

الخطوات لبدء مشروع Spring Boot

واستخدام لتكوين الخطوات هذه اتبع، على مبنى `spring-boot-maven-plugin` مشروع في `spring-boot-maven-plugin` لاستخدام شامل: دليل أدناه تطبيقيك. وإدارة تشغلي، لبناء، الإضافة

1. مشروع جديد مشروع

عادةً: يتضمن هذا `spring-boot-maven-plugin` كمشروع مهيأ لمشروعك أن من تأكد، `spring-boot-maven-plugin` استخدام قبل

به: `spring-boot-starter-parent` الموصى من الاستمرار

إدارة لوالد `spring-boot-starter-parent` بإعداد قم بك، الخاص `pom.xml` ملف في الإضافة. وإصدارات

مثال:

```
<parent>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-parent</artifactId>
  <version>2.5.4</version> <!-- Spring Boot -->
  <relativePath/> <!-- -->
</parent>
```

`spring-boot-dependencies` استخدام بديل،

في `spring-boot-dependencies` بإستيراد قم، `spring-boot-starter-parent` استخدام تستطع لم إذا `dependencyManagement` قسم

مثال:

```
<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-dependencies</artifactId>
      <version>2.5.4</version> <!-- Spring Boot -->
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>
```

تلقائياً. الإصدارات يدير حيث السهولة، أجل من به الموصى هو `spring-boot-starter-parent` استخدام

2. بک ال خاص pom.xml ملف إلى spring-boot-maven-plugin أضف

بك، الخ خاص pom.xml ملف من `<build><plugins>` قسم في إعلانها عليك الإضافة، لاستخدام

□ spring-boot-starter-parent: تستخدم كنات إذا

الوالد. يديره حيث إل إصدار، تحديد دون إل إضافة أضف

مثال:

```
<build>
  <plugins>
    <plugin>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-maven-plugin</artifactId>
    </plugin>
  </plugins>
</build>
```

spring-boot-starter-parent: تستخدم تـكـنـ لـم إذا

المستخدم. [] إصدار مع متطابق صريح، بشكل الإصدار بتحديث ق

مثال:

```
<build>
  <plugins>
    <plugin>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-maven-plugin</artifactId>
      <version>2.5.4</version> <!--           Spring Boot           -->
    </plugin>
  </plugins>
</build>
```

3. إضافة أهداف استخدم

أدناه بـك. الخاص `spring-boot-maven-plugin` تطبيقات وإدارة تشغيل، بناء، في لمساعدتك أهداف عدة `spring-boot-maven-plugin` وتوفر الأهداف: هذه من استخدام الأكثر

```
| spring-boot:run
```

..... مثل دمدج ویب خادم باسٹخدام من مباشرة بك الخصاص تطبیق یجری

□ **والتجربة. للتنمية مفيد**

الأم: □

```
mvn spring-boot:run
```

□ **spring-boot:repackage**

□ يتضمن للتنفيذ قابل أو □□□□ □□□□ إلى mvn package بواسطة يولد الذي □□□□ أو □□□□ ملف تعبئة يعيد الاعتمادات. جميع

□ مهيئة. الإضافة كانت إذا package مرحلة خلال تلقائيًا تنفيذه يتم الهدف هذه

الأم: □

```
mvn package
```

□ باسخدام: التطبيق تشغيل يمكنك التنفيذ، بعد

```
java -jar target/myapp.jar
```

□ **spring-boot:start** و **spring-boot:stop**

□ pre-integration-test و post-integration-test مراحل خلال التطبيق وإيقاف لبدء التكمال لاختبارات استخدام التواليف. على

مثال: □

```
mvn spring-boot:start
```

```
mvn spring-boot:stop
```

□ **spring-boot:build-info**

□ الإصدار. البناء، وقت □ مثل البناء معلومات على يحتوي build-info.properties ملف يولد

□ تعليقات أو □□□□□□ □□□□□□ من BuildProperties □□□□□□ باستخدام تطبيق في المعلومات هذه إلى الوصول يمكن @Value.

الأم: □

```
mvn spring-boot:build-info
```

□ اختياري □ الإضافة تكوين تخصيص 4.

□ الاختصاصات بعض أدناه .pom.xml في التكوين خيارات بإضافة spring-boot-maven-plugin سلوك تخصيص يمكن الشائعة:

□ **الرئسية الفئة تحديد:**

□ يدويًا. بتحديد هذا قم تلقائيًا، الرئسية الفئة اكتشاف من الإضافة يتمكن لم إذا

مثال: □

□□□□ □□□□: من الاعتمادات استبعاد □

مثال:

الطبيق حجج تعين

□ pom.xml في مثال:

4

```

        <configuration>
            <arguments>
                <argument>--server.port=8081</argument>
            </arguments>
        </configuration>
    </plugin>
</plugins>
</build>

```

الأمثلة: سطر عبر الحجة بتسليم قم بديلاً،

```
mvn spring-boot:run -Dspring-boot.run.arguments=--server.port=8081
```

ملفات البناء:

war في pom.xml هو بأكبر الخصاصات عبئية نوع أن من تأكد خارجي،

```
<packaging>war</packaging>
```

الاعتمادات أضف البيئية. من توفرها تم إذا

```
<dependencies>
```

```
<dependency>
```

```
<groupId>org.springframework.boot</groupId>
```

```
<artifactId>spring-boot-starter-tomcat</artifactId>
```

```
<scope>provided</scope>
```

```
</dependency>
```

```
</dependencies>
```

5. رئيسية ملاحظات

الافتراضي السلوك:

قابل لجعله package مرحلة خلال تلقائي

original. مثل تصنيف مع الأصلي المنتج ويضاف النهائي، المنتج تجميعي يتم

التمهيدية عملية:

السريعة. والتجربة للتمهيدية mvn spring-boot:run استخدم

للتنفيذ. قابل أو لبناء mvn package استخدم

التكامل اختبارات:

التكامل. اختبارات أثناء التطبيق حيادية دورة لإدارة spring-boot:stop و spring-boot:start استخدم

ملخص

من الاستمرار خلال من عادة، `spring-boot-maven-plugin` مشروع هو مشروعك أن من تأكد. 1. `spring-boot-maven-plugin` لاستخدام `spring-boot-dependencies` قسم إلى إضافة أضف. 2. `spring-boot-starter-parent` أو استخدام `mvn package` لتشغيل `mvn spring-boot:run` مثال: `mvn spring-boot:run` أوامر استخدام. 3. بك. الخاص `pom.xml` ملف من `spring-boot:build-info` و `spring-boot:stop`، `spring-boot:start` مثل أخرى أهداف `mvn` قابل `spring-boot:build-info` لبناء `pom.xml` في إضافة تكوين بتخصيص قم اختياري، 4. الحاجة. حسب `spring-boot:build-info` باستخدام فعال بشكل `spring-boot:build-info` تطبيقات وإدارة تشغيل، بناء، لك تتيج الإعدادات هذه.