



課程安排

- 1~2 堂 Springboot/Thymeleaf
- 3~5 實作一
- 6~8 實作二



chool

Spring Boot



段維瀚 老師





Spring Boot

- Spring Boot是由Pivotal團隊提供的全新框架，其設計目的是用來簡化Spring應用的建立、執行、除錯、部署等。使用Spring Boot可以做到專注於Spring應用的開發，而無需過多關注XML的配置。





Spring Boot

- 習慣 優於 配置
 - Spring Boot 擴展了 Spring framework，透過建立 Spring application 省去了冗長的設定檔(configurations) 能更快更有效率地開發應用系統，讓企業專案快速執行起來。





Spring Tools | 4

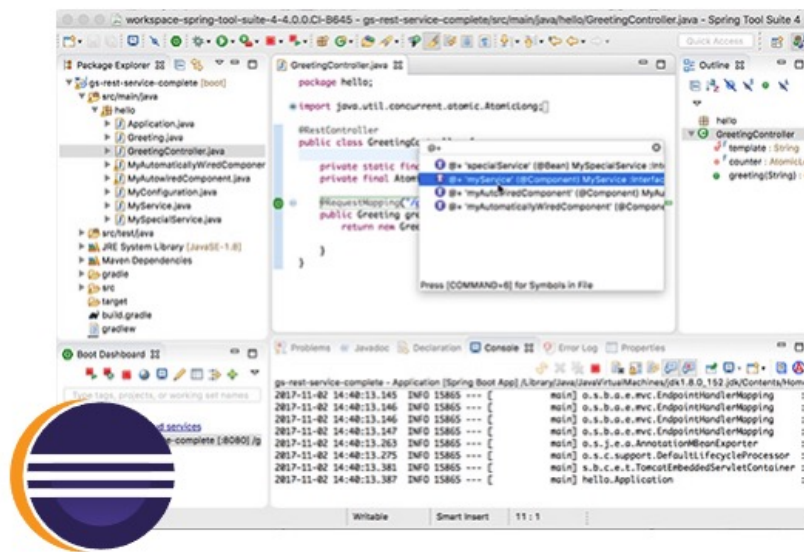
Spring Tools 4 for Eclipse

The all-new Spring Tool Suite 4.
Free. Open source.

4.12.0 - LINUX X86_64

4.12.0 - MACOS X86_64

4.12.0 - WINDOWS X86_64





pom.xml

```
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-thymeleaf</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-tomcat</artifactId>
    <scope>provided</scope>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-configuration-processor</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
  </dependency>
  <dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <scope>runtime</scope>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-test</artifactId>
    <scope>test</scope>
  </dependency>
</dependencies>
```





YAML 語法

- **YAML基本語法** (/'jæmə1/ , 尾音類似camel駱駝)
 - 使用縮排表示層級關係
 - 大小寫區分
 - key: value (注意「:」後有空格)
- **YAML 數據結構**
 - 字面值：鍵值與數值
 - 物件：鍵值與數值的集合
 - 數組陣列





application.properties

```
server.port=8081

server.servlet.context-path=/sts
server.servlet.encoding.charset=UTF-8

spring.mvc.hiddenmethod.filter.enabled=true

spring.messages.basename=base,errors
spring.messages.encoding=UTF-8

spring.thymeleaf.cache=false
spring.thymeleaf.mode=HTML
spring.thymeleaf.encoding=UTF-8

spring.datasource.url=jdbc:mysql://localhost:3306/finweb?serverTimezone=UTC&characterEncoding=utf-8&useUnicode=true
spring.datasource.username=root
spring.datasource.password=12345678
spring.datasource.driverClassName=com.mysql.cj.jdbc.Driver

spring.jpa.database=MYSQL
spring.jpa.show-sql=true
spring.jpa.hibernate.ddl-auto=update
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL8Dialect
spring.jpa.properties.hibernate.format_sql=true
```




application.yml

```
server:
  port: 8080
  servlet:
    context-path: /sts

spring:
  datasource:
    driver-class-name: com.mysql.cj.jdbc.Driver
    url: jdbc:mysql://localhost:3306/web?serverTimezone=UTC&characterEncoding=utf-8&useUnicode=true
    username: root
    password: 12345678
  jpa:
    database: MYSQL
    show-sql: true
    hibernate:
      ddl-auto: update
    properties:
      hibernate:
        dialect: org.hibernate.dialect.MySQL5Dialect
        format_sql: true
```





YAML、Properties 相互轉換

- <https://www.toyaml.com/index1.html>





Thymeleaf



<https://www.thymeleaf.org/>

Thymeleaf是適用於Web和獨立環境的現代伺服器端Java模板引擎

Natural templates

HTML templates written in Thymeleaf still look and work like HTML, letting the actual templates that are run in your application keep working as useful design artifacts.

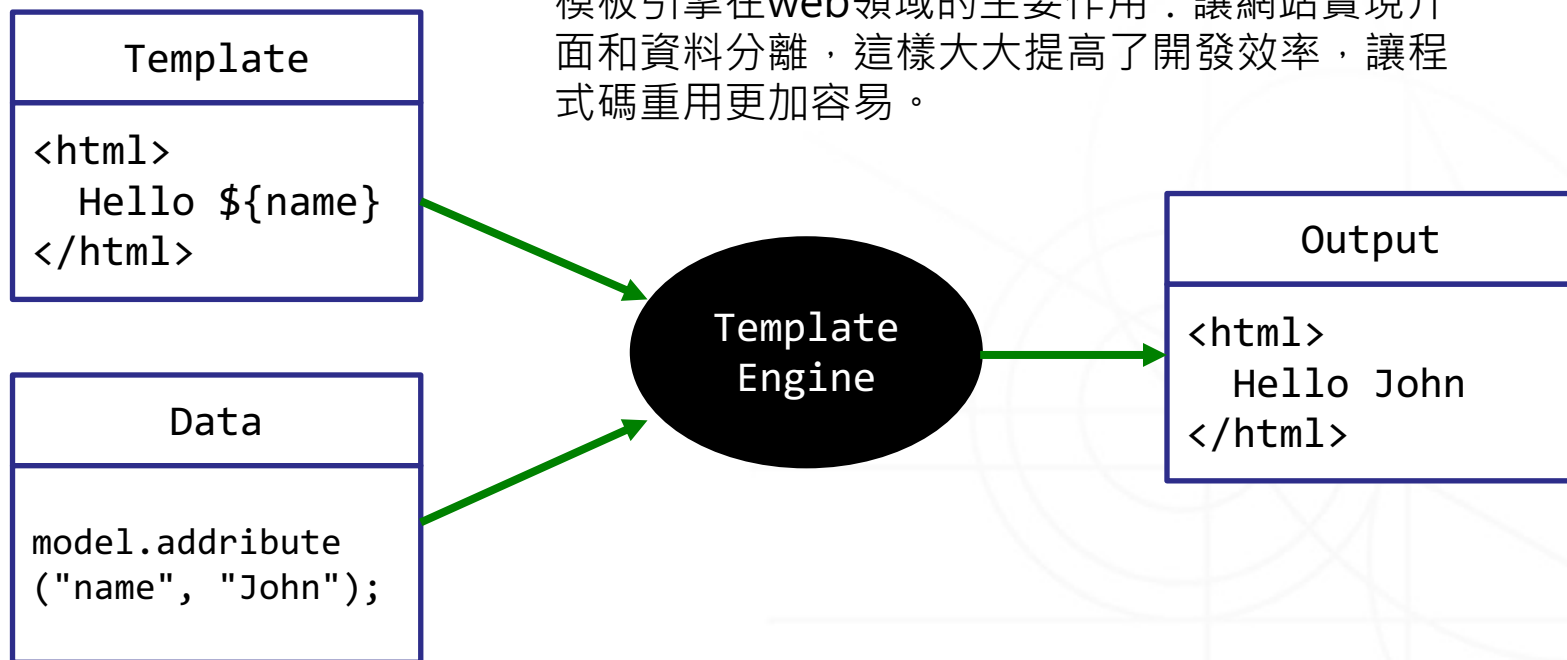
```
1  <table>
2    <thead>
3      <tr>
4        <th th:text="#{msgs.headers.name}">Name</th>
5        <th th:text="#{msgs.headers.price}">Price</th>
6      </tr>
7    </thead>
8    <tbody>
9      <tr th:each="prod: ${allProducts}">
10        <td th:text="${prod.name}">Oranges</td>
11        <td th:text="${#numbers.formatDecimal(prod.price, 1, 2)}">0.99</td>
12      </tr>
13    </tbody>
14  </table>
```



模板引擎

- JSP、Thymeleaf、Freemarker

模板引擎在web領域的主要作用：讓網站實現介面和資料分離，這樣大大提高了開發效率，讓程式碼重用更加容易。





Thymeleaf

- Thymeleaf作為被Springboot官方大力推薦的模板引擎
 - 動靜分離
 - Thymeleaf選用html作為模板頁，這是任何一款其他模板引擎做不到的！
 - 開箱即用：
 - Thymeleaf提供標準和Spring標準兩種方言，可以直接套用模板來實現。
 - 目前預設版本：2.x





Thymeleaf

- 編寫框架
 - pom.xml 加入 Thymeleaf 的依賴
 - application.yml 整合配置內容
 - controller 用來繫結 Thymeleaf html 頁面
 - static 存放靜態資源
 - css、javascript、靜態/傳統 html
 - templates 存放模板引擎
 - Thymeleaf 的 html 檔案



Thymeleaf

```
<tr th:each="user, stat : ${users}">
  <td th:text="${stat.index}"></td>
  <td th:text="${stat.count}"></td>
  <td th:text="${user.id}"></td>
  <td th:text="${user.name}"></td>
  <td th:text="${user.password}"></td>
  <td th:text="${user.birth}"></td>
  <td th:text="${user.height}"></td>
  <td th:text="${user.weight}"></td>
  <td>
    <a th:href="@{/user/{id}(id=${user.id})}">編輯</a>
  </td>
  <td>
    <a th:href="@{/user/delete/{id}(id=${user.id})}">刪除</a>
  </td>
</tr>
```


Thymeleaf

• 宣告

- `<html xmlns:th="http://www.thymeleaf.org">`

標籤	作用	示例
th:id	替換id	<code><input th:id="\${user.id}"/></code>
th:text	文字替換	<code><p text:="\${user.name}">big sai</p></code>
th:utext	支援html的文字替換	<code><p utext:="\${htmlcontent}">content</p></code>
th:object	替換物件	<code><div th:object="\${user}"></div></code>
th:value	替換值	<code><input th:value="\${user.name}" ></code>
th:each	迭代	<code><tr th:each="student:\${user}" ></code>
th:href	替換超連結	<code><a th:href="@{index.html}">超連結</code>
th:src	替換資源	<code><script type="text/javascript" th:src="@{index.js}"></script></code>



ThymeLeaf

- 表達式

- $\${}$ 變數表達式
- $*\{\}$ 可選變數表達式
- $\#\{\}$ 資源變數表達式
- $@\{\}$ 鏈結表達式
- 字串連接、數學運算、布林運算、?: 運算





ThymeLeaf

- 迭代狀態參數配置
 - index: 當前迭代索引，從0開始
 - count: 當前迭代序號，從1開始
 - size: 元素的總量
 - current: 當前正在迭代的元素
 - even/odd: 迭代是奇數列還是偶數列
 - first: 當前迭代是否是第一個
 - last: 當前迭代是否是最後一個





ThymeLeaf

- 條件判斷表達式可以是如下的形式：
 - 布林類型返回 true/false
 - 數值類型且值不是0，返回 true
 - 字元類型且值不是0，返回 true
 - 字串類型且值不是
 - "false"、"off"、"no"，返回 true
 - 之外其他類型其值不是 null，返回 true
 - 物件對象的值是null，返回 false



Thymeleaf 3 新語法

```
<dependency>
  <groupId>org.thymeleaf</groupId>
  <artifactId>thymeleaf</artifactId>
  <version>3.0.11.RELEASE</version>
</dependency>
<dependency>
  <groupId>org.thymeleaf</groupId>
  <artifactId>thymeleaf-spring5</artifactId>
  <version>3.0.11.RELEASE</version>
</dependency>
```

```
<dependency>
  <groupId>org.thymeleaf.extras</groupId>
  <artifactId>thymeleaf-extras-java8time</artifactId>
  <version>3.0.4.RELEASE</version>
</dependency>
```

支援 java 8 Date

- **#dates** : *java.util.Date* 對象的實用方法
- **#calendars** : 類似於 **#dates** , 用於 *java.util.Calendar* 對象
- **#numbers** : 格式化數字對象的實用方法
- **#strings** : *String* 對象的實用方法
- **#objects** : *Java Object* 類的通用
- **#bools** : 布爾評估的實用方法
- **#arrays** : 數組的實用方法
- **#lists** : 列表的實用方法
- **#sets** : 集合的實用方法
- **#maps** : 地圖的實用方法
- **#aggregates** : 用於在數組或集合上創建聚合的實用方法
- **#messages** : 用於在變量表達式中獲取外部化消息的實用方法

學習參考網址 : <https://www.baeldung.com/spring-thymeleaf-3-expressions>
<https://www.baeldung.com/dates-in-thymeleaf>



Bootstrap

使用全球最流行的前端
開發工具 Bootstrap，
快速設計及自定義響應
式網站





Bootstrap

於網頁<header>中配置：

```
<link rel="stylesheet"  
href="https://maxcdn.bootstrapcdn.com/bootstrap/  
3.3.7/css/bootstrap.min.css"/>
```



Bootstrap 網格

柵格参数

通过下表可以详细查看 Bootstrap 的柵格系统是如何在多种屏幕设备上工作的。

	超小屏幕 手机 (<768px)	小屏幕 平板 (≥768px)	中等屏幕 桌面显示器 (≥992px)	大屏幕 大桌面显示器 (≥1200px)
柵格系统行为	总是水平排列			
<code>.container</code> 最大宽度	None (自动)	750px	970px	1170px
类前缀	<code>.col-xs-</code>	<code>.col-sm-</code>	<code>.col-md-</code>	<code>.col-lg-</code>
列 (column) 数	12			
最大列 (column) 宽	自动	~62px	~81px	~97px
槽 (gutter) 宽	30px (每列左右均有 15px)			
可嵌套	是			
偏移 (Offsets)	是			
列排序	是			