**Configuring a Servlet with Eclipse**

This section will guide you to:

* Configure a Servlet with Eclipse

**Development Environment**

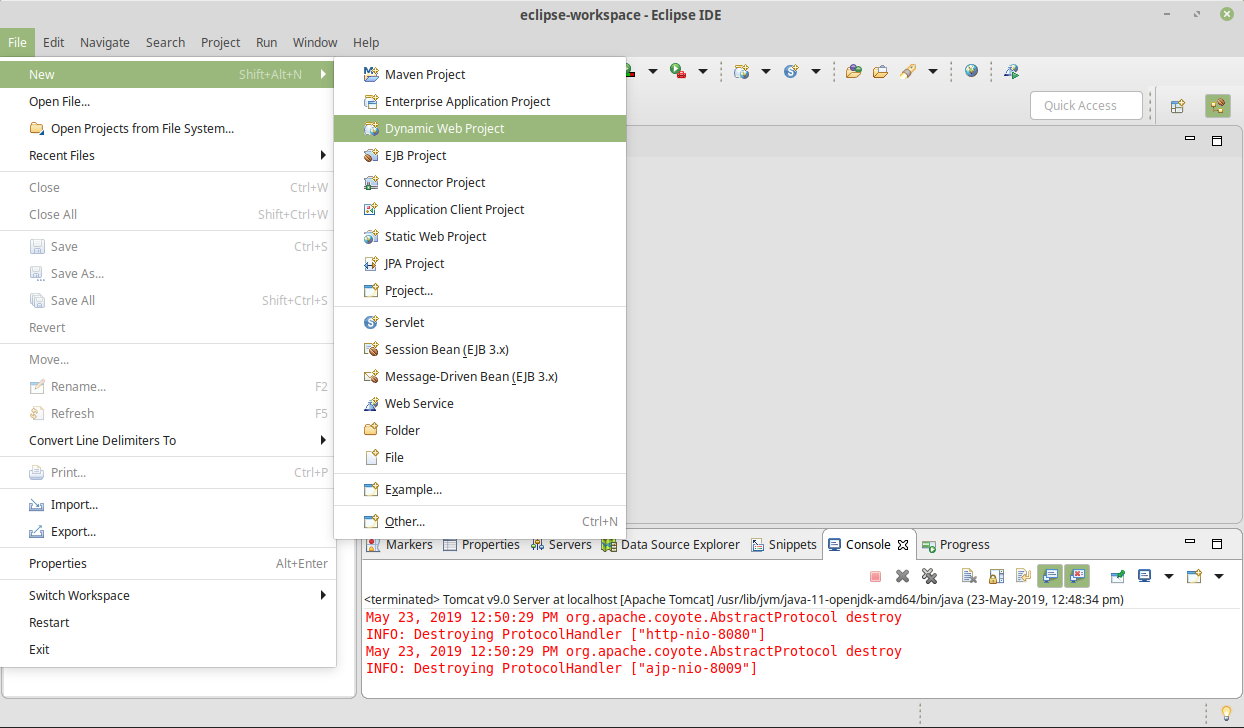
* Eclipse IDE for Enterprise Java Developers v2019-03 (4.11.0)
* Apache Tomcat Server v9.0
* JRE: OpenJDK Runtime Environment 11.0.2

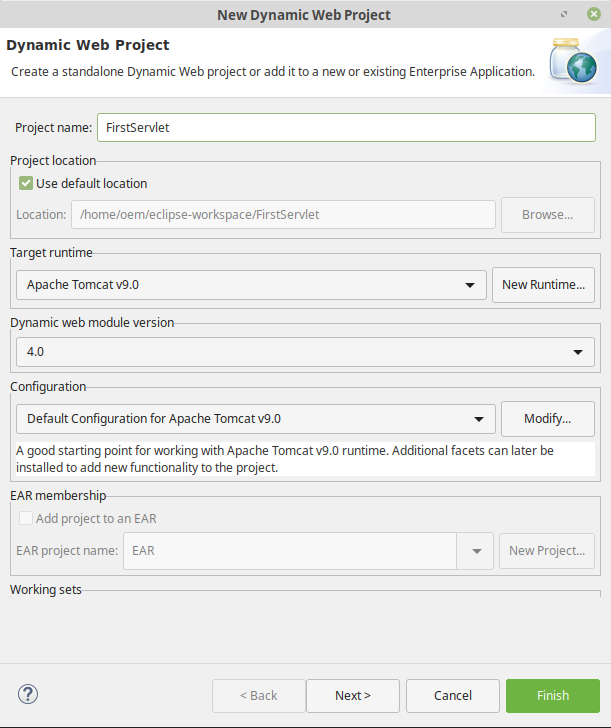
This lab has seven subsections, namely:

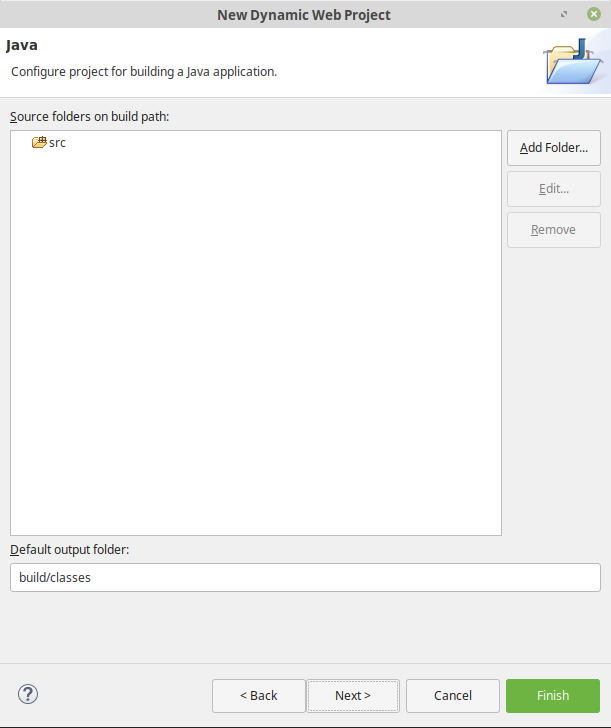
* + 1. Creating a dynamic web project
    2. Creating a servlet
    3. Checking for servlet-api.jar
    4. Building the project
    5. Publishing and starting the project
    6. Running the project
    7. Pushing the code to your GitHub repositories

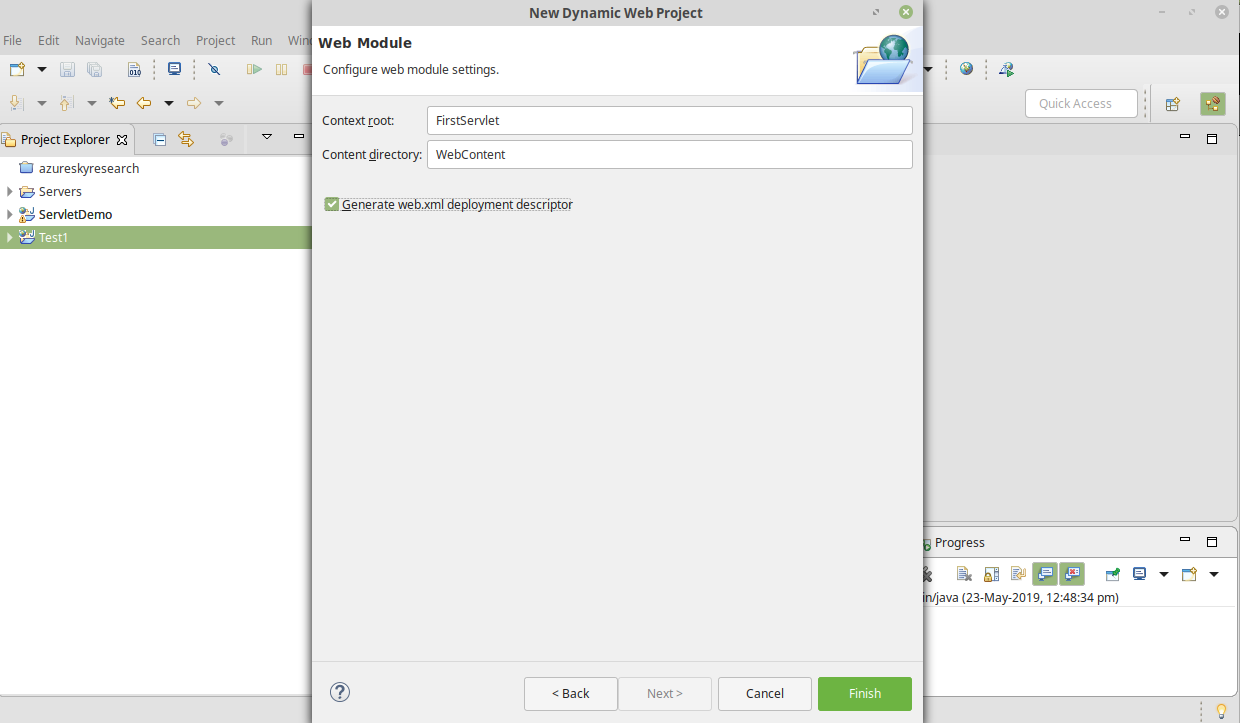
**Step 1.2.1:** Creating a dynamic web project

* Open Eclipse
* Go to the **File** menu. Choose **New->Dynamic Web Project**
* Enter the project name as **FirstServlet**. Click on **Next**
* Enter nothing in the next screen and click on **Next**
* Check the checkbox **Generate web.xml deployment descriptor** and click on **Finish**
* This will create the project files in the Project Explorer



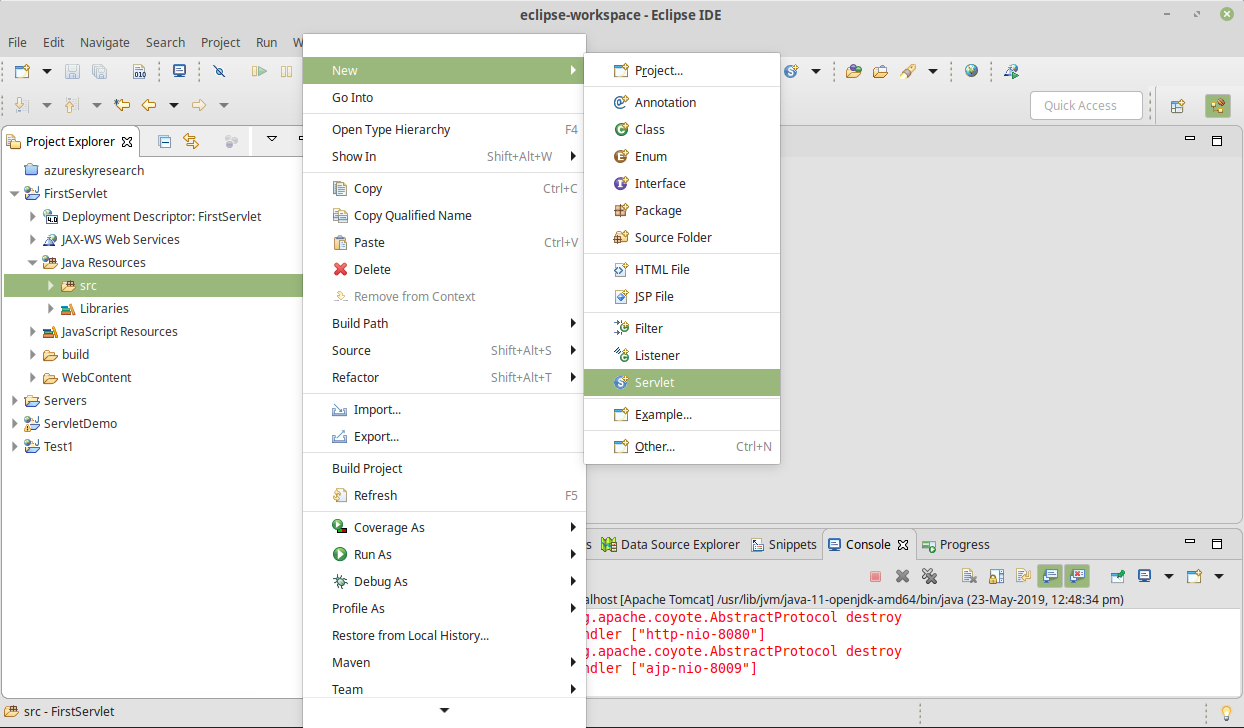


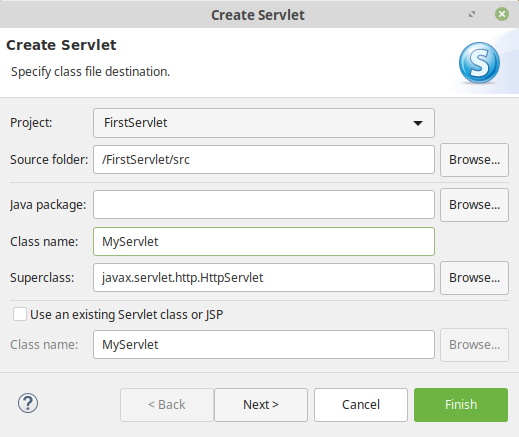
****

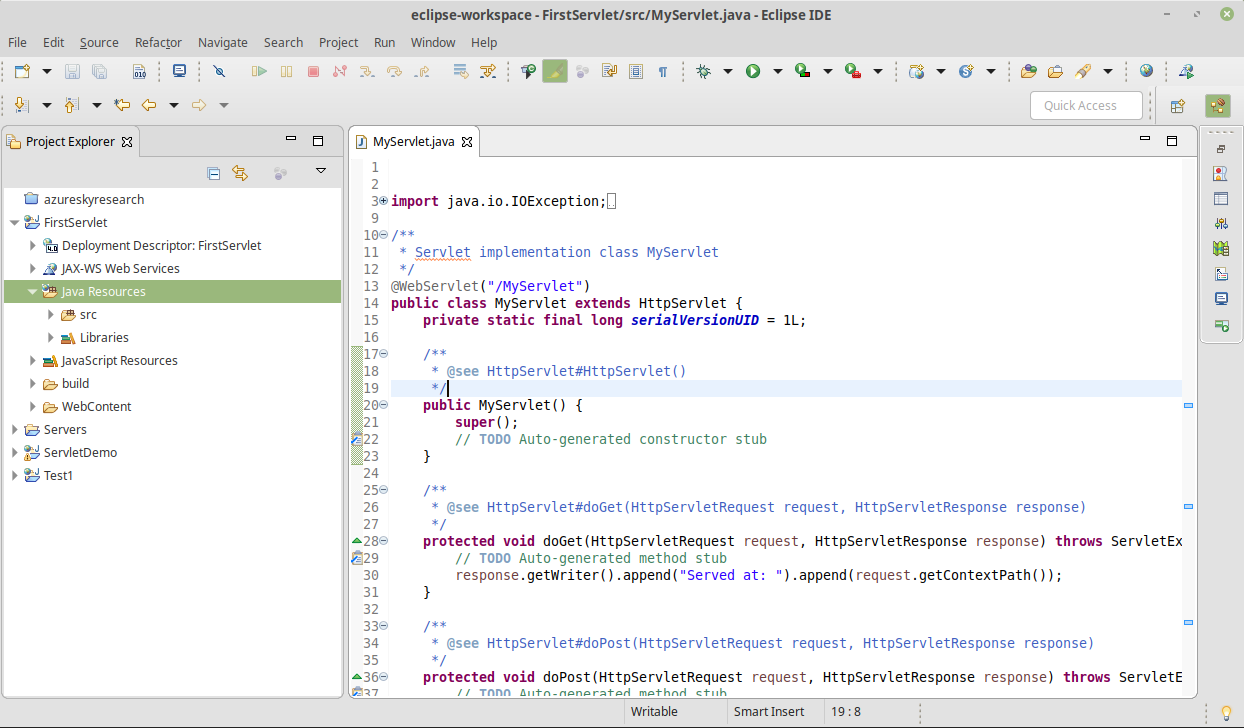
****

**Step 1.2.2:** Creating a servlet

* In the Project Explorer, expand **ServletConcept->Java Resources**
* Right click on **src** and choose **New->Servlet**
* In **Class Name,** enter **FirstServlet** and click on **Finish**



****

****

**Step 1.2.3:** Checking for servlet-api.jar

* Before building the project, we need to add servlet-api.jar to the project
* Servlet-api.jar file is already present in your practice lab. (Refer FSD: Lab Guide - Phase 2)
* To add it to the project, follow the below mentioned steps:
  + In the Project Explorer, right click on **FirstServlet** and choose **Properties**
  + Select **Java Build Path** from the options on the left
  + Click on **Libraries** tab on the right
  + Under **ClassPath,** expand the node that says **Apache Tomcat**
  + If there is an existing entry for servlet-api.jar, then click on **Cancel** and exit the window
  + If it is not there, then click on **Classpath** entry and click on **Add External JARs** button on the right
  + From the file list, select servlet-api.jar file and click **Ok**
  + Click on **Apply and Close**

**Step 1.2.4:** Building the project

* From the **Project** menu at the top, click on **Build**
* If any compile errors are shown, fix them as required

**Step 1.2.5:** Publishing and starting the project

* If you do not see the **Servers** tab near the bottom of the IDE, go to **Window** menu and click on **Show View->Servers**
* Right click on the **Server** entry and choose **Add and Remove**
* Click the **Add** button to move **FirstServlet** from the **Available** list to the **Configured** list
* Click on **Finish**
* Right click on the **Server** entry and click on **Publish**
* Right click on the **Server** entry and click on **Start**
* This will start the server

**Step 1.2.6:** Running the project

* To run the project, open a web browser and type: [**http://localhost:8080/ServletConcept**](http://localhost:8080/ServletConcept)

**Step 1.2.7:** Pushing the code to your GitHub repositories

* Open your command prompt and navigate to the folder where you have created your files.

**cd <folder path>**

* Initialize your repository using the following command:

**git init**

* Add all the files to your git repository using the following command:

**git add .**

* Commit the changes using the following command:

**git commit . -m “Changes have been committed.”**

* Push the files to the folder you initially created using the following command:

**git push -u origin master**