

خادم تطبيقي باستخدام Java

باستخدام Java نحن نريد إنشاء خادم تطبيقي ونوزيعه على المواقع المختلفة! يعني دايمارش نريد إنشاء خادم هذا يخلق لتوسيعه. والمشروع لبناء على بتكوين بالفعل قمت أنك ستفترض . مثلاً، باستخدام ملف إلئي ويوبنيه إلئي، الوصول عنده بـ servlet-4.0 المميزة.

خطوة خطوة دليل

1. بناء مشروع

لإنشاءه، استخدام أو يدوياً البنية هذه إنشاء يمكنك . باستخدام قياسيه ويجب تطبيقي بنية بتكوين ابدأ

المجلدات: بنية

```
SimpleServletApp/
  pom.xml
  src/
    main/
      java/
        com/
          example/
            HelloServlet.java
  webapp/
    WEB-INF/
      web.xml
```

الحاجة: حسب بتعديله قم ثم البنية، إنشاء الأمر هذا بـ شغيل قم **اختياري**: مع إنشاء

```
mvn archetype:generate -DgroupId=com.example -DartifactId=simple-servlet-app -DarchetypeArtifactId=maven-
```

القادمة. الخطوات في متعدلاً وافتوي، بنية يخلق هذا

2. كتابة كود

التألي: المحتوى مع HelloServlet.java في src/main/java/com/example/ باسم ملف إنشاء

```
package com.example;

import javax.servlet.http.HttpServlet;
```

```
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.IOException;

public class HelloServlet extends HttpServlet {
    @Override
    protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws IOException {
        resp.setContentType("text/plain");
        resp.getWriter().write("Hello World!");
    }
}
```

البس يطّة doGet طريقة يستخدم عادي. بنص! لطلب ات يس تج يب هذا شرح: web.xml تكويين مع للت وافق ال تعليقات استخدام وي جن ب ال صريح.

الـ `web.xml` ملف إنشاء.

الاتالي: الـ **web.xml** في **src/main/webapp/WEB-INF** باسم ملف 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.xsd"
          version="4.0">
    <servlet>
        <servlet-name>HelloServlet</servlet-name>
        <servlet-class>com.example.HelloServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>HelloServlet</servlet-name>
        <url-pattern>/hello</url-pattern>
    </servlet-mapping>
</web-app>
```

تعليقات نستخدم لـ `اننا ضروري هذا /hello`. `HelloServlet` فئة `web.xml` ملف يحدد شرح: `@WebServlet`.

4. ملف تكوين pom.xml

الثالث: الـ `SimpleServletApp` مع `pom.xml` تحديث 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>simple-servlet-app</artifactId>
    <version>1.0-SNAPSHOT</version>
    <packaging>war</packaging>

    <properties>
        <maven.compiler.source>1.8</maven.compiler.source>
        <maven.compiler.target>1.8</maven.compiler.target>
    </properties>

    <dependencies>
        <!-- Servlet API (provided by WLP) -->
        <dependency>
            <groupId>javax.servlet</groupId>
            <artifactId>javax.servlet-api</artifactId>
            <version>4.0.1</version>
            <scope>provided</scope>
        </dependency>
    </dependencies>

    <build>
        <plugins>
            <!-- Maven WAR Plugin to build the WAR file -->
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-war-plugin</artifactId>
                <version>3.3.1</version>
                <configuration>
                    <finalName>myapp</finalName>
                </configuration>
            </plugin>
            <!-- Liberty Maven Plugin for deployment -->
            <plugin>
                <groupId>io.openliberty.tools</groupId>
                <artifactId>liberty-maven-plugin</artifactId>

```

```
<version>3.3.4</version>

<configuration>
    <installDirectory>/opt/ibm/wlp</installDirectory>
    <serverName>myServer</serverName>
    <appsDirectory>dropins</appsDirectory>
    <looseApplication>false</looseApplication>
    <stripVersion>true</stripVersion>
</configuration>
</plugin>
</plugins>
</build>
</project>
```

شرح:

الموسوعة الكنسية

مان SimpleServletApp ملحفه بناء قم / سيرفر خدام است باس:

```
mvn clean package
```

للتوزيع. ويعد web.xml ملفاً إلخ target/myapp.war، يجمع هذا الملف كل تفاصيل التطبيق.

٦. توزیع علی و تشغیلہ

الخطوة الخامسة: إنشاء ملف `server.xml` في المجلدة `conf` وإدخال الميزة `myServer` التي تم تأكيد خادم أن من مفعوله.

```
<featureManager>  
    <feature>ser</feature>  
</featureManager>
```

```
mvn liberty:run
```

مٰدٰت حدیثی

بالفعل. يعمل يكن لم إذا المقدمة في المدخلات الخادم يبدأ
تلقياً. مجلد إلى myapp.war يوزع
ايقافه. يتم حتى يعمل الخادم يظل

مثل: سجل رسالة عن البحث التوزيع: من تحقق

```
[AUDIT      ] CWWKT0016I: Web application available (default_host): http://localhost:9080/myapp/
```

في إعادة سجلات /opt/ibm/wlp/usr/servers/myServer/logs/console.log.

7. الوصول إلى التطبيق

إلى: وادهب وي ب متصفح افتح

<http://localhost:9080/myapp/hello>

المتوقع: الخروج

Hello World!

8. إيقاف الخادم

الواجهة. في Ctrl+C على بالضغط لإيقاف قم المقدمة، في الخادم يعمل mvn liberty:run لأن

ملاحظات

المطلبات الأساسية لـ

نظامك. على ومكوناً مثبتاً يكون أن يجب
بتعديل قم الخادم مثل هناك يكون أن يجب /opt/ibm/wlp, في مثبتاً يكون أن يجب
installDirectory و serverName أو /usr/local/wlp defaultServer. مثل مختلفاً إذا pom.xml في
فديل myServer. installDirectory و serverName أو /usr/local/wlp defaultServer.
فديل myServer. في مفعولة.servlet-4.0 المميزة تكون أن يجب

توزيع: دليل

منفصل: بشكل وتوسيع لبناء

```
mvn clean package
```

```
mvn liberty:deploy
```

الأمر: لزم إذاً دويًا الخادم بتشغل قم

```
/opt/ibm/wlp/bin/server start myServer
```

قم و `<httpEndpoint>` من تحقق مختلـفـاـ، `مـيـنـاءـ يـسـتـخـدـمـ` خـادـمـ كـانـ إـذـاـ الـمـيـنـاءـ:ـتـكـوـينـ

ـلـذـلـكـ،ـوـفـقـاـ `بـعـدـيـلـ`

إلى الـ `pom.xml` مثل `<finalName>app</finalName>` بتغيير رقم الـ `app`:
جذر `/app`.

الأخطاء: إصلاح

فی pom.xml فشل إذا installDirectory و serverName من تتحقق الالتوزيع،

الأخطاء. عن opt.ibm.wlp/usr/servers/myServer/logs/messages.log في السجلات من تتحقق إعدادات مع متوافقاً أحدث، أو 8 [] maven.compiler. اسخدام من تأكد

ملخص

تکوین web.xml و HelloServlet.java بسیط می باشد. ۱. کیفیة: الدلیل هذا یوضّح pom.xml مع myapp.war إلی التطبيقات بناء. ۲. ۳. ۴. mvn clean package إلی الوصول باستخدام liberty:run. ۵. http://localhost:9080/myapp/hello! فی

على تطبيق وتنزييع لتطويق باستخدام ممكّون مبسّطاً هنا هذا يوفر