



SPRING BOOT

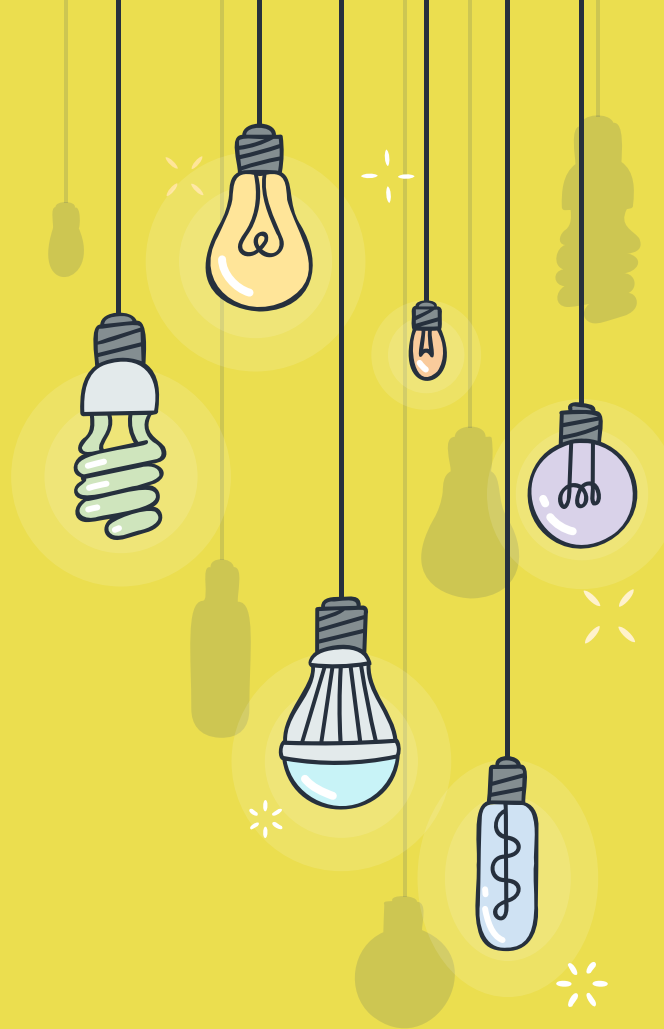
1. THYMELEAF 의 개요

2. THYMELEAF 의 사용



1

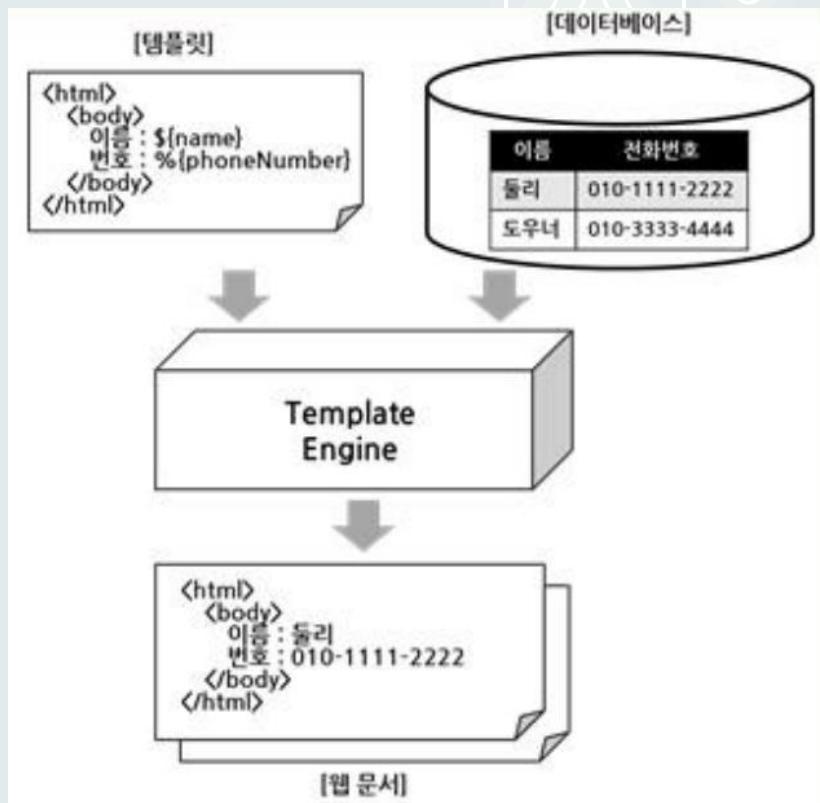
THYMELEAF 개요



* 템플릿 엔진

+ 템플릿 엔진이란?

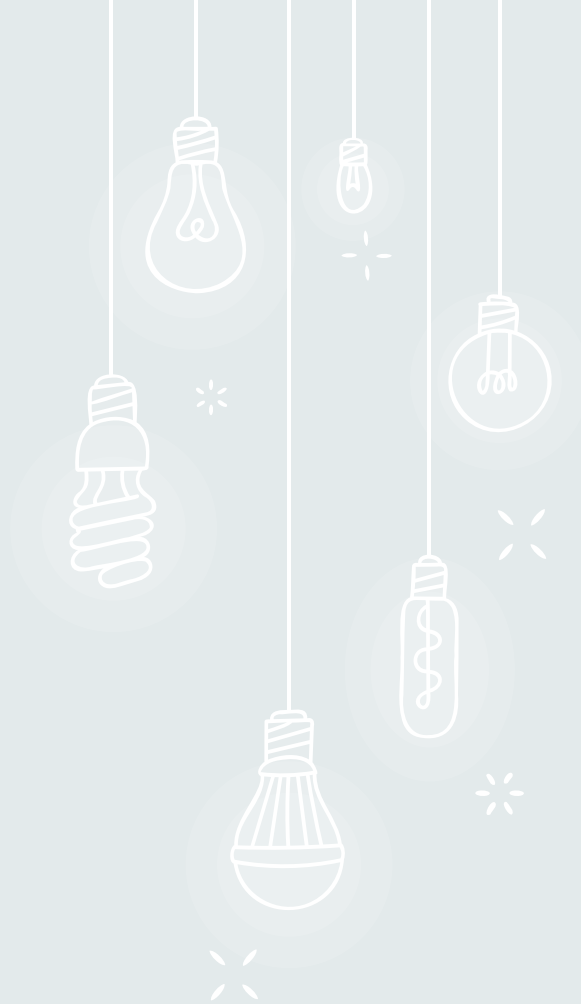
- × 사용자에게 제공되는 화면과 데이터 처리 로직을 분리
- ◆ 개발 및 유지보수의 편의성 증대
- × EL과 JSTL이 결합된 JSP도 템플릿 엔진에 포함됨



* 스프링 부트의 템플릿 엔진

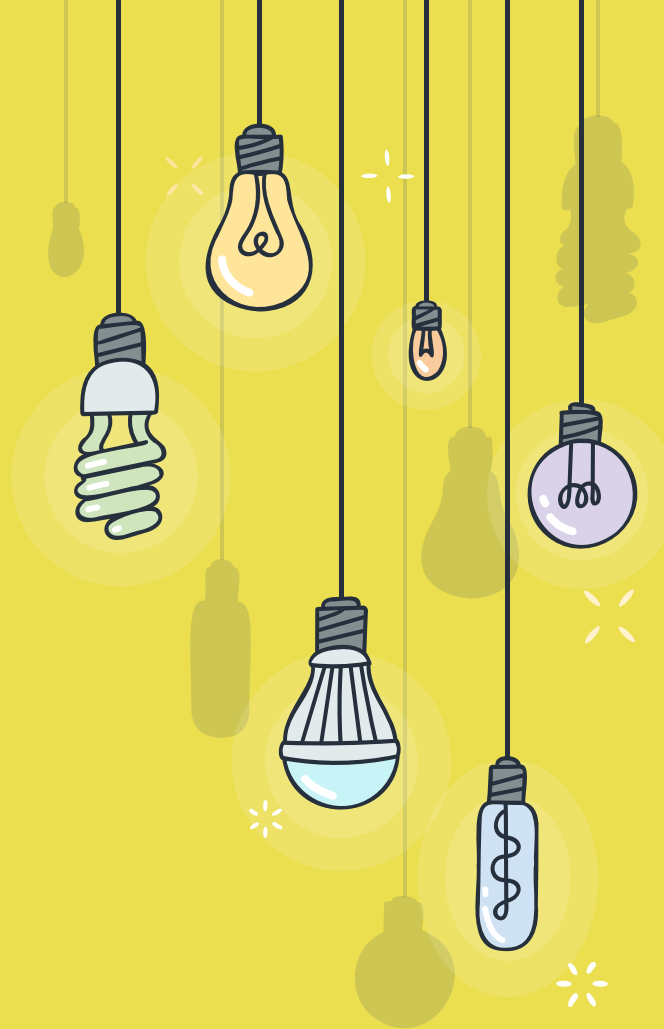
+ 스프링 부트가 지원하는 템플릿 엔진

- × Thymeleaf
- × Apache Freemarker
- × Mustache
- × Groovy Templates



2

THYMELEAF의 사용



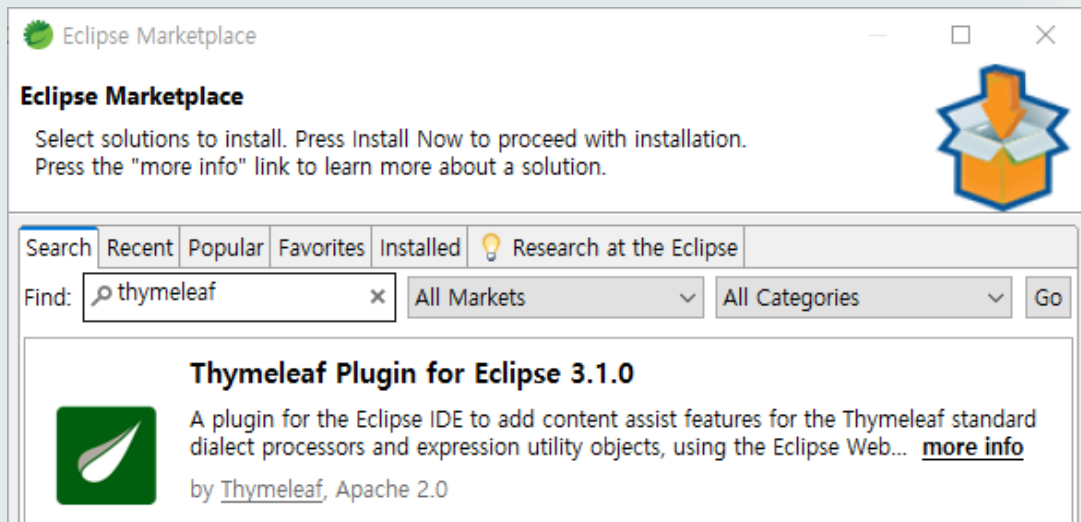
* THYMELEAF 기본

+ 템플릿 파일 작성

- × Thymeleaf가 인지하는 템플릿 파일의 기본위치

src/main/resources/templates 폴더

- × 속성을 자동완성 하려면 플러그인을 설치



* THYMELEAF 문법

+ Thymeleaf를 적용하기 위해서는 “th:” 접두사로 시작

속성	의미	사용예
th:text	단순 텍스트 출력	<code></code>
		<code>greeting=환영합니다 -> 환영합니다</code>
th:with	동적 변수 값 출력	<code><div th:with="address=\${name}+'의 주소는 강남'" th:text={address}></div></code>
		<code>name=홍길동 -> <div>홍길동의 주소는 강남</div></code>

* THYMELAF 문법

+ 조건식(if)

속성	의미	사용예
th:if th:unless	조건문 if~else	<code>관리자</code> <code>사용자</code>
		role=Admin -> <code>관리자</code> 나머지는 모두 -> <code>사용자</code>

* THYMELEAF 문법

+ 조건식(switch)

속성	의미	사용예
th:switch th:case	다중분기 switch~case	<pre><div th:switch="{search}"> 제목 내용 작성자 </div></pre>
		<pre>search=T -> 제목 search=C -> 내용 search=W -> 작성자</pre>

* THYMELAF 문법

+ 반복문(each)

속성	의미	사용예
th:each	반복문	<pre><tr th:each="board:\${boardList}"> 반복할 내용 </tr></pre>
		boardList컬렉션에 3개의 게시글이 들어있다면 <pre><tr> 반복할 내용 반복할 내용 반복할 내용 </tr></pre>

× 속성에 상태변수 선언 가능

◆ state라는 상태 변수로 게시글에 순서대로 번호 부여가능

* THYMELEAF 문법

+ 하이퍼링크 설정 및 파라미터 전달

- × 하이퍼링크 설정 : @{링크주소}
- × 파라미터 전달 : 키=값

```
<tr th:each="board, state : ${boardList}">
  <td th:text="${state.count}"></td>
  <td>
    <a th:href="@{boardDetail(boardno=${board.boardno})}"
      th:text="${board.title}">
    </a>
  </td>
  <td th:text="${board.writer}"></td>
</tr>
```

* THYMELAF 문법

+ 파라미터 여러 개 전달은 콤마(,)를 이용

```
<a th:href="@{login(id=${member.id}, pwd=${member.pwd})}"  
  th:text="로그인">  
</a>
```

THANKS!

+ Any questions?

