**pom.xml**

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<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.myimg.app.ws</groupId>

<artifactId>my-imagepro</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>war</packaging>

<name>my-imagepro Maven Webapp</name>

<!-- FIXME change it to the project's website -->

<url>http://www.example.com</url>

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<maven.compiler.source>1.7</maven.compiler.source>

<maven.compiler.target>1.7</maven.compiler.target>

</properties>

<dependencies>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.11</version>

<scope>test</scope>

</dependency>

<dependency>

<groupId>org.glassfish.jersey.core</groupId>

<artifactId>jersey-server</artifactId>

<version>2.20</version>

</dependency>

<dependency>

<groupId>org.glassfish.jersey.containers</groupId>

<artifactId>jersey-container-servlet</artifactId>

<version>2.20</version>

</dependency>

</dependencies>

<build>

<finalName>my-imagepro</finalName>

<pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->

<plugins>

<plugin>

<artifactId>maven-clean-plugin</artifactId>

<version>3.1.0</version>

</plugin>

<!-- see http://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin\_bindings\_for\_war\_packaging -->

<plugin>

<artifactId>maven-resources-plugin</artifactId>

<version>3.0.2</version>

</plugin>

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.8.0</version>

</plugin>

<plugin>

<artifactId>maven-surefire-plugin</artifactId>

<version>2.22.1</version>

</plugin>

<plugin>

<artifactId>maven-war-plugin</artifactId>

<version>3.2.2</version>

</plugin>

<plugin>

<artifactId>maven-install-plugin</artifactId>

<version>2.5.2</version>

</plugin>

<plugin>

<artifactId>maven-deploy-plugin</artifactId>

<version>2.8.2</version>

</plugin>

</plugins>

</pluginManagement>

</build>

</project>

**web.xml**

<!DOCTYPE web-app PUBLIC

"-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"

"http://java.sun.com/dtd/web-app\_2\_3.dtd" >

<web-app>

<display-name>Archetype Created Web Application</display-name>

<servlet>

<servlet-name>JerseyDemo</servlet-name>

<servlet-class>org.glassfish.jersey.servlet.ServletContainer</servlet-class>

<!-- Define the ResourceConfig class -->

<init-param>

<param-name>javax.ws.rs.Application</param-name>

<param-value>service.server</param-value>

</init-param>

<load-on-startup>1</load-on-startup>

</servlet>

<!-- Map all the URLs to the Jersey ServletContainer -->

<servlet-mapping>

<servlet-name>JerseyDemo</servlet-name>

<url-pattern>/\*</url-pattern>

</servlet-mapping>

</web-app>

**client.java**

**package** app;

**import** java.io.File;

**import** javax.ws.rs.GET;

**import** javax.ws.rs.Path;

**import** javax.ws.rs.Produces;

**import** javax.ws.rs.core.Response;

**import** javax.ws.rs.core.Response.ResponseBuilder;

@Path("/Jos")

**public** **class** client {

@GET

@Path("/doc")

@Produces("image/jpg")

**public** Response getStudentFile()

{

File file=**new** File("d:\\img.jpg");

ResponseBuilder response=Response.*ok*((Object)file);

response.header("Content-Disposition","attachment;filename-DemoFile.txt");

**return** response.build();

}

}

**server.java**

**package** service;

**import** org.glassfish.jersey.server.ResourceConfig;

**public** **class** server **extends** ResourceConfig

{

**public** server()

{

packages("app");

}

}