ol>
21 </body>
22 </html>
```

여기까지 작성하세요.

## 2.사용자정의 properties파일에서 가져오기

프로퍼티 파일 생성 : resources/my.properties

```
1 my.id=nakja
2 my.pass=1234
3 my.address=Seoul
4 my.age=12
5 |
```

Config파일 : com.edu.springboot.PropertyConfig.java

```
1 package com.edu.springboot;
2
3 import org.springframework.beans.factory.config.PropertiesFactoryBean;
4
5 @Configuration
6 public class PropertyConfig {
7
8     @Bean(name="myprops")
9     public PropertiesFactoryBean propertiesFactoryBean() {
10         PropertiesFactoryBean propertiesFactoryBean =
11             new PropertiesFactoryBean();
12         ClassPathResource classPathResource =
13             new ClassPathResource("my.properties");
14         propertiesFactoryBean.setLocation(classPathResource);
15         return propertiesFactoryBean;
16     }
17 }
```

Controller : com.edu.springboot.MainController.java

```
51 @Value("#{myprops['my.id']}")
52 private String myId;
53 @Value("#{myprops['my.pass']}")
54 private String myPass;
55 @Value("#{myprops['my.address']}")
56 private String myAddress;
57 @Value("#{myprops['my.age']}")
58 private String myAge;
59
60 @RequestMapping("/link2")
61 public String link2(Model model) {
62
63     System.out.println("myId="+ myId);
64     System.out.println("myPass="+ myPass);
65     System.out.println("myAddress="+ myAddress);
66     System.out.println("myAge="+ myAge);
67 }
```

```
68     model.addAttribute("myId", myId);
69     model.addAttribute("myPass", myPass);
70     model.addAttribute("myAddress", myAddress);
71     model.addAttribute("myAge", myAge);
72
73     return "link2";
74 }
75 }
```

View : [webapp/WEB-INF/views/link2.jsp](#)

```
10 <body>
11     <h2>사용자정의 properties파일에서 가져오기</h2>
12     <ul>
13         <li>myId : ${myId }</li>
14         <li>myPass : ${myPass }</li>
15         <li>myAddress : ${myAddress }</li>
16         <li>myAge : ${myAge }</li>
17     </ul>
18 </body>
19 </html>
```