UNIVERSITI MALAYSIA TERENGGANU

CSM3023 WEB BASED APPLICATION DEVELOPMENT (K1)

BACHELOR OF COMPUTER SCIENCE (MOBILE COMPUTING) WITH HONORS

LAB 1 – INTRODUCTION TO SERVLET, JSP AND MYSQL DATABASE

SEMESTER 4 2024/2025

Prepared for:

DR. MOHAMAD NOR HASSAN

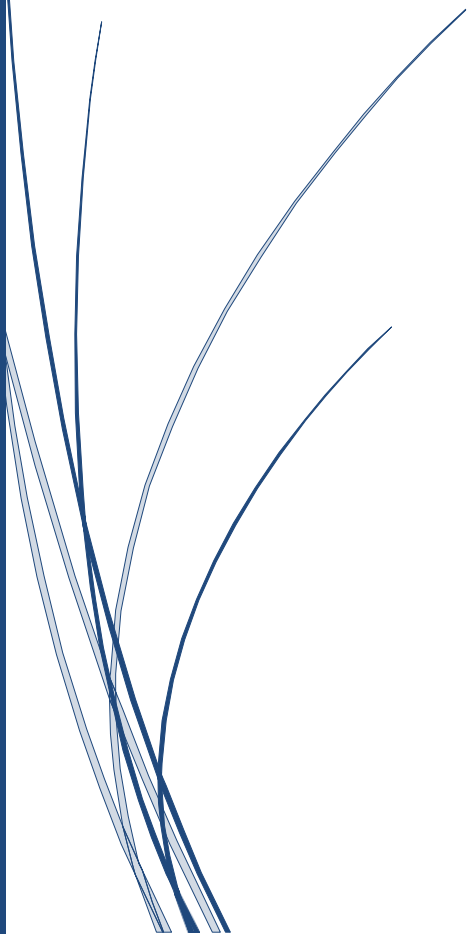
Prepared by:

NUR ARINA BINTI ABDUL MALEK (S65361)

Week 1

Introduction to Servlet, JSP and MySQL Database

Web Based Application Development



Lecturers

PUSAT PENGAJIAN INFORMATIK DAN MATEMATIK GUNAAN (PPIMG), UNIVERSITI MALAYSIA TERENGGANU (UMT)

Revision History

Revision Date	Previous Revision Date	Summary of Changes	Changes Marked
		First Issue	Mohamad Nor Hassan
		Second Issue	Dr Rabiei Mamat Dr Faizah Aplop Dr Fouad Ts Dr Rosmayati Mohemad Fakhrul Adli Mohd Zaki
21/02/2019		Addition of Revision History, Table of Contents, Formatting Cover Page	Fakhrul Adli Mohd Zaki
24/02/2019	21/02/2019	Updated Task 1, English translation for the instruction, Addition of Java Servlet task. Fix spelling and grammar.	Fakhrul Adli Mohd Zaki
30/3/2021		Added "Troubleshooting Notes: Fixing phpMyAdmin Error" Added new topic "Managing Apache Tomcat" Added new topic "Netbeans 12.3 IDE Installation" Updated existing topic "Link Netbeans" to "Apache Tomcat and Writing a Simple Java Servlet" Updated page numbers	Fakhrul Adli Mohd Zaki

Table of Contents

Task 1: Apache Tomcat and MySQL Installation Using XAMPP 5

Task 2: Change the Default Root Password of MySQL Database 11

Task 3: Managing Apache Tomcat 18

Task 4: Netbeans 12.3 IDE Installation 22

Task 5: Linking Netbeans to Apache Tomcat and Writing a Simple Java Servlet 25

Task 6: Writing a Simple JSP Program 38

Task 7: Use Java Reference Datatype/Class Wrapper in JSP 44

Task 8: Using JSP Implicit object in JSP page 47

Task 9: Populate Array values into HTML's Table 51

Arahan:

Manual makmal ini adalah untuk kegunaan pelajar-pelajar Pusat Pengajian Informatik dan Matematik Gunaan (PPIMG), Universiti Malaysia Terengganu (UMT) sahaja. Tidak dibenarkan mencetak dan mengedar manual ini tanpa kebenaran rasmi daripada penulis.

Sila ikuti langkah demi langkah sebagaimana yang dinyatakan di dalam manual. Tandakan (√) setiap langkah yang telah selesai dibuat dan tulis kesimpulan bagi setiap aktiviti yang telah selesai dijalankan.

Instruction:

This laboratory manual is for use by the students of the School of Informatics and Applied Mathematics (PPIMG), Universiti Malaysia Terengganu (UMT) only. It is not permissible to print and distribute this manual without the official authorisation of the author.

Please follow step by step as described in the manual. Tick (√) each step completed and write the conclusions for each completed activity.

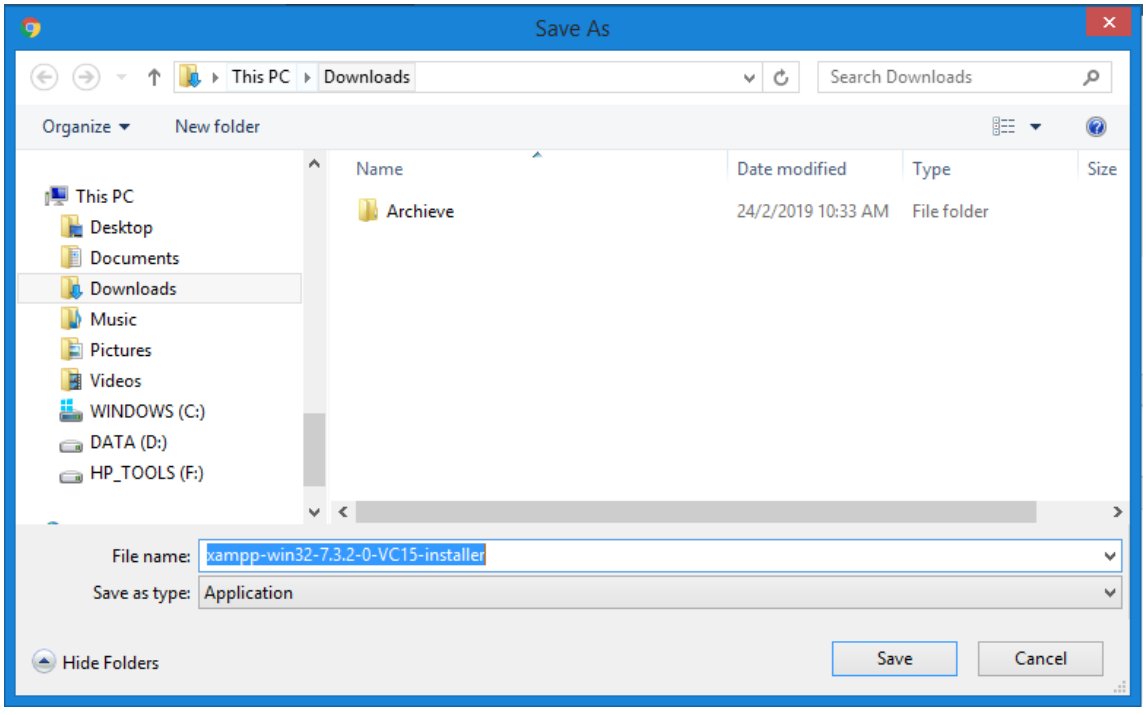
Task 1: Apache Tomcat and MySQL Installation Using XAMPP

Objective	:	Installation of Apache Tomcat and MySQL
Problem Description	:	To install Apache Tomcat and MySQL
Estimated time	:	25 minutes

1. Go to the browser and type URL
   http://www.apachefriends.org/en/xampp.html

2. Click to XAMPP for Windows.

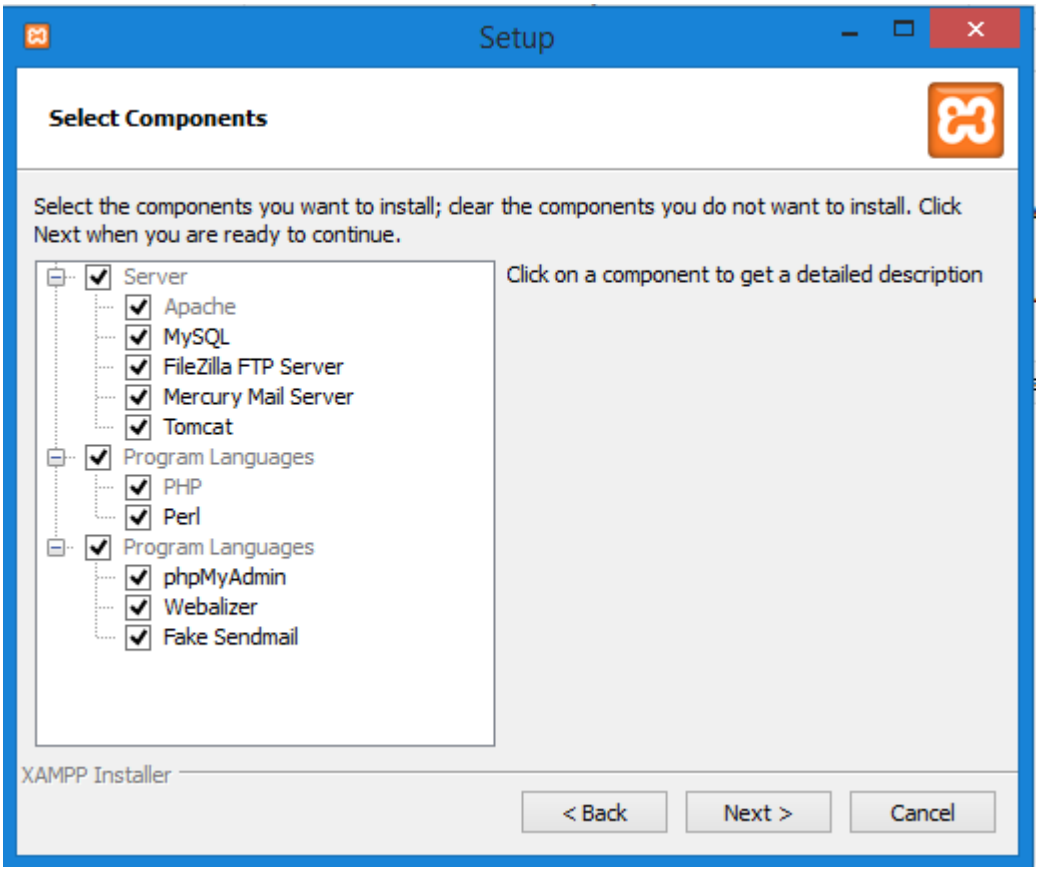
3. Save to a specific directory — for example, the Downloads folder.



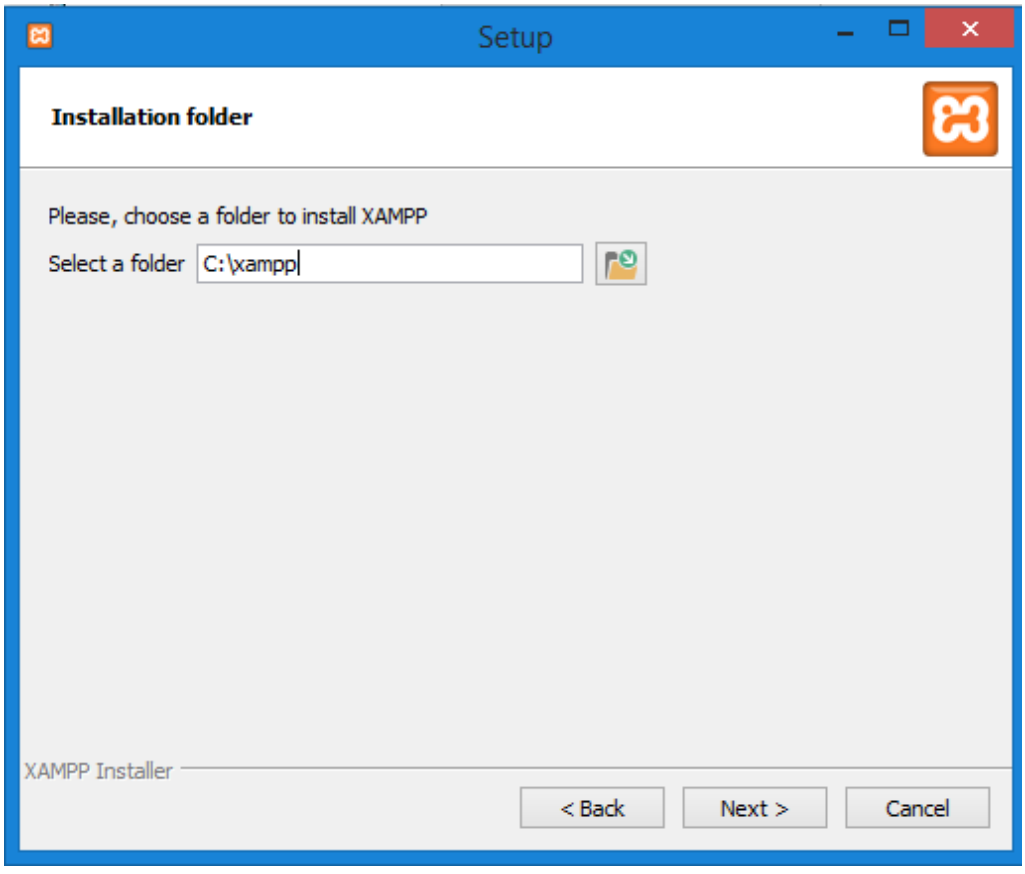
5. Run file xampp-win32-**[X.X.X-VCXX]**-installer.exe
Note: [X.X.X-VCXX] refers to the current version of the installer. It might differ with the version you see on the XAMPP website.

6. Click the Next button.

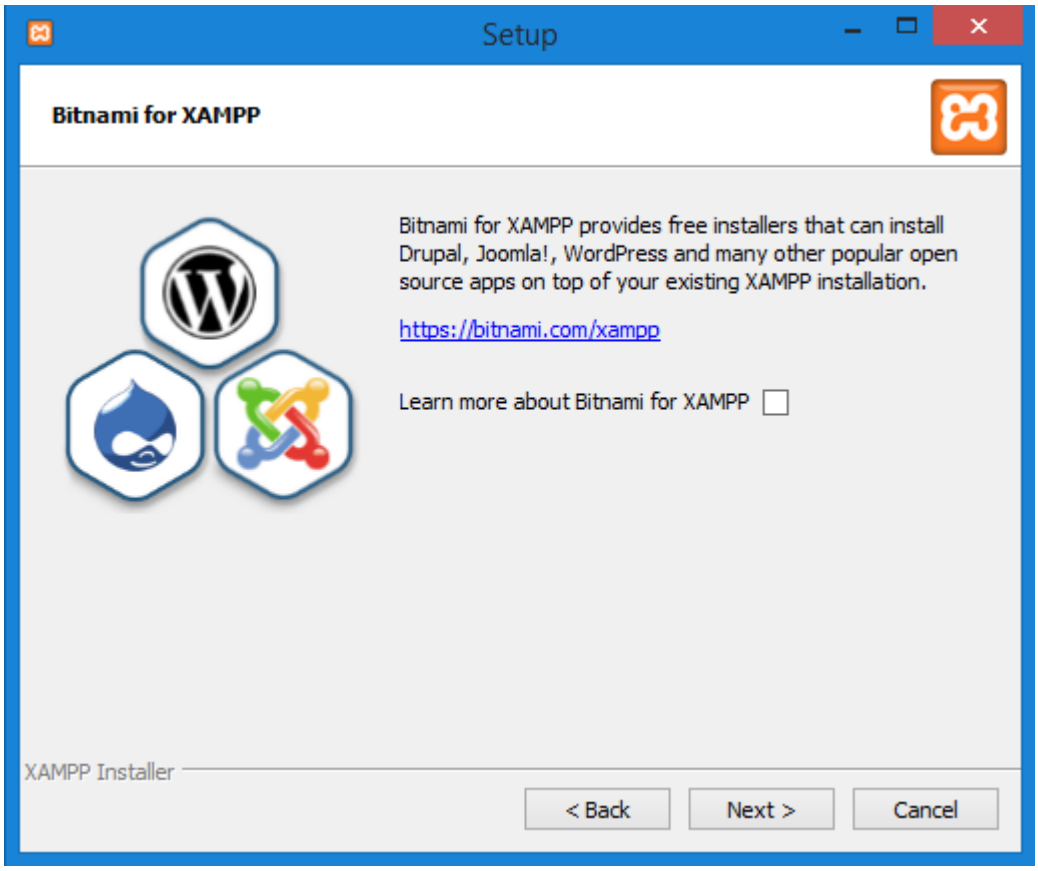
7. Please ensure you choose **Tomcat** and **MySQL**.



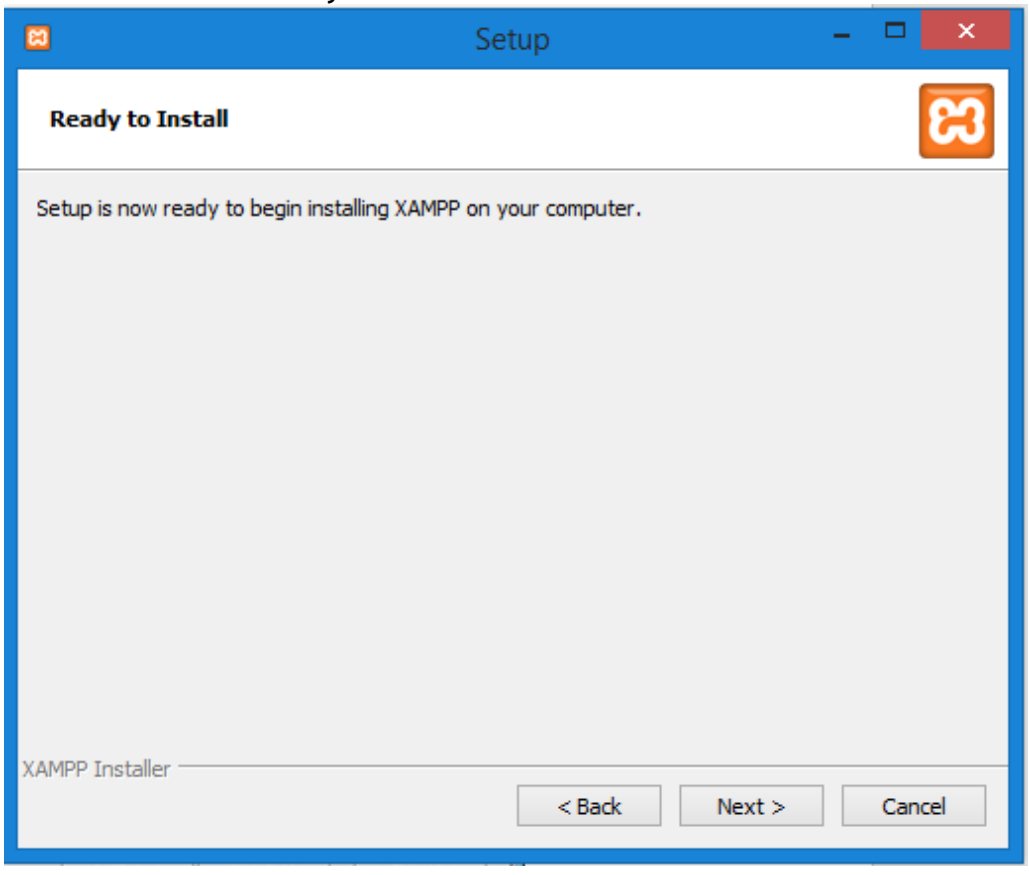
8. Choose a folder as C: \xampp and click the Next > button.



10. Untick “Learn more about Bitnami for XAMPP” and click Next button.

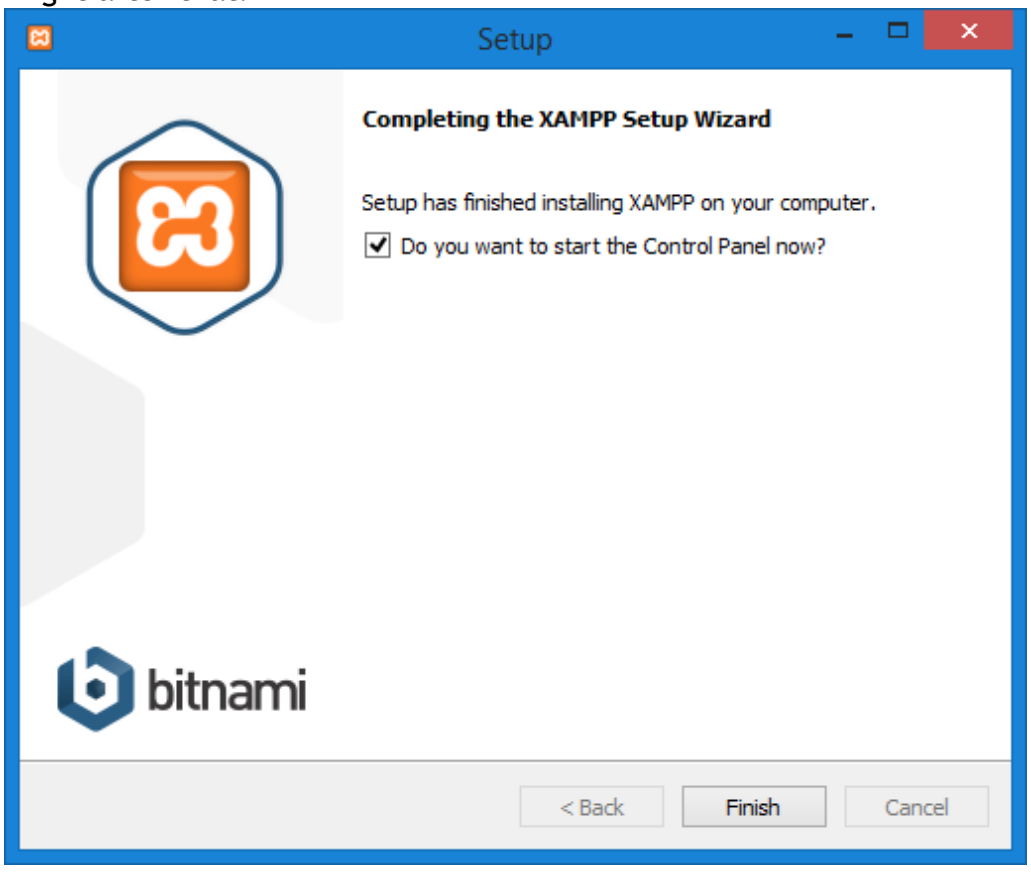


11. XAMPP is now ready to install.

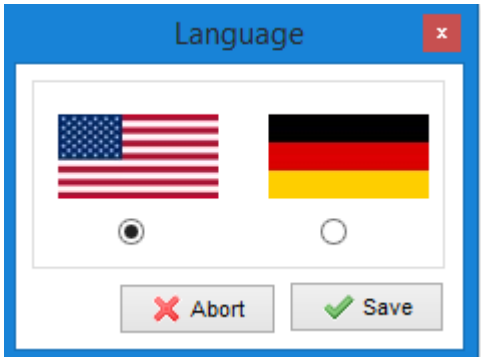


12. Wait for the installation process to complete.

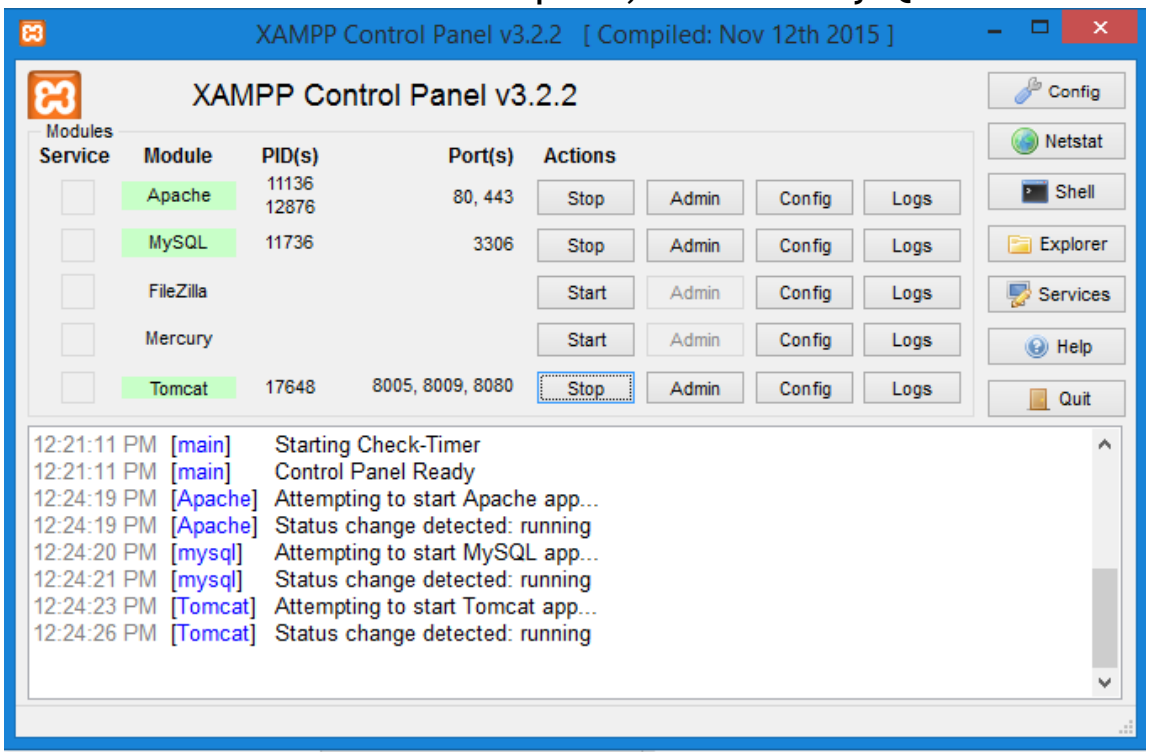
13. Click Finish button once completed and the Control Panel should start right after that.



14. Only for first time launching, select English as the default language for XAMPP. Click the Save button.



15. Click the Start button to launch **Apache, Tomcat** dan **MySQL** service.



Note: Why do we need Apache? We will use phpmyadmin as a tool to manage our database in the upcoming tasks. You may use other tools such as MySQL Workbench, Netbeans Database Manager etc. Do not confuse with the functions of various instruments used during the JSP development. Try to understand the context of the usage of each of them.

Reflection

What have you learnt from this exercise?

Task 2: Change the Default Root Password of MySQL Database

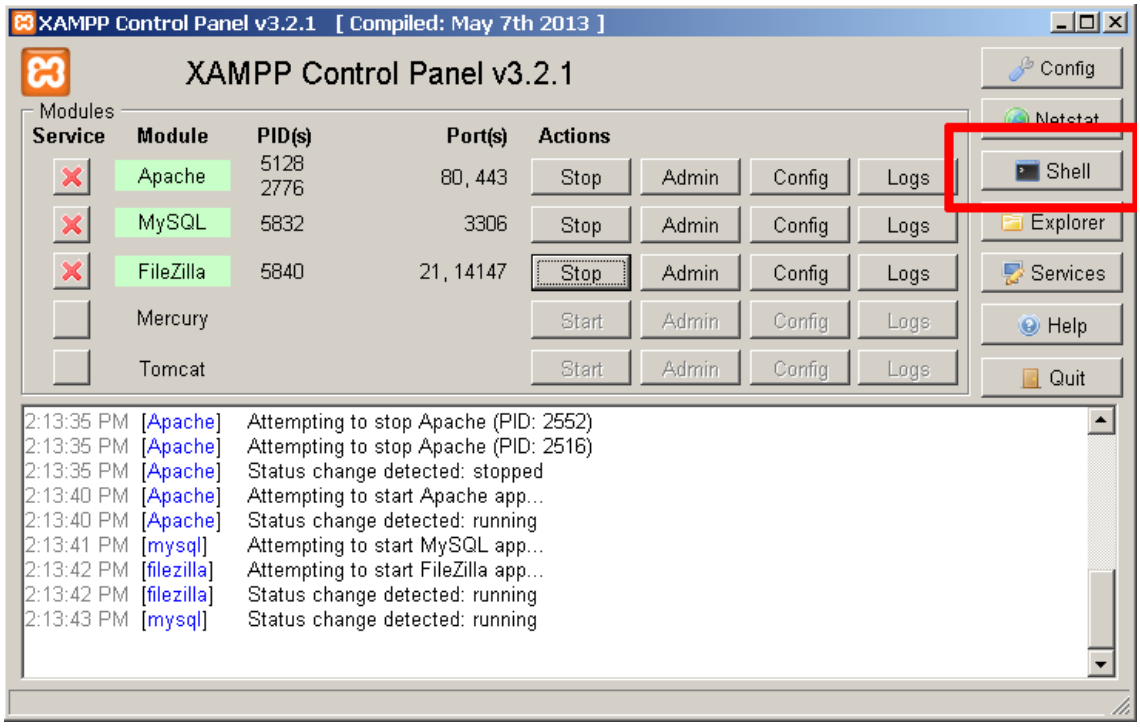
Objective:	Change the root password of MySQL Database
Problem Description:	To reset a new root password for MySQL Database
Estimated time:	5 minutes

By default, the MySQL installation that ships with XAMPP have an empty root password. It is a severe security risk, especially if you plan to use XAMPP in production scenarios.

To change the MySQL root password, follow these steps:

1. Ensure that the MySQL server is running.

2. Open your Windows command prompt by clicking the "Shell" button in the XAMPP control panel.

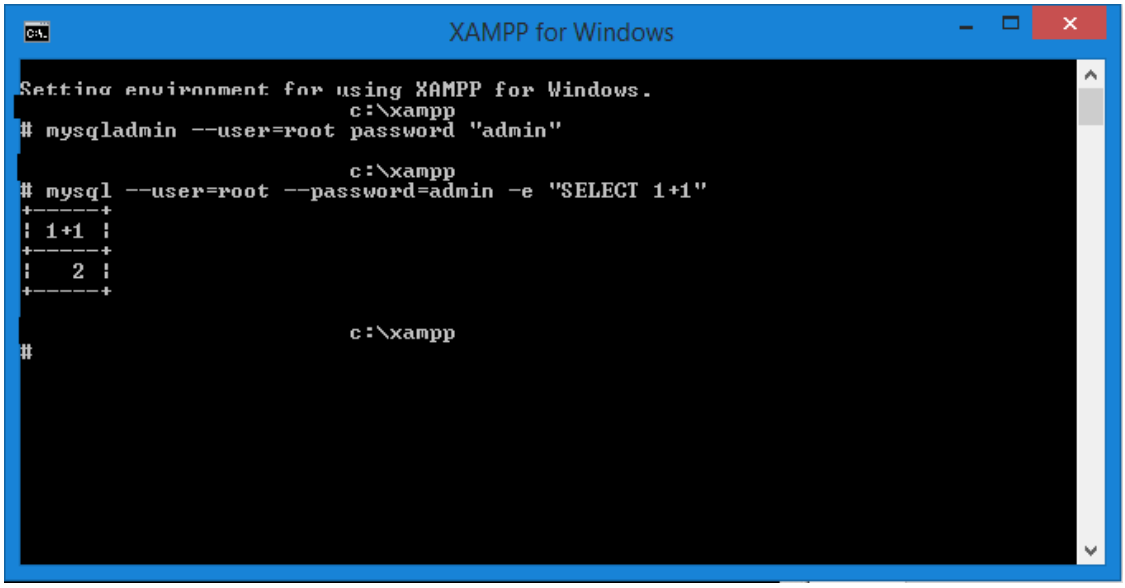


3. Use the mysqladmin command-line utility to alter the MySQL password, using the following syntax (Note: We will use “admin” as our password during the lesson):

```
mysqladmin --user=root password "admin"
```

4. To test that your password change has been accepted, by attempting to connect to the MySQL server using the MySQL command-line client in the same directory. For example, you could use the command below to connect to the server and return the results of a calculation:

   ```
   mysql --user=root --password=admin -e "SELECT 1+1"
   ```



5. To update the password in the future, you can use the following syntax:

   ```
   mysqladmin --user=root --password=oldpassword
   password "newpassword"
   ```

6. To test that your password change has been accepted, by attempting to connect to the MySQL server using the MySQL command-line client in the same directory. For example, you could use the command below to connect to the server and return the results of a calculation:

   ```
   mysql --user=root --password=gue55me -e "SELECT 1+1
   ```

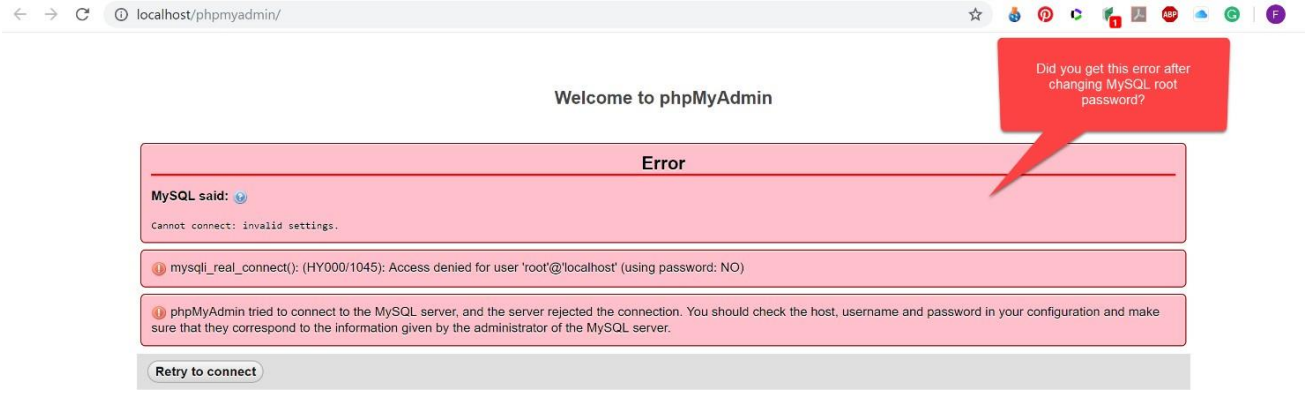
IMPORTANT:
To avoid forgetting the password during our lesson, please set the MySQL password as ***admin*** ONLY..!

Remark: The above instructions are taken from Apache Friends Documentation. You may retrieve the documentation at http://localhost/dashboard/docs/reset-mysql-password.html after done with the XAMPP installation.

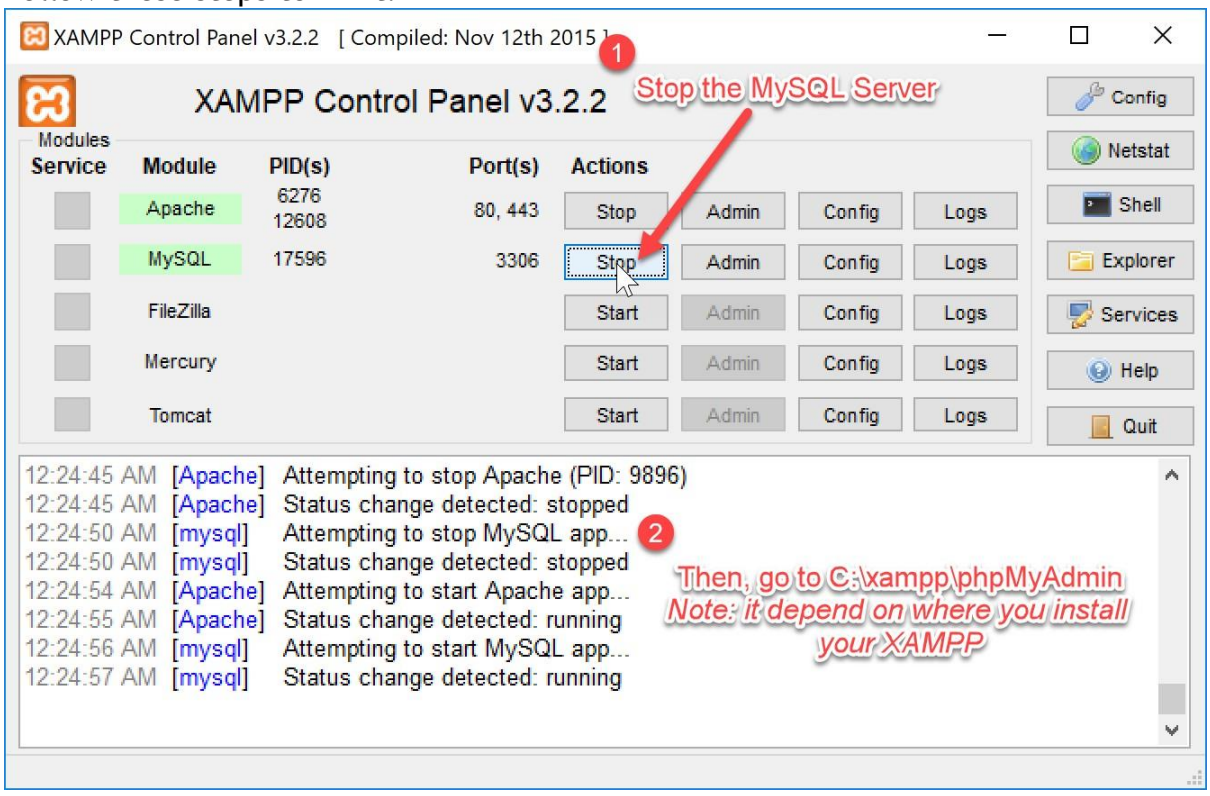
Reflection
What have you learned from this exercise?

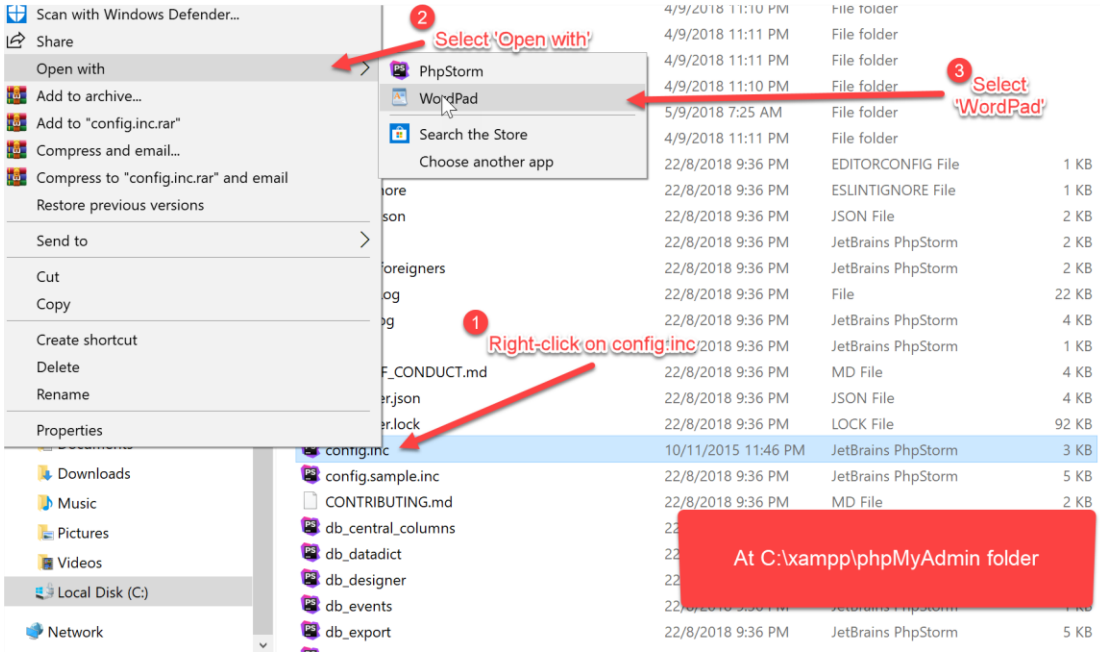
Troubleshooting Notes: Fixing phpMyAdmin Error

Did you get an error page when opening the phpMyAdmin?



Follow these steps to fix it.



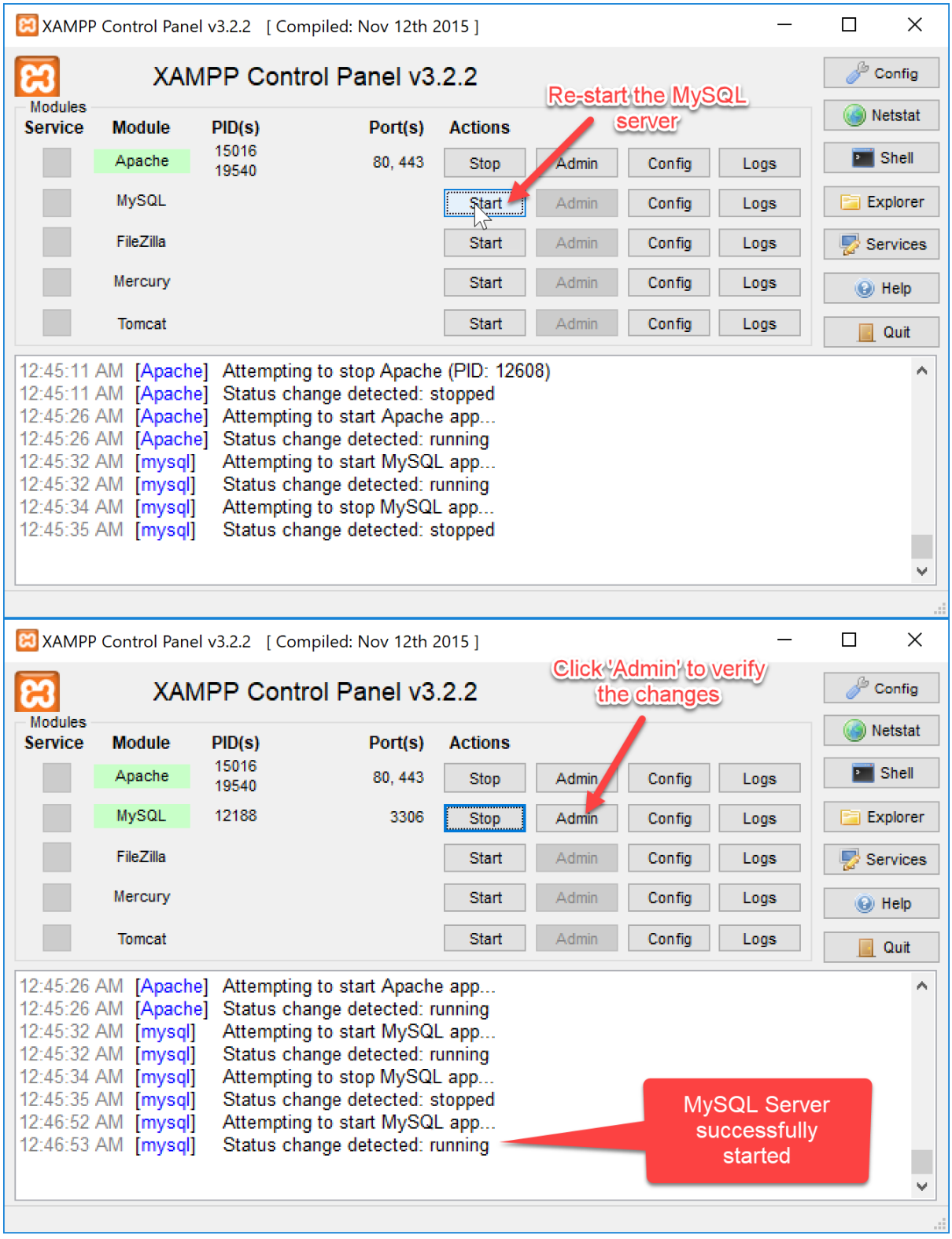


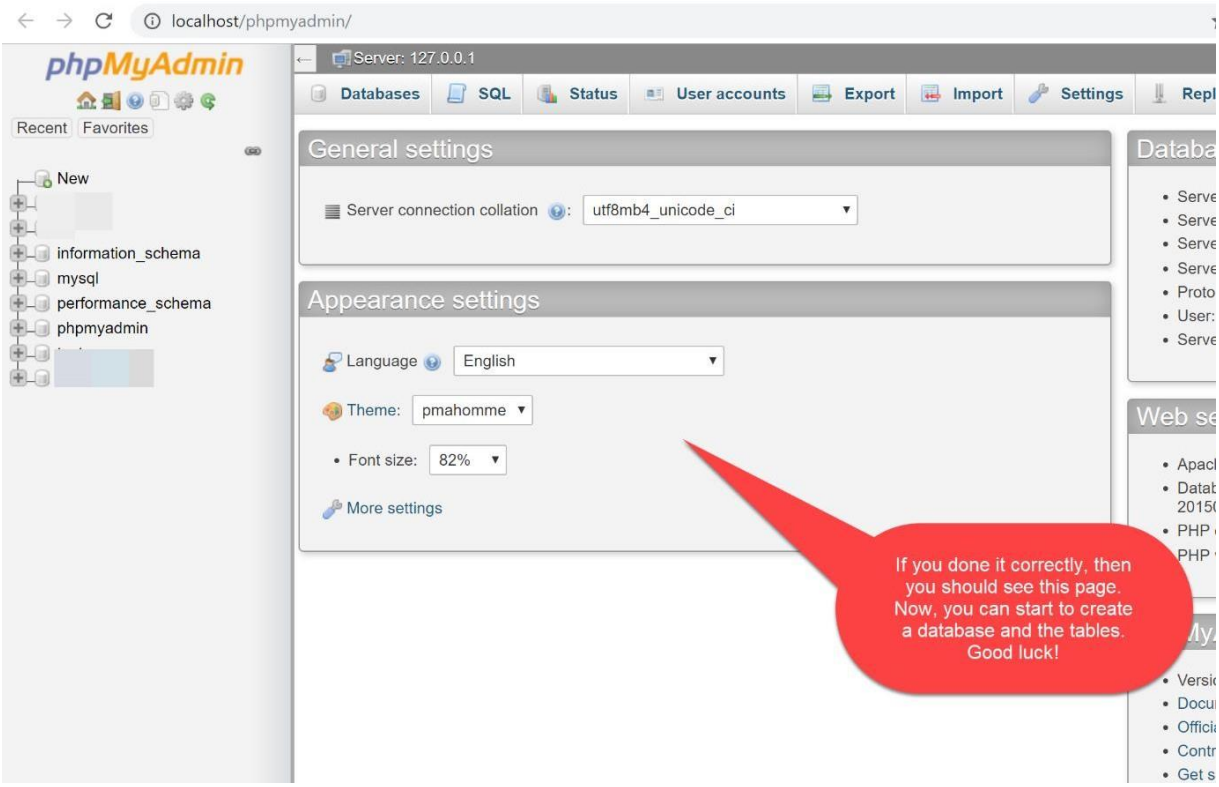

```
password in
 * cookie
 */
$cfg['blowfish_secret'] = 'xampp'; /* YOU SHOULD CHANGE THIS
FOR A MORE SECURE COOKIE AUTH! */

/*
 * Servers configuration
 */
$i = 0;

/*
 * First server
 */
$i++;

/* Authentication type and info */
$cfg['Servers'][$i]['auth_type'] = 'config';
$cfg['Servers'][$i]['user'] = 'root';
$cfg['Servers'][$i]['password'] = 'admin';
$cfg['Servers'][$i]['extension'] = 'mysqli';
$cfg['Servers'][$i]['AllowNoPassword'] = false;
$cfg['Lang'] = '';

/* Bind to the localhost ipv4 address and tcp */
$cfg['Servers'][$i]['host'] = '127.0.0.1';
$cfg['Servers'][$i]['connect_type'] = 'tcp';

/* User for advanced features */
$cfg['Servers'][$i]['controluser'] = 'pma';
$cfg['Servers'][$i]['controlpass'] = '';

/* Advanced phpMyAdmin features */
$cfg['Servers'][$i]['pmadb'] = 'phpmyadmin';
$cfg['Servers'][$i]['bookmarktable'] = 'pma__bookmark';
$cfg['Servers'][$i]['relation'] = 'pma__relation';
$cfg['Servers'][$i]['table_info'] = 'pma__table_info';
$cfg['Servers'][$i]['table_coords'] = 'pma__table_coords';
$cfg['Servers'][$i]['pdf_pages'] = 'pma__pdf_pages';
$cfg['Servers'][$i]['column_info'] = 'pma__column_info';
```

1 Put your root password here. In this example, 'admin' is the root password.

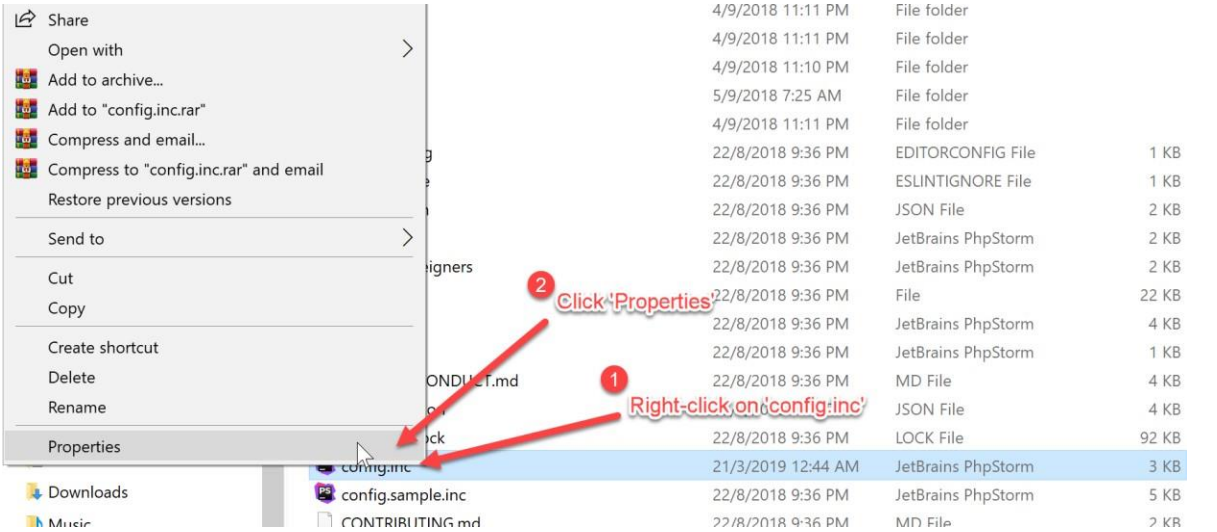
2 Change the value from 'true' to 'false'

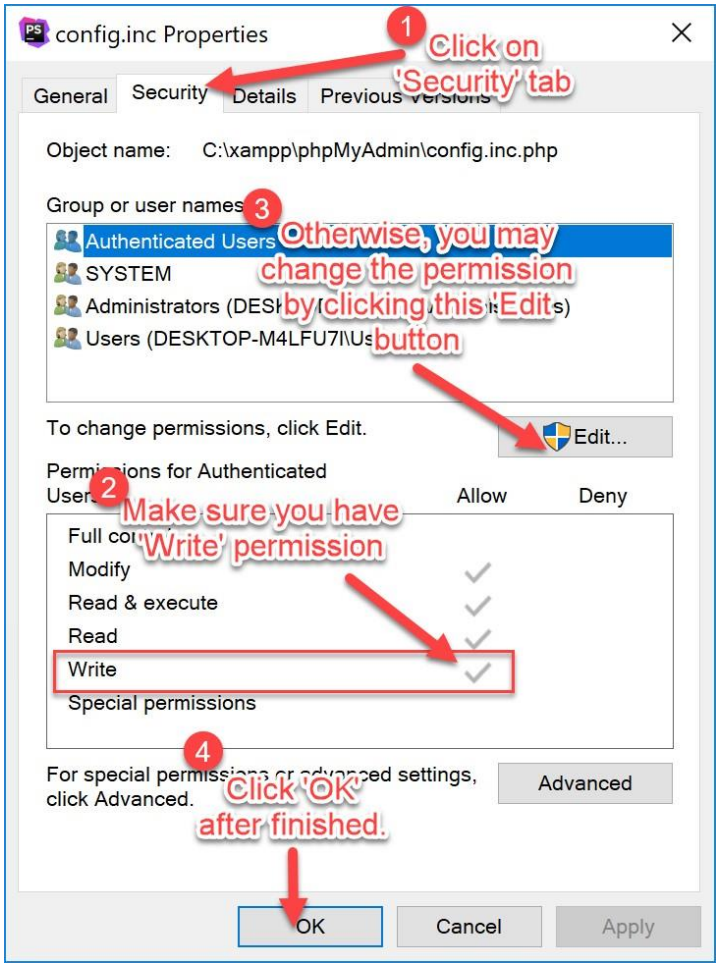
3 Save the file


XAMPP Control Panel v3.2.2 [Compiled: Nov 12th 2015]
XAMPP Control Panel v3.2.2
Re-start the MySQL server
Modules
Service Module PID(s) Port(s) Actions
Apache 15016 19540 80, 443 Stop Admin Config Logs
MySQL Start Admin Config Logs
FileZilla Start Admin Config Logs
Mercury Start Admin Config Logs
Tomcat Start Admin Config Logs
Config
Netstat
Shell
Explorer
Services
Help
Quit
12:45:11 AM [Apache] Attempting to stop Apache (PID: 12608)
12:45:11 AM [Apache] Status change detected: stopped
12:45:26 AM [Apache] Attempting to start Apache app...
12:45:26 AM [Apache] Status change detected: running
12:45:32 AM [mysql] Attempting to start MySQL app...
12:45:32 AM [mysql] Status change detected: running
12:45:34 AM [mysql] Attempting to stop MySQL app...
12:45:35 AM [mysql] Status change detected: stopped
XAMPP Control Panel v3.2.2 [Compiled: Nov 12th 2015]
XAMPP Control Panel v3.2.2
Click 'Admin' to verify the changes
Modules
Service Module PID(s) Port(s) Actions
Apache 15016 19540 80, 443 Stop Admin Config Logs
MySQL 12188 3306 Stop Admin Config Logs
FileZilla Start Admin Config Logs
Mercury Start Admin Config Logs
Tomcat Start Admin Config Logs
Config
Netstat
Shell
Explorer
Services
Help
Quit
12:45:26 AM [Apache] Attempting to start Apache app...
12:45:26 AM [Apache] Status change detected: running
12:45:32 AM [mysql] Attempting to start MySQL app...
12:45:32 AM [mysql] Status change detected: running
12:45:34 AM [mysql] Attempting to stop MySQL app...
12:45:35 AM [mysql] Status change detected: stopped
12:46:52 AM [mysql] Attempting to start MySQL app...
12:46:53 AM [mysql] Status change detected: running
MySQL Server successfully started



If you can't edit or save the *config.inc*, follow the steps below:



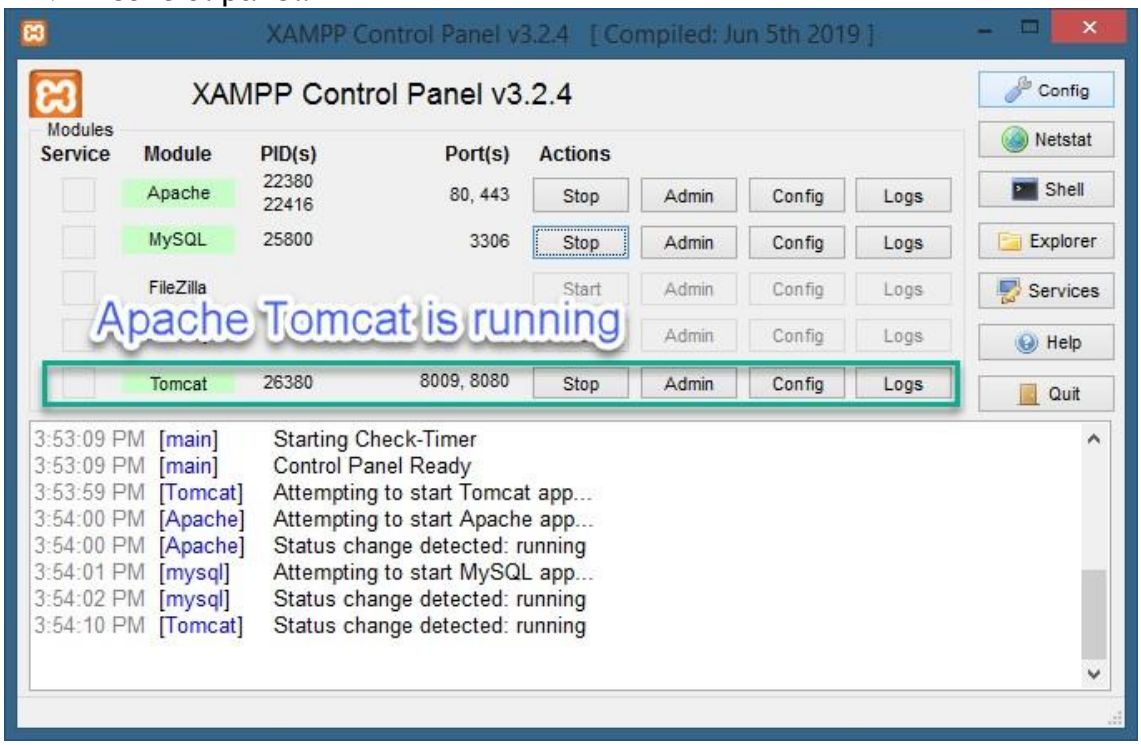

config.inc Properties
1
Click on 'Security' tab
General
Security
Details
Previous versions
Object name: C:\xampp\phpMyAdmin\config.inc.php
Group or user names:
3
Otherwise, you may change the permission by clicking this 'Edit' button
Authenticated Users
SYSTEM
Administrators (DESK
Users (DESKTOP-M4LFU7I\Us
To change permissions, click Edit.
Edit...
Permissions for Authenticated Users
2
Make sure you have 'Write' permission
Allow
Deny
Full control
Modify
Read & execute
Read
Write
Special permissions
For special permissions or advanced settings, click Advanced.
Advanced
4
Click 'OK' after finished.
OK
Cancel
Apply

Task 3: Managing Apache Tomcat

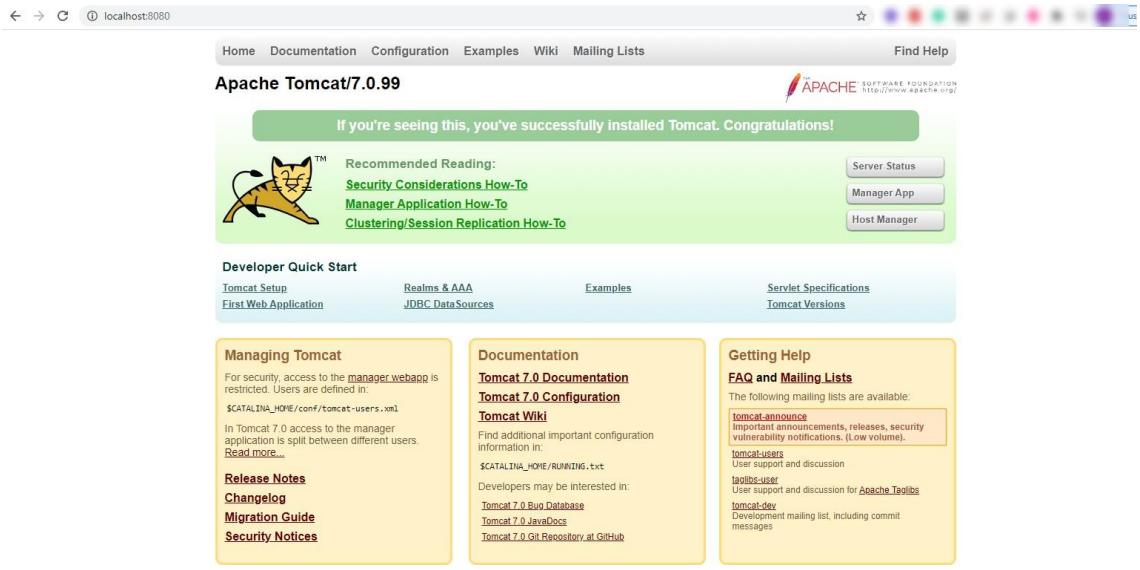
Objective:	Testing the access and add a new user to Apache Tomcat in XAMPP
Problem Description:	To configure user access for Apache Tomcat in XAMPP and test the access at localhost.
Estimated time:	30 minutes

In the previous task, we have successfully installed Apache Tomcat which is part of XAMPP software package.

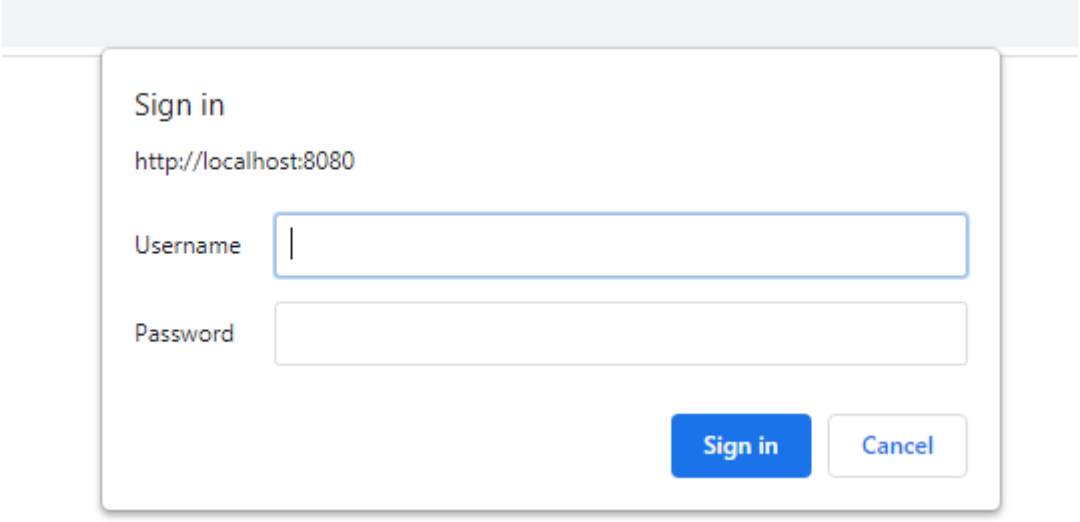
1. Make sure Apache Tomcat is running. You can see that status from XAMPP control panel.



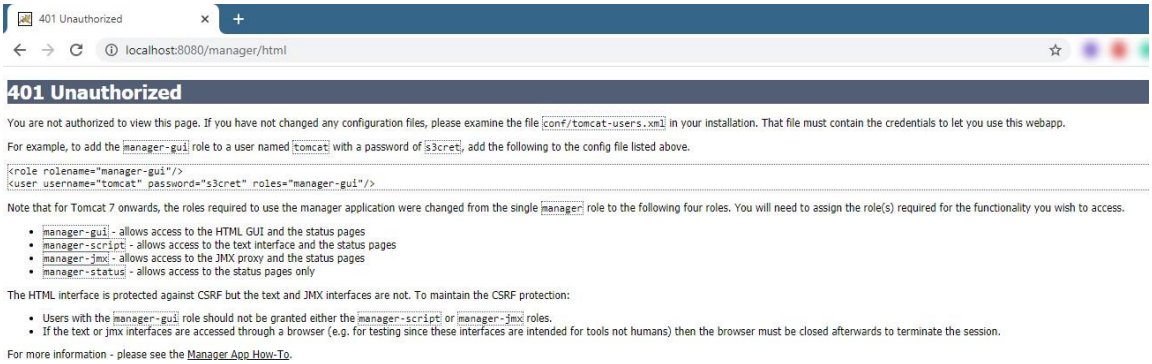
2. Open the browser, and go to http://localhost:8080/. If Apache Tomcat is running correctly, you will see a homepage as follows:



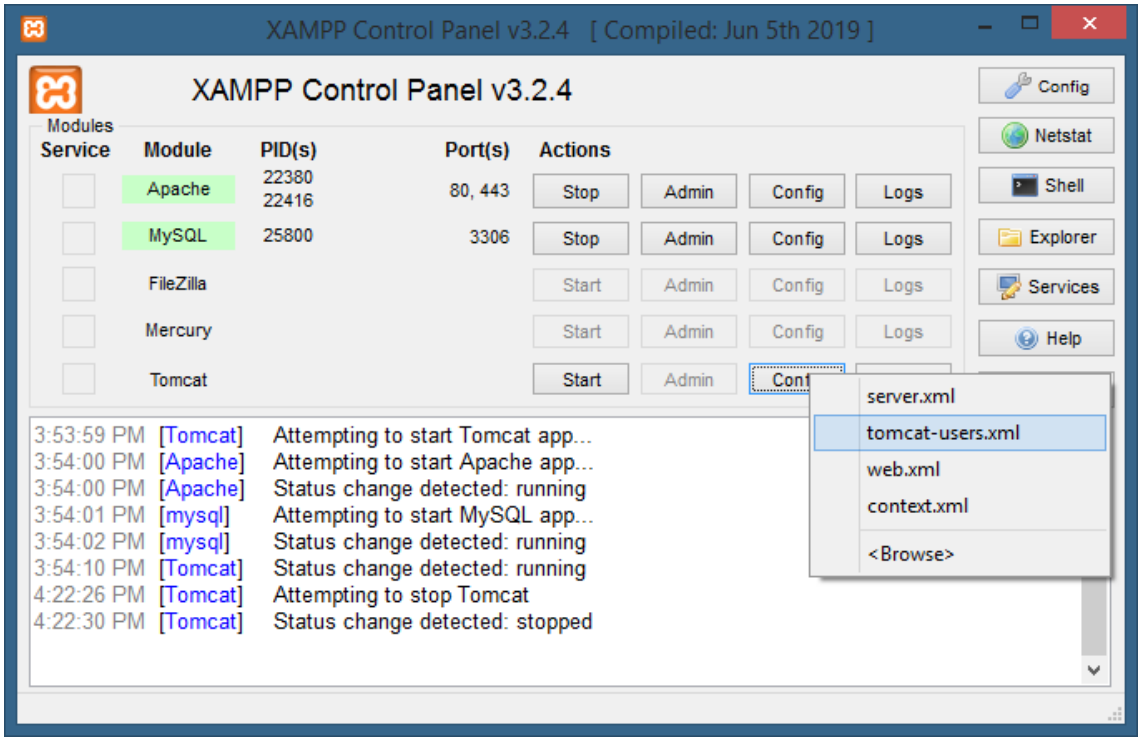
3. Click on the “Manager App” of Apache Tomcat/7.0.99 homepage and there will appear a page ask for username and password. Do not panic, just click “Cancel”.



4. As our Apache Tomcat currently does not has any defined user, a page below will be shown on the browser.



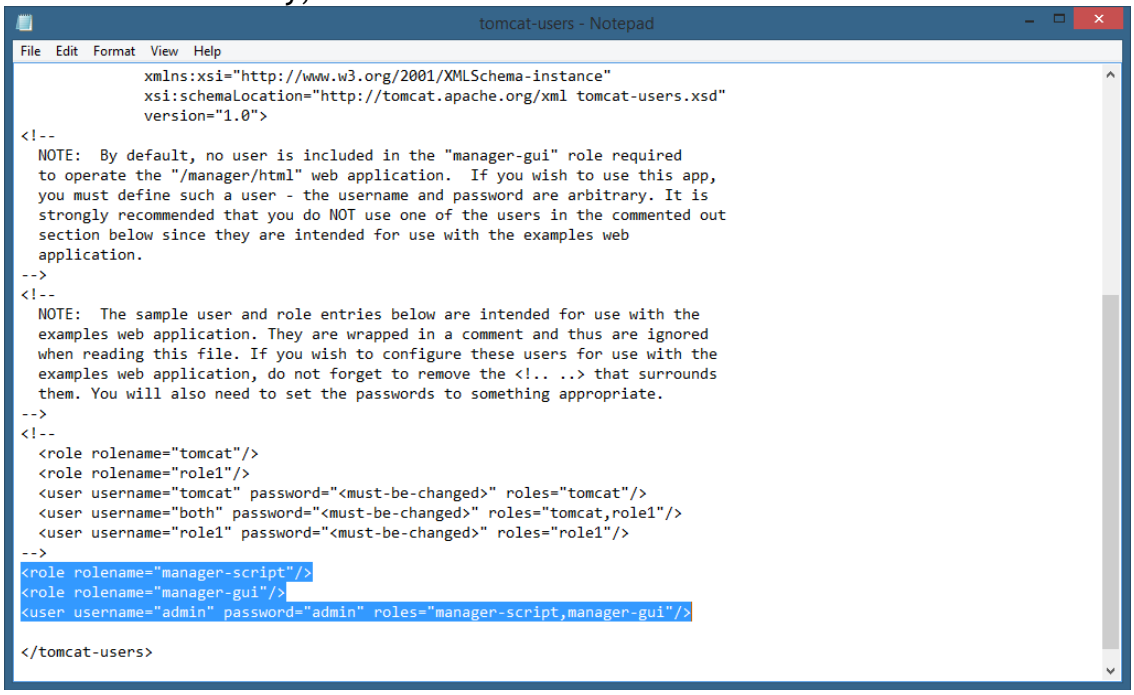
5. To define a new user, click **Stop** Apache Tomcat server at the XAMPP Control Panel and click “Config” > tomcat-users.xml



6. The tomcat-users.xml will be opened using the default editor in your system. Scroll down the file until you see the </tomcat-users> tag. Add the tags below at the top of </tomcat-users> tag.

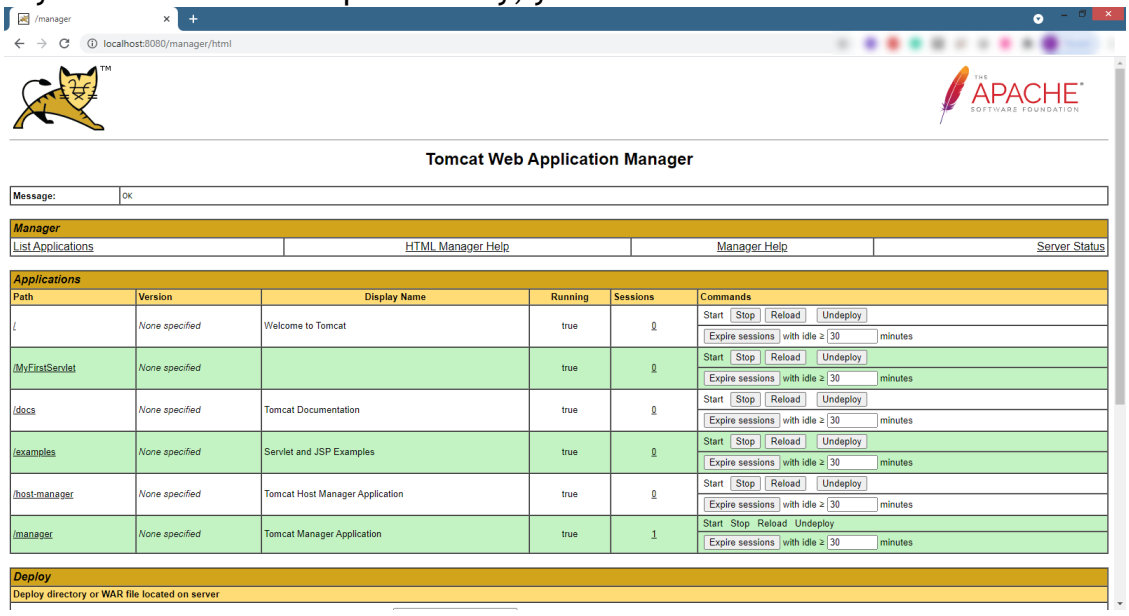
```
<role rolename="manager-script"/>
<role rolename="manager-gui"/>
<user username="admin" password="admin"
roles="manager-script,manager-gui"/>
```

7. Save the file. Finally, tomcat-users.xml will look like this



8. We have added a new user “admin” with a password “admin”. Be sure to change the password when you deploy your application on a production server.

9. Now, **start** the Apache Tomcat and repeat Step 2 and Step 3. This time, put the username as “admin” and password as “admin”.
10. If you did the above steps correctly, you will see the screen as below:

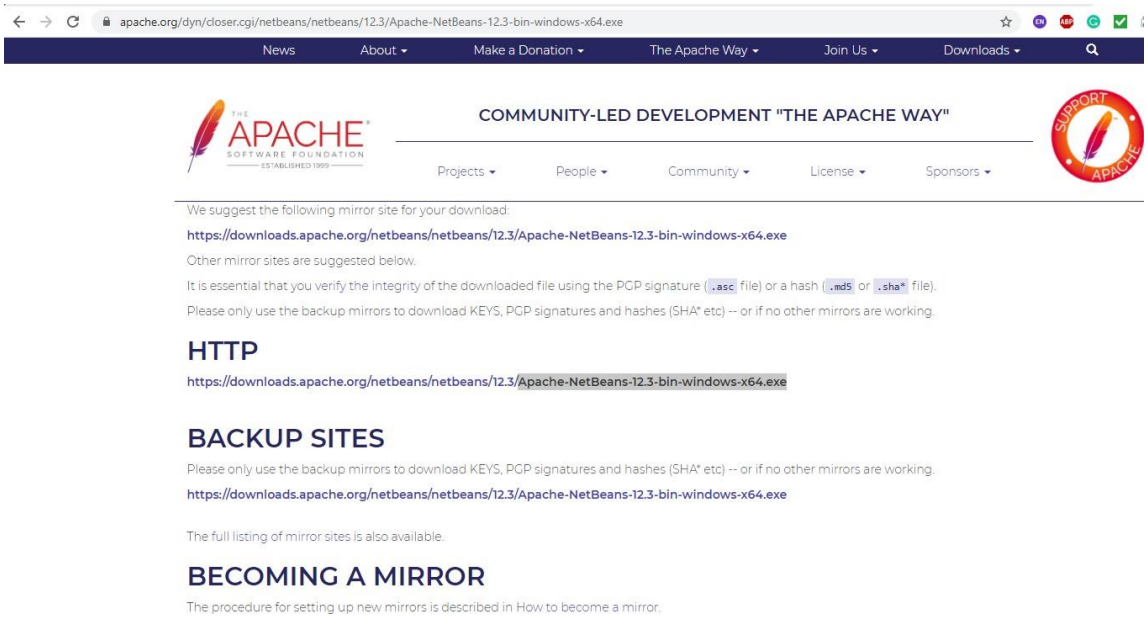


Task 4: Netbeans 12.3 IDE Installation

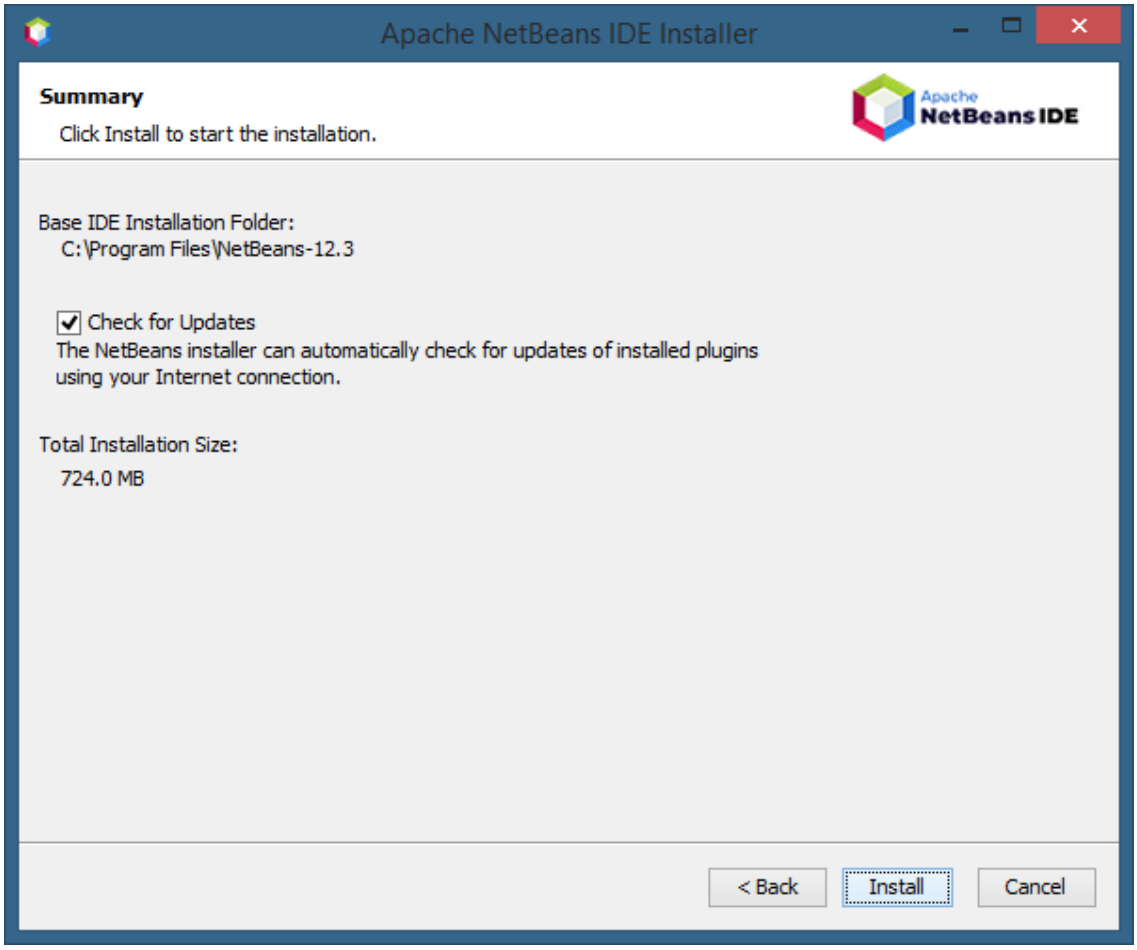
Objective:	Netbeans 12.3 IDE Installation
Problem Description:	To setup a proper environment for Java Web Application development.
Estimated time:	30 minutes

To start the Netbeans 12.3 installation process, follow the steps below:

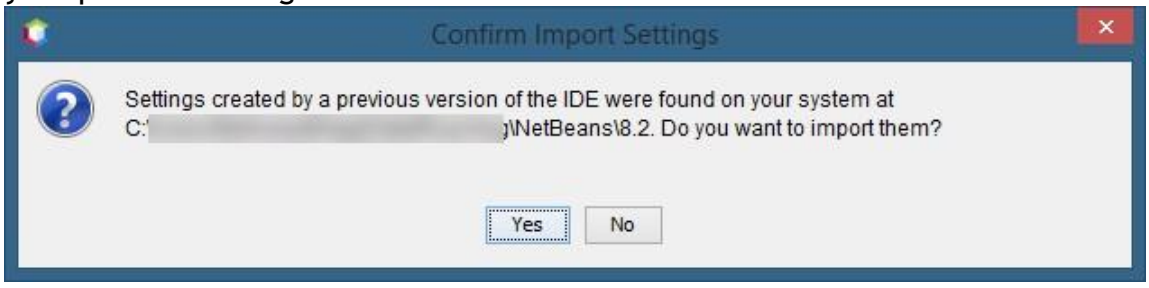
11. Go to https://netbeans.apache.org/download/index.html
12. Click "Download". Choose the installer that appropriate with your operating system and computer architecture (x32 or x64). A correct installation program could be something like this *Apache-NetBeans-12.3-bin-windows-x64.exe*



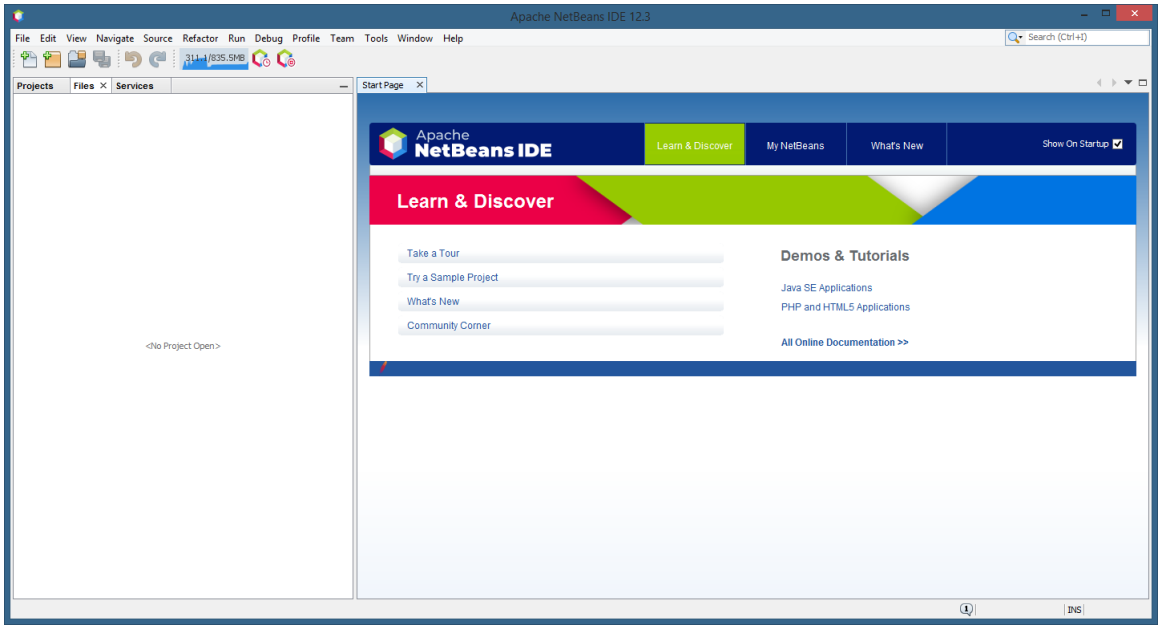
13. After the download finish, double-click on the installer and click "Next" until you reach the following screen:



14. Make sure you tick the “Check for Updates”, then click Install. Wait until the installation finish.
15. Run the Netbeans. If you have previous installation, the pop-up windows as below may appear and simply click “Yes”. This will import your previous configuration to a new one.



16. Finally, you will get the following screen.



17. Now, you are ready to start developing your first Java web application. Well done!

Task 5: Linking Netbeans to Apache Tomcat and Writing a Simple Java Servlet

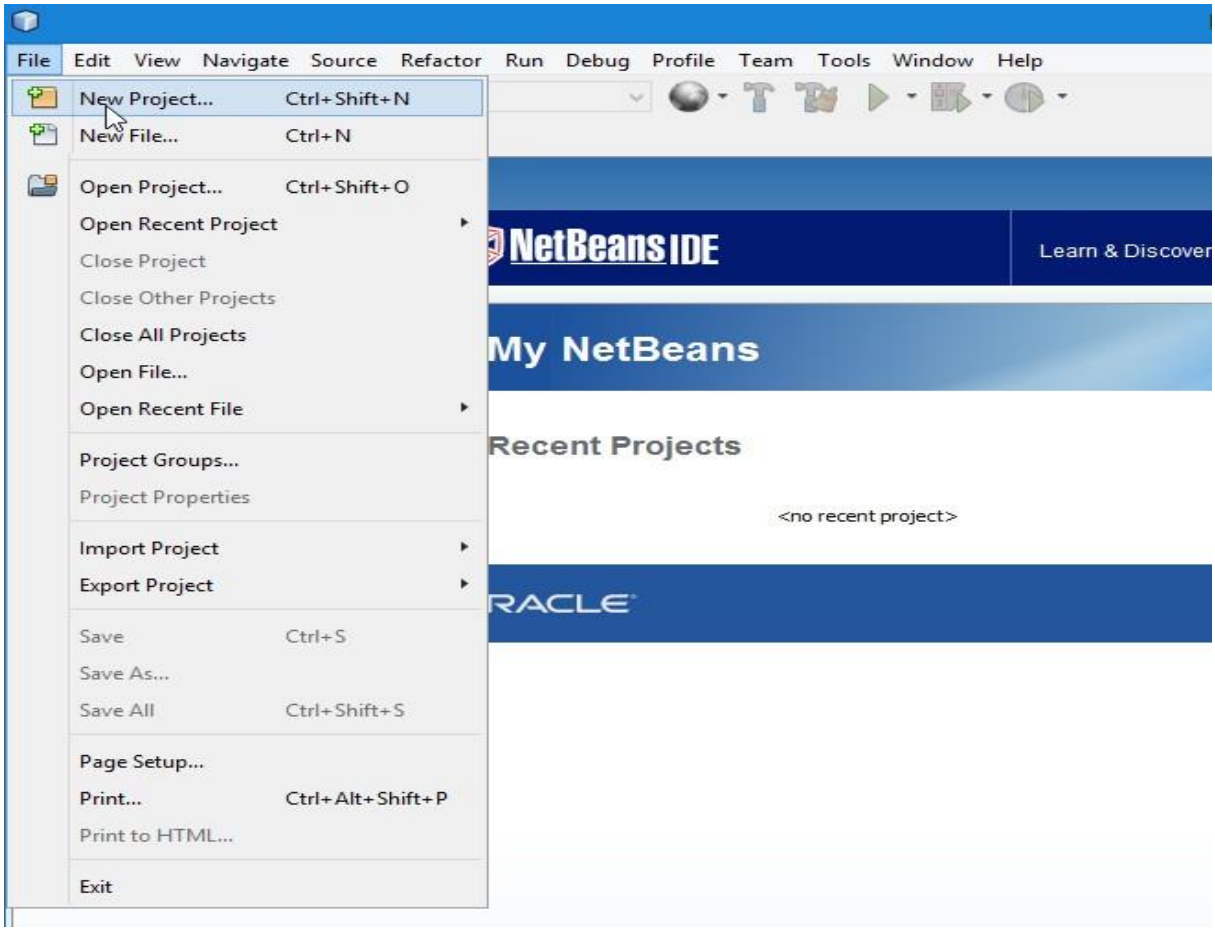
Objective:	Link Netbeans to Apache Tomcat and Writing a Simple Java Servlet
Problem Description:	To write a simple Java Servlet
Estimated time:	15 minutes

1. Create a directory *C:\[MATRICNUMBER]*.

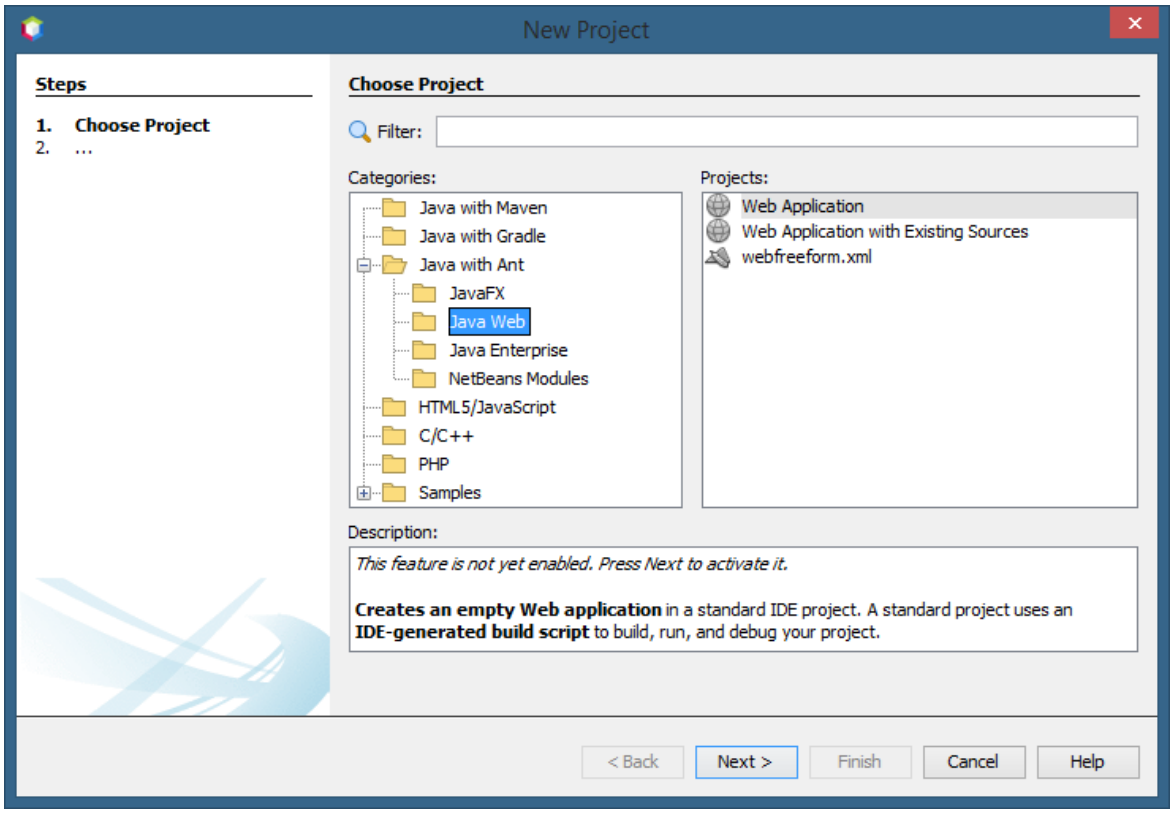
2. Go to *C:\[MATRICNUMBER]* Lab's directory and create sub-directory as *Lab 1 - Servlet*.

3. Open your NetBeans.

4. Go to File -> New Project

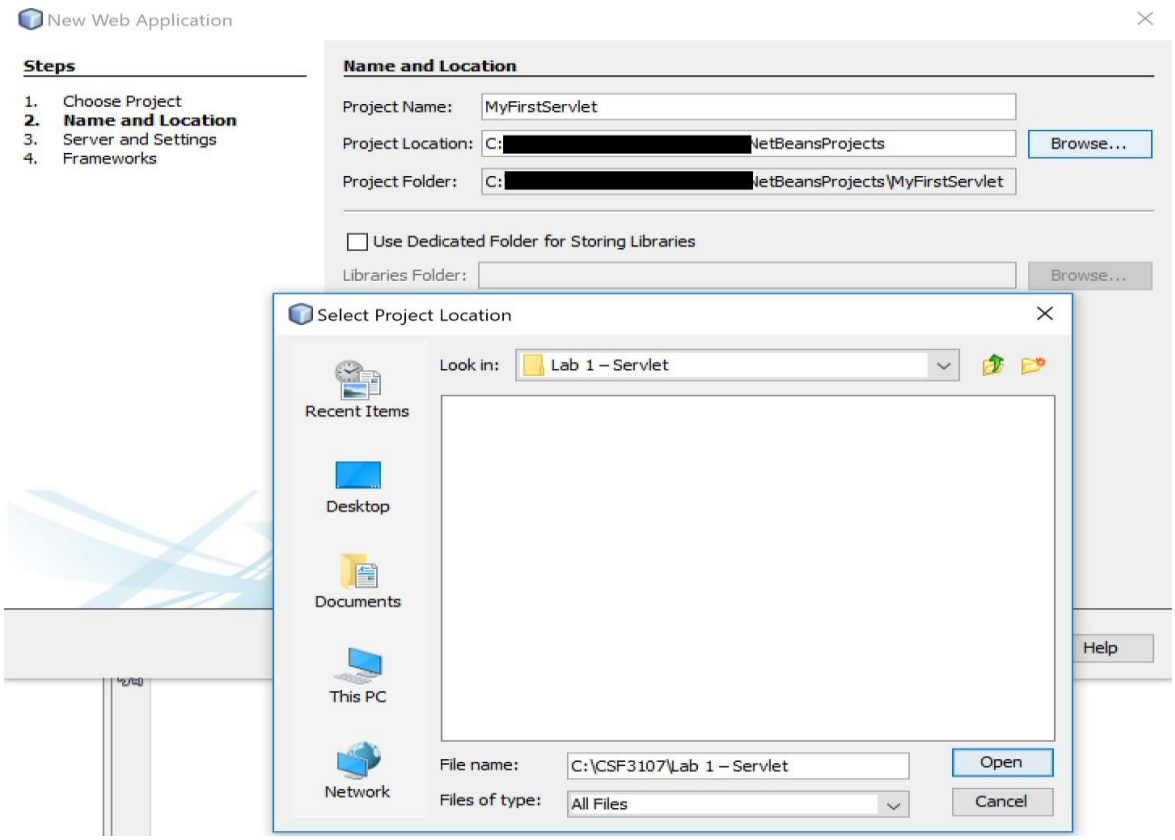


5. Select Java with Ant -> Java Web -> Web Application and click Next.

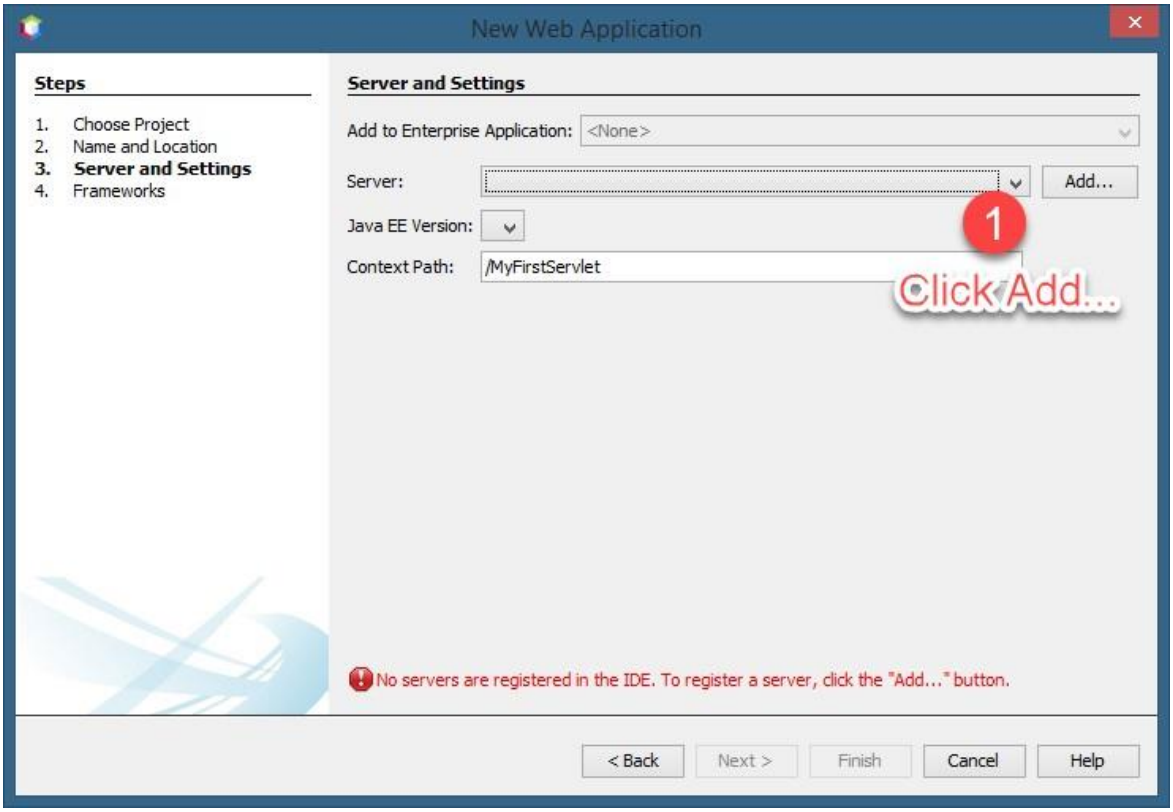


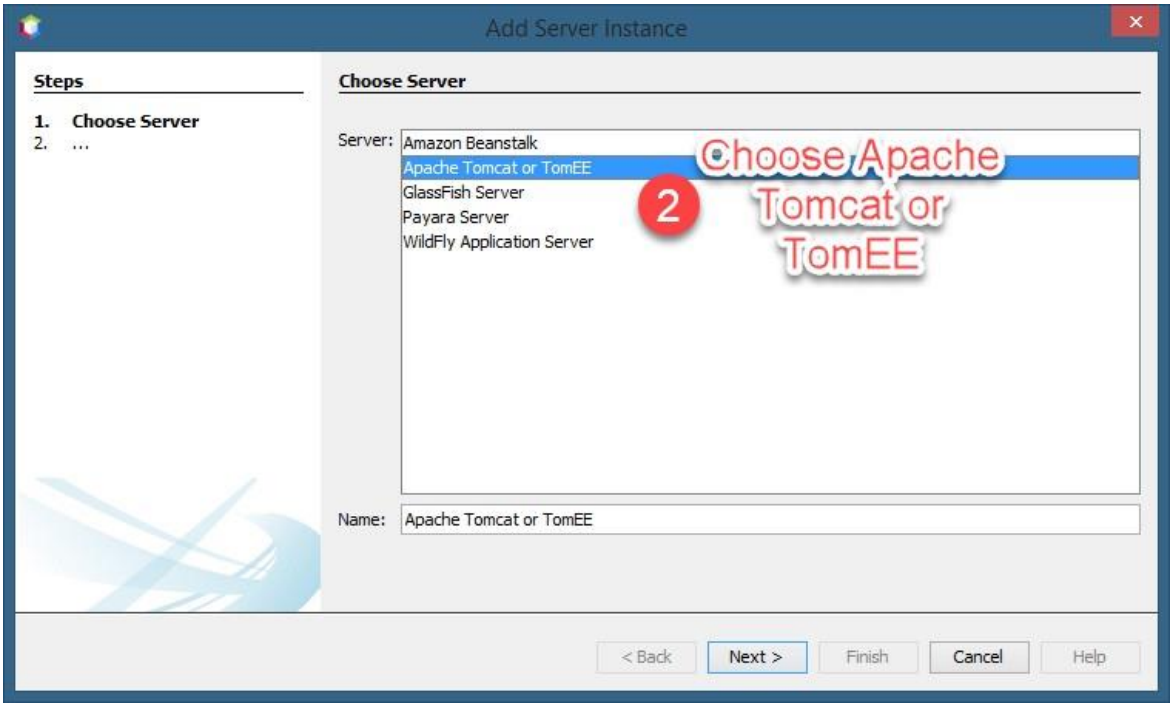
1. Type Project Name: *MyFirstServlet*.

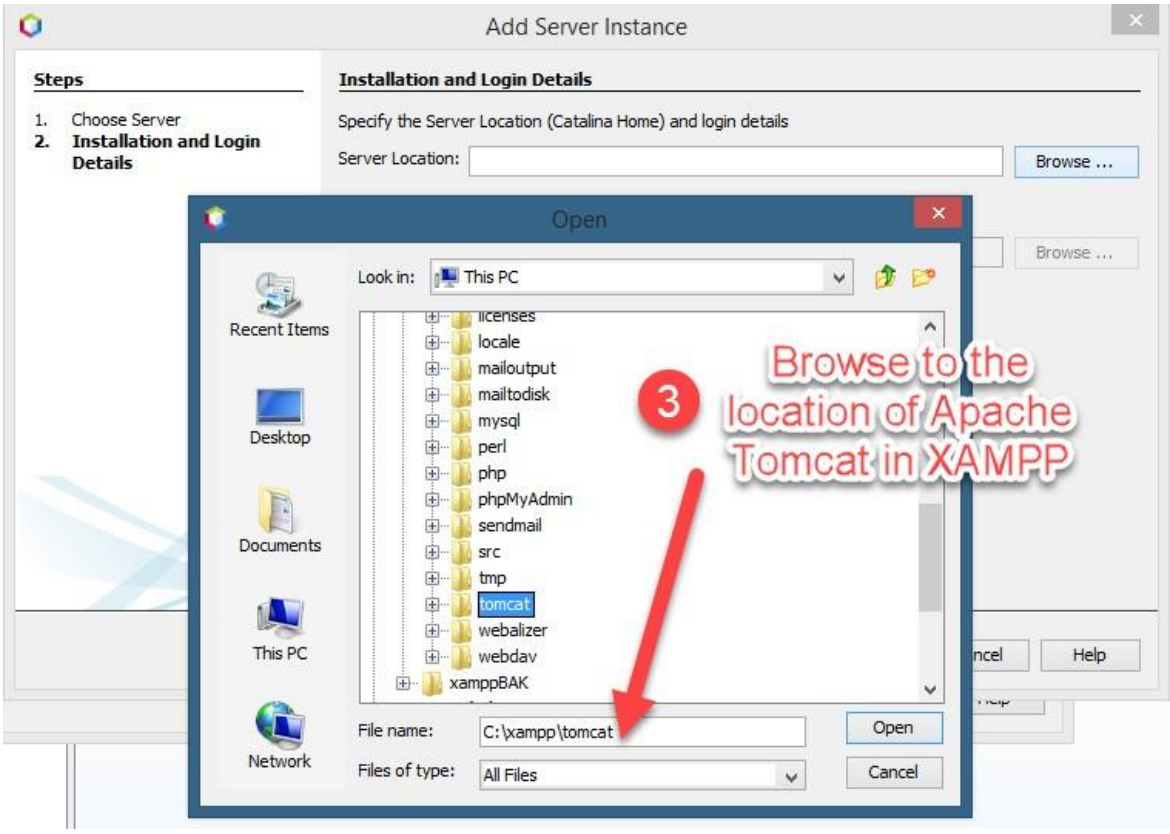
2. Click Browse and choose Project Location: *C:\[MATRICNUMBER]\Lab 1 - Servlet*. Then click the Next button.

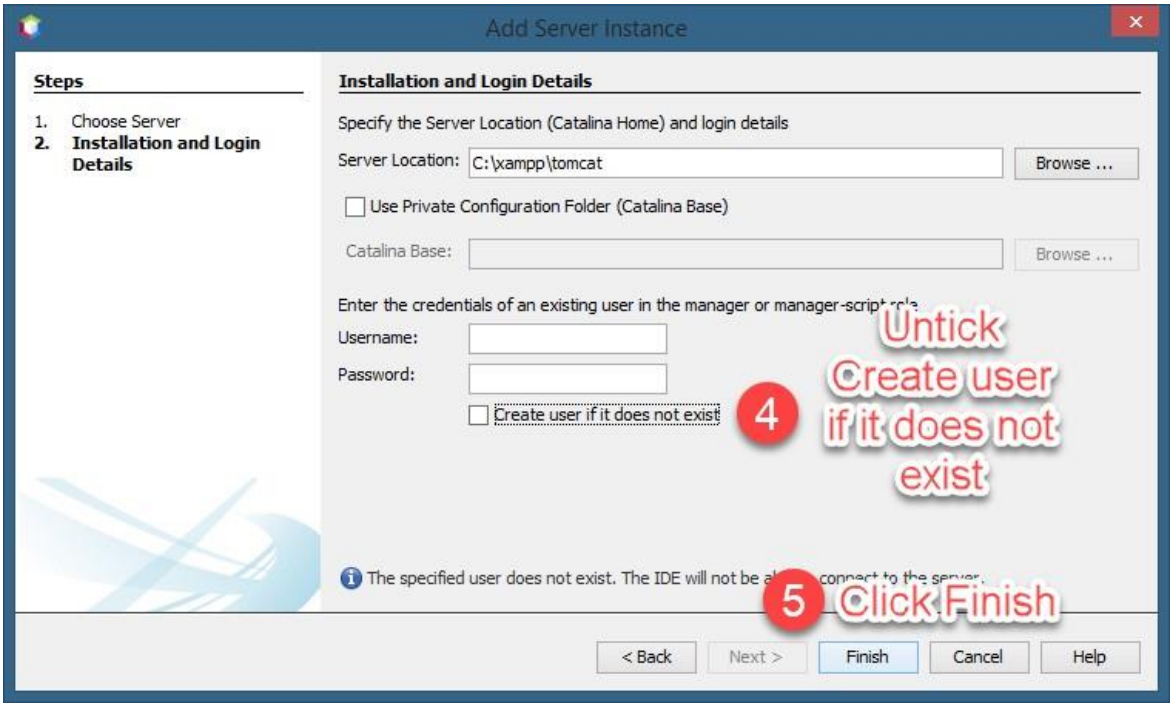

New Web Application
Steps
1. Choose Project
2. Name and Location
3. Server and Settings
4. Frameworks
Name and Location
Project Name: MyFirstServlet
Project Location: C:[redacted]NetBeansProjects
Browse...
Project Folder: C:[redacted]NetBeansProjects\MyFirstServlet
Use Dedicated Folder for Storing Libraries
Libraries Folder:
Browse...
Help
Select Project Location
Look in: Lab 1 – Servlet
Recent Items
Desktop
Documents
This PC
Network
File name: C:\CSF3107\Lab 1 – Servlet
Files of type: All Files
Open
Cancel

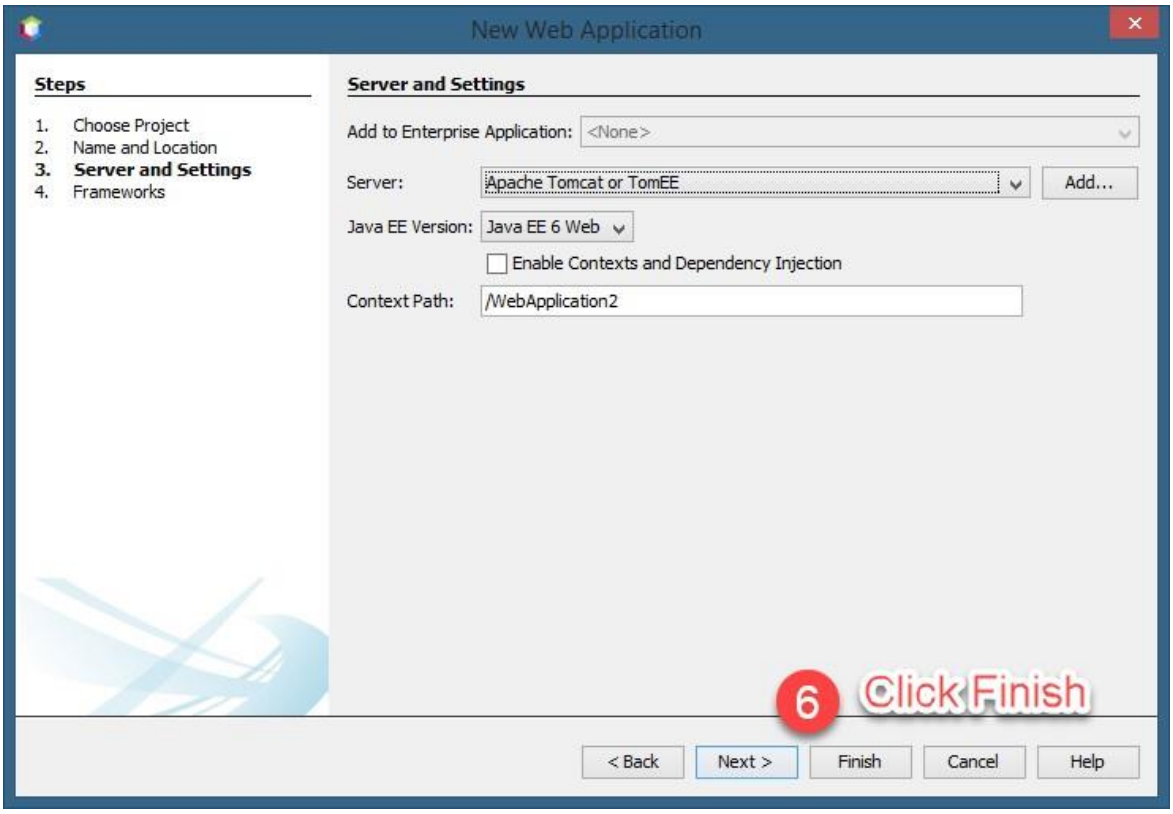
3. Before we can run our Java Web Application, we need to link it to Apache Tomcat Server. Follow the diagrams below to link Netbeans with Apache Tomcat in XAMPP.





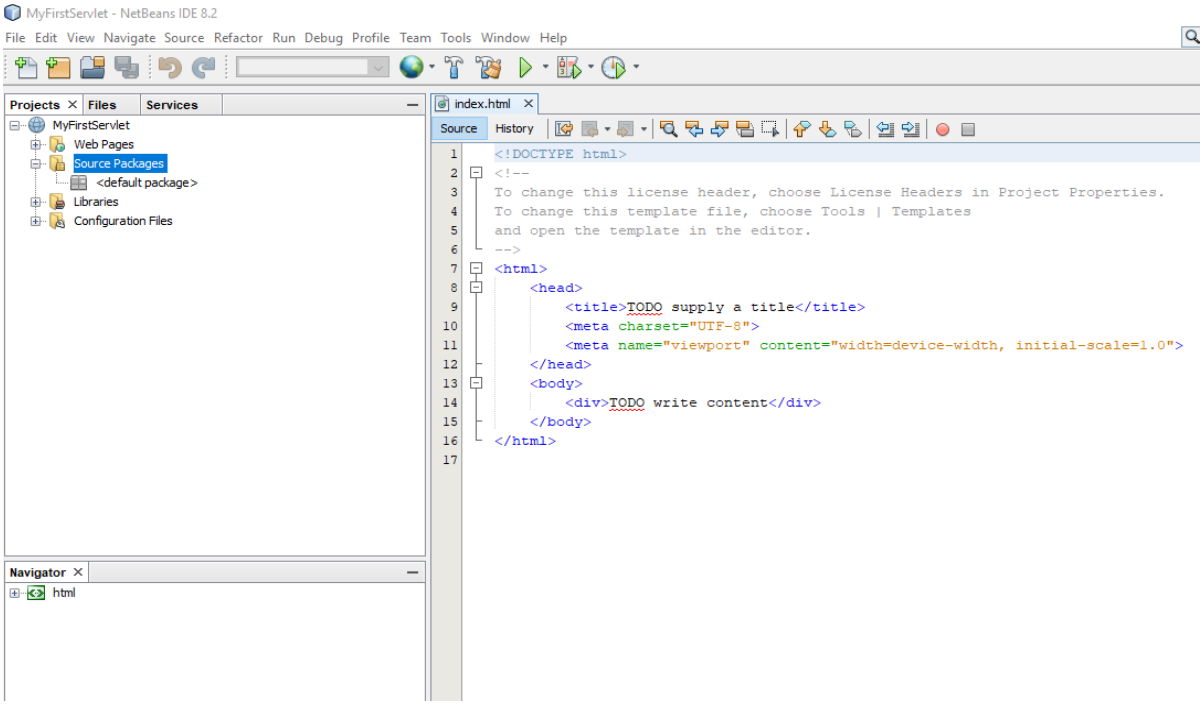

Add Server Instance
Steps
1. Choose Server
2. Installation and Login Details
Installation and Login Details
Specify the Server Location (Catalina Home) and login details
Server Location:
Browse ...
Open
Look in: This PC
Recent Items
Desktop
Documents
This PC
Network
licenses
locale
mailoutput
mailtodisk
mysql
perl
php
phpMyAdmin
sendmail
src
tmp
tomcat
webalizer
webdav
xamppBAK
3
Browse to the location of Apache Tomcat in XAMPP
File name: C:\xampp\tomcat
Files of type: All Files
Open
Cancel
Help


Add Server Instance
Steps
1. Choose Server
2. Installation and Login Details
Installation and Login Details
Specify the Server Location (Catalina Home) and login details
Server Location: C:\xampp\tomcat
Browse ...
Use Private Configuration Folder (Catalina Base)
Catalina Base:
Browse ...
Enter the credentials of an existing user in the manager or manager-script role
Username:
Password:
Create user if it does not exist
4
Untick Create user if it does not exist
The specified user does not exist. The IDE will not be able to connect to the server.
5
Click Finish
< Back
Next >
Finish
Cancel
Help

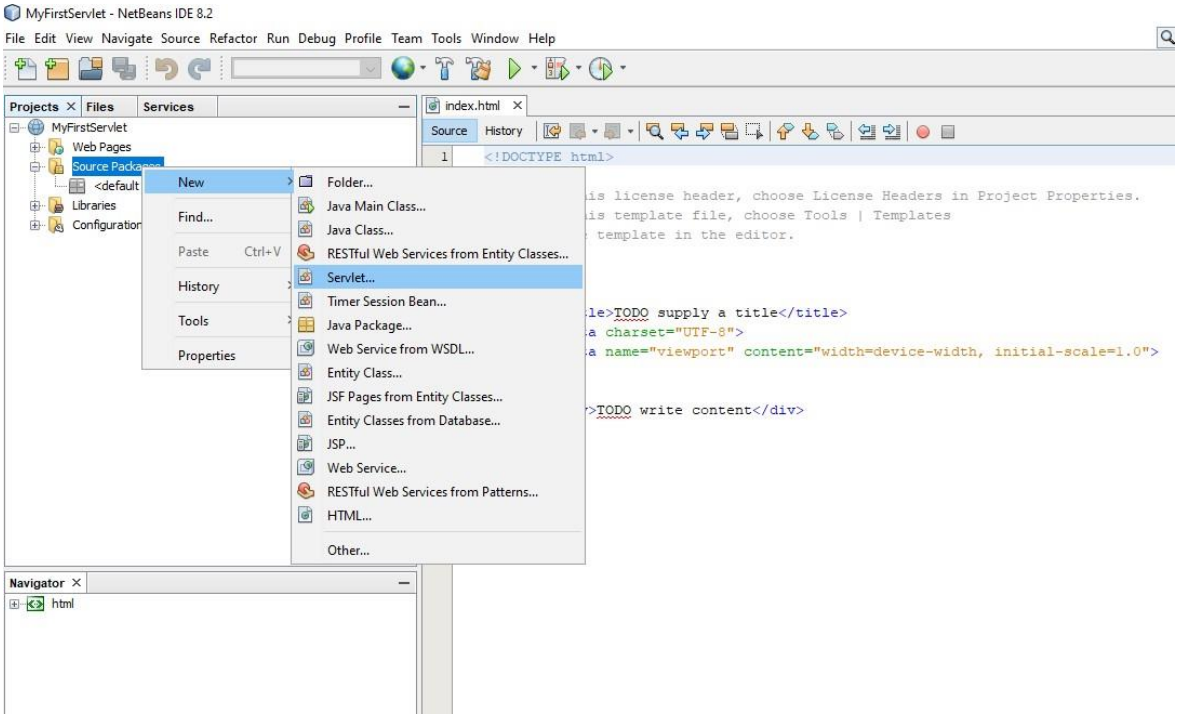


Note: the above steps only need to be ran for the first time you link the Netbeans to Apache Tomcat in XAMPP. After this, you just need to select it from the screen.

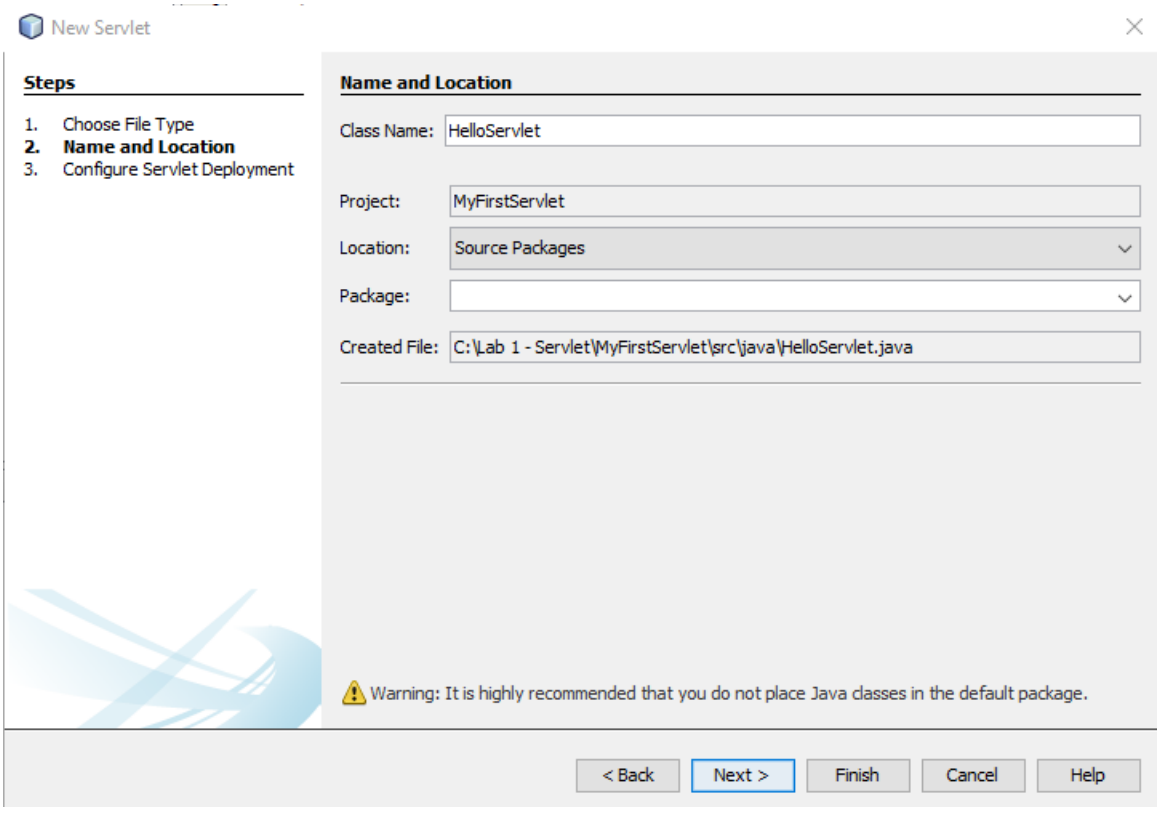
4. If everything okay, you will see the following screen:



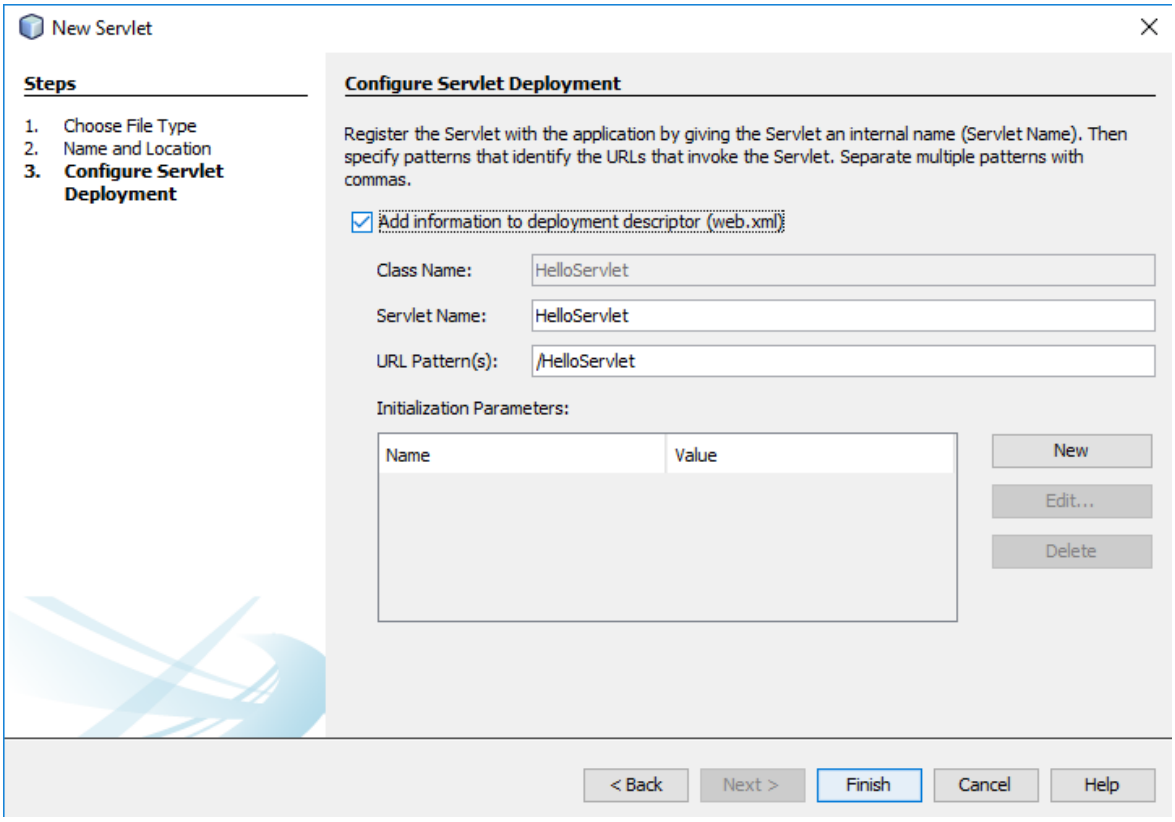
5. Create a new Servlet file.



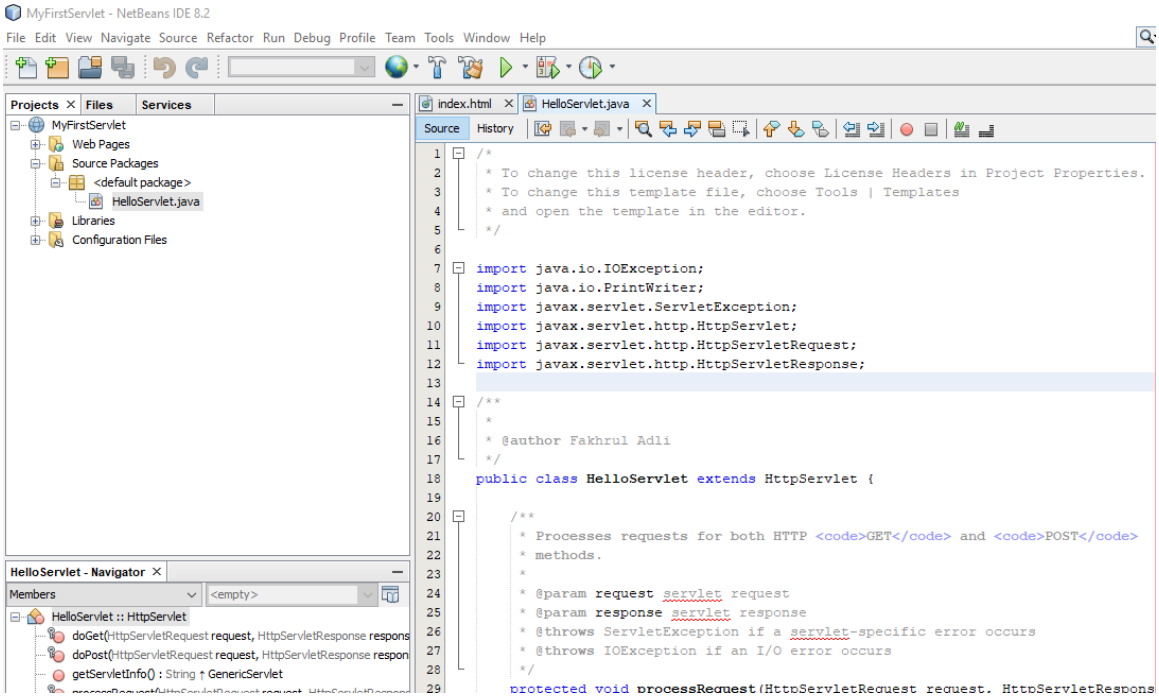
6. Name your servlet as *HelloServlet*.



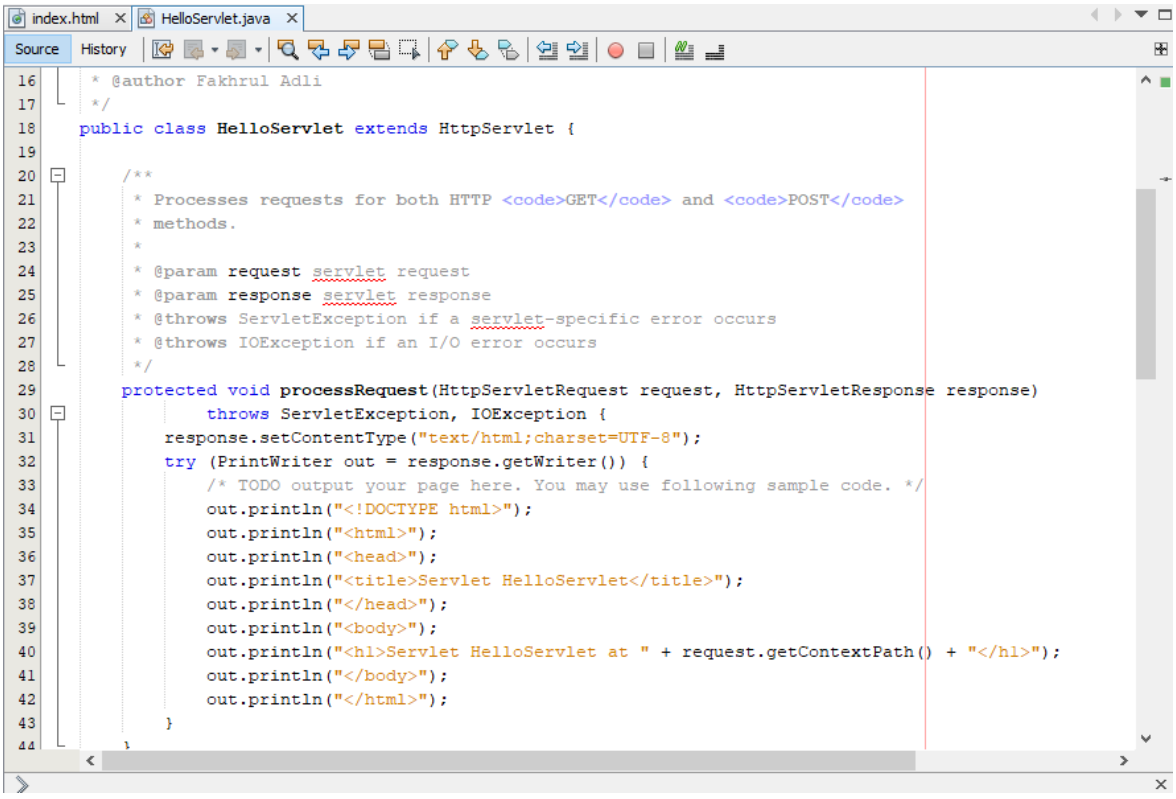
7. Tick the “Add information to deployment descriptor (web.xml)”



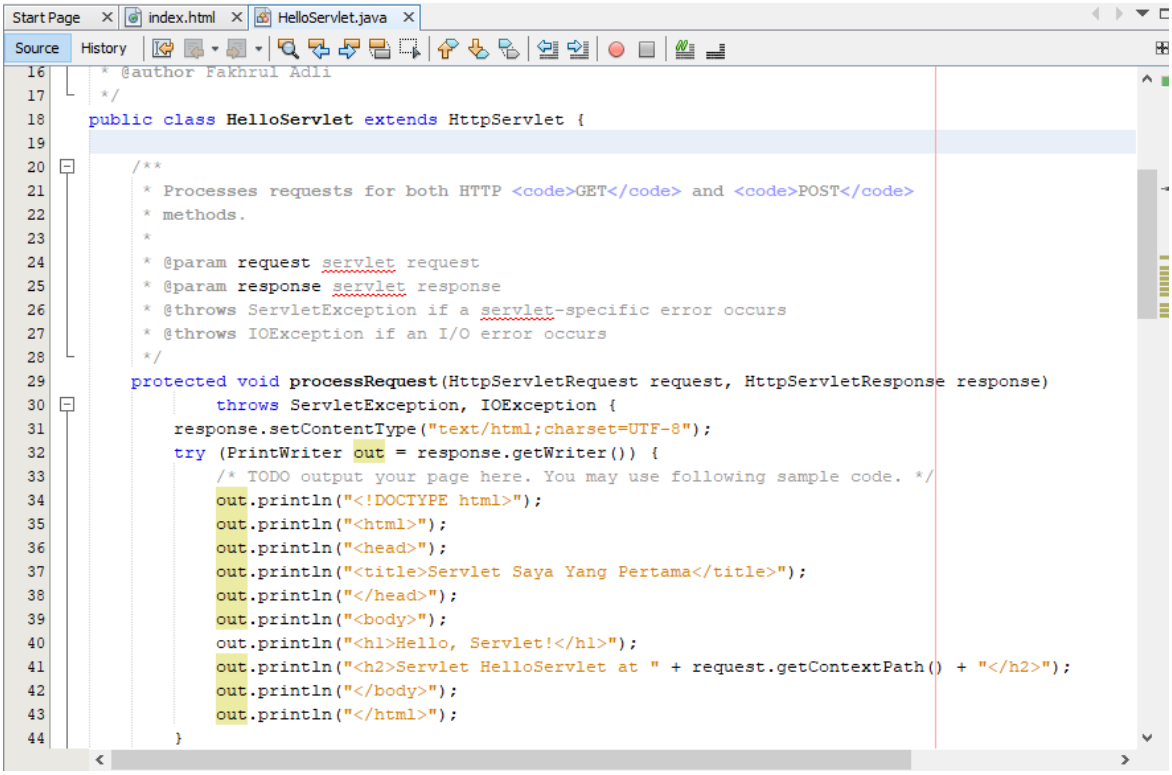
8. NetBeans has produced a new file named *HelloServlet.java*. You may see the location of it on the left side of NetBeans editor. On the right, are the servlet codes generated by NetBeans.



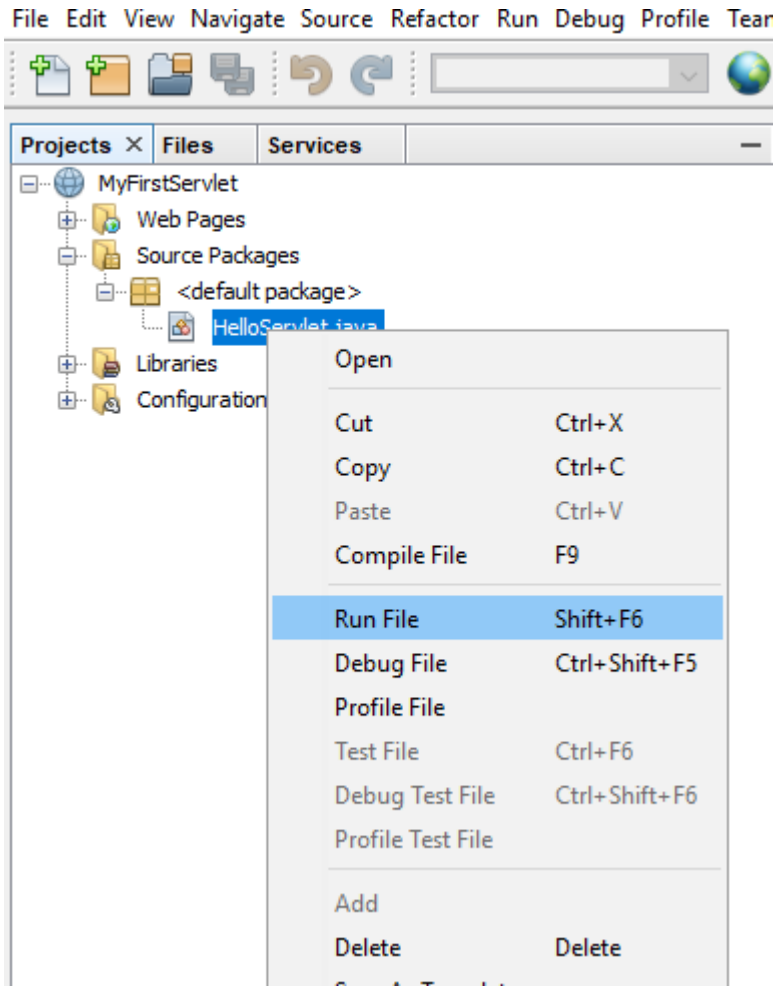
9. Browse through the servlet file until you find the processRequest() method.



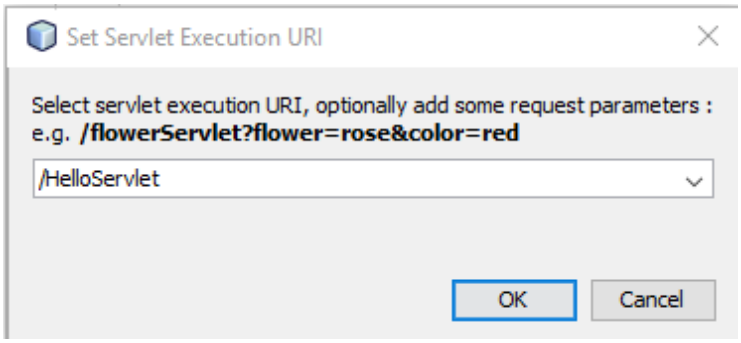
10. We are going to make a simple modification to the existing codes. Modify the them as follows. Refer to line 37, 40 and 41.



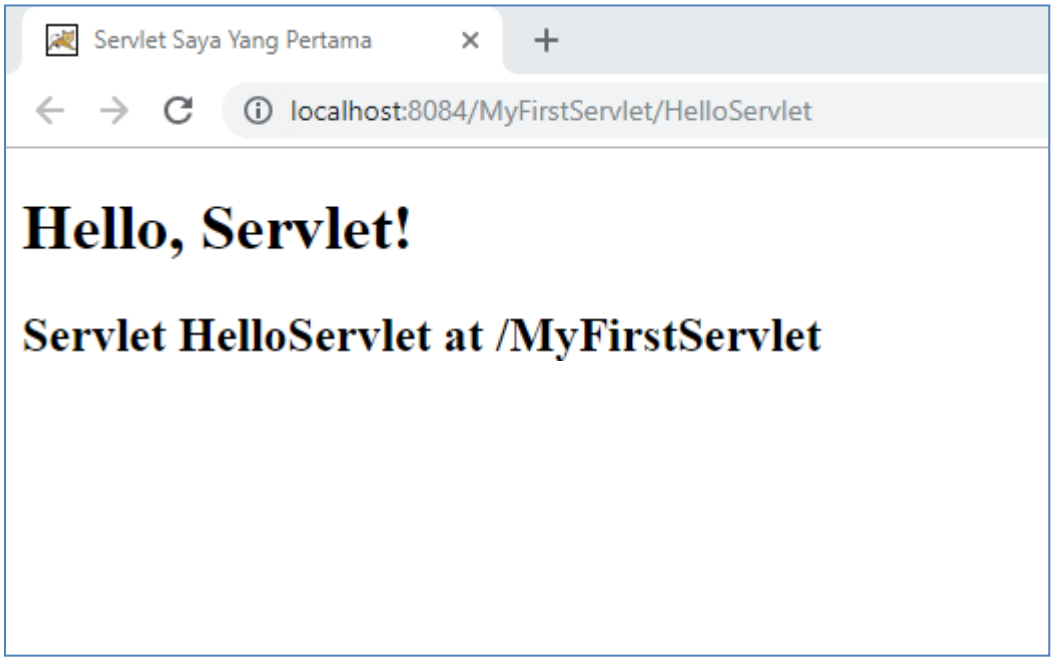
11. After finished your modification on the previous step, run *HelloServlet.java* by right-clicking on it and select *Run File*.



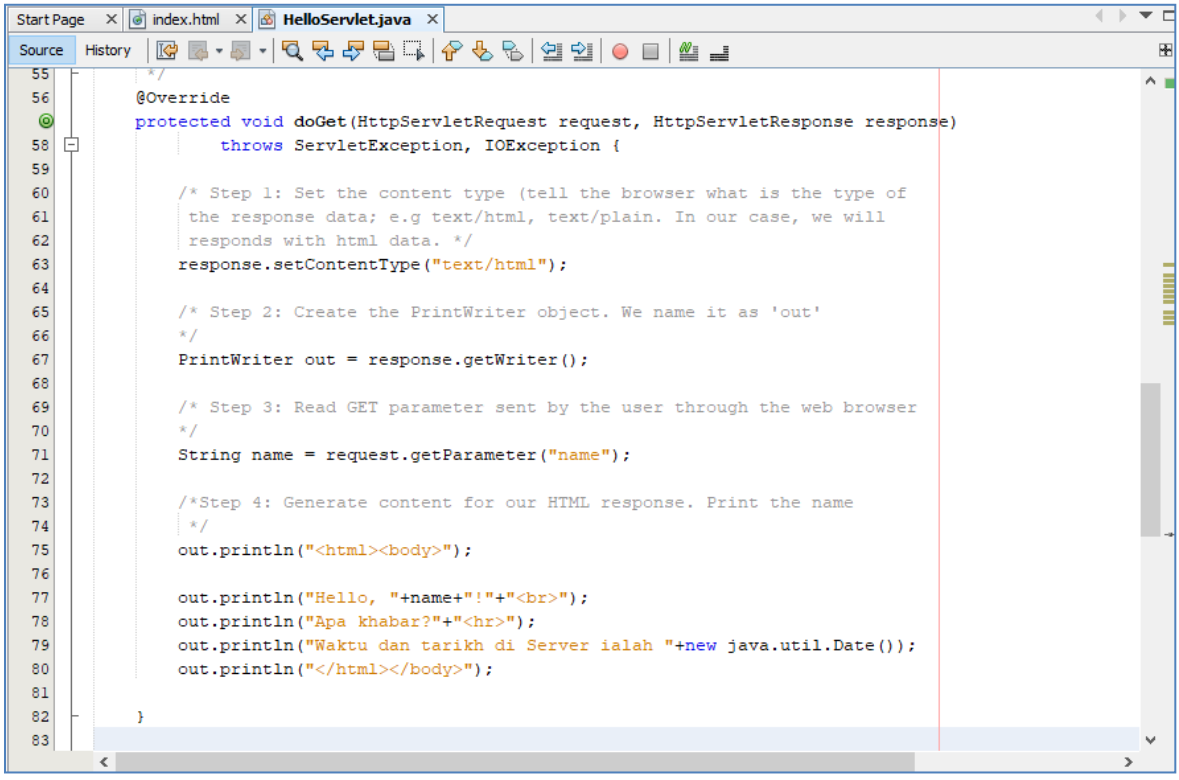
12. Before you can see the output, a dialogue box will appear. This dialogue box allows us to add some request parameter. At this moment, we will not supply any parameter. Just click OK button.



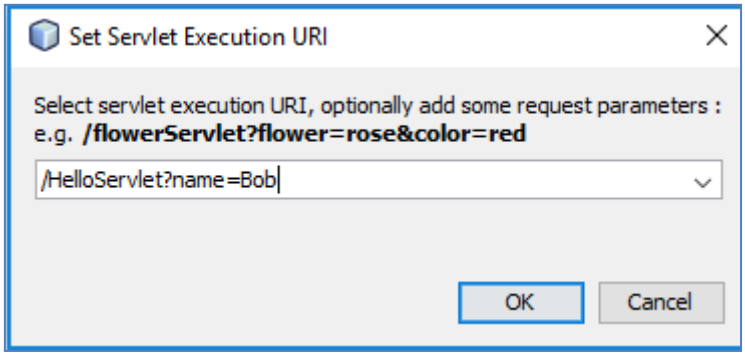
13. You will see the following output on your browser.



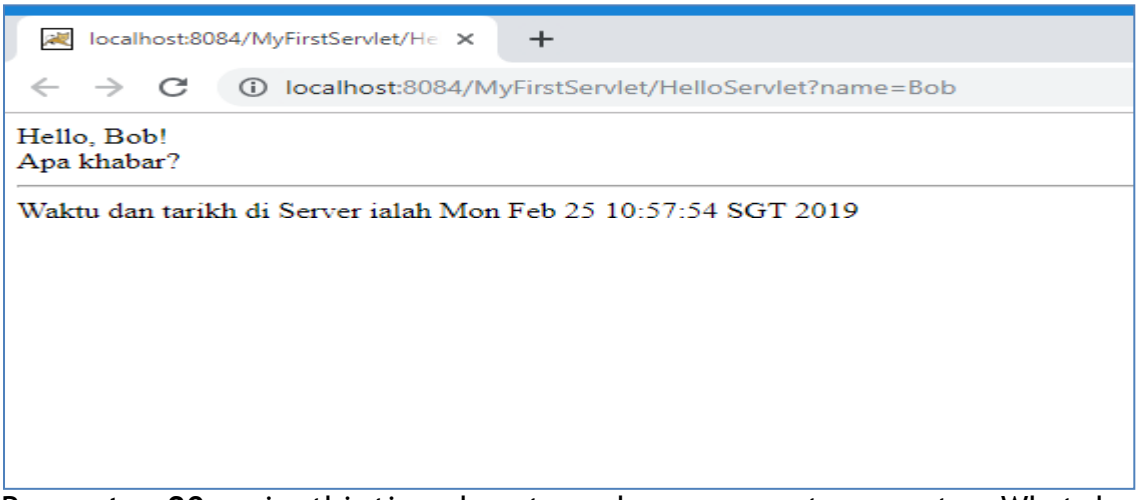
14. Now, we will make further modification to our servlet file. This time, we aregoing to modify the *doGet()* method. Browse through the codes in *HelloServlet.java* until you find the desired method. Type the codes as provided in the screenshot below. Read the comments carefully and try to understand what each of the codes does. **Remark: Please remove processRequest() method from the body of *doGet()* method.**



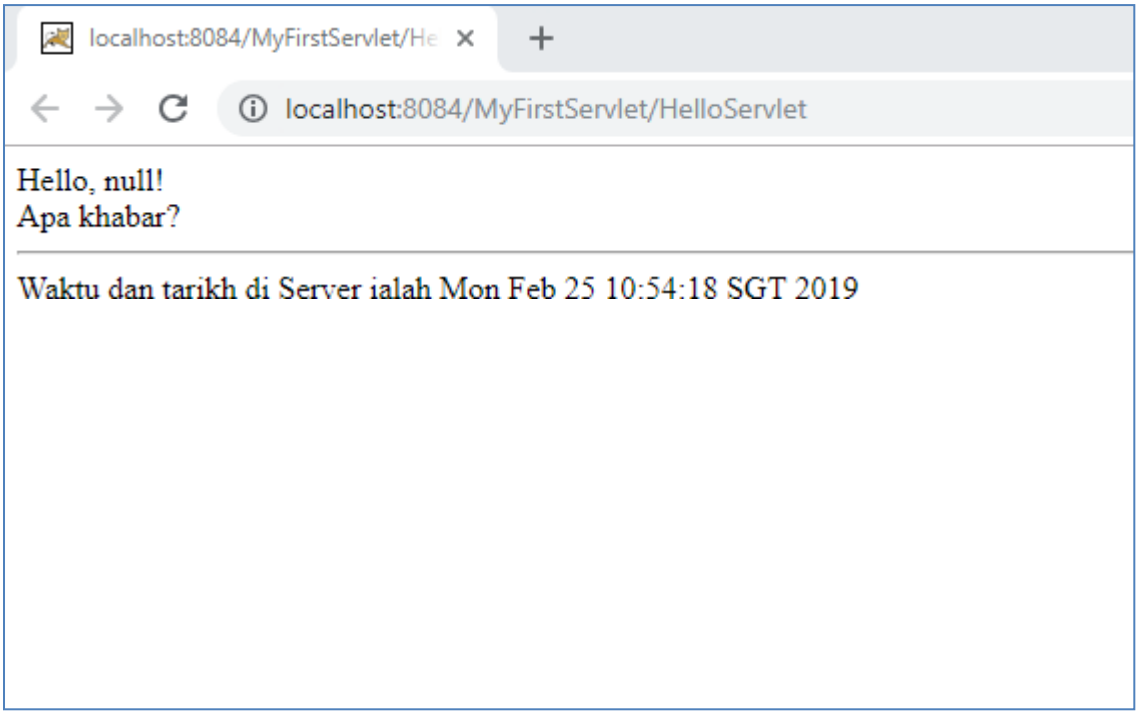
15. After finish, right click on *HelloServlet.java* and click Run. As you have seen previously, a dialogue box shows. This time, we will supply a value *Bob* to the parameter *name*. Then, click the OK button.



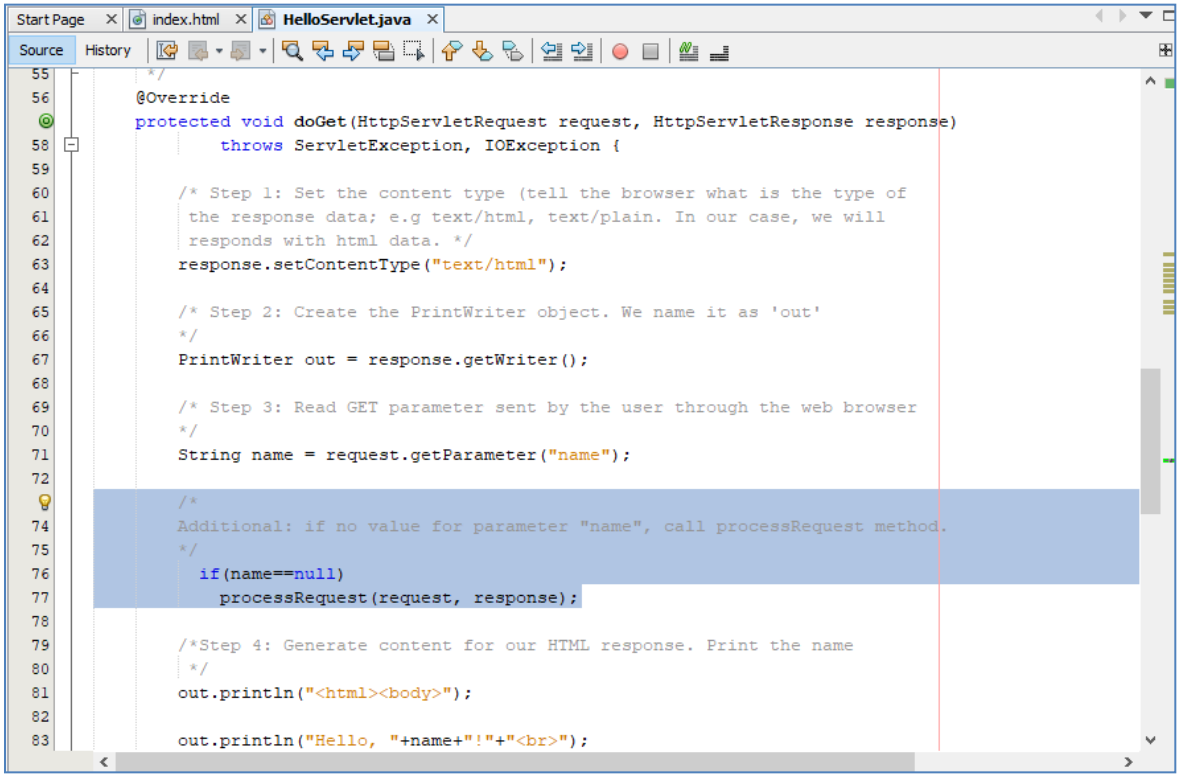
16. If everything goes well, you will see the output as below:



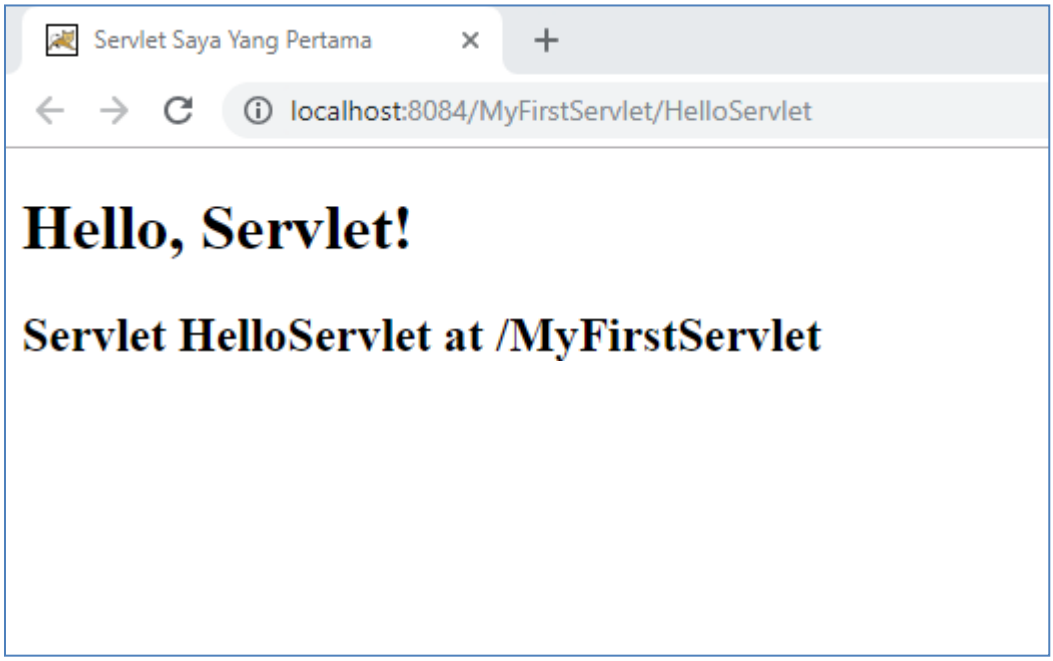
17. Rerun **step 20** again, this time do not supply any request parameter. What do you see from the output? Do you see something like the following screenshot? How to avoid the *null* from being displayed?



18. You can upgrade your code in HelloServlet.java by putting the following codes into it. By doing this, if no value supplied to the parameter *name*, the request will be passed to *processRequest()* method, and this will avoid from the *null* value appears on the browser.



19. So, if you rerun the file and without supplying any parameter, you will see the output as follows:

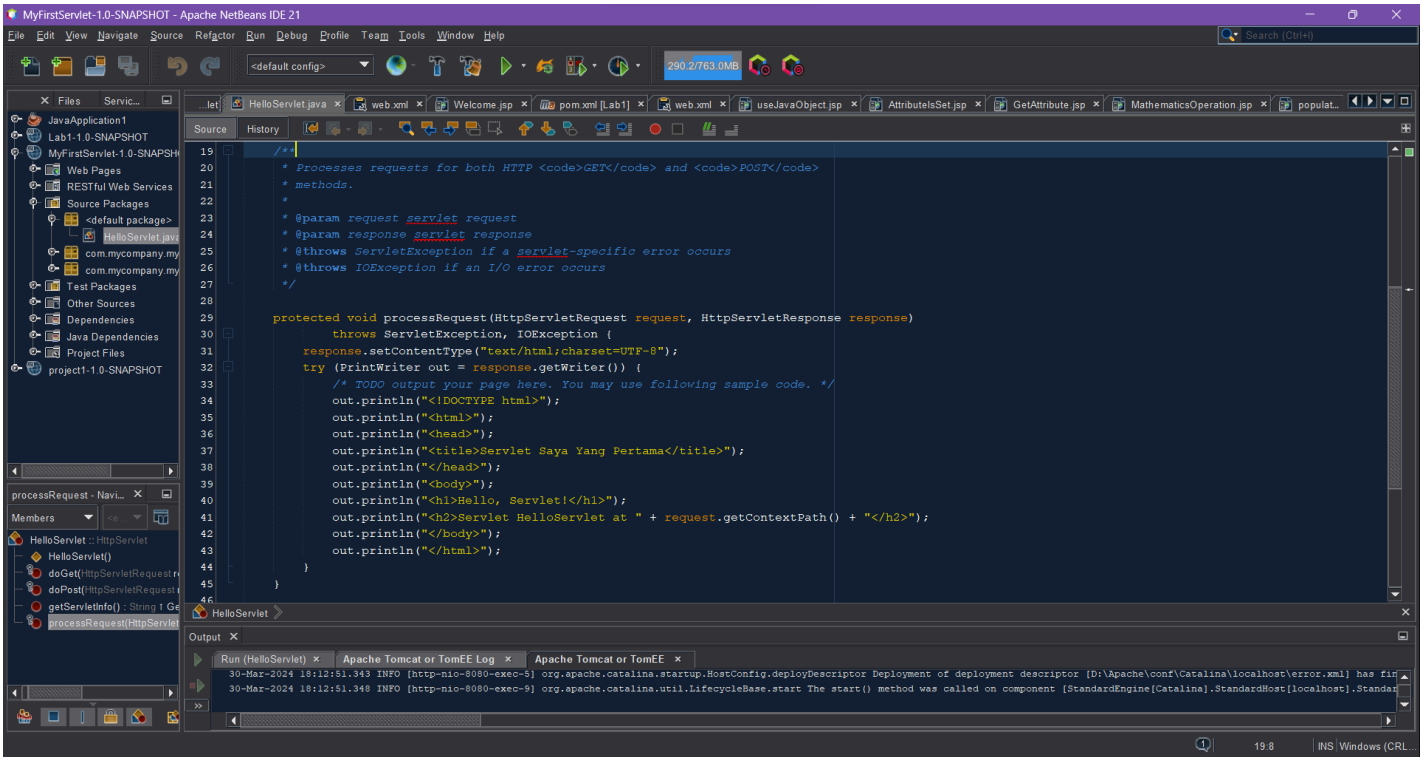


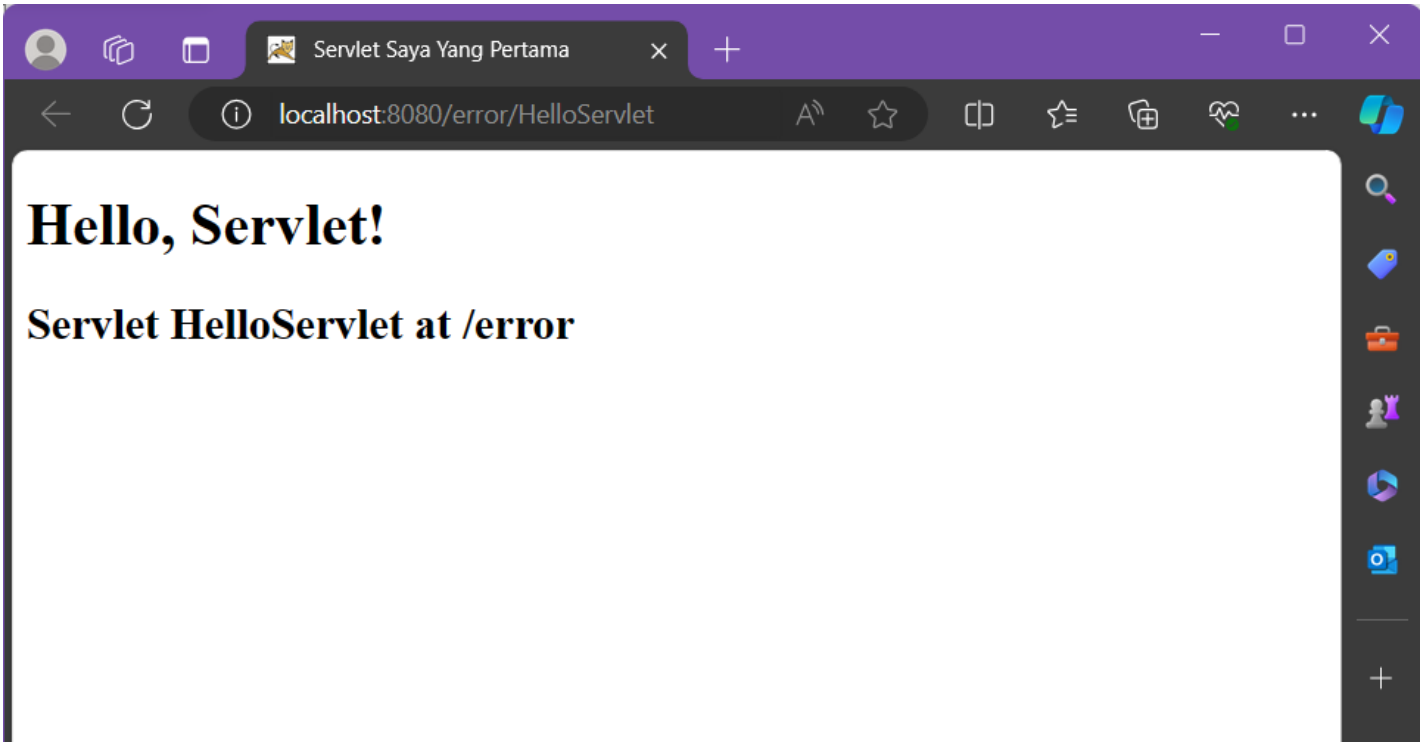
It is the same output as can be seen in **Step 18**: why?

Yes , cause the *processRequest()* method can avoid from the *null* value appears on the browser

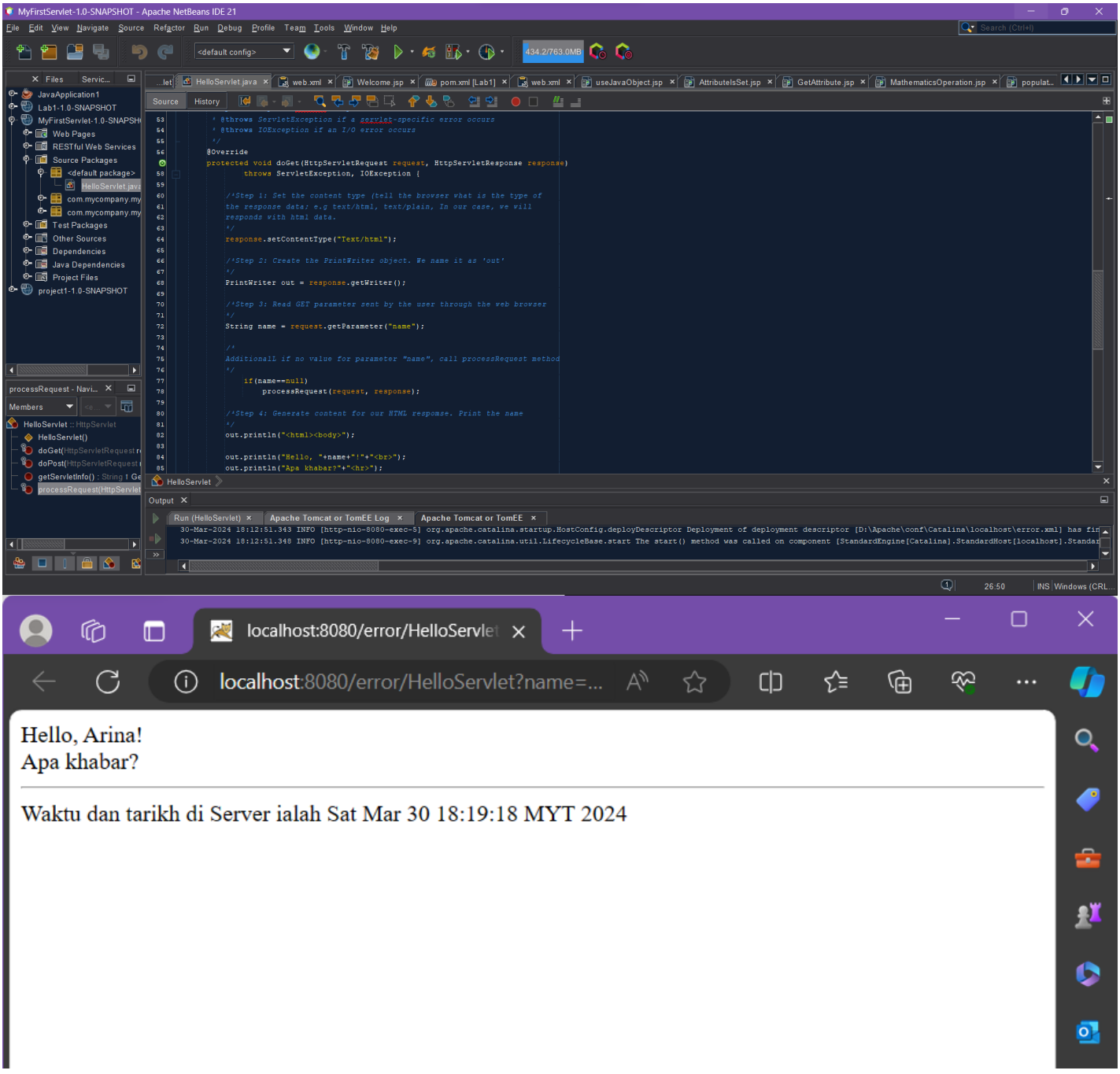
Source code

13.





17


MyFirstServlet-1.0-SNAPSHOT - Apache NetBeans IDE 21
HelloServlet.java
web.xml
Welcome.jsp
pom.xml [Lab1]
useJavaObject.jsp
AttributeIsSet.jsp
GetAttribute.jsp
MathematicsOperation.jsp
Source
History
HelloServlet
Output
Run (HelloServlet)
Apache Tomcat or TomEE Log
Apache Tomcat or TomEE
30-Mar-2024 18:12:51.343 INFO [http-nio-8080-exec-5] org.apache.catalina.startup.HostConfig.deployDescriptor Deployment of deployment descriptor [D:\Apache\conf\Catalina\localhost\error.xml] has fin
30-Mar-2024 18:12:51.348 INFO [http-nio-8080-exec-9] org.apache.catalina.util.LifecycleBase.start The start() method was called on component [StandardEngine[Catalina].StandardHost[localhost].Standar

localhost:8080/error/HelloServlet
localhost:8080/error/HelloServlet?name=...
Hello, Arina!
Apa khabar?
Waktu dan tarikh di Server ialah Sat Mar 30 18:19:18 MYT 2024

Task 6: Writing a Simple JSP Program

Objective : Writing a simple plain JSP program

Problem

Description : Write a simple plain JSP program to display "Welcome to [MATRICNUMBER]..!"

Estimated time : 15 minutes

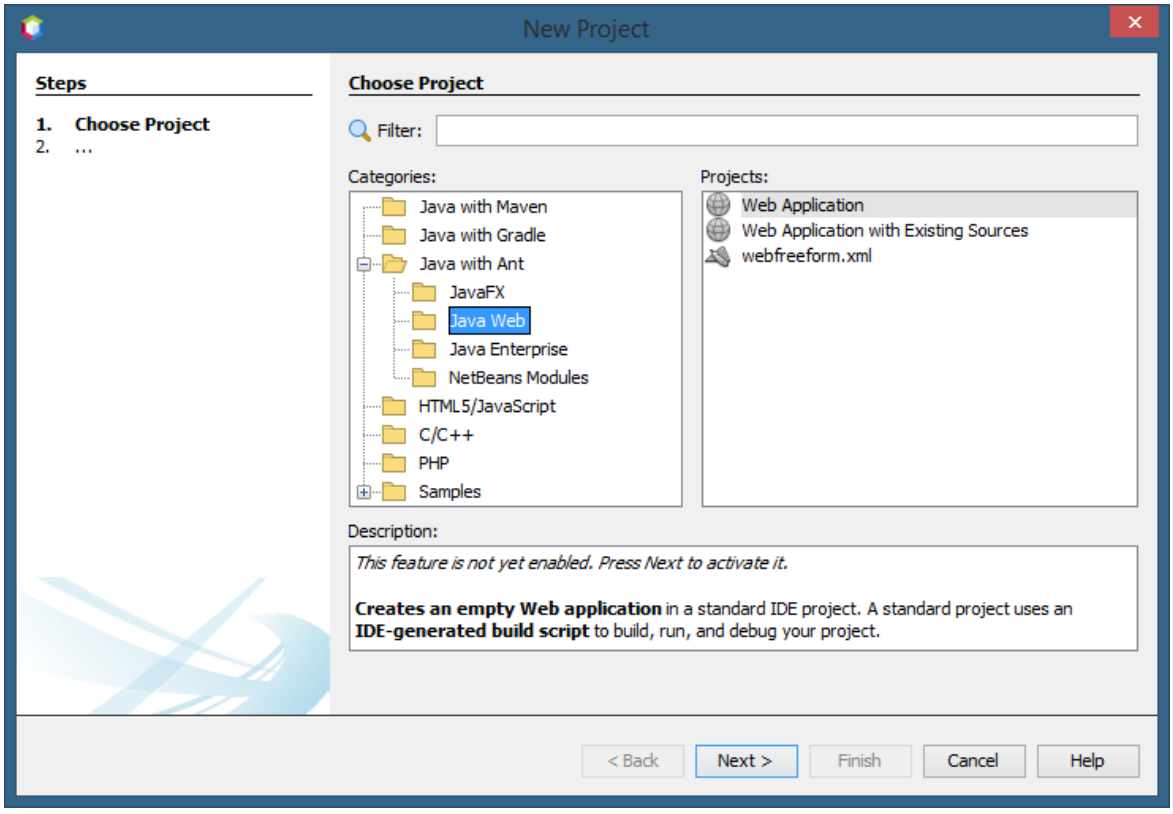
1. Create a directory C:\[MatricNumber] Lab

2. Go to C:\ [MatricNumber] Lab's directory and create sub-directory as Lab 1 - JSP.

3. Open your NetBeans.

4. Go to File -> New Project

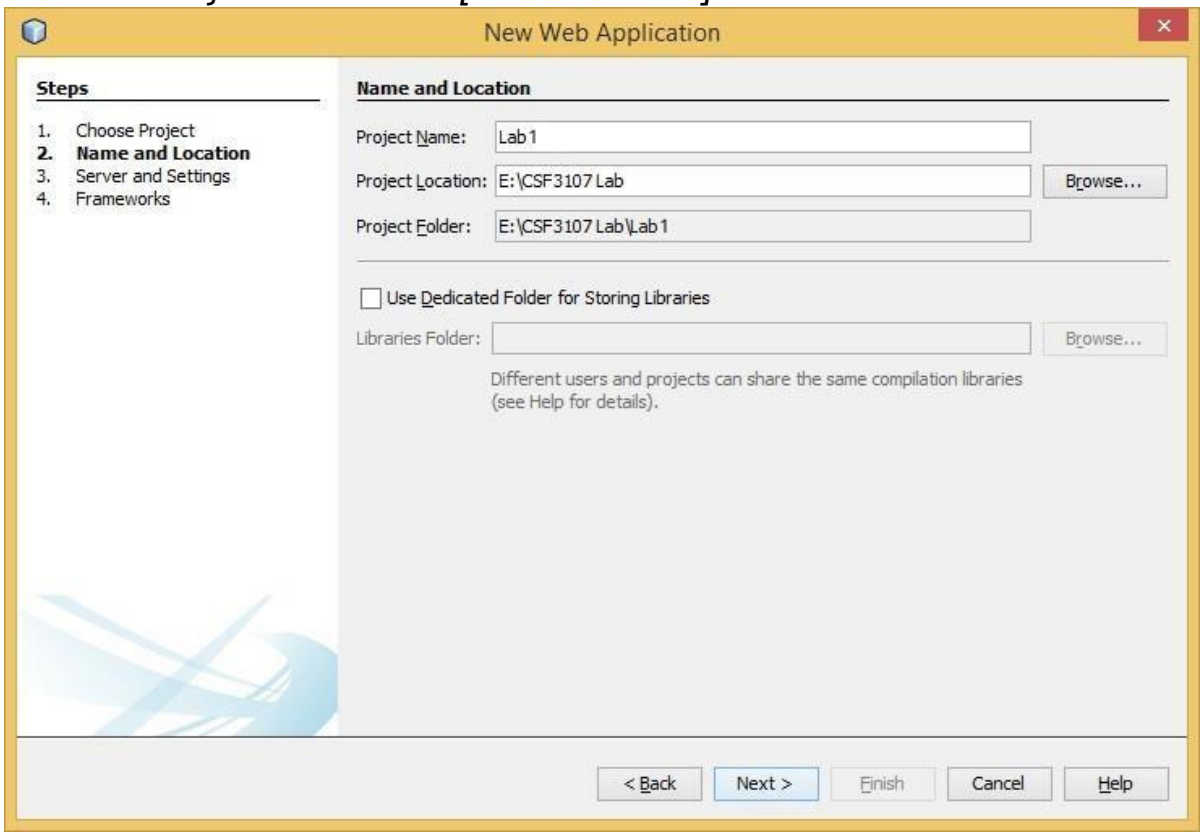
5. Select Select Java with Ant -> Java Web -> Web Application and click Next.



6. Click the Next button.

7. Type Project Name: *Lab1*.

8. Choose Project Location: *C:\[MATRICNUMBER]\Lab 1*.

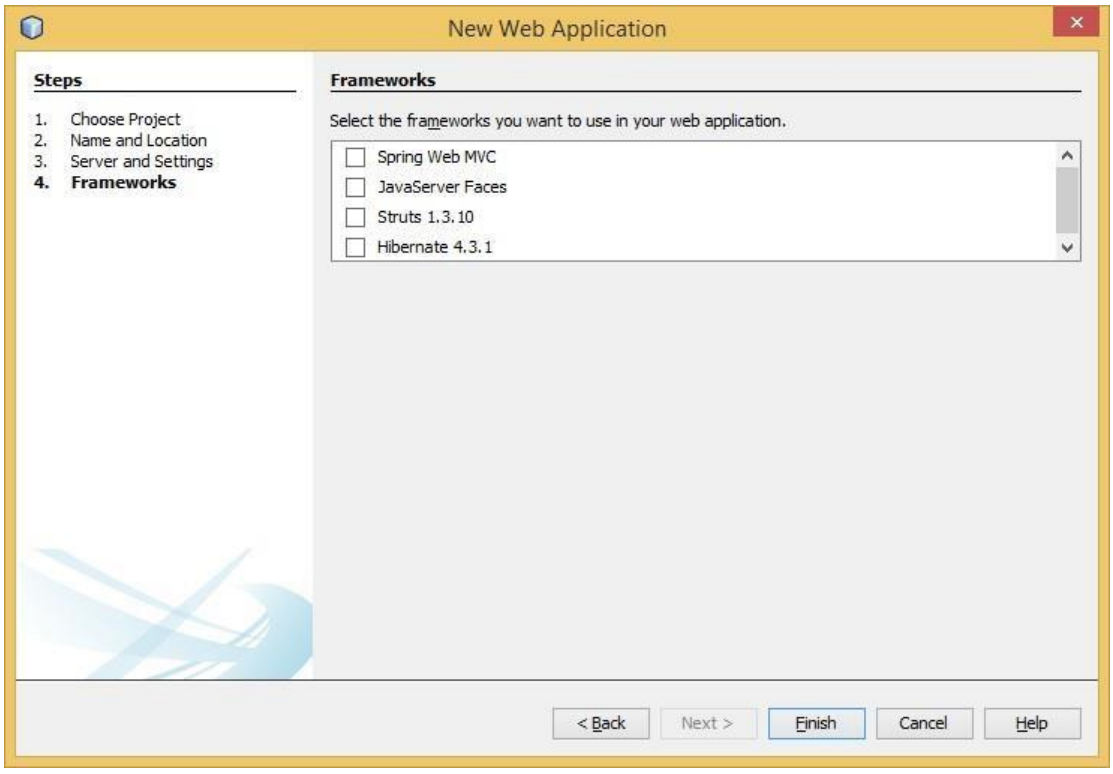


9. Click the *Next* button.

10. Select Server: *Apache Tomcat or TomEE*

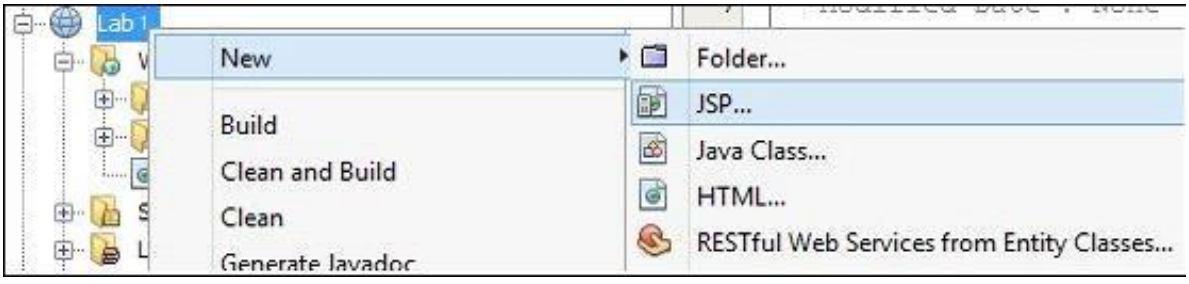
11. Select Java EE Version: Java EE 6 Web.

12. Click the Next button.

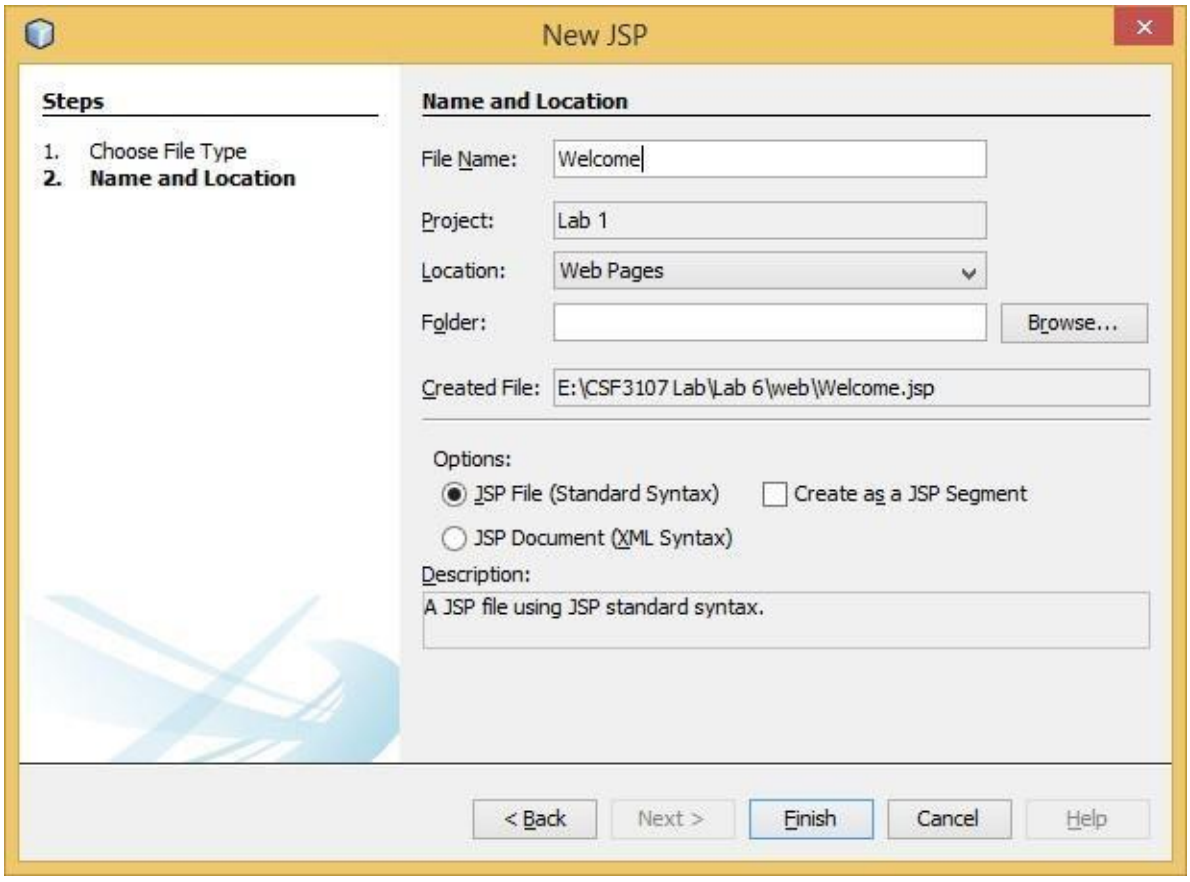


13. Click the Finish button.

14. Create a new JSP's file.



15. Type file name as *Welcome*.



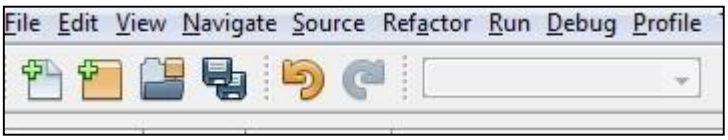
16. Click the *Finish* button.

17. Type title as *[MATRICNUMBER] – Web Programming 2*

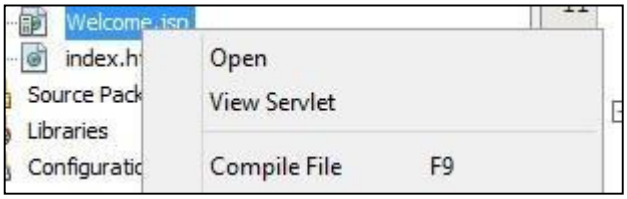
18. Type header1 as *Welcome to [MATRICNUMBER]…!*.

```
<%--
    Document   : Welcome.jsp
    Created on : 29-Mar-2016, 09:46:05
    Author     : Mohamad Nor Hassan
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>CSF3107 - Web Programming 2</title>
    </head>
    <body>
        <h1>Welcome to CSF3107...!</h1>
    </body>
</html>
```

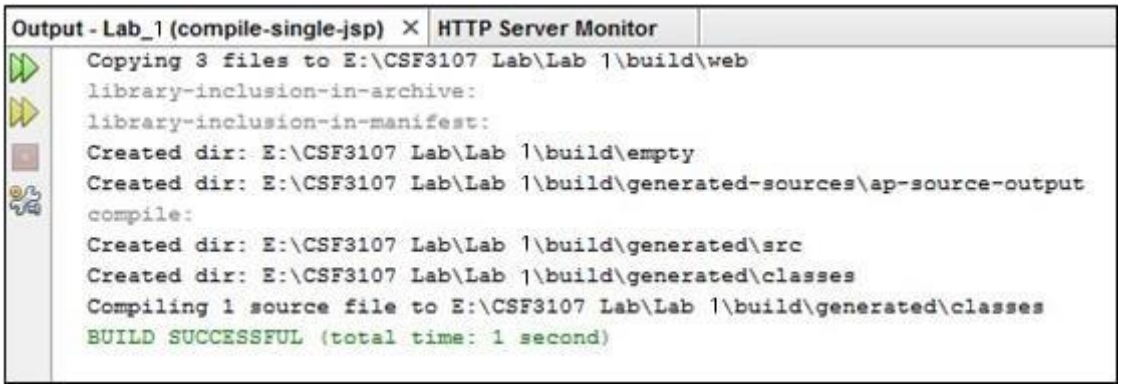
19. Click *SaveAll* icon



20. Right-click file *Welcome.jsp* and click *Compile File* (F9).

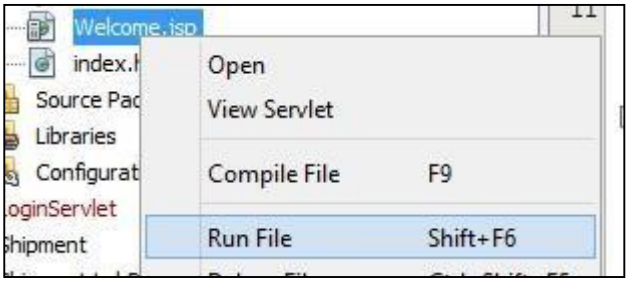


21. You will get notification message the bottom of Netbeans IDE with the green colour.



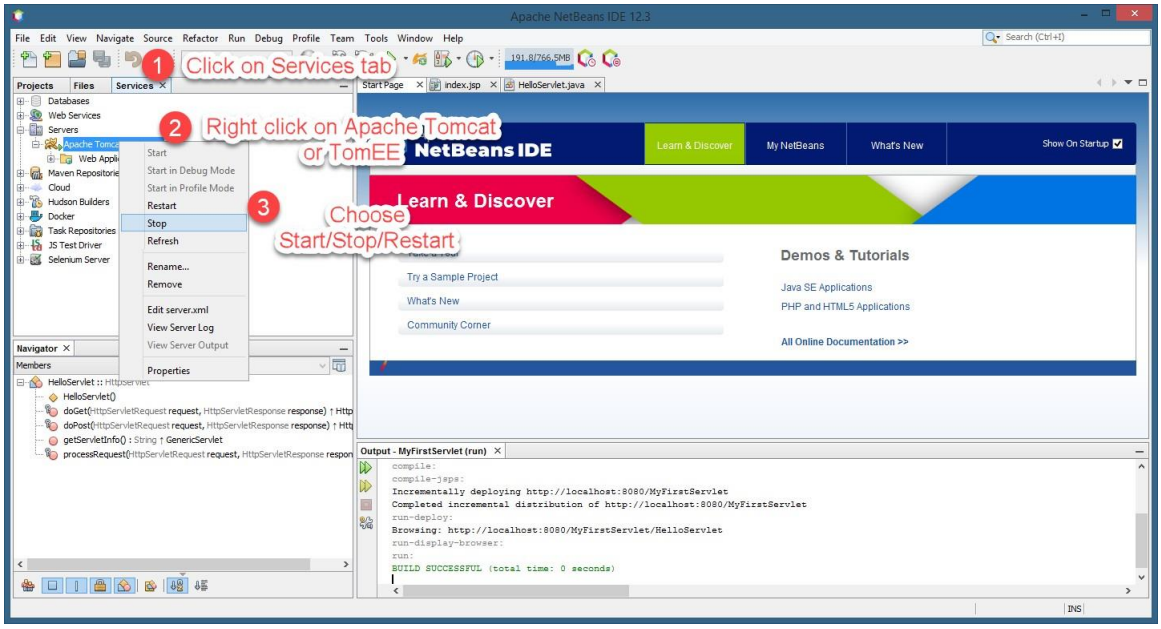
Note: Before running any JSP's files for the first time upon opening your Netbeans IDE, you need to start your web server (i.e., Apache Tomcat).
Note: Avoid these steps if Apache Tomcat already starts.

26. Go *to the Project*'s tab. Then right click *Welcome.jsp* file and click *Run File* (Shift+F6).



27. The output will appear in a web browser.

Note: Beside using XAMPP to start or stop the Apache Tomcat, you may also do it directly from Netbeans as shown in the figure below.



Reflection

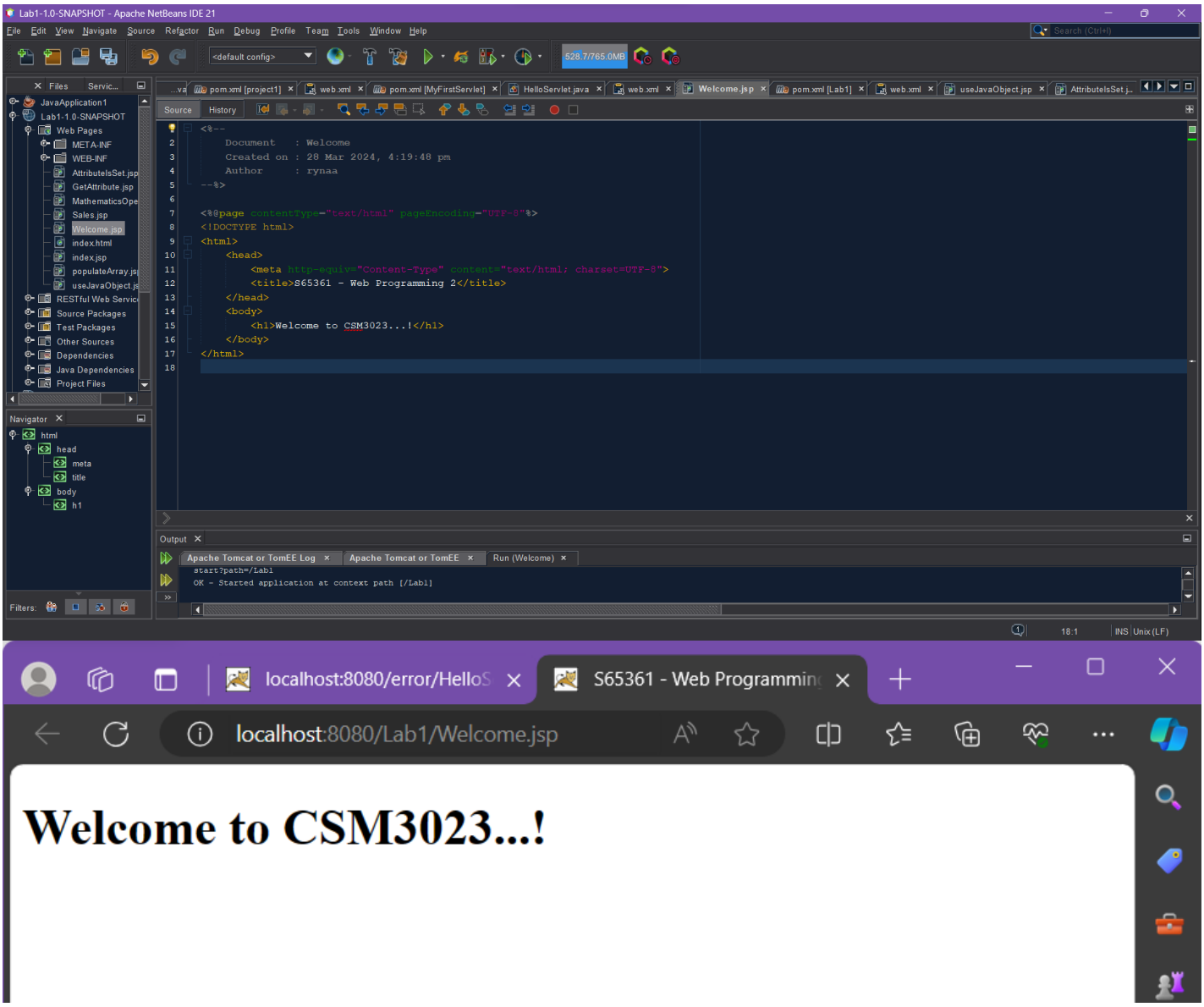
1. What have you learned from this exercise?

   Writing a simple JSP program to get the output in the browser.

2. Explain the general concept of how the JSP's file work?
   Java Server Pages is a server-side technology which is used for creating web applications. It also used to create dynamic web content. It also consist of both HTML tags and JSP tags.

3. Based on your observation of the previous tasks (Task 3 and Task 4), what are the differences you can find between servlet and JSP?
Servlet is a java code while JSP is a HTML-based compilation code.

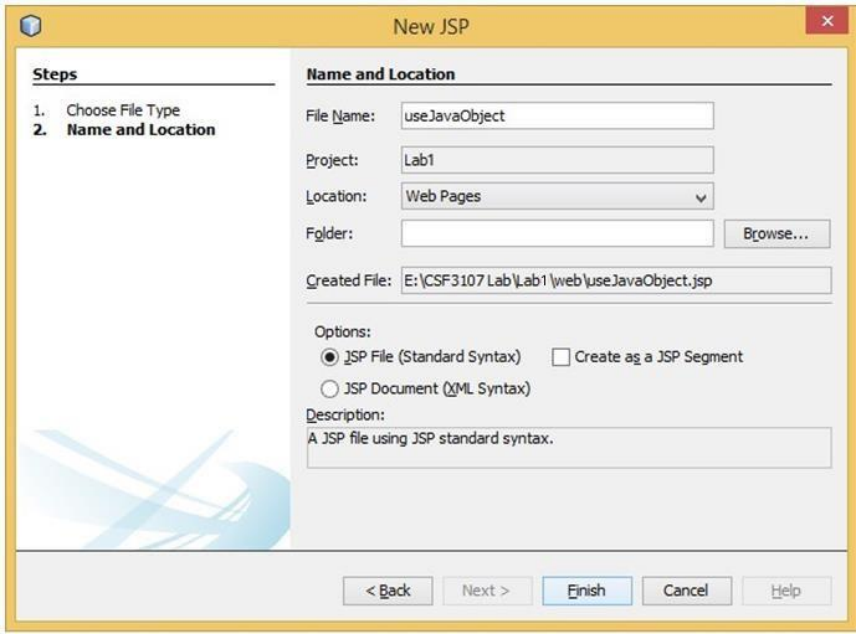
Source code


Lab1-1.0-SNAPSHOT - Apache NetBeans IDE 21
Welcome.jsp
Document : Welcome
Created on : 28 Mar 2024, 4:19:48 pm
Author : rynaa
<title>S65361 - Web Programming 2</title>
<h1>Welcome to CSM3023...!</h1>
localhost:8080/Lab1/Welcome.jsp
Welcome to CSM3023...!

Task 7: Use Java Reference Datatype/Class Wrapper in JSP

Objective	:	Using Java's object in JSP page.
Problem Description	:	Display the current date, perform auto refresh header in JSP's page.
Estimated time	:	20 minutes

1. Go to project *Lab1*.
2. Create a new JSP's file as *useJavaObject*.



3. Click the *Finish* button.
4. Change the title *Using Java's object in JSP page*.
5. Change the <h1> as *Display Current Date and* perform auto refresh header.

6. Add *Java util* package for Date reference.

```
7    <%@page contentType="text/html" pageEncoding="UTF-8"%>
8    <%@page import="java.util.Date.*"%>
9
```

7. Write a Java Scriplet to create Date's object and display the current date and time.

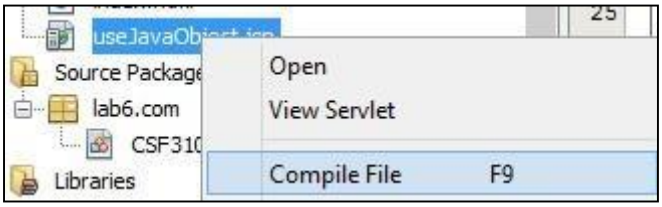
```
<body>
    <h1>Display Current Date and perform simple Mathematics operations </h1>

    <%
        Date todayDate = new Date();
        out.print("<p>Current date and time is " + todayDate.toString() + "</p>");
    %>
</body>
```

8. Continue writing a code to perform auto refresh header.

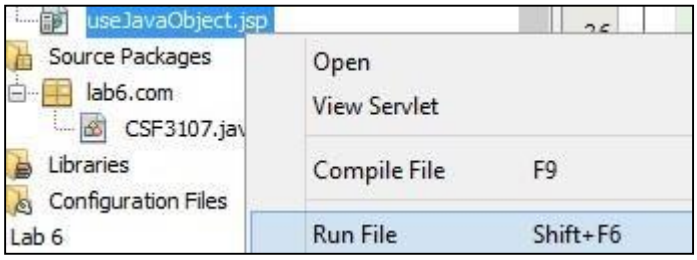
```
<%
    // Set refresh, autoload time as 5 seconds
    response.setIntHeader("Refresh", 5);

    %>
```

9. Save your file.

10. Right-click *useJavaObject.jsp* and compile the program.



11. Finally, right-click *useJavaObject.jsp* and choose Run File to run the program.



12. Review the output display in the browser.

Reflection

1. What have you learnt from this exercise?

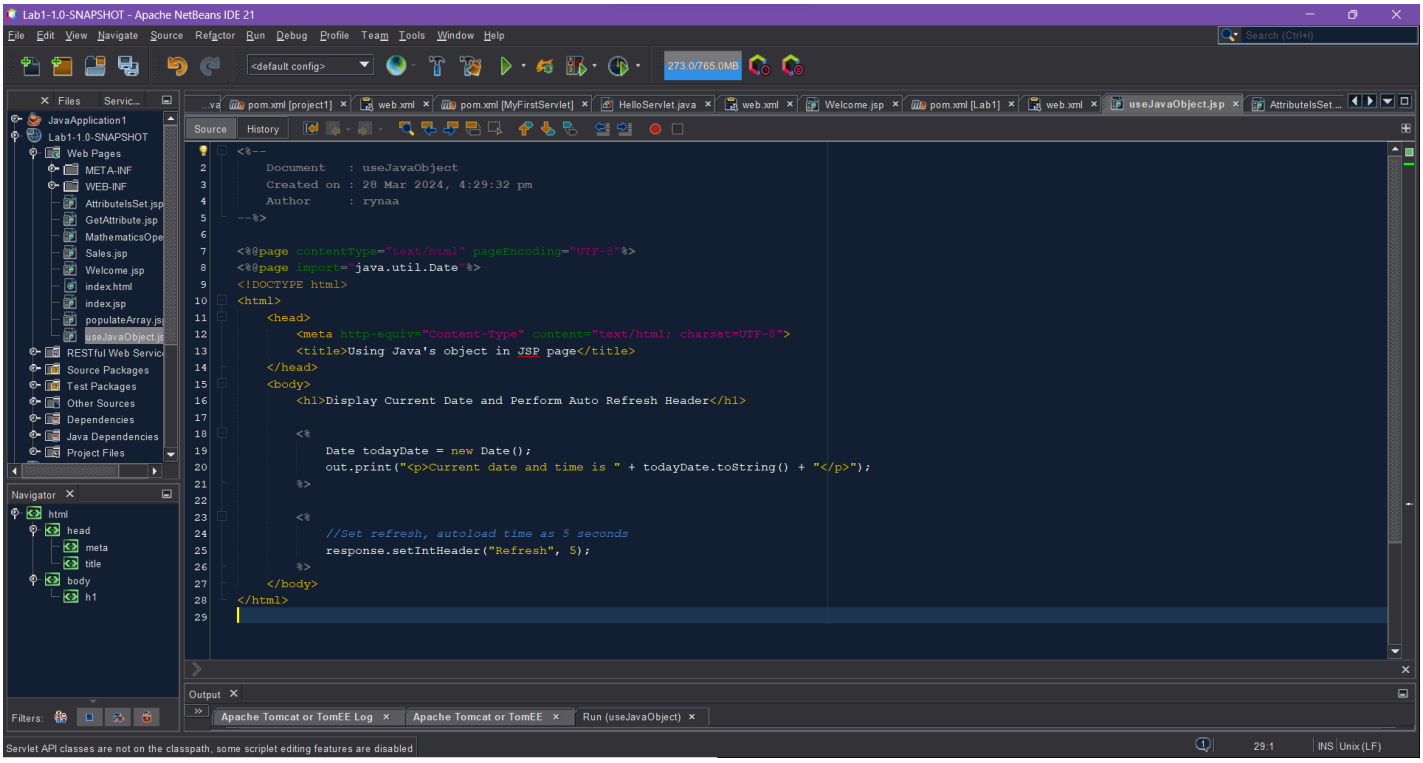
   Use Java Reference Datatype/Class Wrapper in JSP such as Date and time.

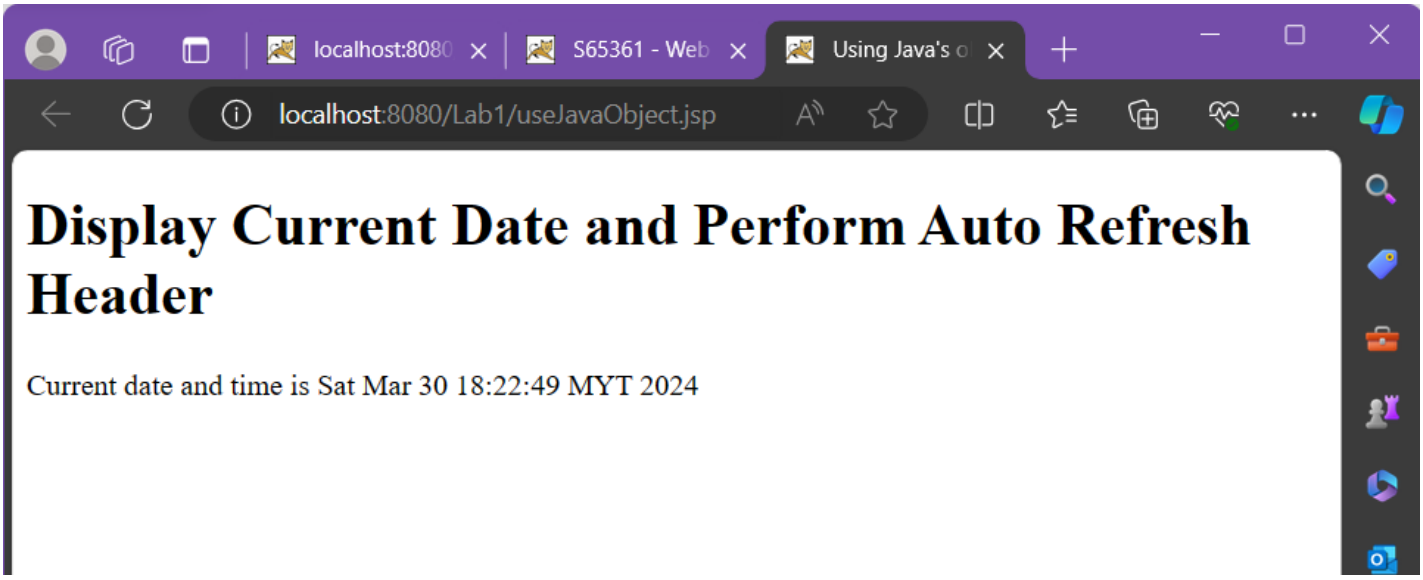
2. What is Java Scriplet?
   Scriplet is a code used to contain any code fragment that is valid for the scripting language used in a page. The syntax for a scriplet is <% scripting-language-statements %>.

3. How to use Java code in your JSP's page?

   By using JSP expressions, JSP scriplets and JSP declarations.

Source code


Lab1-1.0-SNAPSHOT - Apache NetBeans IDE 21
useJavaObject.jsp
<%--
Document : useJavaObject
Created on : 28 Mar 2024, 4:29:32 pm
Author : rynaa
--%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.util.Date"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Using Java's object in JSP page</title>
</head>
<body>
<h1>Display Current Date and Perform Auto Refresh Header</h1>
<%
Date todayDate = new Date();
out.print(" Current date and time is " + todayDate.toString() + " ");
%>
<%
//Set refresh, autoload time as 5 seconds
response.setIntHeader("Refresh", 5);
%>
</body>
</html>
Servlet API classes are not on the classpath, some scriplet editing features are disabled


localhost:8080/Lab1/useJavaObject.jsp
Display Current Date and Perform Auto Refresh Header
Current date and time is Sat Mar 30 18:22:49 MYT 2024

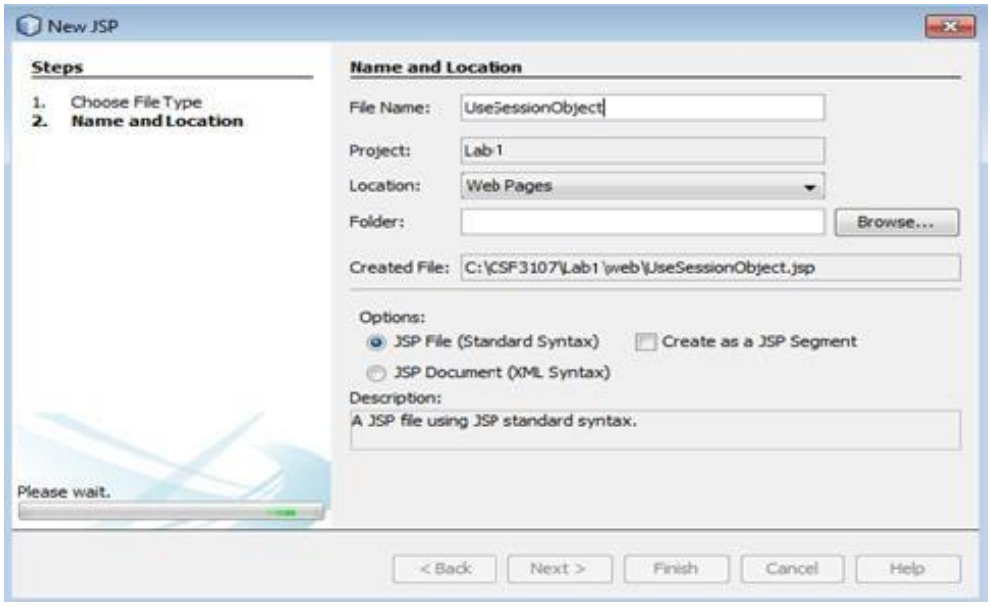
Task 8: Using JSP Implicit object in JSP page

Objective	:	Using JSP Implicit object (Session) in JSP page.
Problem Description	:	Using Session object, perform simple Mathematics operations in JSP's page.
Estimated time	:	30 minutes

1. Go to Project *Lab1*.

2. To create a JSP's page, right click *Lab1*-> *New* -> *JSP*.

3. Create a new JSP's file as *AttributeIsSet*.



4. Click the *Finish* button.

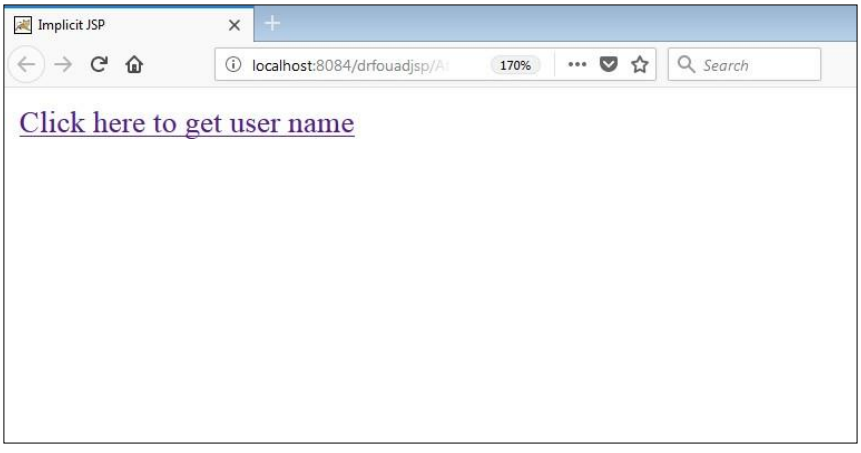
5. Source code for *AttributeIsSet.jsp* will appear.

6. Write the code below:

```
<%--
    Document   : jsp
    Created on : 20-Feb-2018
    Author     : Dr. Faizah Aplop
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Implicit JSP</title>
    </head>
    <body>

        <% session.setAttribute("user", "Fouad Abdulameer");%>
        <a href="GetAttribute.jsp">Click here to get user name </a>

    </body>
</html>
```

9. Save and compile *AttributeIsSet.jsp* file.

10. Run the *AttributeIsSet.jsp* file, and you should get the interface as below:



11. Repeat step 1 and step 2.

12. Key-in File Name: *GetAttribute*.

13. Click the *Finish* button.

14. Source code for *GetAttribute.jsp* will appear.

15. Write the *GetAttribute* code.

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Implicit JSP</title>
    </head>
    <body>

        <%
            String name = (String) session.getAttribute("user");
            out.println("User Name is: " + name);
        %>


    </body>
</html>
```

17. Compile *GetAttribute.jsp* file.

18. Run the *AttributeIsSet.jsp* file and click on the link.

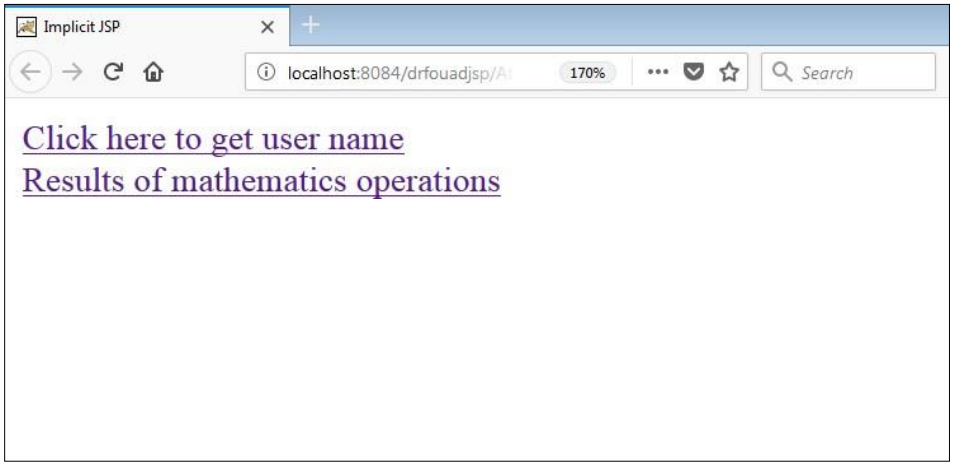
19. Add math package to perform simple Mathematics operation in your page.

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.math.*"%>
```

20. Create a new JSP's file as *MathematicsOperations* to perform addition, multiplication and find the square roots of the number.

```
<%
    int num1 = 25;
    int num2 = 10;
    int addition_output;
    int multiply_output;
    double squareroot = 0.00;

    java.util.Formatter myFormat = new java.util.Formatter();

    //perform basic arithematics operations,,,
    addition_output = num1 + num2;
    multiply_output = num1 * num2;

    //Find square root for variable num1..
    squareroot = (double)(Math.sqrt(num1));

    out.print("<p>Addition num1 and num2 is " + addition_output + "</p>");
    out.print("<p>Multiplication num1 and num2 is " + multiply_output + "</p>");

    out.print("<p></p>");
    out.print("<p>Square root of " + num1 + " is " + myFormat.format("%.2f", squareroot) +  "</p>");
%>
```

21. You should get the interface as below:



22. Review the outputs display in the browser.

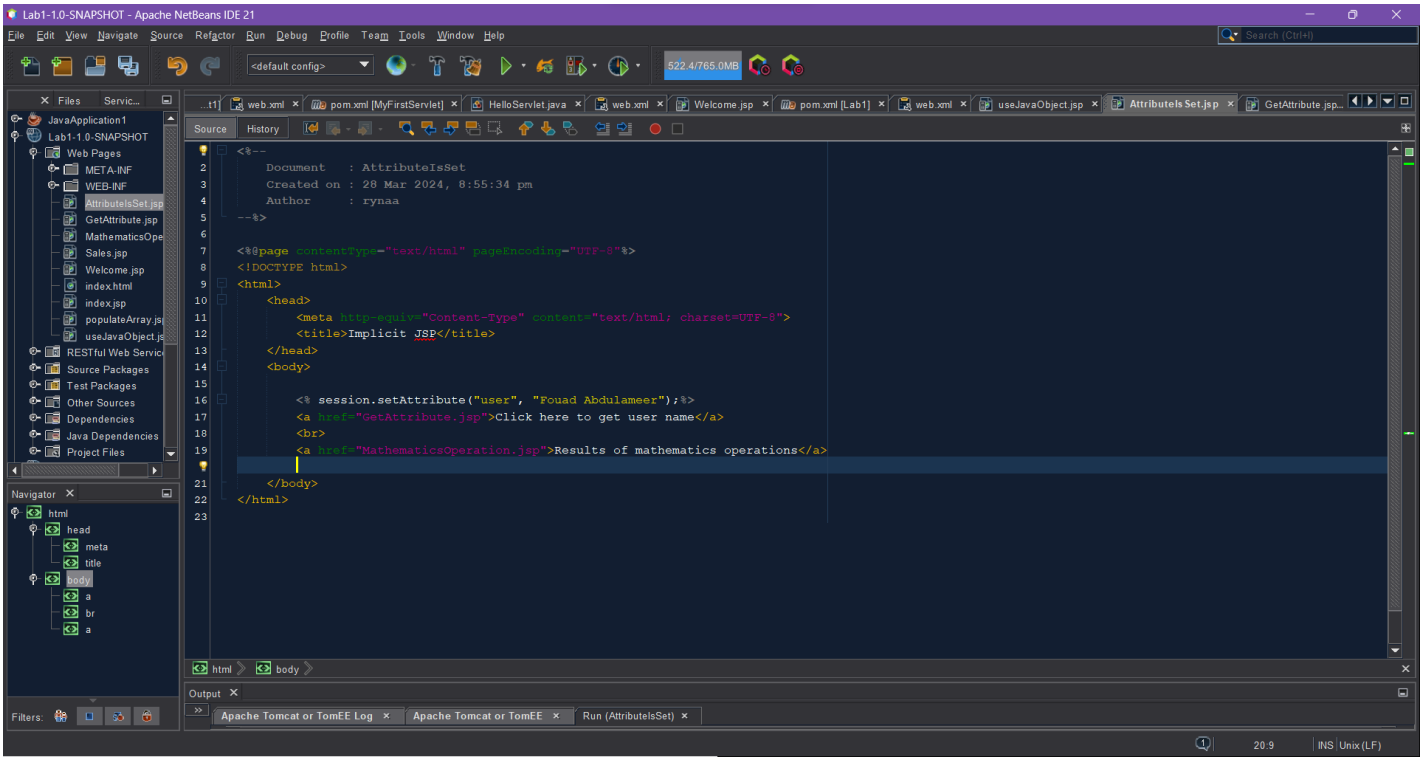
Reflection

1. How do you want to submit specific information from one form to next form?
   Use session Attributes

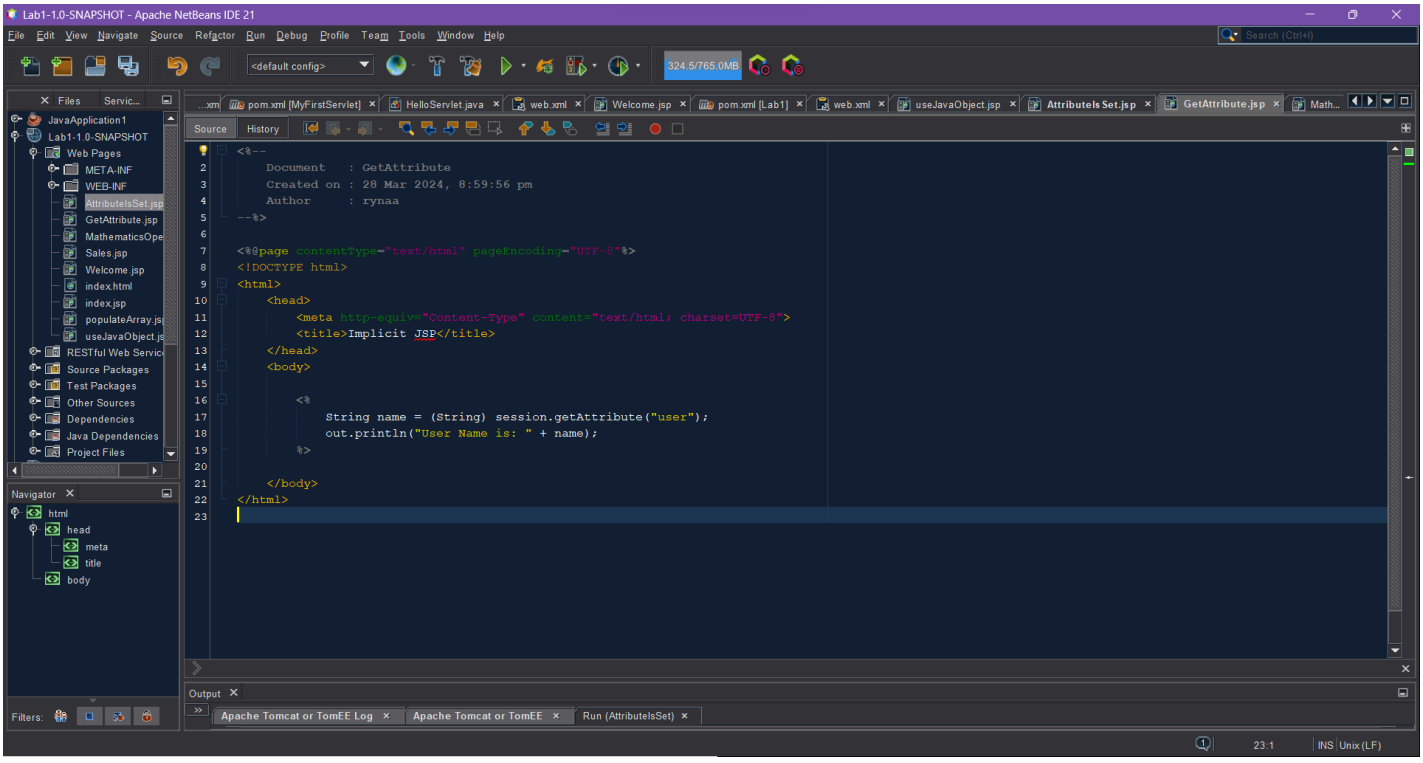
2. What happened if the field name you specify in *request.getParameter ("field_name")* in the second page is different from the field name you defined in the first page?
   It will return 'null'.

Source code

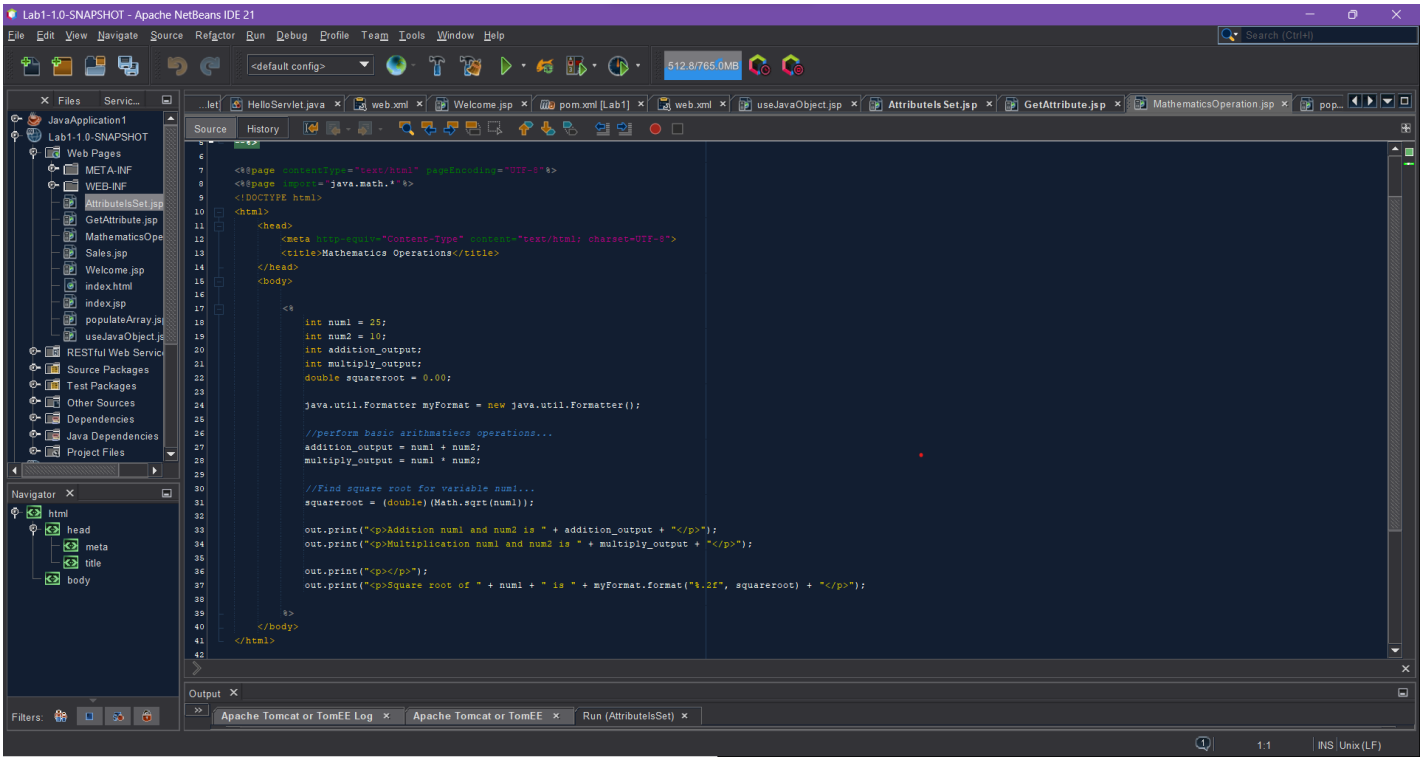
AttributeIsSet.jsp

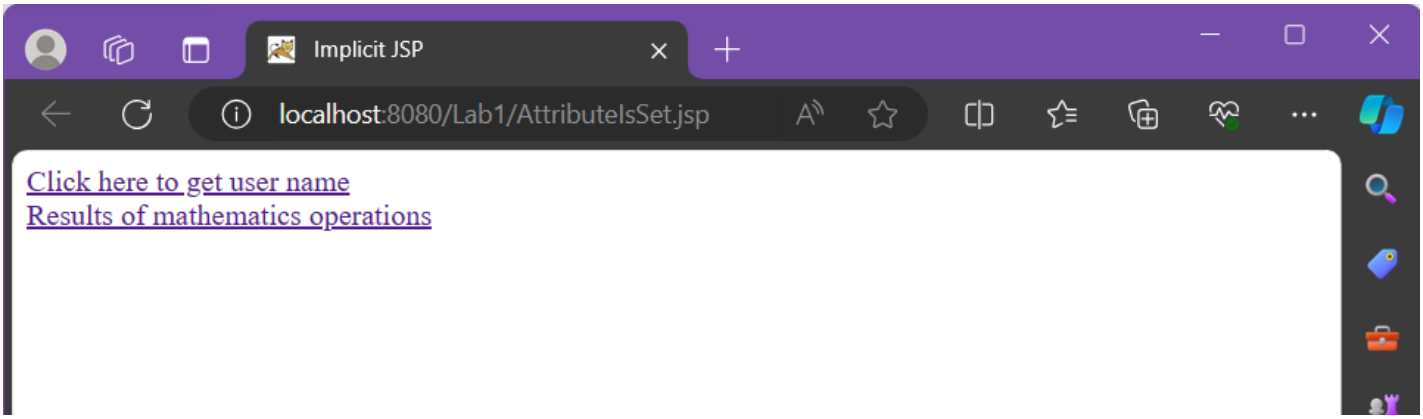


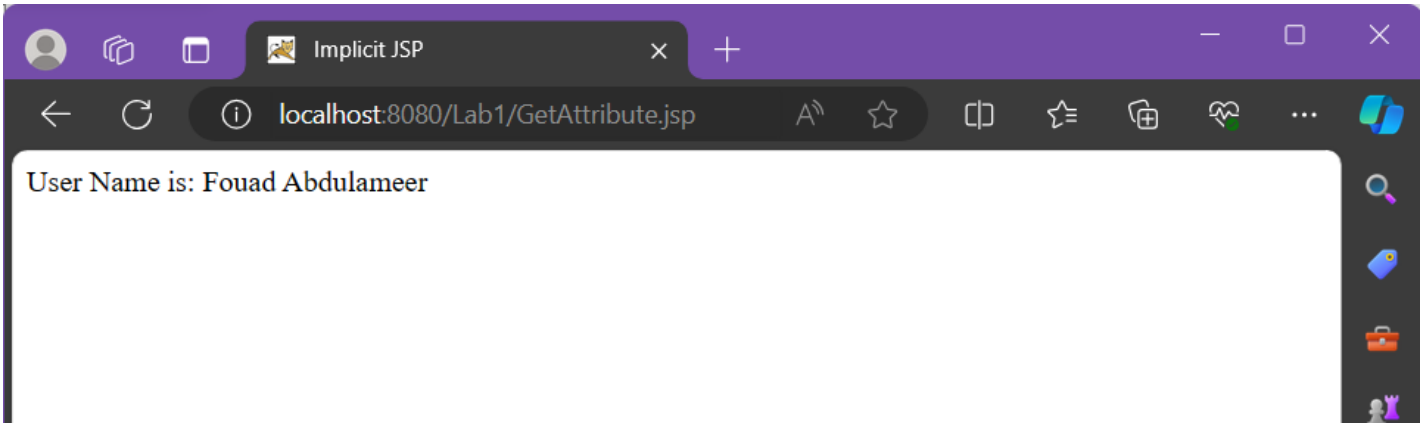
GetAttribute.jsp

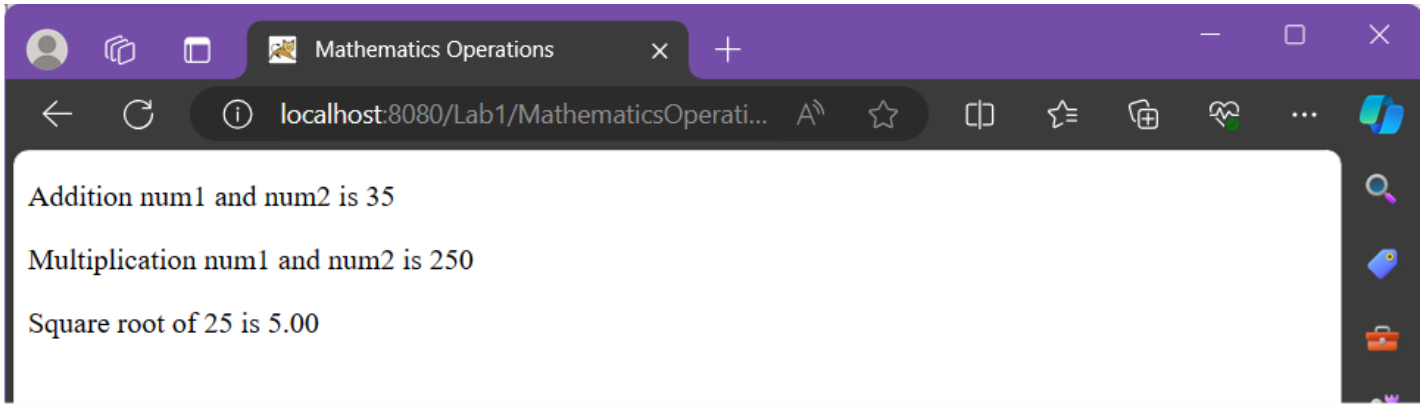


MathematicsOperations.Jsp





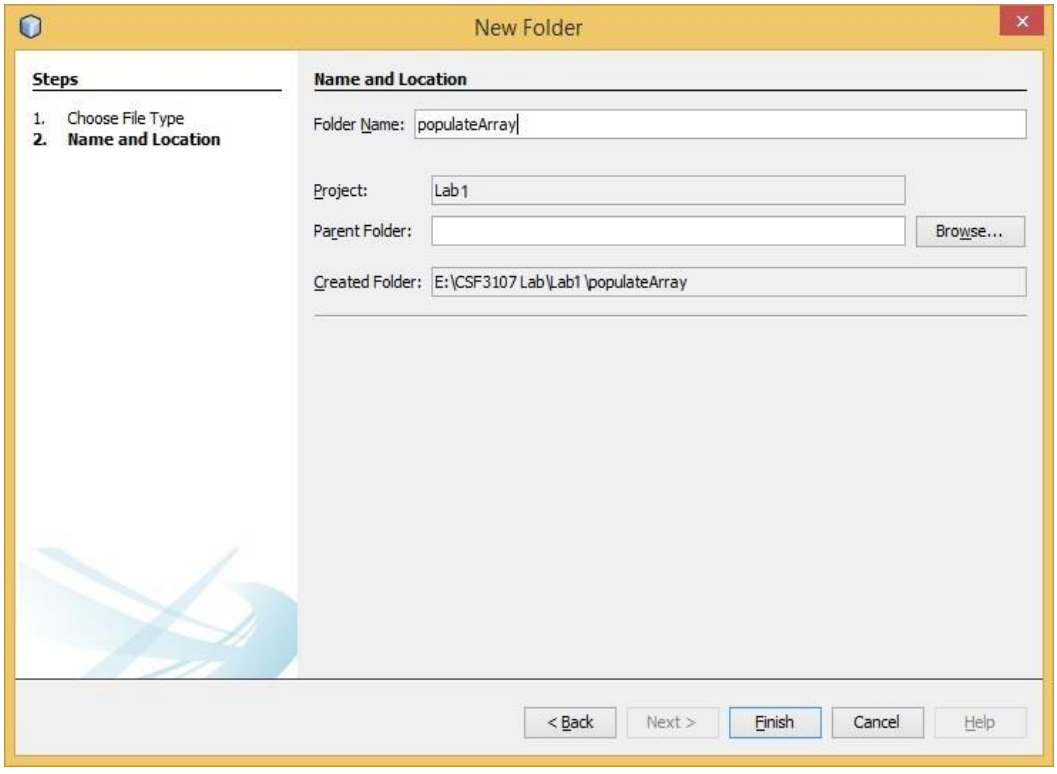



Mathematics Operations
localhost:8080/Lab1/MathematicsOperati...
Addition num1 and num2 is 35
Multiplication num1 and num2 is 250
Square root of 25 is 5.00

Task 9: Populate Array values into HTML's Table

Objective	:	Read Java array and populate it into HTML's table.
Problem Description	:	i. Create a 2D array that store sales data. ii. Then, read an array and populate into HTML's table.
Estimated time	:	50 minutes

1. Go to Project *Lab1*.
2. To create a JSP's page, right click *Lab1* -> *New* -> *JSP*.
3. Key-in File Name: *populateArray*.



4. Click the *Finish* button.

5. Prepare standard HTML'a markup for page *populateArray.jsp*.

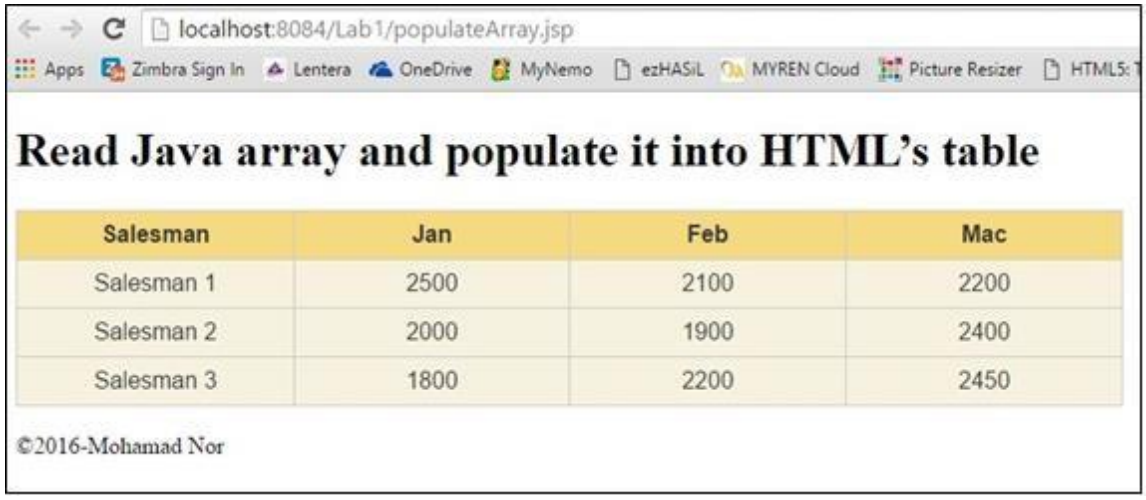
6. Write a Java Scriplet and store the following information into an array;

	Jan	Feb	Mac
Salesman 1	2500	2100	2200
Salesman 2	2000	1900	2400
Salesman 3	1800	2200	2450

7. Read the array and populate its value into HTML's table.

8. Save and compile *populateArray.jsp* file.

9. Run the *populateArray.jsp* file and sample of output shows as below:



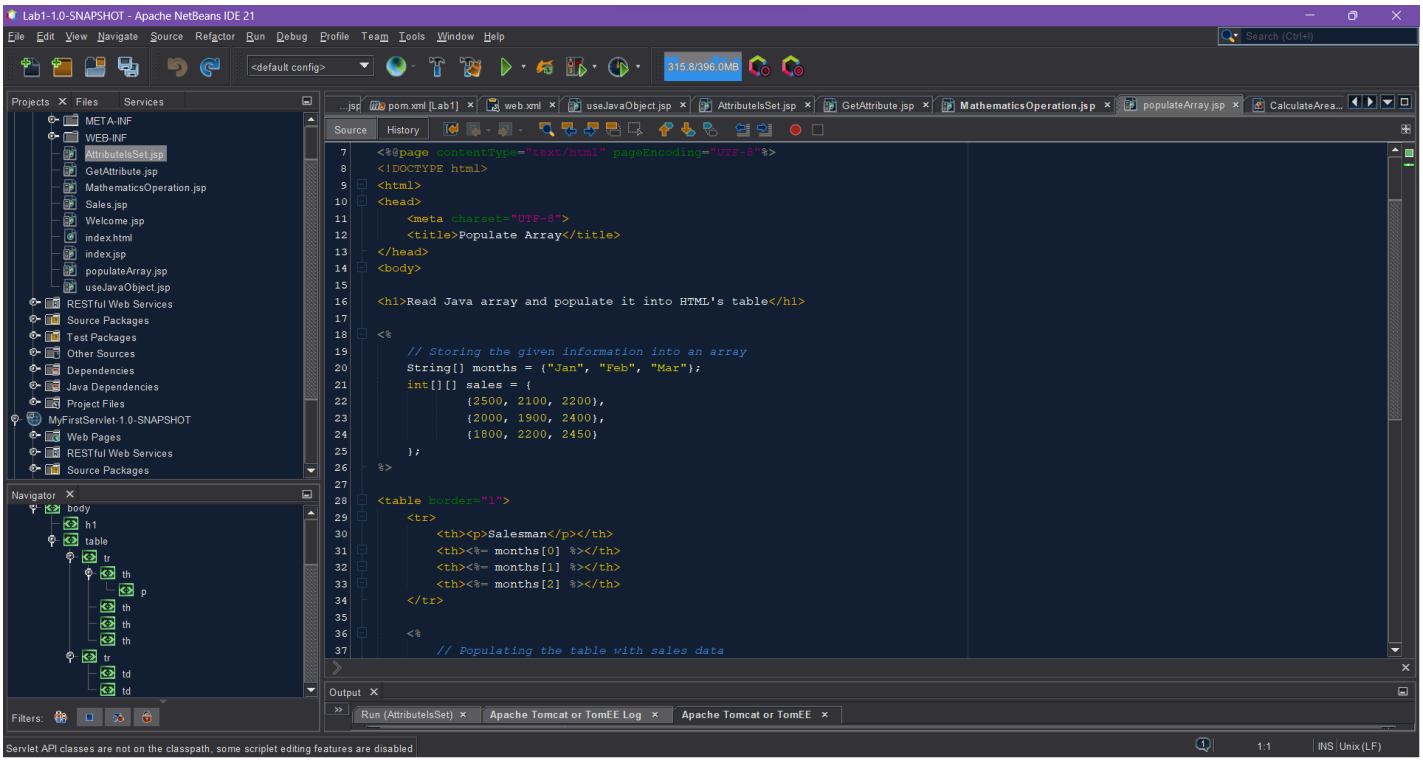
Reflection

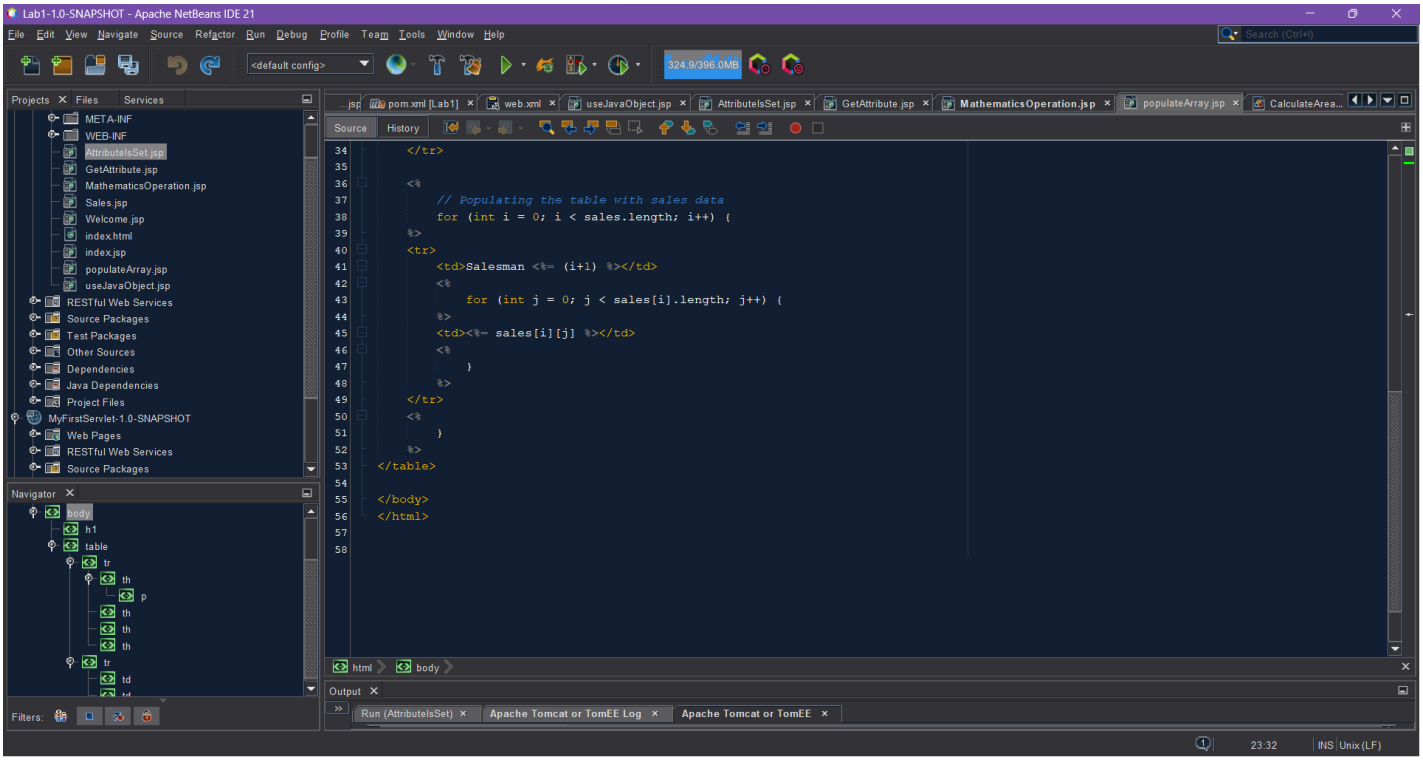
1. Write a sample syntax to declare 2D Java array.
   String[][] StringArray = new String[2][4];

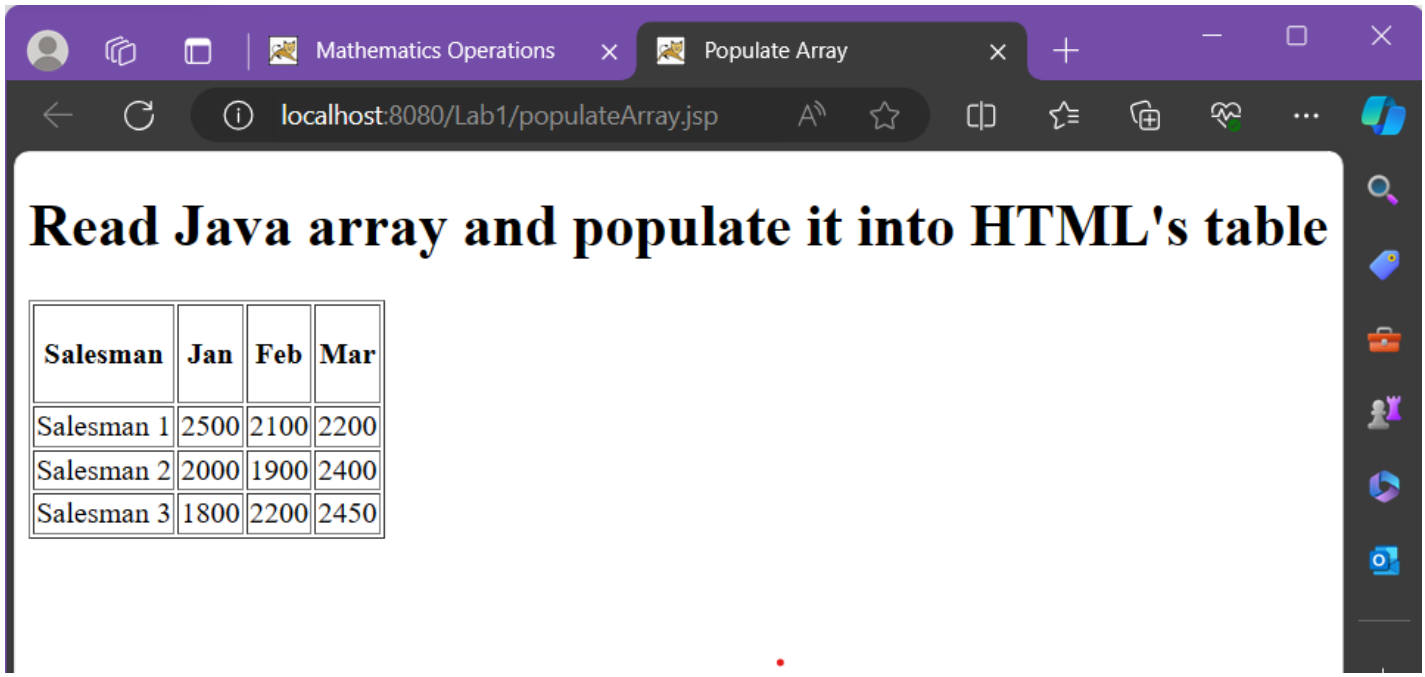
2. Define a sequence of steps on how you accomplish Task 7.

3. What is the difference between HTML's page and JSP's page?
   HTML pages are processed on the client-side by web browsers. It was static and do not involve server-side processing while JSP pages are processed on the server-side by a JSP container such as Tomcat.

Source code

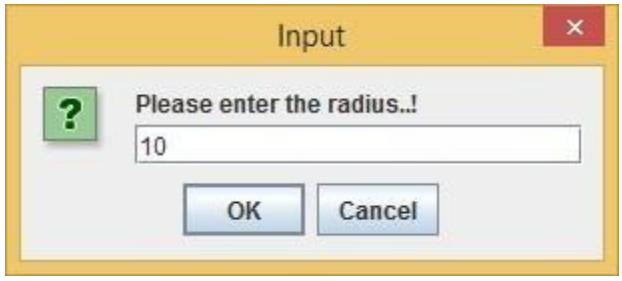

Lab1-1.0-SNAPSHOT - Apache NetBeans IDE 21
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Projects Files Services
META-INF
WEB-INF
AttributeIsSet.jsp
GetAttribute.jsp
MathematicsOperation.jsp
Sales.jsp
Welcome.jsp
index.html
index.jsp
populateArray.jsp
useJavaObject.jsp
RESTful Web Services
Source Packages
Test Packages
Other Sources
Dependencies
Java Dependencies
Project Files
MyFirstServlet-1.0-SNAPSHOT
Web Pages
RESTful Web Services
Source Packages
Navigator
body
h1
table
tr
th
p
th
th
th
tr
td
td
Filters:
pom.xml [Lab1]
web.xml
useJavaObject.jsp
AttributeIsSet.jsp
GetAttribute.jsp
MathematicsOperation.jsp
populateArray.jsp
CalculateArea...
Source
History
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Populate Array</title>
</head>
<body>
<h1>Read Java array and populate it into HTML's table</h1>
<%
// Storing the given information into an array
String[] months = {"Jan", "Feb", "Mar"};
int[][] sales = {
{2500, 2100, 2200},
{2000, 1900, 2400},
{1800, 2200, 2450}
};
%>
<table border="1">
<tr>
<th> Salesman </th>
<th><%= months[0] %></th>
<th><%= months[1] %></th>
<th><%= months[2] %></th>
</tr>
<%
// Populating the table with sales data
Output
Run (AttributeIsSet)
Apache Tomcat or TomEE Log
Apache Tomcat or TomEE
Servlet API classes are not on the classpath, some scriptlet editing features are disabled
1:1
INS Unix (LF)


Lab1-1.0-SNAPSHOT - Apache NetBeans IDE 21
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Projects Files Services
Navigator
body
h1
table
tr
th
p
th
th
th
tr
td
Filters:
Source
History
</tr>
<%
// Populating the table with sales data
for (int i = 0; i < sales.length; i++) {
%>
<tr>
<td>Salesman <%= (i+1) %></td>
<%
for (int j = 0; j < sales[i].length; j++) {
%>
<td><%= sales[i][j] %></td>
<%
}
%>
</tr>
<%
}
%>
</table>
</body>
</html>
html
body
Output
Run (AttributeIsSet)
Apache Tomcat or TomEE Log
Apache Tomcat or TomEE
23:32
INS Unix (LF)


Mathematics Operations
Populate Array
localhost:8080/Lab1/populateArray.jsp
Read Java array and populate it into HTML's table
Salesman Jan Feb Mar
Salesman 1 2500 2100 2200
Salesman 2 2000 1900 2400
Salesman 3 1800 2200 2450

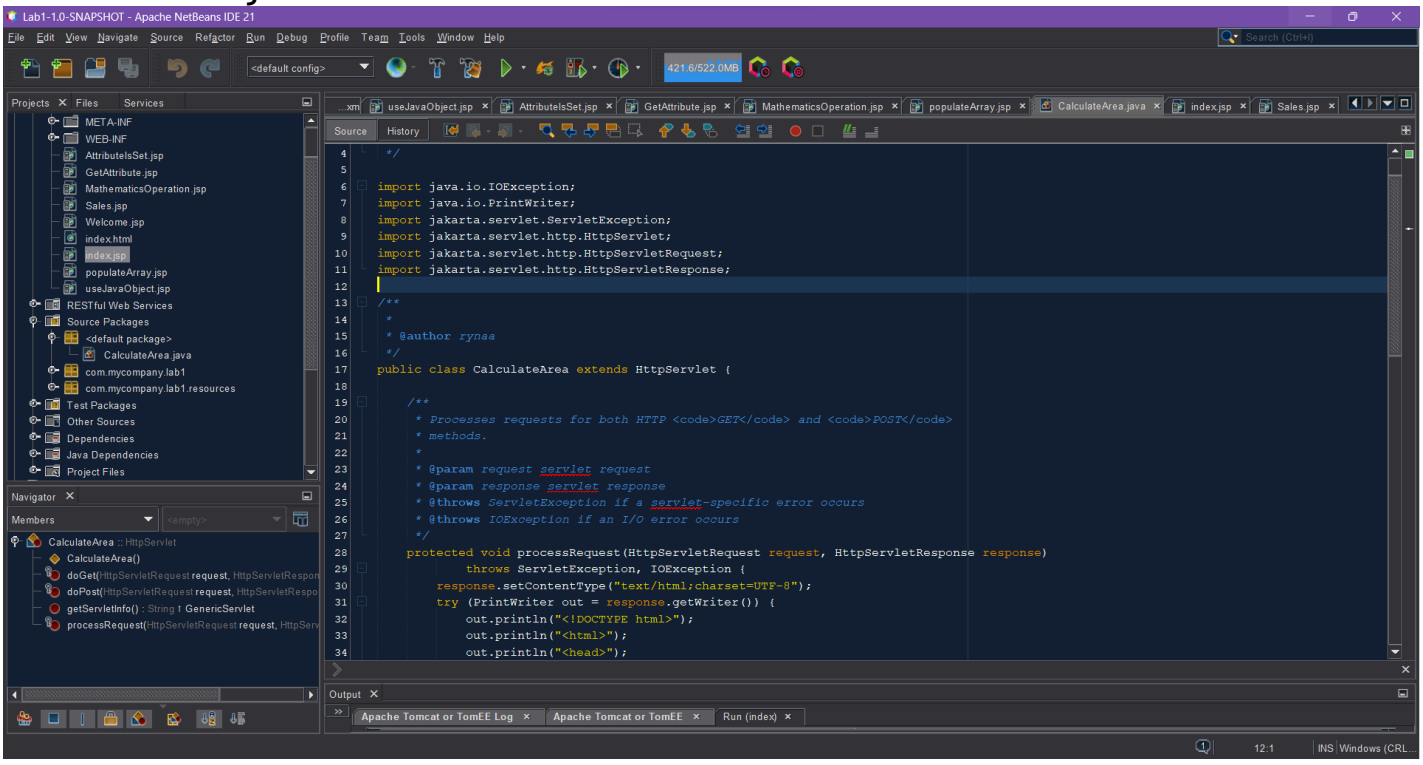
Exercise

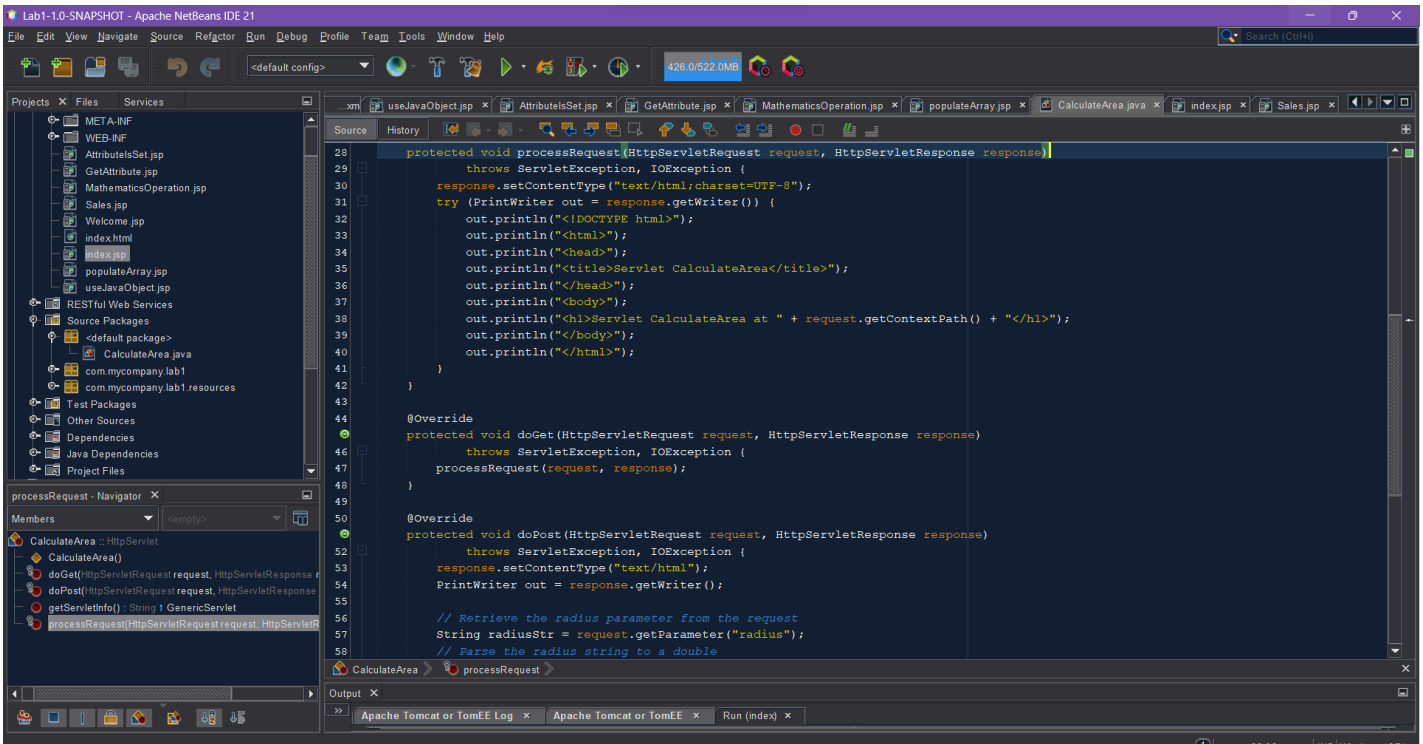
1. The following program is developed using Java Standard Edition.

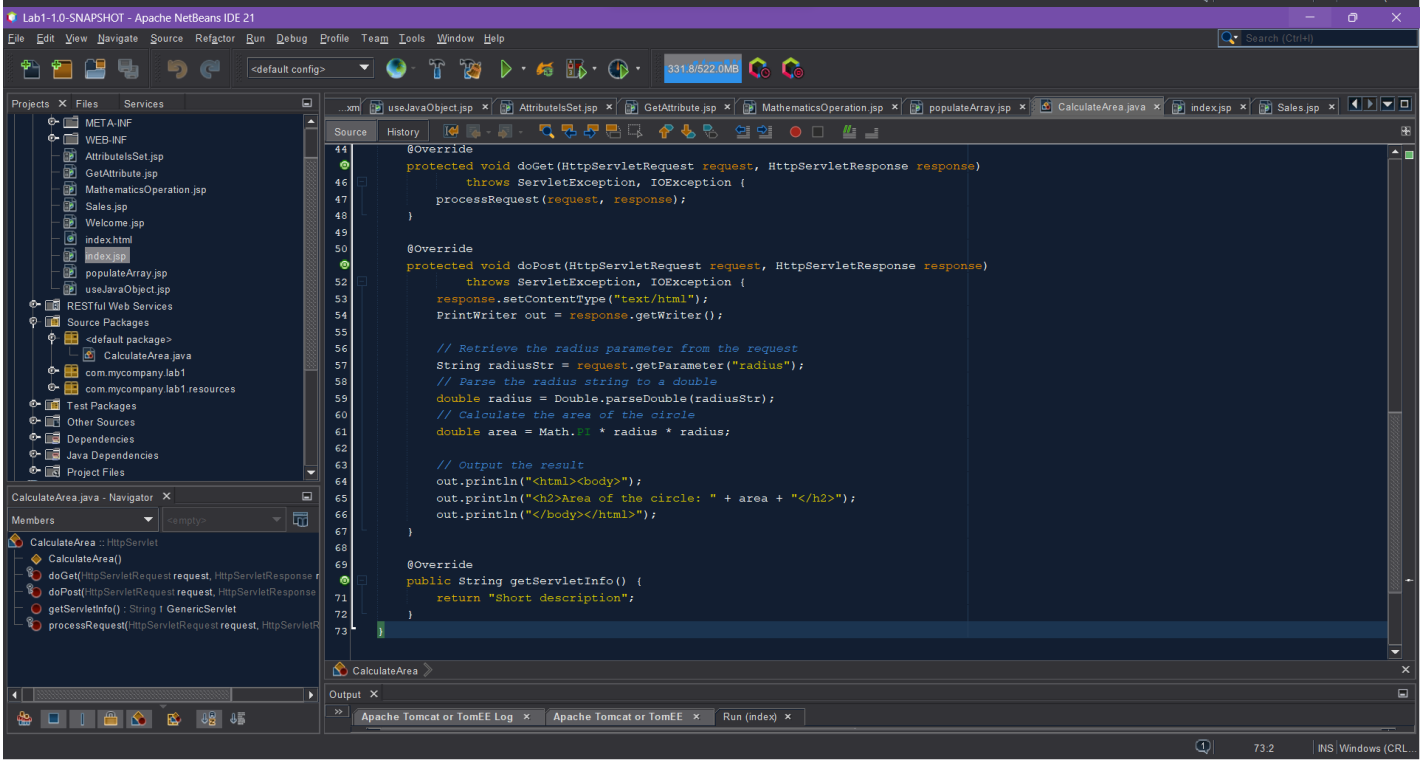


Convert this program into Web-Based application where the user can dynamically key-in radius and submit the request to get the area of a circle. You must ensure to accept only number as a radius.

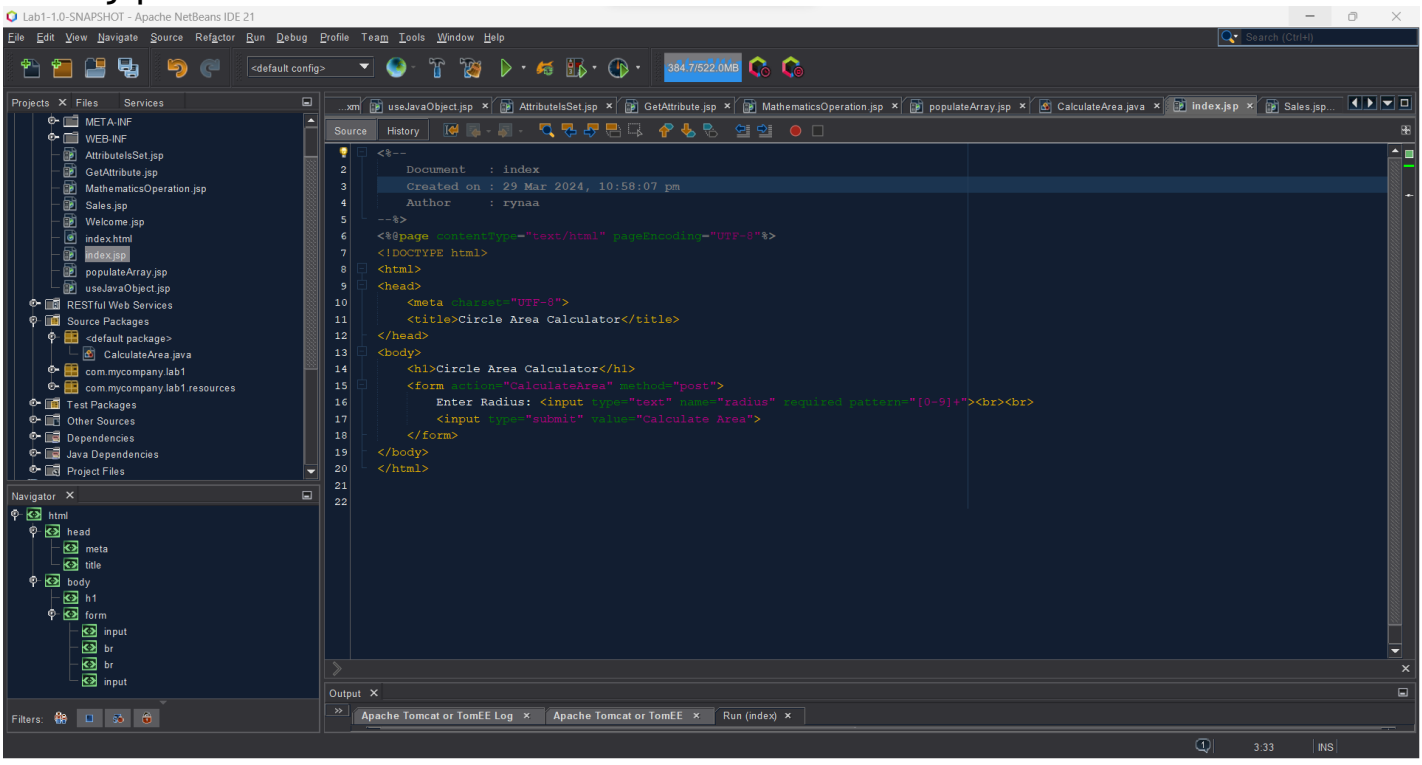
Source code
calculateArea.java



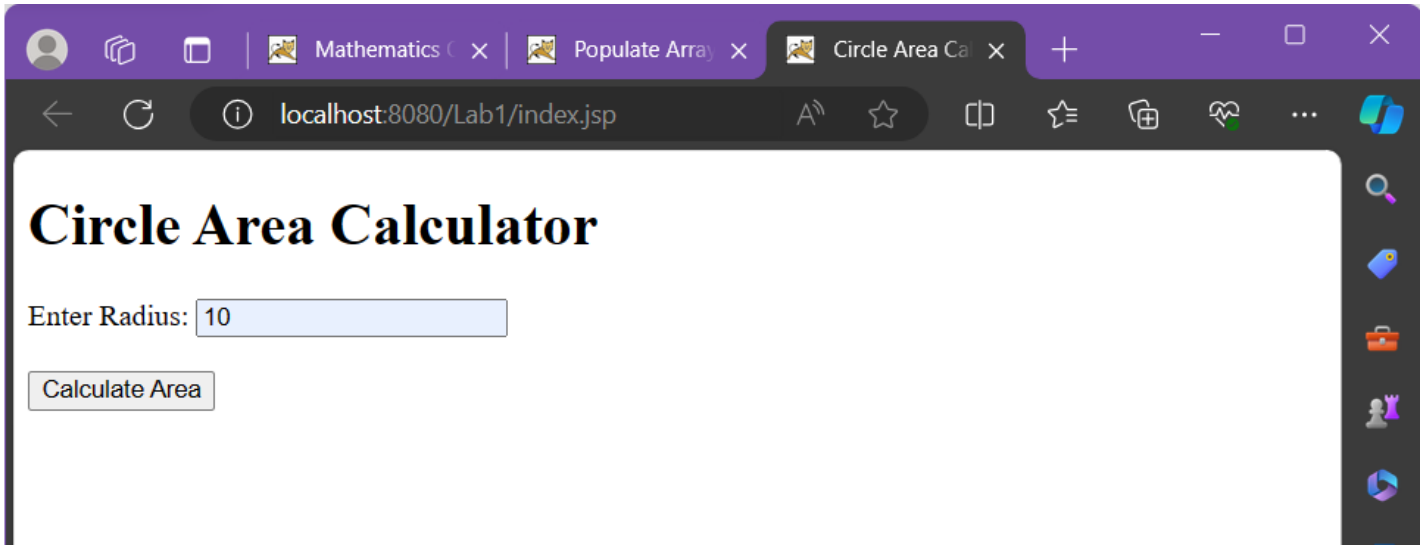

Lab1-1.0-SNAPSHOT - Apache NetBeans IDE 21
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Projects Files Services
META-INF
WEB-INF
AttributeIsSet.jsp
GetAttribute.jsp
MathematicsOperation.jsp
Sales.jsp
Welcome.jsp
index.html
index.jsp
populateArray.jsp
useJavaObject.jsp
RESTful Web Services
Source Packages
<default package>
CalculateArea.java
com.mycompany.lab1
com.mycompany.lab1.resources
Test Packages
Other Sources
Dependencies
Java Dependencies
Project Files
processRequest - Navigator
Members
CalculateArea :: HttpServlet
CalculateArea()
doGet(HttpServletRequest request, HttpServletResponse
doPost(HttpServletRequest request, HttpServletResponse
getServletInfo() : String ↑ GenericServlet
processRequest(HttpServletRequest request, HttpServletR
useJavaObject.jsp
AttributeIsSet.jsp
GetAttribute.jsp
MathematicsOperation.jsp
populateArray.jsp
CalculateArea.java
index.jsp
Sales.jsp
Source History
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
response.setContentType("text/html;charset=UTF-8");
try (PrintWriter out = response.getWriter()) {
out.println("<!DOCTYPE html>");
out.println("<html>");
out.println("<head>");
out.println("<title>Servlet CalculateArea</title>");
out.println("</head>");
out.println("<body>");
out.println("<h1>Servlet CalculateArea at " + request.getContextPath() + "</h1>");
out.println("</body>");
out.println("</html>");
}
}
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
processRequest(request, response);
}
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
response.setContentType("text/html");
PrintWriter out = response.getWriter();
// Retrieve the radius parameter from the request
String radiusStr = request.getParameter("radius");
// Parse the radius string to a double
CalculateArea processRequest
Output
Apache Tomcat or TomEE Log
Apache Tomcat or TomEE
Run (index)
28:92
INS Windows (CRL...


Lab1-1.0-SNAPSHOT - Apache NetBeans IDE 21
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Projects Files Services
META-INF
WEB-INF
AttributeIsSet.jsp
GetAttribute.jsp
MathematicsOperation.jsp
Sales.jsp
Welcome.jsp
index.html
index.jsp
populateArray.jsp
useJavaObject.jsp
RESTful Web Services
Source Packages
<default package>
CalculateArea.java
com.mycompany.lab1
com.mycompany.lab1.resources
Test Packages
Other Sources
Dependencies
Java Dependencies
Project Files
CalculateArea.java - Navigator
Members
CalculateArea :: HttpServlet
CalculateArea()
doGet(HttpServletRequest request, HttpServletResponse
doPost(HttpServletRequest request, HttpServletResponse
getServletInfo() : String ↑ GenericServlet
processRequest(HttpServletRequest request, HttpServletR
useJavaObject.jsp
AttributeIsSet.jsp
GetAttribute.jsp
MathematicsOperation.jsp
populateArray.jsp
CalculateArea.java
index.jsp
Sales.jsp
Source History
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
processRequest(request, response);
}
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
response.setContentType("text/html");
PrintWriter out = response.getWriter();
// Retrieve the radius parameter from the request
String radiusStr = request.getParameter("radius");
// Parse the radius string to a double
double radius = Double.parseDouble(radiusStr);
// Calculate the area of the circle
double area = Math.PI * radius * radius;
// Output the result
out.println("<html><body>");
out.println("<h2>Area of the circle: " + area + "</h2>");
out.println("</body></html>");
}
@Override
public String getServletInfo() {
return "Short description";
}
}
CalculateArea
Output
Apache Tomcat or TomEE Log
Apache Tomcat or TomEE
Run (index)
73:2
INS Windows (CRL...

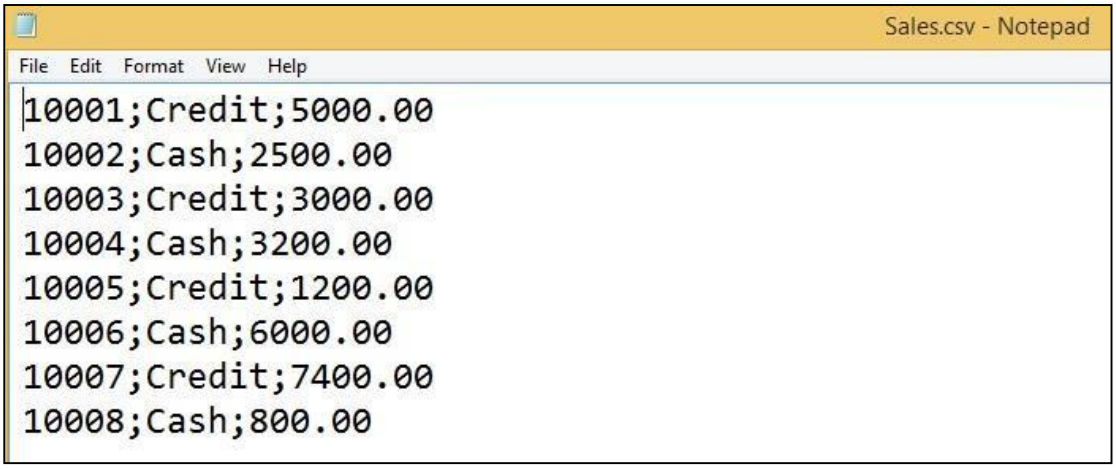
Index.jsp


Lab1-1.0-SNAPSHOT - Apache NetBeans IDE 21
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Projects Files Services
META-INF
WEB-INF
AttributeIsSet.jsp
GetAttribute.jsp
MathematicsOperation.jsp
Sales.jsp
Welcome.jsp
index.html
index.jsp
populateArray.jsp
useJavaObject.jsp
RESTful Web Services
Source Packages
<default package>
CalculateArea.java
com.mycompany.lab1
com.mycompany.lab1.resources
Test Packages
Other Sources
Dependencies
Java Dependencies
Project Files
Navigator
html
head
meta
title
body
h1
form
input
br
br
input
Source History
<%--
Document : index
Created on : 29 Mar 2024, 10:58:07 pm
Author : rynaa
--%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Circle Area Calculator</title>
</head>
<body>
<h1>Circle Area Calculator</h1>
<form action="CalculateArea" method="post">
Enter Radius: <input type="text" name="radius" required pattern="[0-9]+">

<input type="submit" value="Calculate Area">
</form>
</body>
</html>
Output
Apache Tomcat or TomEE Log
Apache Tomcat or TomEE
Run (index)


Mathematics
Populate Array
Circle Area Ca
localhost:8080/Lab1/index.jsp
Circle Area Calculator
Enter Radius: 10
Calculate Area

2. Prepare the following file in the Notepad and save it as *Sales.csv*.

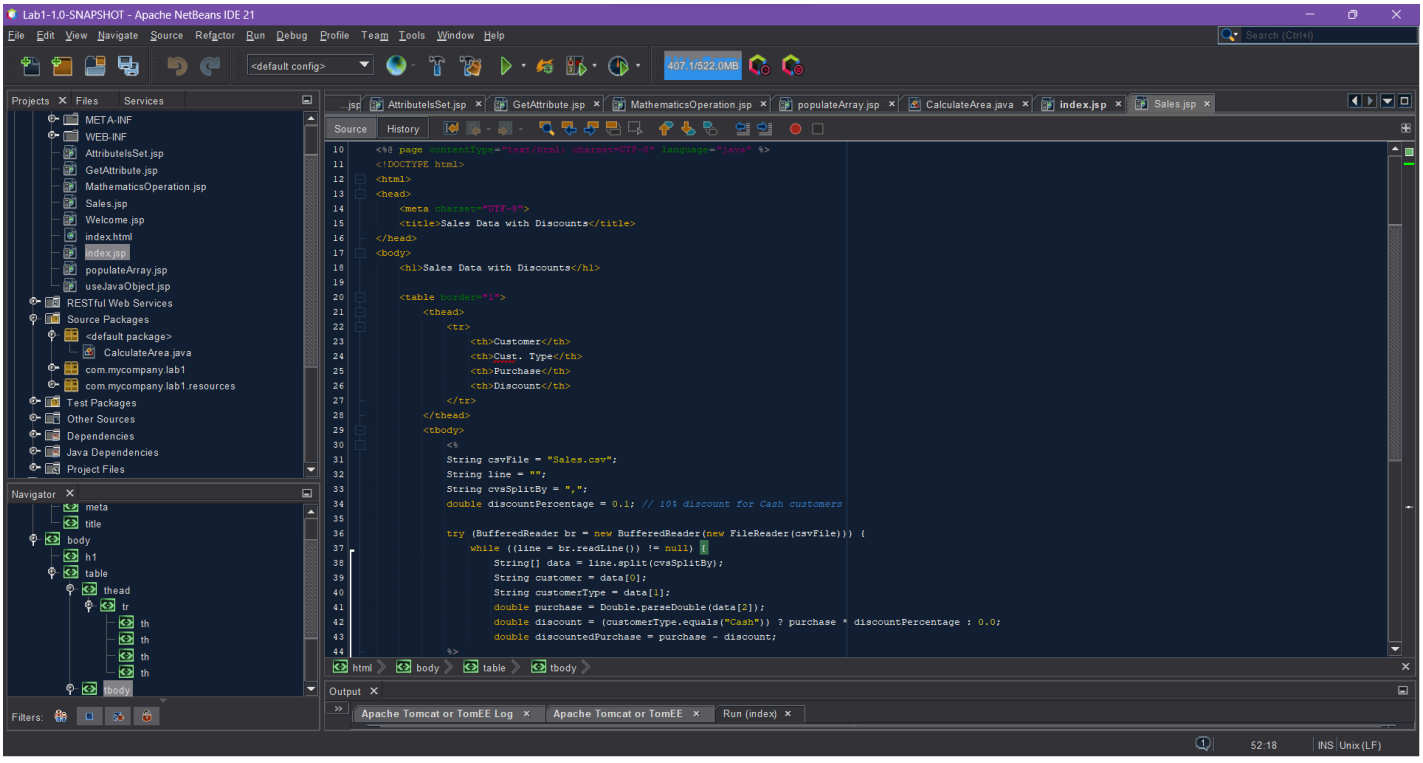


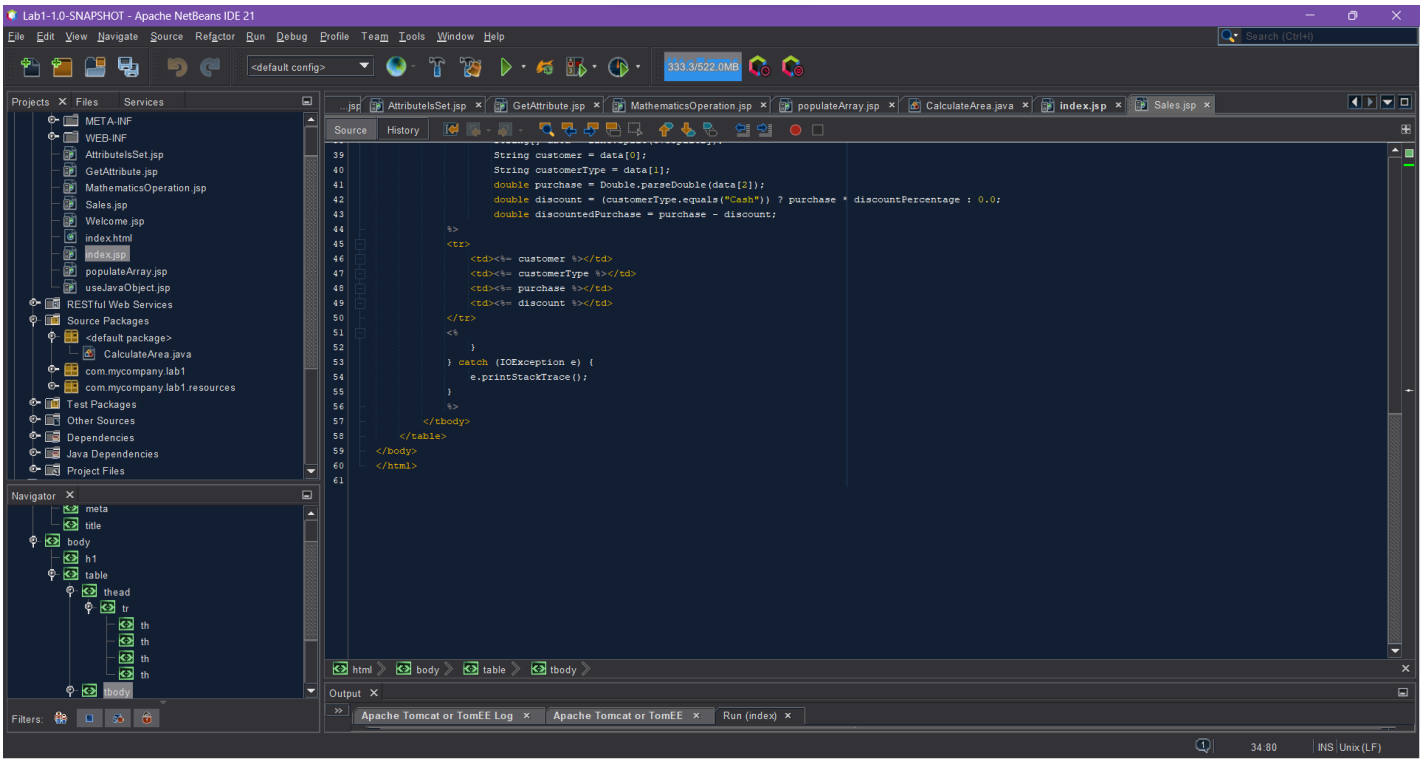
Create a simple JSP's page to read Sales.csv file. Upon reading Sales.csv's file, calculate the discount and populate HTML's table in the JSP's page.

Customer	Cust. Type	Purchase	Discount
10001	Credit	5000	0.00
10002	Cash	2500	250.00
10003	Credit	3000	0.00
10004	Cash	3200	320.00
10005	Credit	1200	0.00
10006	Cash	6000	600.00
10007	Credit	7400	0.00
10008	Cash	800	80.00

Cash customers are eligible to get a 10% discount.

Source code




Lab1-1.0-SNAPSHOT - Apache NetBeans IDE 21
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Projects Files Services
META-INF
WEB-INF
AttributeIsSet.jsp
GetAttribute.jsp
MathematicsOperation.jsp
Sales.jsp
Welcome.jsp
index.html
index.jsp
populateArray.jsp
useJavaObject.jsp
RESTful Web Services
Source Packages
<default package>
CalculateArea.java
com.mycompany.lab1
com.mycompany.lab1.resources
Test Packages
Other Sources
Dependencies
Java Dependencies
Project Files
Navigator
meta
title
body
h1
table
thead
tr
th
th
th
th
tbody
Filters:
AttributeIsSet.jsp
GetAttribute.jsp
MathematicsOperation.jsp
populateArray.jsp
CalculateArea.java
index.jsp
Sales.jsp
Source History
String customer = data[0];
String customerType = data[1];
double purchase = Double.parseDouble(data[2]);
double discount = (customerType.equals("Cash")) ? purchase * discountPercentage : 0.0;
double discountedPurchase = purchase - discount;
%>
<tr>
<td><%= customer %></td>
<td><%= customerType %></td>
<td><%= purchase %></td>
<td><%= discount %></td>
</tr>
<%
}
} catch (IOException e) {
e.printStackTrace();
}
%>
</tbody>
</table>
</body>
</html>
html body table tbody
Output
Apache Tomcat or TomEE Log
Apache Tomcat or TomEE
Run (index)
34:80
INS Unix (LF)