

Maven for webapps

=====

```
archetype => maven-archetype-webapp
zipping   => war
```

Creating a maven project for webapplications

=====

```
mvn archetype:generate -DgroupId=ineuron -DartifactId=WishApp -Dversion=1.0 -
-DarchetypeArtifactId=maven-archetype-webapp -DinteractiveMode=false -Dpackaging=war -
-Dpackage=in.ineuron.controller
```

Directory structure

=====

```
WishApp
|
src/main/java(folder should be created manually)
|=> in.ineuron.controller.WishServlet.java(servlet)
src/main/resources(empty folder without any file)
src/main/webapp
|=> WEB-INF
|=> web.xml
|=> index.jsp(UI)
pom.xml
|=> servlet-api.jar
```

WishServlet.java

=====

```
package in.ineuron.controller;

import java.io.IOException;
import java.io.PrintWriter;
import java.time.LocalDateTime;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

@WebServlet("/wish")
public class WishServlet extends HttpServlet {

    private static final long serialVersionUID = 1L;

    @Override
    public void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        response.setContentType("text/html");
        PrintWriter pw = response.getWriter();
        LocalDateTime dt = LocalDateTime.now();
        int hour = dt.getHour();

        pw.println("<h1><center>Today date and time is :: " + dt +
"</center></h1>");
        if (hour < 12)
```

```

        pw.println("<h2><center>Good Morning </center></h2>");
    else if (hour < 16)
        pw.println("<h2><center>Good Afternoon</center></h2>");
    else if (hour < 20)
        pw.println("<h2><center>Good Evening</center></h2>");
    else
        pw.println("<h2><center>Good night</center></h2>");

    pw.close();
}

@Override
public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
    doGet(request, response);
}
}

```

pom.xml

=====

```

<dependencies>
    <dependency>
        <groupId>javax.servlet</groupId>
        <artifactId>javax.servlet-api</artifactId>
        <version>3.0.1</version>
        <scope>provided</scope>
    </dependency>
</dependencies>

```

index.jsp

=====

```

<html>
<body>
<h1 style='color:blue;text-align:center;'>HTML 2 SERVLET COMMUNICATION</h1>
<h2 style='color:red; text-align: center;'>
    <a href='./wish'>Generate Wish Message</a>
</h2>
</body>
</html>

```

D:\Mavenpgms\WishApp>mvn package

```

[INFO] Scanning for projects...
[INFO]
[INFO] -----< iNeuron:WishApp >-----
[INFO] Building WishApp Maven Webapp 1.0
[INFO]    from pom.xml
[INFO] -----[ war ]-----
[INFO]
[INFO] --- resources:3.3.0:resources (default-resources) @ WishApp ---
[WARNING] Using platform encoding (Cp1252 actually) to copy filtered resources,
i.e. build is platform dependent!
[INFO] Copying 0 resource
[INFO]
[INFO] --- compiler:3.10.1:compile (default-compile) @ WishApp ---
[INFO] Changes detected - recompiling the module!
[WARNING] File encoding has not been set, using platform encoding Cp1252, i.e.
build is platform dependent!
[INFO] Compiling 1 source file to D:\Mavenpgms\WishApp\target\classes
[INFO]

```

```

[INFO] --- resources:3.3.0:testResources (default-testResources) @ WishApp ---
[WARNING] Using platform encoding (Cp1252 actually) to copy filtered resources,
i.e. build is platform dependent!
[INFO] skip non existing resourceDirectory D:\Mavenpgms\WishApp\src\test\resources
[INFO]
[INFO] --- compiler:3.10.1:testCompile (default-testCompile) @ WishApp ---
[INFO] No sources to compile
[INFO]
[INFO] --- surefire:3.0.0:test (default-test) @ WishApp ---
[INFO] No tests to run.
[INFO]
[INFO] --- war:3.3.2:war (default-war) @ WishApp ---
[INFO] Packaging webapp
[INFO] Assembling webapp [WishApp] in [D:\Mavenpgms\WishApp\target\WishApp]
[INFO] Processing war project
[INFO] Copying webapp resources [D:\Mavenpgms\WishApp\src\main\webapp]
[INFO] Building war: D:\Mavenpgms\WishApp\target\WishApp.war
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 3.516 s
[INFO] Finished at: 2023-04-06T20:09:26+05:30
[INFO] -----

```

Note: gives war file copy that war file to webapps folder of tomcat server (HARD deployment)

=> run tomcat and send the request to check the response
<http://localhost:9999/WishApp>

Usage of one Maven project code in another Maven project code

mvn install => it creates the jar for the entire class and keeping the jar file in the local repository.

[under .m2 folder --> check for groupId]

D:\Mavenpgms\MathProj2> mvn tomcat7:run

|=> MathProj2 will be installed as jar file in .m2

repository with [GAV]

pom.xml

=====

```

<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/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>ineuron</groupId>
  <artifactId>WishMsgApp</artifactId>
  <packaging>war</packaging>
  <version>1.0-SNAPSHOT</version>
  <name>WishMsgApp Maven Webapp</name>
  <url>http://maven.apache.org</url>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>
  </dependencies>

```

```

</dependency>
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>javax.servlet-api</artifactId>
    <version>3.0.1</version>
    <scope>provided</scope>
</dependency>

<!-- Our project jar---->
<dependency>
    <groupId>ineuron</groupId>
    <artifactId>MathProj</artifactId>
    <version>1.0</version>
</dependency>
</dependencies>

<!-- build for maven plugin to deploy our project---->
<build>
    <finalName>WishMsgApp</finalName>
    <plugins>
        <plugin>
            <groupId>org.apache.tomcat.maven</groupId>
            <artifactId>tomcat7-maven-plugin</artifactId>
            <version>2.1</version>
            <configuration>
                <server>tomcat</server>
                <url>http://localhost:2020/manager</url>
                <path>/WishMsgApp</path>
                <port>9999</port>
            </configuration>
        </plugin>
    </plugins>
</build>
</project>

```

WishServlet.java

=====

```
package in.ineuron.controller;
```

```
import java.io.IOException;
import java.io.PrintWriter;
import java.time.LocalDateTime;
```

```
import in.ineuron.Arithmetic;//import the other project class(coming from local repository jar)
```

```
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
```

```
@WebServlet("/wish")
public class WishServlet extends HttpServlet {
```

```
    private static final long serialVersionUID = 1L;
```

```
    @Override
```

```
    public void doGet(HttpServletRequest request, HttpServletResponse response)
```

```
throws ServletException, IOException {
```

```
    response.setContentType("text/html");
    PrintWriter pw = response.getWriter();
    LocalDateTime dt = LocalDateTime.now();
    int hour = dt.getHour();
```

```
    //usage of local repository jar class
    Arithmetic ar = new Arithmetic();
    int sum = ar.add(10,20);
```

```
    pw.println("<h1><center>Today date and time is :: " + dt +
"</center></h1>");
    if (hour < 12)
        pw.println("<h2><center>Good Morning </center></h2>");
    else if (hour < 16)
        pw.println("<h2><center>Good Afternoon</center></h2>");
    else if (hour < 20)
        pw.println("<h2><center>Good Evening</center></h2>");
    else
        pw.println("<h2><center>Good night</center></h2>");

    pw.println("<h1 style='color:red;'><center>The sum is ::
"+sum+"</center></h1>");
    pw.close();
}

@Override
public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
    doGet(request, response);
}
}
```

```
cmd> mvn tomcat7:run
```

open browser and send the request
<http://localhost:9999/WishMsgApp/>

Test the App with this url: <http://localhost:9999/WishMsgApp/>

Commands to manipulate WAR file on Tomcat.

```
» mvn tomcat7:deploy //deployment
» mvn tomcat7:undeploy //undeployment
» mvn tomcat7:redploy //redeployment
```

List out the dependency scope in Maven?

The various dependency scope used in Maven are:

Compile: It is the default scope, and it indicates what dependency is available in the classpath of the project

Provided: It indicates that the dependency is provided by JDK or web server or container at runtime

Runtime: This tells that the dependency is not needed for compilation but is required during execution

Test: It says dependency is available only for the test compilation and execution phases

System: it indicates you have to provide the system path

Import: This Indicates that the identified or specified POM should be replaced with

the dependencies in that POM's section.

Rule

====

Controller(V0)

|

Service(DT0)

|

DAO(B0)