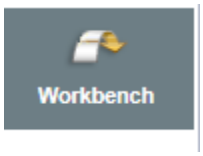
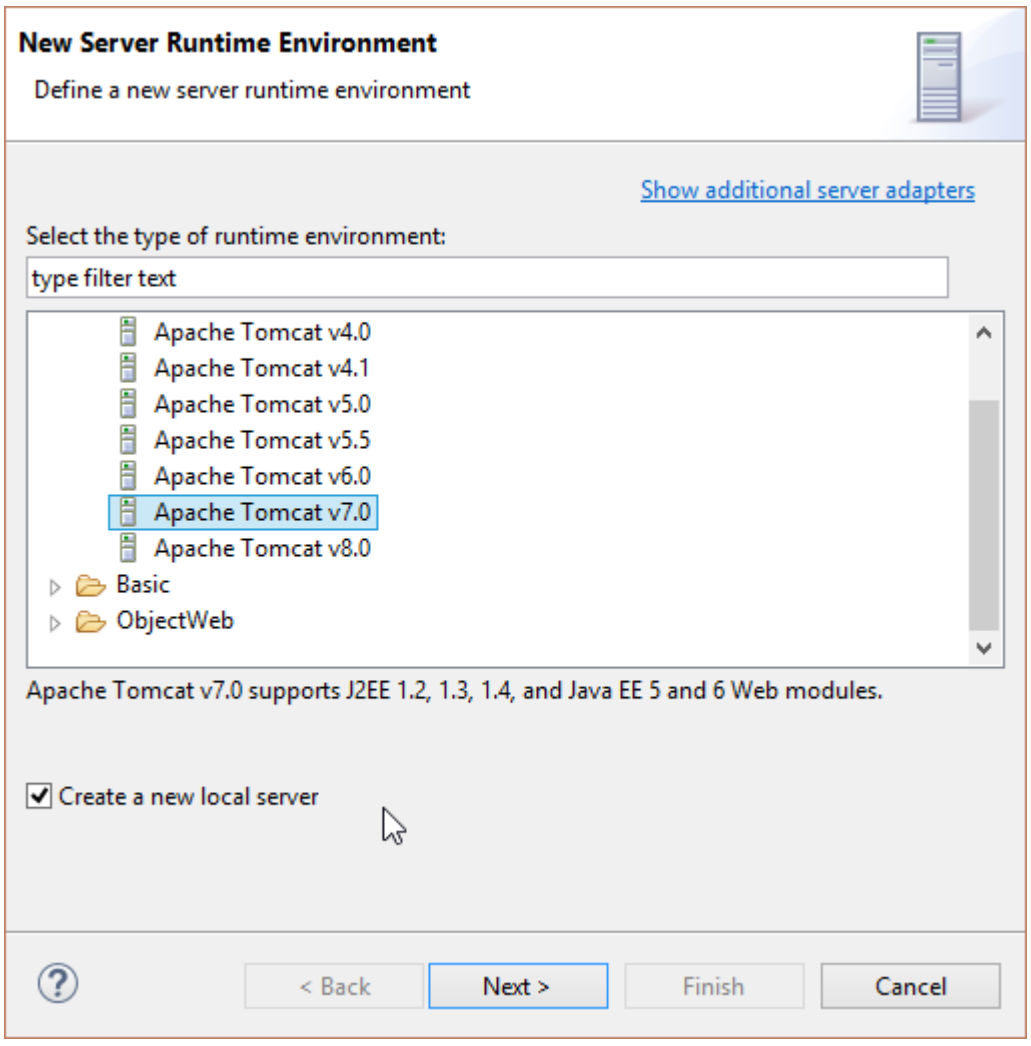
LAB 12 – Hello Cliche

1. **Launch Eclipse**
    a. Run **`eclipse.exe`** from the eclipse directory.
    b. Accept the default workspace location (or specify an alternate location) and click OK.
    c. Click the Workbench icon to close out the welcome screen.

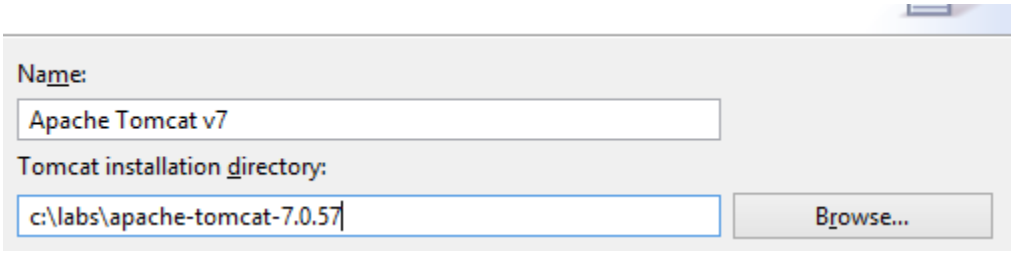


2. **Create a Project**
    a. File > New > Dynamic Web Project
    b. **Project Name**: `ClicheMessage`
    c. Click "New Runtime"
    d. Select "Apache Tomcat v7.0" (make sure "Create new local server" is checked) and click **Finish.**



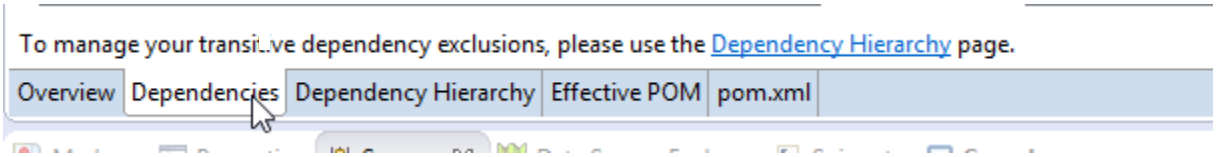
    e. Click **Next**.

f. Point to the location of the Tomcat directory.

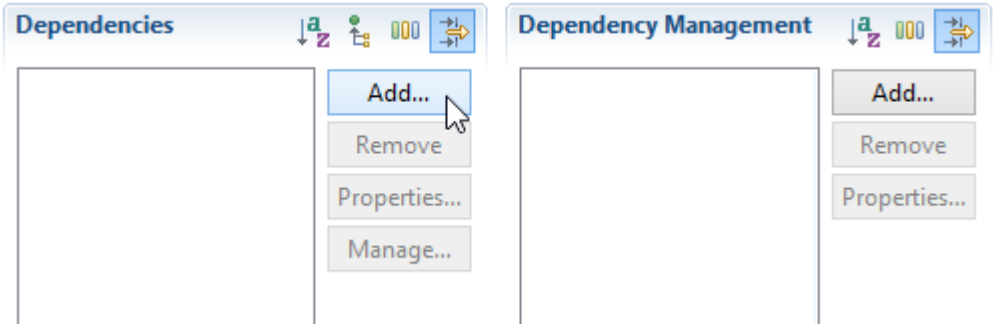


g. Click **Finish**

3. **Convert project into a Maven project**
   a. Right-click on the project. *Configure > Convert to Maven Project*
   b. Accept defaults and click **Finish**.
4. **Set Maven dependencies to include required libraries**
   a. Expand project in the Project Explorer view
   b. Double-click `pom.xml` (this is the Maven config file)
   c. Click on the "Dependencies" tab



d. Add each dependency by clicking the "Add" button on the left.



e. Add the following dependencies:

Group Id	Artifact Id	Version
asm	asm	3.3.1
org.json	json	20140107
com.sun.jersey	jersey-bundle	1.19
com.sun.jersey	jersey-core	1.19
com.sun.jersey	Jersey-server	1.19

f. Save `pom.xml` (*File > Save* or *Ctrl + S*).

5. Create a service implementation class

a. Right-click on the project. *New > Class*

b. **Package:** `com.labs.rest`

c. **Name:** `HelloCliche`

d. Click **Finish**.

6. Complete the source code for the service

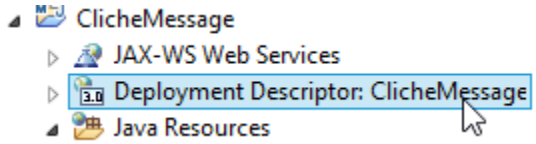
```
package com.labs.rest;

import javax.ws.rs.GET;
import javax.ws.rs.POST;
import javax.ws.rs.Path;

@Path("/hello")
public class HelloCliche {

   @GET
   public String getSomething() {
         return "Hello GET!";
   }

   @POST
   public String postSomething() {
         return "Hello POST!";
   }

}
```

a. Save `HelloCliche.java` (*File > Save* or *Ctrl + S*).

7. Configure the service container

a. Right-click the project. *Java EE Tools > Generate Deployment Descriptor Stub*

b. Open the project's Deployment Descriptor.

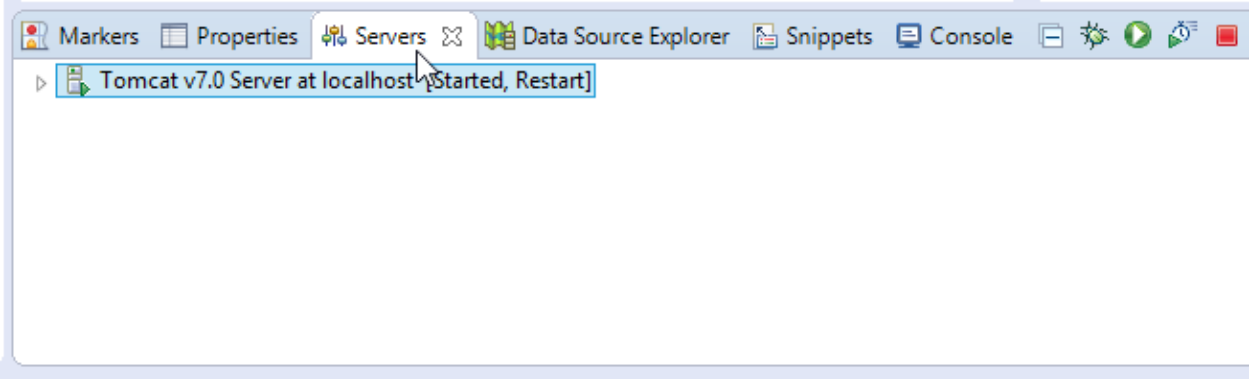


c. Add a `Servlet` and `ServletMapping` definitions below the `welcome-file-list` element.

```
<servlet>
  <servlet-name>Cliche Rest Service</servlet-name>
  <servlet-class>
    com.sun.jersey.spi.container.servlet.ServletContainer
  </servlet-class>
  <load-on-startup>1</load-on-startup>
</servlet>

<servlet-mapping>
  <servlet-name>Cliche Rest Service</servlet-name>
  <url-pattern>/cliche/*</url-pattern>
</servlet-mapping>
```

8. Deploy project to server

a. Click the *Servers* tab directly below the editor.



b. Right-click on the Tomcat server. Select *"Add and Remove…"*

c. Click the project name on the left, click the "Add" button, and click **Finish**.

9. Test the service

a. Click the green play button on the right side of the *Servers* tab.

b. Test the service using your browser.

   i. Launch browser of your choice.

   ii. Navigate to:
   `http://localhost:8080/ClicheMessage/cliche/hello/`

   iii. Result should be: Hello GET!

c. Test the service using a simple form.

   i. Right-click on *WebContent* folder and select "Import…"

   ii. *General > File System*

   iii. Click "Browse" and select the *Lab12* folder.

   iv. Select the `RestTester.html` file.

   v. Click **Finish**.

   vi. Navigate your browser to:
   `http://localhost:8080/ClicheMessage/RestTester.html`

   vii. Experiment with GET and POST requests