

**JPassion.com**

**JASMINE CONSEIL** 

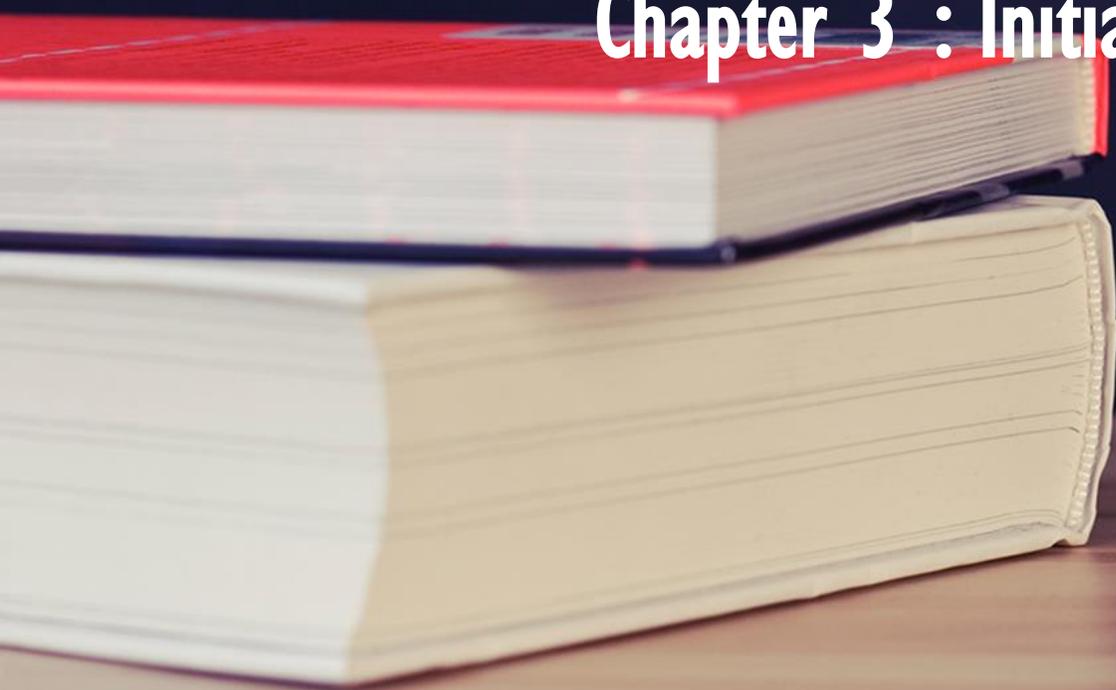
# JAVA EE 8 Application Development in Practice

## Chapter 3 : Initializing your JEE Development Environment

Karim DJAAFAR

JEE Architect Evangelist

Last Version date :  
December 2018



# AGENDA

---

- Tools and Framework presentation:
  - Eclipse for Java EE Developers
  - Maven Archetype
  - Angular
  - Payara 5 Java EE server
  - Vagrant and Docker



# OBJECTIVES OF THIS COURSE

---

- See the tool stack which will be used in this course
- Understand the role of each tool and his usage in the development environment
- Understand the design and the Architecture overview of the application



# Eclipse for Java EE

# Eclipse IDE

- A good IDE is essential for better productivity while coding
- Eclipse is one such IDE, which has great editor features and many integration points with JEE technologies
- The primary purpose of this section is to show you how to develop JEE applications using Eclipse
- So the following is a quick introduction to Eclipse, if you are not already familiar with it

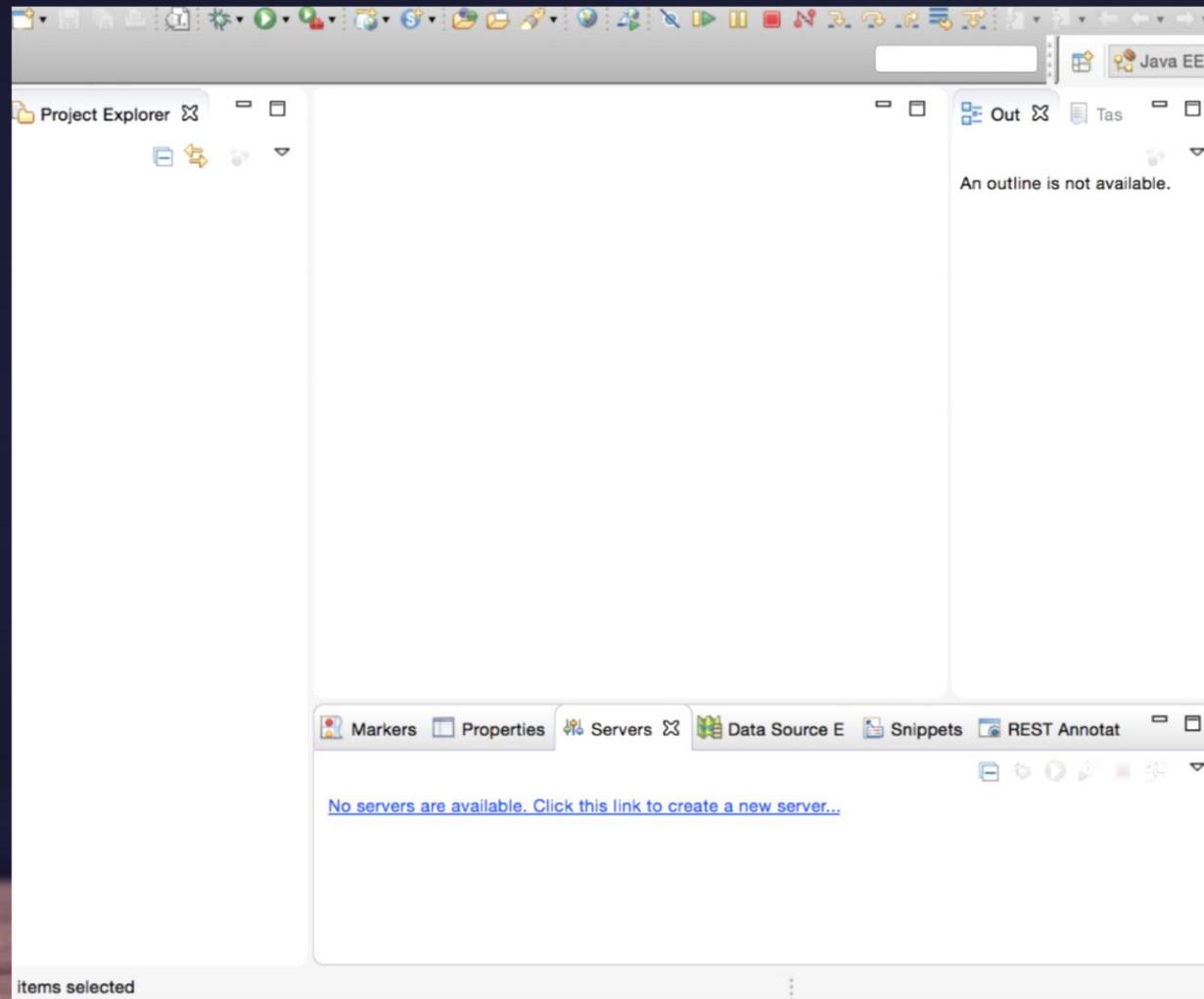


# What is Eclipse

- Eclipse is an open source IDE for developing applications in many different programming languages
- It is quite popular for developing many different types of Java applications
- Its architecture is pluggable—there is a core IDE component and many different plugins can be added to it
- Along with editor support, Eclipse has plugins to interact with many of the external systems used during development :
  - examples include source control systems such as SVN and Git, build tools such as Maven, managing servers such as Payara and GlassFish, database explorers, memory and CPU profilers, Vagrant support ...
- All those external systems are displayed as views



# Default Eclipse view for JEE Development



# Workspace

- The Eclipse workspace is a collection of projects, settings, and preferences
- It is a folder where Eclipse stores this information
- You must create a workspace to start using Eclipse
- You can create multiple workspaces, but only one can be opened at a time by one running instance of Eclipse
- However, you can launch multiple instances of Eclipse with different workspaces



# Plugin

- Eclipse has pluggable architecture
- Many of the features of Eclipse are implemented as plugins, for example, editor plugins for Java and many other languages, plugins for SVN and Git, and many more
- The default installation of Eclipse comes with many built-in plugins and you can add more plugins for the features you want later



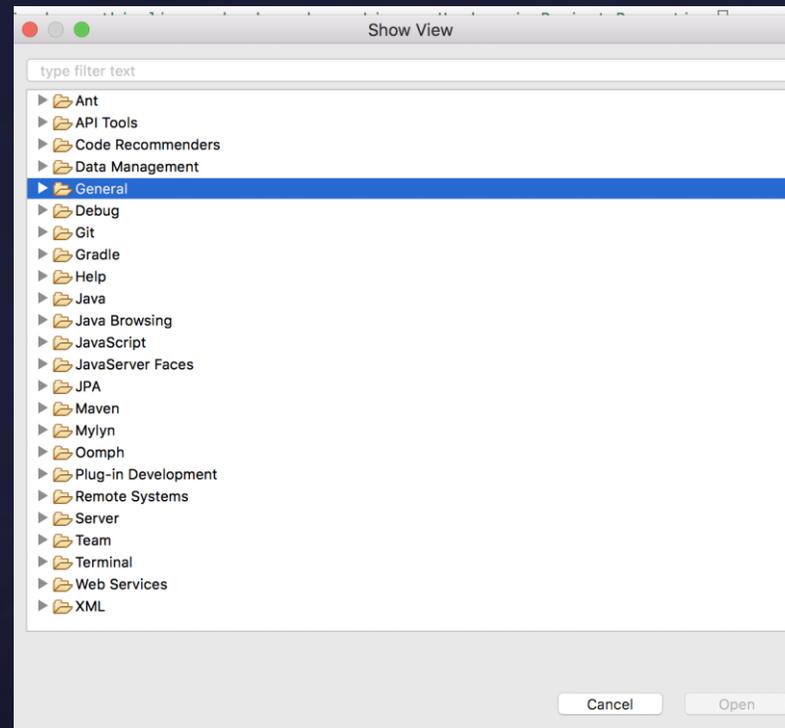
# Editors and views

- The Eclipse workspace is a collection of projects, settings, and preferences
- It is a folder where Eclipse stores this information
- You must create a workspace to start using Eclipse
- You can create multiple workspaces, but only one can be opened at a time by one running instance of Eclipse
- However, you can launch multiple instances of Eclipse with different workspaces



# Show All Eclipse Views

- To see all views in a given Eclipse installation, open the Window | Show View | Other menu:



# Build-in Tools and supported frameworks



Built-in tools

Build tools (Maven, Gradle)  
Version Control  
Code coverage



JVM Languages

Java  
Scala  
Groovy



Frameworks and JEE Servers

Spring  
Java EE  
Apache TomEE  
Payara



Web Development

JavaScript  
HTML/CSS  
AngularJS  
Node.js  
React

# **Build Tool : Describing our Material**



# Maven

- Build Automation tool:
  - Code, Build, Test and Report
- Open Source (<http://maven.apache.org/> )
- **Convention over Configuration** : Describes how software will be build
- Describes software dependencies using a POM XML descriptor

# Archetype support

- Standardize directory project structure
- it's a template project template from which other projects are created
- The main benefit of using archetypes is to standardize project development and to enable developers to easily follow best practices while bootstrapping their projects faste
- We will use JEE 8 Archetype to enhance our development productivity



# Selecting a JEE Server



# Selecting a JEE Server



# What is Payara

- Payara is a lightweight Java EE 8 server
- Payara Server is a drop in replacement for GlassFish Server with guaranteed quarterly releases
- Very simple installation
- Ability to create a self-contained JAR that can run the server and application at the same time
- Support Java EE 8 and Microservices
- We will use it in our use case application



**Choosing a Tool which create a virtual and  
isolated and repeatable environment :  
Vagrant**



# What is Vagrant

- With a virtualized environment, the development environments can also mimic the production environment
- No more needing to worry if something will work when it get deployed, if it is being developed on a machine with the exact same software configuration
- Even if you deploy on a Linux machine but develop on Windows, your virtualized environment can be Linux, running the same distribution as your production environment
- **Vagrant** (<http://www.vagrantup.com/>) is a powerful development tool, which lets you manage and support the virtualization of your development environment
- Instead of running all your projects locally on your own computer, having to juggle the different requirements and dependencies of each project, Vagrant lets you run each project in its own dedicated virtual environment, a box
- Vagrant can be used in combination with **Docker** container to construct build JEE isolated and repeatable development environments



# Choosing a Front end Framework: AngularJS



# What is AngularJS

- AngularJS is a efficient framework that can create Rich Internet Applications (RIA) used to build Single Page Applications (SPA) that offer an app-like experience as opposed to traditional web pages
- SPAs are web applications that loads a single page at first, and further UI updates are handled by dynamic DOM/page updates rather than page reloads
- Applications written in AngularJS are cross-browser compliant, AngularJS automatically handles JavaScript code suitable for each browser
- AngularJS is open source, completely free, and used by thousands of developers around the world. It is licensed under the Apache license version 2.0
- We will choose this framework instead of the JSF Framework supported by Java EE for is powerful facilities and enhancing support of HTML and client side interactions



# **Lab 1: Installing and configuring our Development environment**



# In Summary



In this module we saw all the tools and frameworks to set up our Java EE environment:

- Payara as JEE Server
- Angular as a Front end framework
- Maven as a build tool
- Vagrant as a support for your virtualized development environment combined to Docker container to create efficient Java EE applications



Your friend to teach you  
Java  
With Passion!

[JPassion.com](http://JPassion.com)