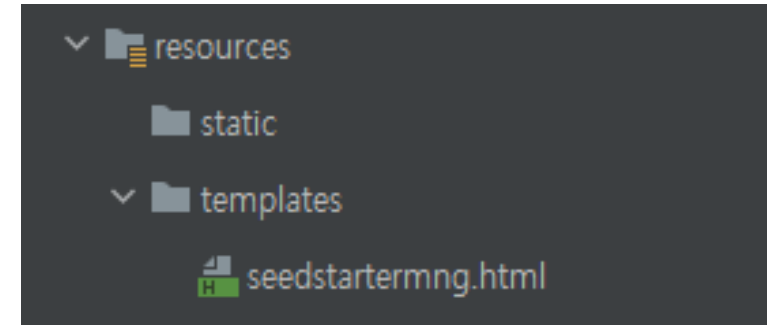


View 구현 1

Thymeleaf

- **SeedStarterMng.html**

- head 정의: bootstrap이용



```
<!DOCTYPE html>
```

```
<html xmlns:th="http://www.thymeleaf.org" th:with="lang=${#locale.language}" th:lang="${lang}">
```

```
<head>
```

```
  <title>SSM</title>
```

```
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">
```

```
  <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KChRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN" crossorigin="anonymous"></script>
```

```
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q" crossorigin="anonymous"></script>
```

```
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js" integrity="sha384-JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYI" crossorigin="anonymous"></script>
```

```
</head>
```

```
</html>
```

Thymeleaf

- SeedStarterMng.html

- 결과 확인

List of Seed Starters



??	??	??	??	??
seedstarter.datePlanted_ko_KR??	seedstarter.covered_ko_KR??	seedstarter.type_ko_KR??	seedstarter.features_ko_KR??	seedstarter.rows_

- 테이블의 컬럼에 이상한 값이 표시

Thymeleaf

- **SeedStarterMng.html**

- body 정의: 테이블 정의(head)

```
<body>
  <div class="container" th:unless="${#lists.isEmpty(seedStarterWithDetail)}">
    <div class="p-3 mb-5 bg-success text-white"><h2>List of Seed Starters</h2></div>
    <p>&#x1F33F;</p>
    <div>
      <table class="table table-sm">
        <thead class="thead-dark">
          <tr>
            <th th:text="#{seedstarter.datePlanted}"></th>
            <th th:text="#{seedstarter.covered}"></th>
            <th th:text="#{seedstarter.type}"></th>
            <th th:text="#{seedstarter.features}"></th>
            <th th:text="#{seedstarter.rows}"></th>
          </tr>
```

...

Thymeleaf

- **Simple expressions**

- Variable Expressions: `${...}`
- Message Expressions: `#{...}`
- Link URL Expressions: `@{...}`

- **Expression Utility Objects**

- `#execInfo`
- `execInfo`
 - `message`
 - `uris`
 - `Dates`
 - `String`
 - `Lists`

Thymeleaf

- **Expression Utility Objects**

- #execInfo
 - message
 - uri
 - Dates
 - String
 - Lists

```
${#lists.toList(object)}
```

```
/*  
 * Compute size  
 */
```

```
${#lists.size(list)}
```

```
/*  
 * Check whether list is empty  
 */
```

```
${#lists.isEmpty(list)}
```

```
/*  
 * Check if element or elements are contained in list  
 */
```

```
${#lists.contains(list, element)}
```

```
${#lists.containsAll(list, elements)}
```

```
/*  
 * Sort a copy of the given list. The members of the list must implement  
 * comparable or you must define a comparator.  
 */
```

```
${#lists.sort(list)}
```

```
${#lists.sort(list, comparator)}
```

Thymeleaf

- Simple expressions

- th:text → html태그에 표시될 텍스트
- th:utext → 변수의 값을 escape 처리하지 않고 표시
- th:if → 조건문, 조건에 따라 화면 표시 여부 결정 가능(css의 display:block, none과 같은 동작)
- th:unless → ~가 아니라면.. if문과 함께 사용 가능하며 이 때는 if 문의 조건과 동일해야 함
- th:each → 반복문
- th:value → input 태그의 값을 지정 `<input type="text" name="st_name" value="kim">`
- th:with → 변수 정의

```
<div th:with="firstArticle=${articles[0]}">
  <a th:text="${firstArticle.name}" th:href="${firstArticle.url}"></a>
</div>
```

Thymeleaf

- **Internationalization(locale)**

- 하나의 웹 사이트가 여러 언어로 표시될 수 있는 기능
- Internationalization (i18n): 기술변경 없이 다른 언어, 다른 지역에서도 사용할 수 있도록 디자인
- Localization (l10n): i18n 된 프로그램 + 특정 지역요소를 추가해서 번역

- **Internationalization with Spring Boot and Thymeleaf**

- Spring Boot에는 i18n이 기본적으로 내장
- message를 이용하여 구현
- message의 목적
 - 클라이언트에 표시될 문자 관리
 - 국제화
 - 에러코드와 에러메시지 관리

Thymeleaf

- **Locale을 위한 messages.properties**

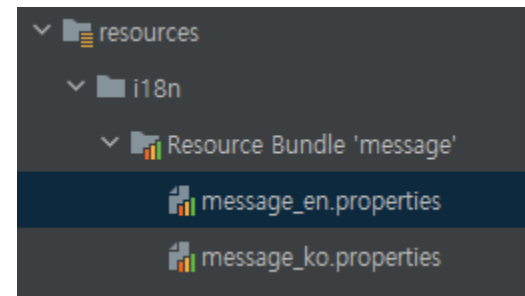
- 스프링 부트의 messages.properties 를 찾는 기본 경로 → resources/

title = Hello

message_en.properties

title = 반갑습니다
seedstarter.datePlanted = 심은날짜
seedstarter.covered = 가림유무
seedstarter.type = 소재
seedstarter.features = 특징
seedstarter.rows = 셀별품종

message_ko.properties



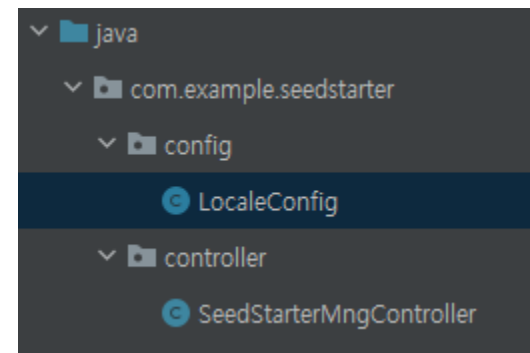
Thymeleaf

- Locale을 위해 웹과 관련된 추가 설정

@Configuration

```
public class LocaleConfig implements WebMvcConfigurer {
```

@EnableWebMvc 붙이면 안됨



- LocaleResolver

@Bean

```
public LocaleResolver localeResolver() {  
    SessionLocaleResolver slr = new SessionLocaleResolver();  
    slr.setDefaultLocale(Locale.KOREA);  
    return slr;  
}
```

- 응용이 현재 어떤 지역에서 사용되고 있는지 해석하기 위해 LocaleResolver가 필요
- session 또는 cookies, the Accept-Language header, a fixed value의 정보를 이용해 현재 지역을 파악할 수 있음

Thymeleaf

- **LocaleChangeInterceptor**

```
@Bean
public LocaleChangeInterceptor localeChangeInterceptor() {
    LocaleChangeInterceptor lci = new LocaleChangeInterceptor();
    lci.setParamName("lang");
    return lci;
}
```

- 요청 html에 붙은 lang을 조회하여 새로운 locale로 변경
- Locale이 기본 Locale과 다를 경우 동작

- **addInterceptors**

- 정의한 인터셉터를 등록
- WebMvcConfigurer를 상속받고 addInterceptors 오버라이딩

```
@Override
public void addInterceptors(InterceptorRegistry registry) {
    registry.addInterceptor(localeChangeInterceptor());
}
```

Thymeleaf

- **messageSource**

- 작성한 언어 리소스의 위치 및 설정 정보를 담아 스프링에게 알려주는 빈

```
@Bean
public MessageSource messageSource() {
    ReloadableResourceBundleMessageSource messageSource = new ReloadableResourceBundleMessageSource();
    messageSource.setBasename("classpath:/i18n/message");
    messageSource.setDefaultEncoding("UTF-8");
    return messageSource;
}
```

- <http://localhost:8080/>
- <http://localhost:8080/?lang=en>

Thymeleaf

- **SeedStarterMng.html**

- body 정의: 테이블 정의(body)

```
<tbody>
  <tr th:each="starter : ${seedStarterWithFeature}">
    <td th:text="${#temporals.format(starter.datePlanted, 'yyyy-MM-dd')}"></td>
    <td th:if="${starter.covered}==true"></td>
    <td th:unless="${starter.covered}==true"></td>
    <td th:text="${starter.type}"></td>
    <td> #<td>
    <td> #<td>
  </tr>
</tbody>
```