Technical Direction Document (TDD)

Introduction to ReactJs

React is a JavaScript library for building user interfaces.It is used to build single-page applications.It allows us to create reusable UI components.

So in this session, we will start building a web application using ReactJs.

Before starting with React.JS, you should have intermediate experience in:

- HTML
- CSS
- JavaScript

What is HTML?

- HTML stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

  Example:

  ```
  <!DOCTYPE html>
  <html>
  <head>
  <title>Page Title</title>
  </head>
  <body>

  <h1>My First Heading</h1>
  <p>My first paragraph.</p>

  </body>
  </html>
  ```

What is CSS:Cascading Style Sheets?

CSS is a language of style rules that we use to apply styling to our HTML content, for example setting background colors and fonts, and laying out our content in multiple columns.

What is JavaScript?

JavaScript is a scripting or programming language that allows you to implement complex features on web pages — every time a web page does more than just sit there and display static information for you to look at — displaying timely content updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, etc.

Setting up a React Environment

To begin creating UI with ReactJs, you'll need to install following:

Vscode
Node.js

After installing VScode,Create a new folder with your project name and open vscode with with that respective folder

If you have npx and Node.js installed, you can create a React application by using **create-react-app.**

If you've previously installed **create-react-app** globally, it is recommended that you uninstall the package to ensure npx always uses the latest version of **create-react-app**.

To uninstall, run this command: `npm uninstall -g create-react-app.`

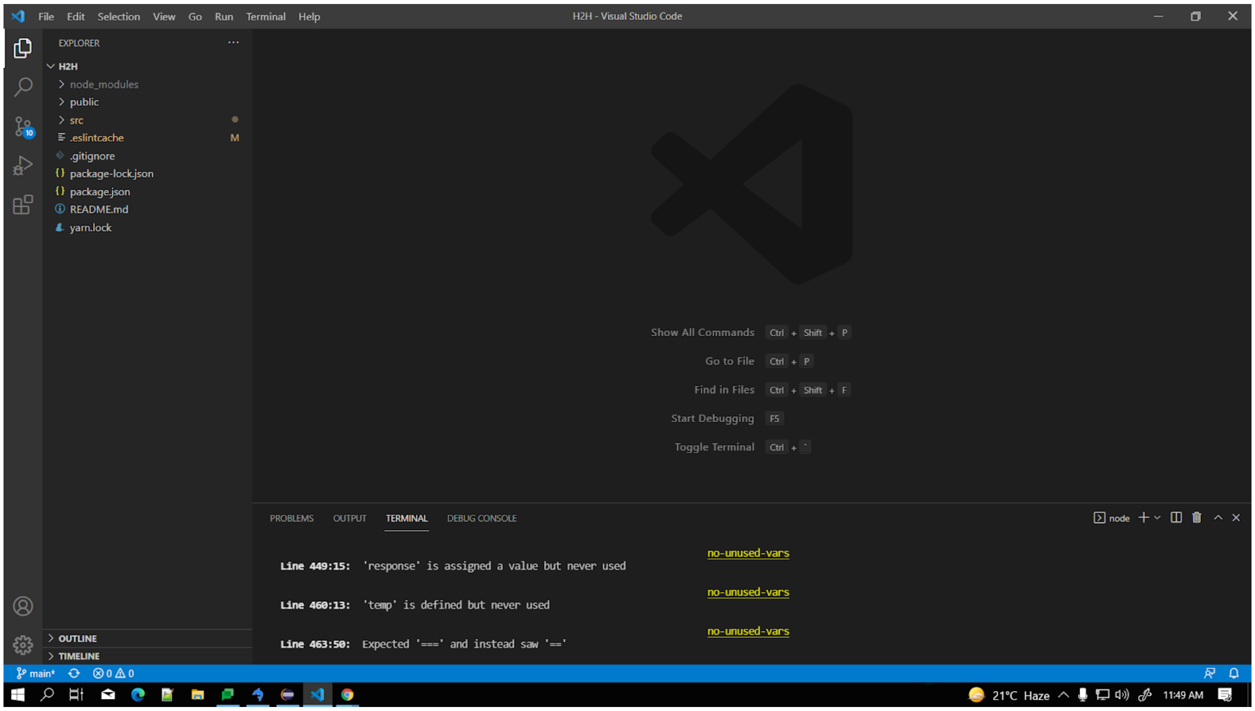
Run this command to create a React application named **my-react-app**:

```
npx create-react-app my-react-app
```

The **create-react-app** will set up everything you need to run a React application.

Run the React Application

VScode:Here Project name is H2H



```
                                                         no-unused-vars
Line 449:15:  'response' is assigned a value but never used

                                                         no-unused-vars
Line 460:13:  'temp' is defined but never used

                                                         no-unused-vars
Line 463:50:  Expected '===' and instead saw '=='
```

Now you are ready to run your first real React application!

Run this command to move to the **my-react-app** directory:

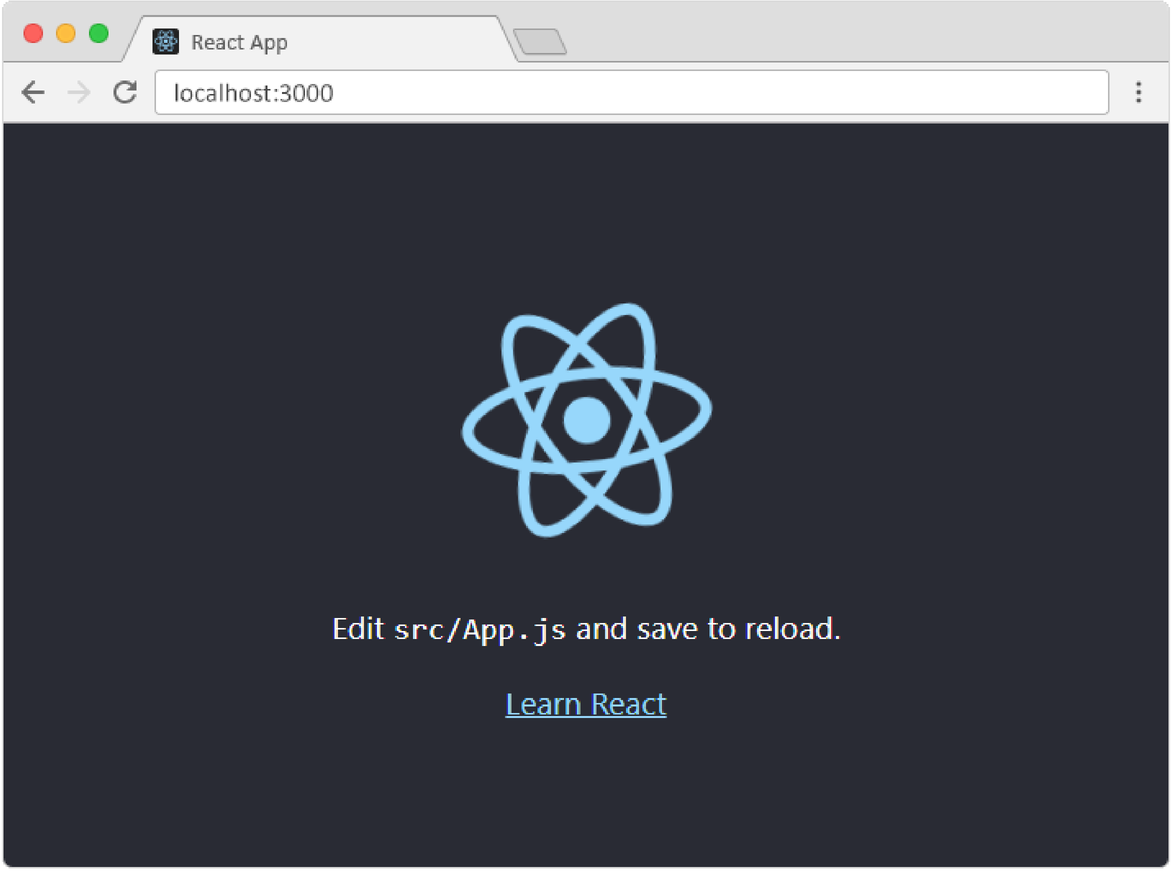
```
cd my-react-app
```

Run this command to run the React application **my-react-app**:

```
npm start
```

A new browser window will pop up with your newly created React App! If not, open your browser and type **localhost:3000** in the address bar.

The result:



Edit `src/App.js` and save to reload.

Learn React

Getting started with your web application: -

Overview: - We are developing a web application which will display a list of invoices. The list should include the invoice data from csv. This web application should allow the users to add, delete, search, update,etc the list of invoices.

Convert CSV to SQL:

Link : Converter

Backend(Java):

Java is a popular programming language, created in 1995.

It is owned by Oracle, and more than 3 billion devices run Java.

It is used for:

- Mobile applications (specially Android apps)
- Desktop applications
- Web applications
- Web servers and application servers
- Games
- Database connection
- And much, much more!

Why Use Java?

- Java works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc.)
- It is one of the most popular programming language in the world
- It is easy to learn and simple to use
- It is open-source and free
- It is secure, fast and powerful
- It has a huge community support (tens of millions of developers)
- Java is an object oriented language which gives a clear structure to programs and allows code to be reused, lowering development costs
- As Java is close to C++ and C#, it makes it easy for programmers to switch to Java or vice versa

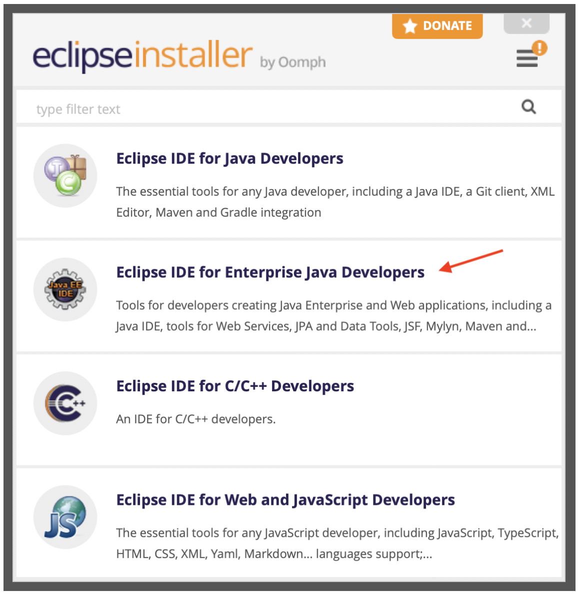
Setting up Java Environment:

Install java development kit (JDK AND JRE) before eclipse

Download Eclipse Installer from http://www.eclipse.org/downloads

Select Eclipse installer and download and install

Below windows pops up



Eclipse IDE for Java Developers

The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Maven and Gradle integration

Eclipse IDE for Enterprise Java Developers

Tools for developers creating Java Enterprise and Web applications, including a Java IDE, tools for Web Services, JPA and Data Tools, JSF, Mylyn, Maven and...

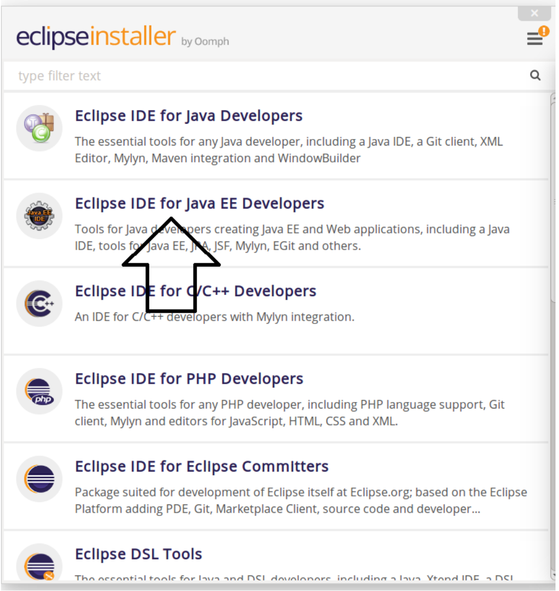
Eclipse IDE for C/C++ Developers

An IDE for C/C++ developers.

Eclipse IDE for Web and JavaScript Developers

The essential tools for any JavaScript developer, including JavaScript, TypeScript, HTML, CSS, XML, Yaml, Markdown... languages support;...

OR



Eclipse IDE for Java Developers

The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Mylyn, Maven integration and WindowBuilder

Eclipse IDE for Java EE Developers

Tools for Java developers creating Java EE and Web applications, including a Java IDE, tools for Java EE, JPA, JSF, Mylyn, EGit and others.

Eclipse IDE for C/C++ Developers

An IDE for C/C++ developers with Mylyn integration.

Eclipse IDE for PHP Developers

The essential tools for any PHP developer, including PHP language support, Git client, Mylyn and editors for JavaScript, HTML, CSS and XML.

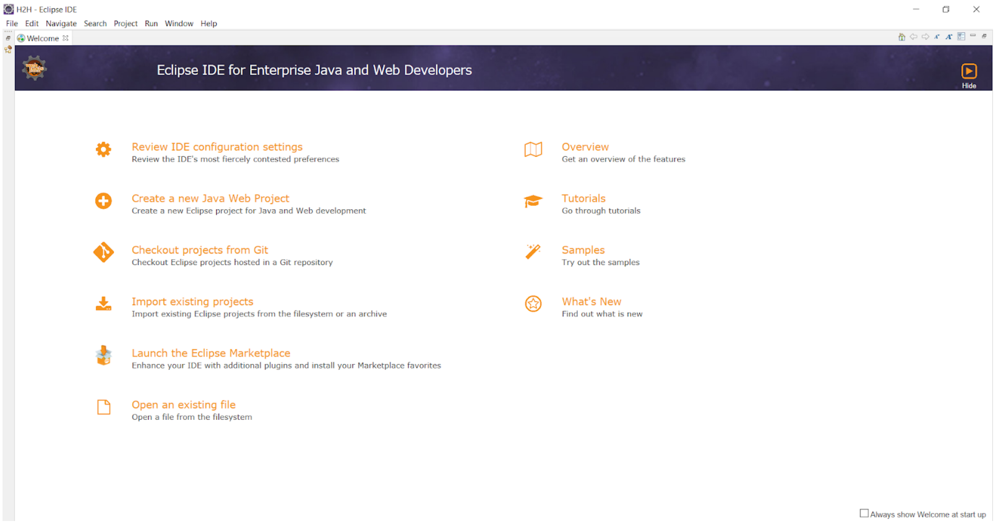
Eclipse IDE for Eclipse Committers

Package suited for development of Eclipse itself at Eclipse.org; based on the Eclipse Platform adding PDE, Git, Marketplace Client, source code and developer...

Eclipse DSL Tools

The essential tools for Java and DSL developers, including a Java, Xtend IDE, a DSL

After installing



Eclipse IDE for Enterprise Java and Web Developers

Review IDE configuration settings
Review the IDE's most fiercely contested preferences

Create a new Java Web Project
Create a new Eclipse project for Java and Web development

Checkout projects from Git
Checkout Eclipse projects hosted in a Git repository

Import existing projects
Import existing Eclipse projects from the filesystem or an archive

Launch the Eclipse Marketplace
Enhance your IDE with additional plugins and install your Marketplace favorites

Open an existing file
Open a file from the filesystem

Overview
Get an overview of the features

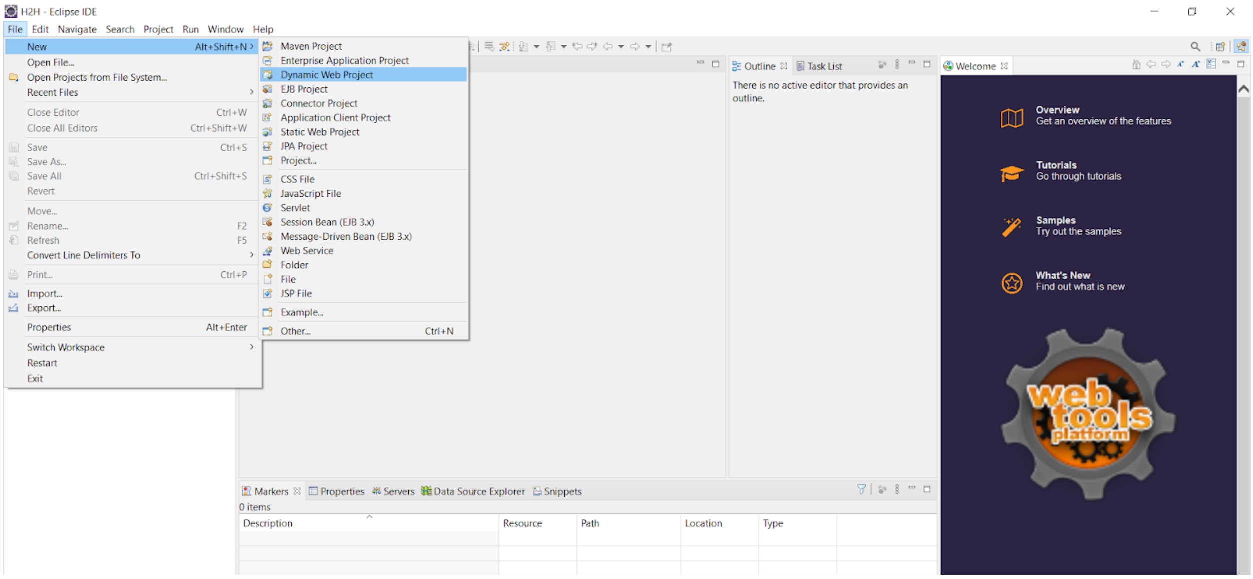
Tutorials
Go through tutorials

Samples
Try out the samples

What's New
Find out what is new

☐ Always show Welcome at start up

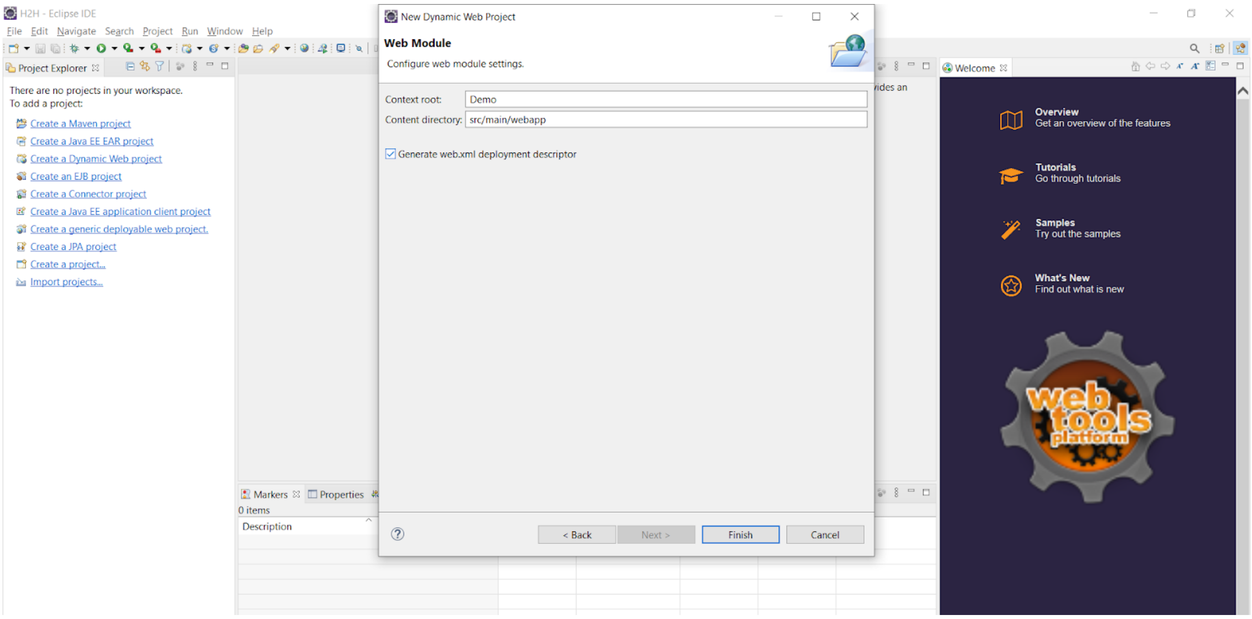
Create a Dynamic Web Project:



There is no active editor that provides an outline.

0 items

Check Generate web.xml



There are no projects in your workspace.
To add a project:

Web Module

Configure web module settings.

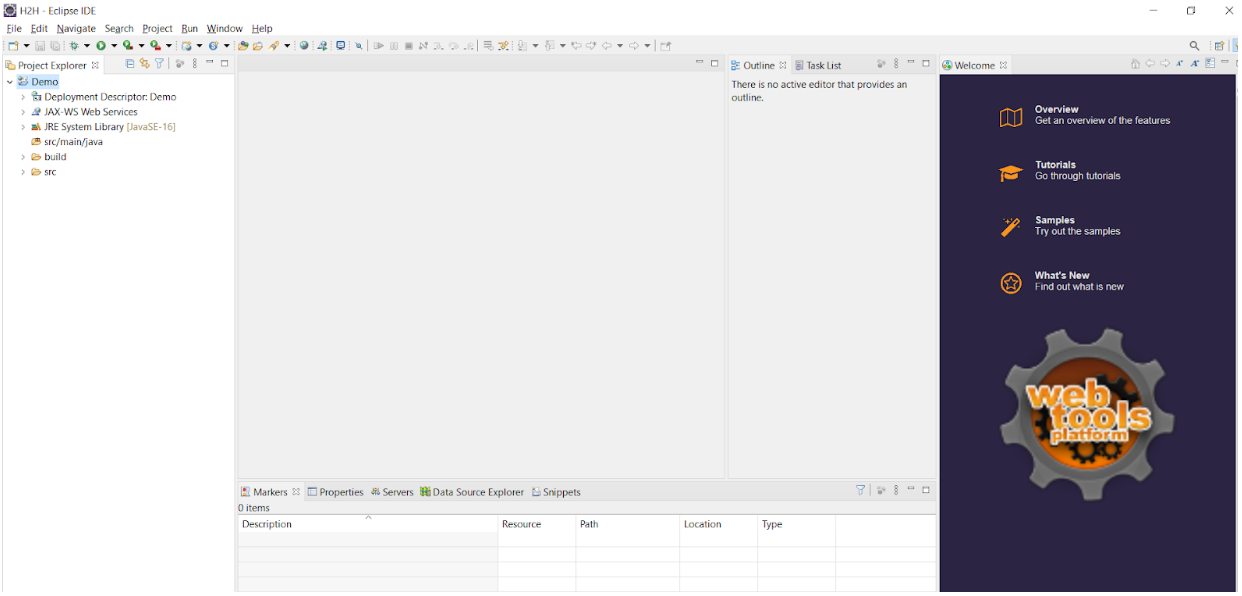
Context root: Demo

Content directory: src/main/webapp

☑ Generate web.xml deployment descriptor

0 items

Final Look:

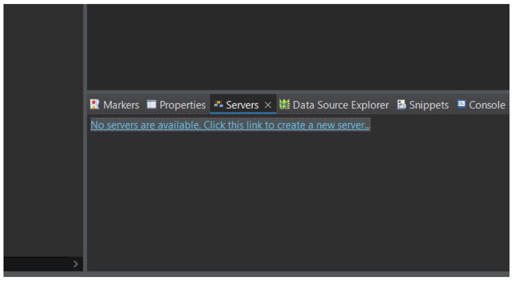


There is no active editor that provides an outline.

Server:

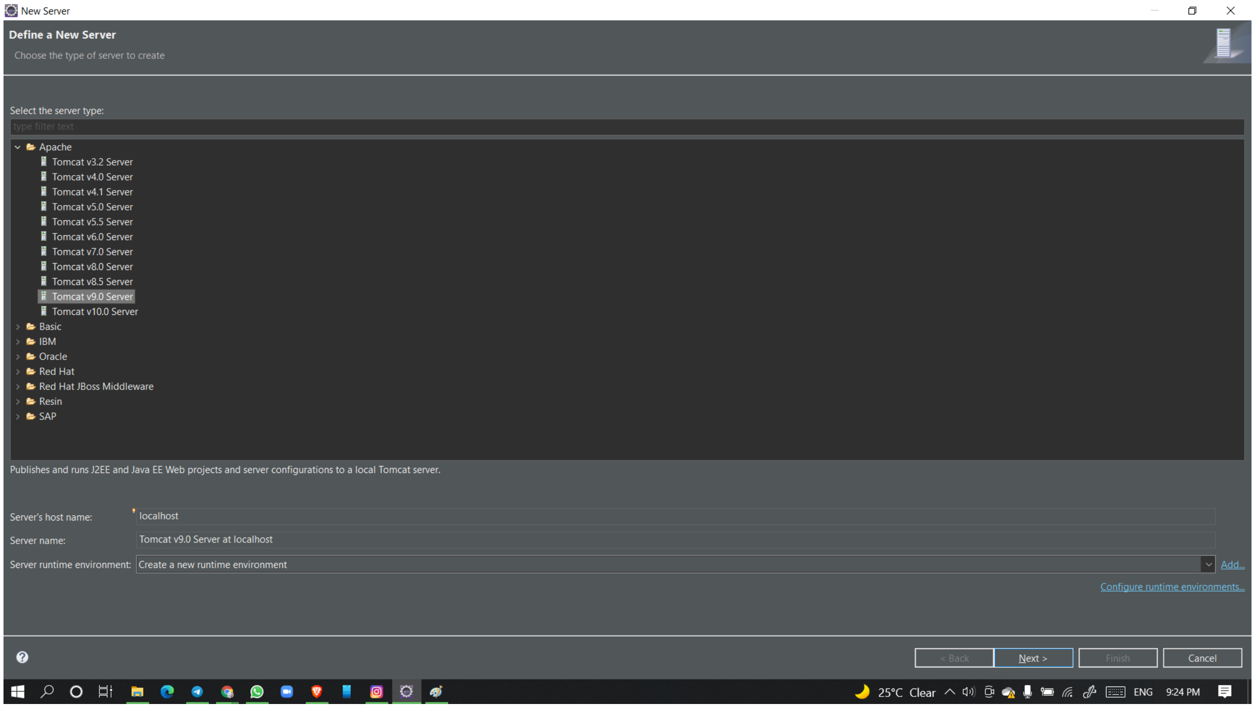
1. If you do not have Apache Tomcat on your machine, you will first need to download and unzip Apache Tomcat Start the Eclipse WTP workbench.
2. Open **Window -> Preferences -> Server -> Installed Runtimes** to create a Tomcat installed runtime.
3. Click on **Add...** to open the **New Server Runtime** dialog, then select your runtime under **Apache** (any version above 8 is fine).

Step:1



No servers are available. Click this link to create a new server...

Step 2: Select version(prefer 9)



New Server

Define a New Server

Choose the type of server to create

Select the server type:

type filter text

- Apache
  - Tomcat v3.2 Server
  - Tomcat v4.0 Server
  - Tomcat v4.1 Server
  - Tomcat v5.0 Server
  - Tomcat v5.5 Server
  - Tomcat v6.0 Server
  - Tomcat v7.0 Server
  - Tomcat v8.0 Server
  - Tomcat v8.5 Server
  - Tomcat v9.0 Server
  - Tomcat v10.0 Server
- Basic
- IBM
- Oracle
- Red Hat
- Red Hat JBoss Middleware
- Resin
- SAP

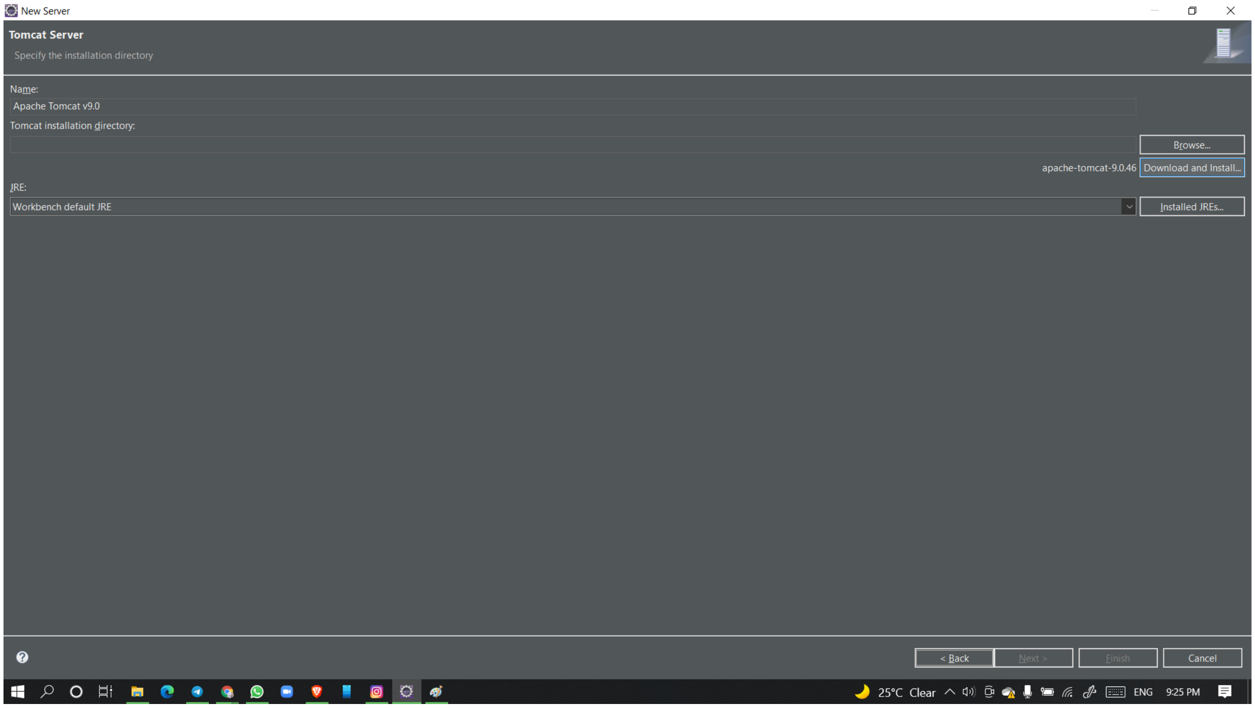
Publishes and runs J2EE and Java EE Web projects and server configurations to a local Tomcat server.

Server's host name: localhost

Server name: Tomcat v9.0 Server at localhost

Server runtime environment: Create a new runtime environment

Step 3:Click Download and install and select path for file to download and wait some time to download



New Server

Tomcat Server

Specify the installation directory

Name:

Apache Tomcat v9.0

Tomcat installation directory:

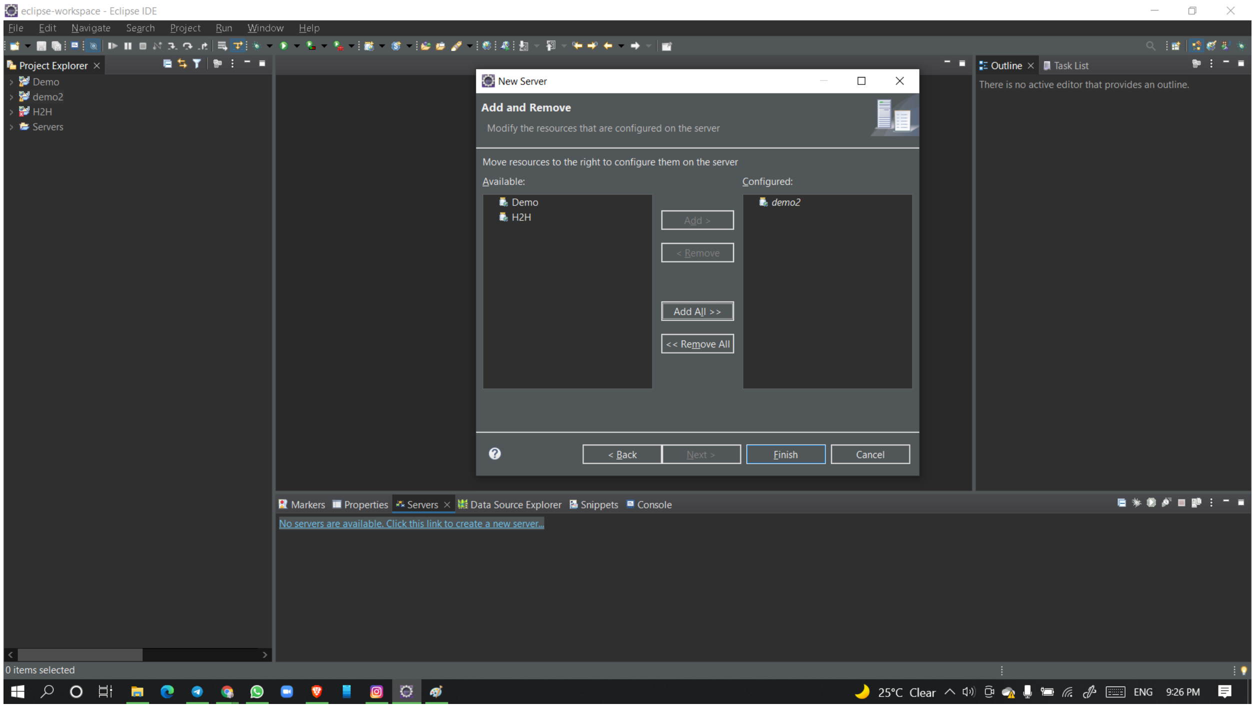
apache-tomcat-9.0.46

JRE:

Workbench default JRE

Step 4:Configure your Project

(unable to configure then click finish without adding and right click on your project name ->properties->project faucets->downgrade the java version to 1.8 ->apply and close and now configure your project)



Add and Remove

Modify the resources that are configured on the server

Move resources to the right to configure them on the server

Available:
- Demo
- H2H

Configured:
- demo2

There is no active editor that provides an outline.

No servers are available. Click this link to create a new server...

0 items selected

Follow the link for more info:

Server Setup