LAB 14 – Client Application

1. **Launch Eclipse**
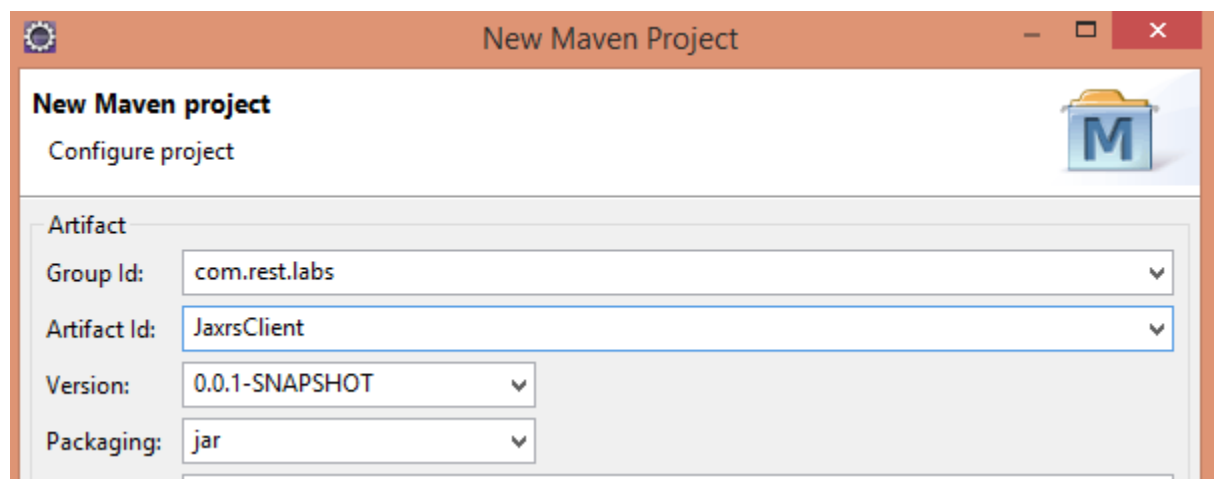   a. Accept the default workspace
2. **Create a Project**
   a. *File > New > Maven Project*
   b. **Project Name**: `Calculator`
   c. Click **Finish**.
3. **Convert project into a Maven project**
   a. Right-click on the project. *Configure > Convert to Maven Project*
   b. Check the first two checkboxes.
   c. Click **Next**.
   d. **Group Id**: `com.rest.labs`
   e. **Artifact Id**: `JaxrsClient`

   f. Click **Finish**.
4. **Set Maven dependency to include required library**
   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 dependency by clicking the "Add" button on the left.
   e. Add one dependency:

Group Id	Artifact Id	Version
com.sun.jersey	jersey-client	1.19

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

5. Create a service client class

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

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

    c. **Name:** `RestClient`

    d. Click **Finish**.

6. Complete the source code for the client

```java
package com.rest.labs;

import com.sun.jersey.api.client.Client;
import com.sun.jersey.api.client.ClientResponse;
import com.sun.jersey.api.client.WebResource;

public class RestClient {

    static Client client = Client.create();
    static String getUrl = ""; //TODO: Set URL

    public static void main(String[] args){
        getRequest();
    }

    public static void getRequest(){
        WebResource webResource = client.resource(getUrl);

        //TODO: Set MIME type inside the accept() method call
        ClientResponse response =
          webResource.accept("").get(ClientResponse.class);

        if( response.getStatus() != 200 ) {
            throw new RuntimeException("HTTP Error: "+ response.getStatus());
        }

        //TODO: Display response data

    }
}
```
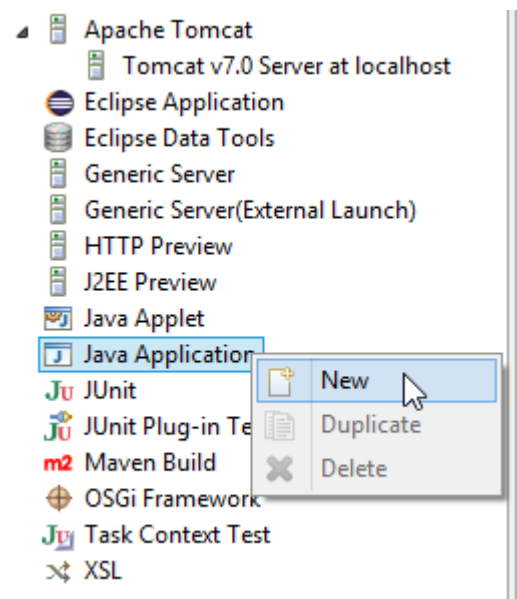
    a. Update the class by following the instructions in the `TODO` comments.

    b. Save `RestClient.java` (*File > Save* or *Ctrl + S*).

7. Setup the Run Configuration to launch the client application

    a. Right-click the project. *Run As > Run Configurations...*

    b. Right-click *Java Application* on the left and select *New*.

Apache Tomcat
    Tomcat v7.0 Server at localhost
Eclipse Application
Eclipse Data Tools
Generic Server
Generic Server(External Launch)
HTTP Preview
J2EE Preview
Java Applet
Java Application   New
JUnit   Duplicate
JUnit Plug-in Te
m2 Maven Build   Delete
OSGi Framework
Task Context Test
XSL

    c. Complete the fields in this window
        i. **Name**: *Java Run Config*
        ii. **Project**: *JaxrsClient*
        iii. **Main class**: `com.rest.labs.RestClient`
    d. Click **Apply**.
    e. Click **Close**.

8. Test the client app

    a. If you are accessing a local service, be sure the server is running.
    b. Right-click the project. *Run As > Java Application*
    c. Check the *Console* view for output from the service call.