

مقدمة

يوفر هذا الكتاب دليلًا عمليًا لإطار بتكويين تقوم أن عليك، مع إشارات إلى كيفية استخدام هذا الإطار لتكويين تطبيقات ويب بسيطة باستخدام Spring MVC. باستخدام هذا الإطار، يمكنك إنشاء تطبيقات ويب بسيطة باستخدام Spring MVC. باستخدام هذا الإطار، يمكنك إنشاء تطبيقات ويب بسيطة باستخدام Spring MVC.

خطوة خطوة دليل

1. المشروع بنية إعداد

يهدف هذا المشروع إلى إنشاء تطبيق ويب بسيط باستخدام Spring MVC. باستخدام هذا الإطار، يمكنك إنشاء تطبيقات ويب بسيطة باستخدام Spring MVC.

```
SimpleSpringMVCApp/  
  src/  
    main/  
      java/  
        com/  
          example/  
            controllers/  
              HomeController.java  
        webapp/  
          WEB-INF/  
            web.xml  
            spring-mvc-config.xml  
          views/  
            home.jsp  
pom.xml    )    Maven)
```

- الوجهات. مثل `src/main/java`: المصدر كود على يحتوي
- ومشاهدات `web.xml`, `spring-mvc-config.xml` ملفات على يحتوي `src/main/webapp/WEB-INF`.

2. الاعتماديات إضافة

مكتبة إلى تحتاج البسيط، مع إشارات إلى كيفية استخدام هذا الإطار لتكويين تطبيقات ويب بسيطة باستخدام Spring MVC.

التالي: بالملف `pom.xml` تعديل أو إنشاء

```

<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.example</groupId>
  <artifactId>SimpleSpringMVCApp</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>war</packaging>

  <dependencies>
    <!-- Spring Web MVC -->
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-webmvc</artifactId>
      <version>5.3.10</version>
    </dependency>
    <!-- Servlet API (provided by the container) -->
    <dependency>
      <groupId>javax.servlet</groupId>
      <artifactId>javax.servlet-api</artifactId>
      <version>4.0.1</version>
      <scope>provided</scope>
    </dependency>
  </dependencies>

  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-war-plugin</artifactId>
        <version>3.3.1</version>
      </plugin>
    </plugins>
  </build>
</project>

```

ملاحظات:

- `<packaging>war</packaging>`: كملف تجميعي يتم العمل مشروع أن يضمن:
- مسار إلى وأضفهم يدويًا و `<scope>provided</scope>` و `<scope>provided</scope>` بتنزيل ق، `<scope>provided</scope>` تستخدم لا كنت إذا العمل مشروع. تصني ف

3. توكوين web.xml في

لمعالجة الـ، الـ الأمامي الموجه هو الذي DispatcherServlet، لتطبيقك التوزيع الوصف هو web.xml ملف الواردة. ال طلبات

التالي: بالملف src/main/webapp/WEB-INF/web.xml إنشاء

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0_1-2.xsd"
  version="4.0">

  <!-- DispatcherServlet -->
  <servlet>
    <servlet-name>spring-mvc</servlet-name>
    <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
    <init-param>
      <param-name>contextConfigLocation</param-name>
      <param-value>/WEB-INF/spring-mvc-config.xml</param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
  </servlet>

  <!-- DispatcherServlet -->
  <servlet-mapping>
    <servlet-name>spring-mvc</servlet-name>
    <url-pattern>/</url-pattern>
  </servlet-mapping>
</web-app>
```

شرح:

- <servlet-class>: إلى ال طلبات يوجه الذي DispatcherServlet، حدد
- <init-param>: ال spring-mvc-config.xml الـ لتوكوين ملف إلى يشير.
- <url-pattern>/</url-pattern>: إلى ال طلبات جميع لمعالجة الـ الذي حدد

4. الـ لتوكوين ملف إنشاء

ومحلل ال موجهات مثل، الـ لتوكوين الـ لتوكوين src/main/webapp/WEB-INF/spring-mvc-config.xml إنشاء ال مشاهدات.

```

<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:context="http://www.springframework.org/schema/context"
       xmlns:mvc="http://www.springframework.org/schema/mvc"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="
           http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd
           http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context.xsd
           http://www.springframework.org/schema/mvc http://www.springframework.org/schema/mvc/spring-mvc.xsd" >

    <!--           -->

    <context:component-scan base-package="com.example.controllers" />

    <!--           MVC           -->
    <mvc:annotation-driven />

    <!--           JSP -->
    <bean class="org.springframework.web.servlet.view.InternalResourceViewResolver">
        <property name="prefix" value="/WEB-INF/views/" />
        <property name="suffix" value=".jsp" />
    </bean>
</beans>

```

شرح:

- `<context:component-scan>`: حزمة فحص يجرى `com.example.controllers` عن للبحث `@Controller`.
- `<mvc:annotation-driven>`: ميزات يتيح `@GetMapping` مثل التسمية على الالقائى `@GetMapping`.
- `InternalResourceViewResolver`: إضافة مع `/WEB-INF/views/` في ملفات إلى المشاهدات أسماء يحدد `.jsp`.

5. بسيط موجه إنشاء.

طلبات لمعالجة موجه إنشاء `HomeController.java` أضف `src/main/java/com/example/controllers/`:

```

package com.example.controllers;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;

@Controller
public class HomeController {

```

```

@GetMapping("/")
public String home() {
    return "home";
}
}

```

شرح:

- @Controller: كواجهة الواجهة هذا يحدد.
- @GetMapping("/"): طلبات يحدد إلى إلى جذر إلى / إلى home() طريقة إلى إلى.
- return "home": الواجهة اسم يعيد "home", إلى إلى /WEB-INF/views/home.jsp.

6. مشاهدة إنشاء

ملف إنشاء في src/main/webapp/WEB-INF/views/home.jsp لإخراج. لعرض بسيط

```

<%@ page contentType="text/html; charset=UTF-8" language="java" %>
<html>
<head>
    <title>Home</title>
</head>
<body>
    <h1>        Spring MVC        Spring Boot</h1>
</body>
</html>

```

7. التطبيق وتعبئة بناء

ملف لبناء المشروع جذر من التالي الأمر بتشغيل قم، تستخدم كنت إذا:

```
mvn clean package
```

target. مجلد في SimpleSpringMVCApp-1.0-SNAPSHOT.war يخلق هذا

- على الحفظ مع يدوي، ملف إلى المشروع وتعبئة ملفات بتجميع قم، تستخدم لا كنت إذا: ملاحظة
- الواجهة. بنيت

8. التطبيق نشر

مجلد إلى SimpleSpringMVCApp-1.0-SNAPSHOT.war نسخ: الواجهة الواجهة حاوية إلى ملف نشر webapps إلى bin مجلد من startup.sh أو startup.bat لتشغيل الواجهة لتشغيل بدء الواجهة. تثبيت من webapps الواجهة.

9. التطبيق إلى الوصول

إلى: وذهب ويب متصفح فتح

<http://localhost:8080/SimpleSpringMVCApp-1.0-SNAPSHOT/>

تري: أن يج ب

Spring MVC Spring Boot

اسم بتغيير قدمت إذا. ملف اسم على يعتمد `/SimpleSpringMVCApp-1.0-SNAPSHOT` السياق مسار: ملاحظة `myapp.war` إلى `http://localhost:8080/myapp/` هو `myapp.war` يصب ح

ملخص

ل-الاعتماديات إضافة. 2. قياسية. بنى مع ويب مشروع إعداد. 1. بدون `spring-mvc-config.xml` لاستخدام `web.xml` في `web.xml`. 4. التكوين تحدي. 3. `spring-mvc-config.xml` ملف في ل-التكوين تحدي. 4. `web.xml` في `web.xml`. 3. `spring-mvc-config.xml` إلى ملف وتعبيبة بناء. 7. للإجابة. `spring-mvc-config.xml`. 5. `spring-mvc-config.xml` لمعالجة موجه إنشاء. 6. الطلبات. 6. `spring-mvc-config.xml`. 5. `spring-mvc-config.xml` حاوية `spring-mvc-config.xml`.

أساسيات لتعلم جديدة طريقة يجعله مما التكوين، على الكامل التوحيه يمنحك النهج هذا