



GUSTAVO CORONEL
DESARROLLA SOFTWARE

DESARROLLO WEB



Configuración de Servlet 3.1 Con Maven y Eclipse 2021-12

Eric Gustavo Coronel Castillo

www.desarrollasoftware.com

gcoronelc@gmail.com

I N S T R U C T O R



CONTENIDO

PRESENTACIÓN.....	3
OBTENER EL SOFTWARE	4
ESTRUCTURA DE CARPETAS.....	5
WORKSPACE DE TRABAJO	6
CONFIGURAR APACHE TOMCAT	7
CREACIÓN DE PROYECTO	9
ARCHIVO POM.XML	11
CONFIGURAR SERVLET 3.1.....	13
ARCHIVO WEB.XML	14
CREACIÓN DE UN SERVLET	15
CURSOS VIRTUALES.....	16
CUPONES.....	16
FUNDAMENTOS DE PROGRAMACIÓN CON JAVA	16
JAVA ORIENTADO A OBJETOS	17
PROGRAMACIÓN CON JAVA JDBC	18
PROGRAMACIÓN CON ORACLE PL/SQL.....	19



PRESENTACIÓN



En esta guía te explicare paso a paso el procedimiento para configurar un proyecto con Maven y Servlet 3.1 utilizando Eclipse IDE 2021-12-4 y Apache Tomcat 9.0.58.



OBTENER EL SOFTWARE



Lo primero que debes hacer es obtener el software.

Para obtener el IDE Eclipse portable debes hacer click en el siguiente enlace:

https://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/2021-12/R/eclipse-jee-2021-12-R-win32-x86_64.zip&mirror_id=1135

Para descargar Apache Tomcat debes hacer click en el siguiente enlace:

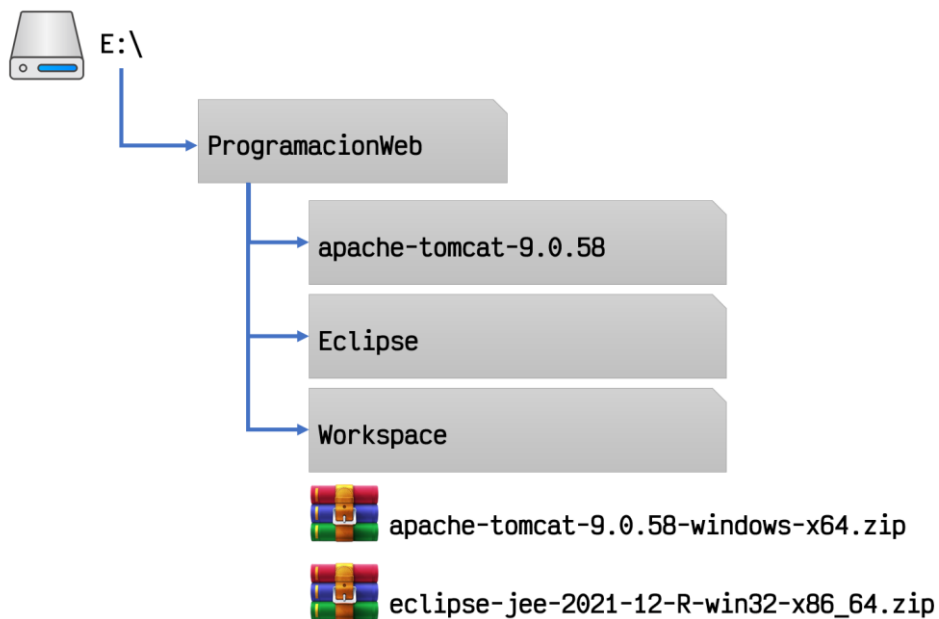
<https://tomcat.apache.org/download-90.cgi>

Los archivos que debes descargar son:

- eclipse-jee-2021-12-R-win32-x86_64.zip
- apache-tomcat-9.0.58-windows-x64.zip



ESTRUCTURA DE CARPETAS

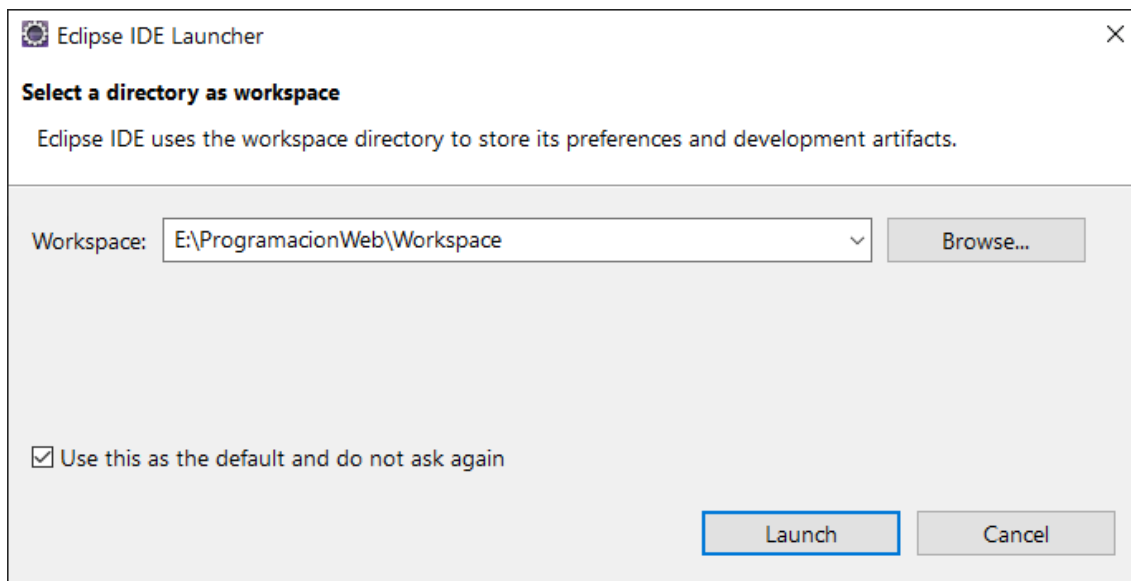


En la imagen anterior tienes la estructura de carpetas que debes tener después de desempaquetar los archivos descargados y crear la carpeta Workspace.

Puedes configurarlo en la unidad C o D, en la imagen se aprecia en la unidad E, pero eso depende de tu forma de trabajo.



WORKSPACE DE TRABAJO

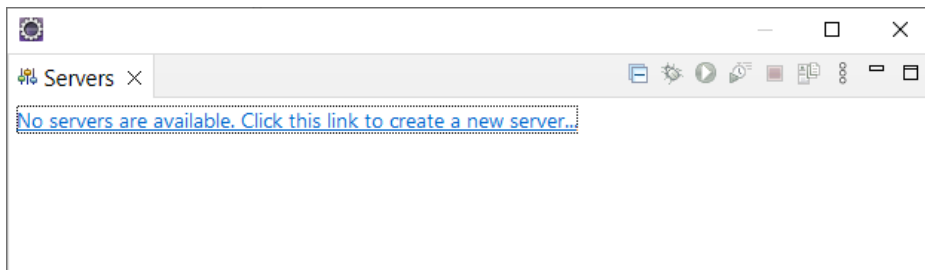


Cuando ejecutas por primera vez tu Eclipse debes configurar la carpeta de trabajo (workspace), tal como se ilustra en la imagen anterior.

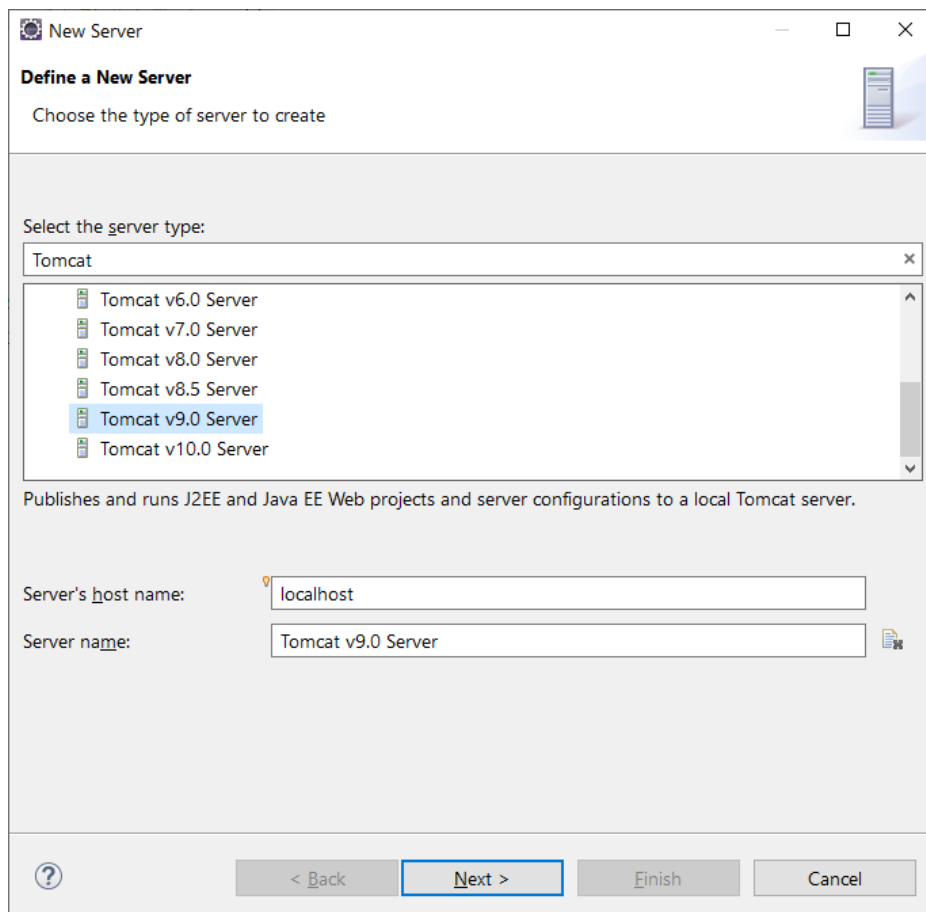


CONFIGURAR APACHE TOMCAT

1. En la ficha Servers debes hacer click en el enlace respectivo, tal como se muestra en la siguiente imagen.

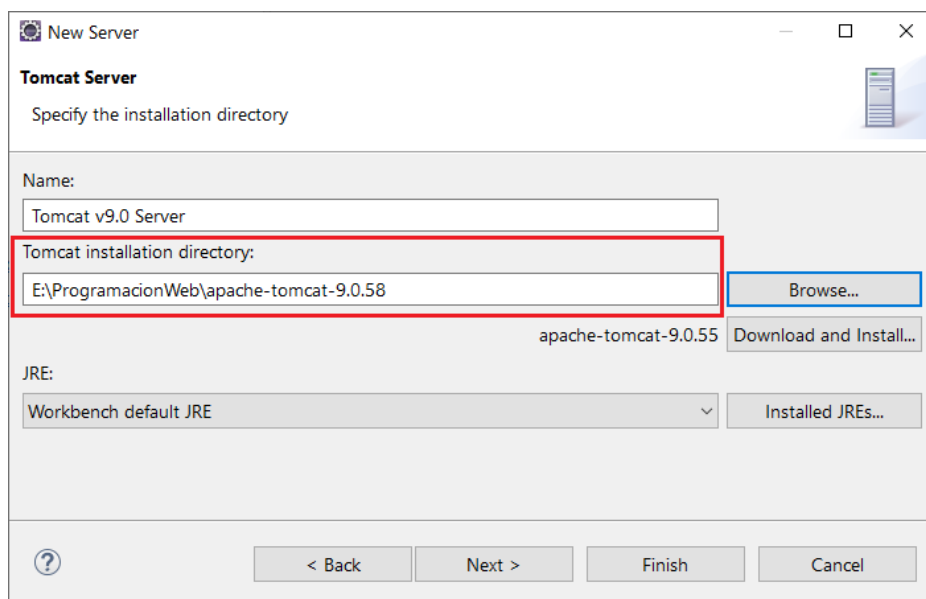


2. En la nueva ventana debes seleccionar **Tomcat v9.0 server**, tal como lo puedes apreciar en la siguiente imagen. Luego haces click en el botón **Next**.





3. En la siguiente venta debes seleccionar la ruta del directorio donde se encuentra Apache Tomcat. Tal como lo puedes apreciar en la siguiente imagen.



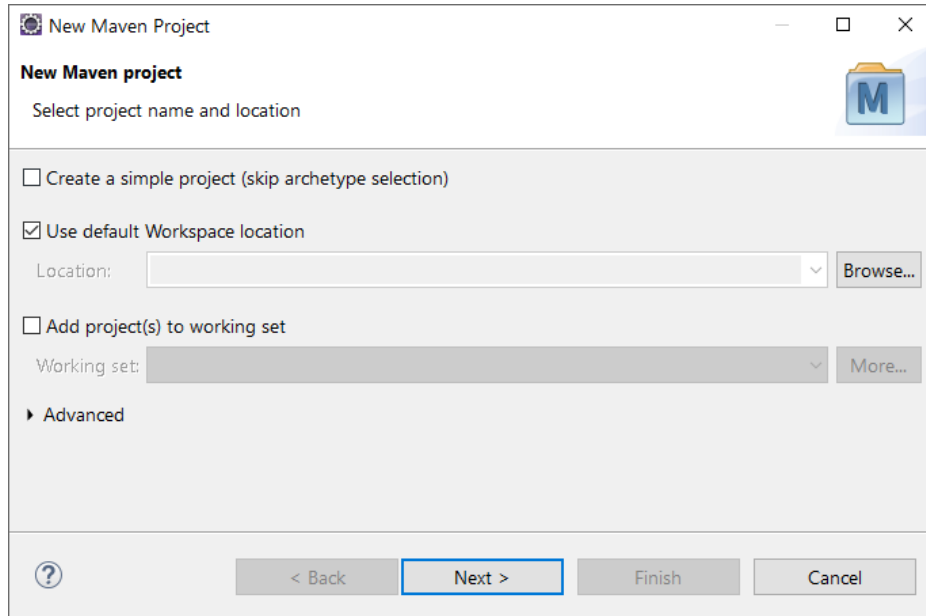
4. Finalmente, haces click en el botón **Finish**. Y ya tienes el Apache Tomcat instalado y configurado.



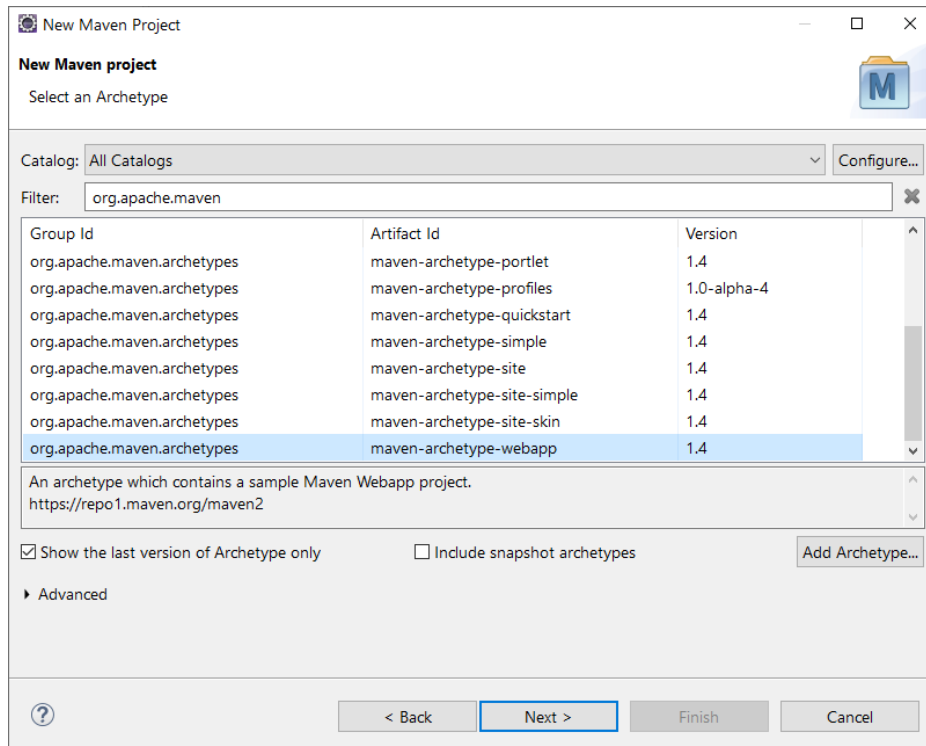


CREACIÓN DE PROYECTO

1. Inicia la creación de un nuevo proyecto con Maven: File -> New -> Maven Project



2. Luego haces click en el botón **Next**, en la nueva ventana debes seleccionar el archetype **maven-archetype-webapp**, tal como se aprecia en la siguiente imagen.





3. En la nueva ventana debes ingresar los siguientes datos:

Group Id: Aquí debes ingresar el dominio de la empresa al revés.

Artifact Id: Aquí debes ingresar el nombre de la aplicación.

En la siguiente imagen tienes un ejemplo ilustrativo.

New Maven Project

New Maven project

Specify Archetype parameters

Group Id: com.egcc

Artifact Id: demo

Version: 0.0.1-SNAPSHOT

Package: com.egcc.demo

Properties available from archetype:

Name	Value
------	-------

Advanced

< Back Next > Finish Cancel

4. Finalmente haces click en el botón **Finish**.



ARCHIVO POM.XML

A continuación, debes hacer algunas configuraciones en el archivo Maven **pom.xml**.

Lo primero que debes hacer es configurar la versión Java con que se va a compilar, en este caso debes ser 1.8, tal como puedes verlo a continuación:

```
<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <maven.compiler.source>1.8</maven.compiler.source>
  <maven.compiler.target>1.8</maven.compiler.target>
</properties>
```

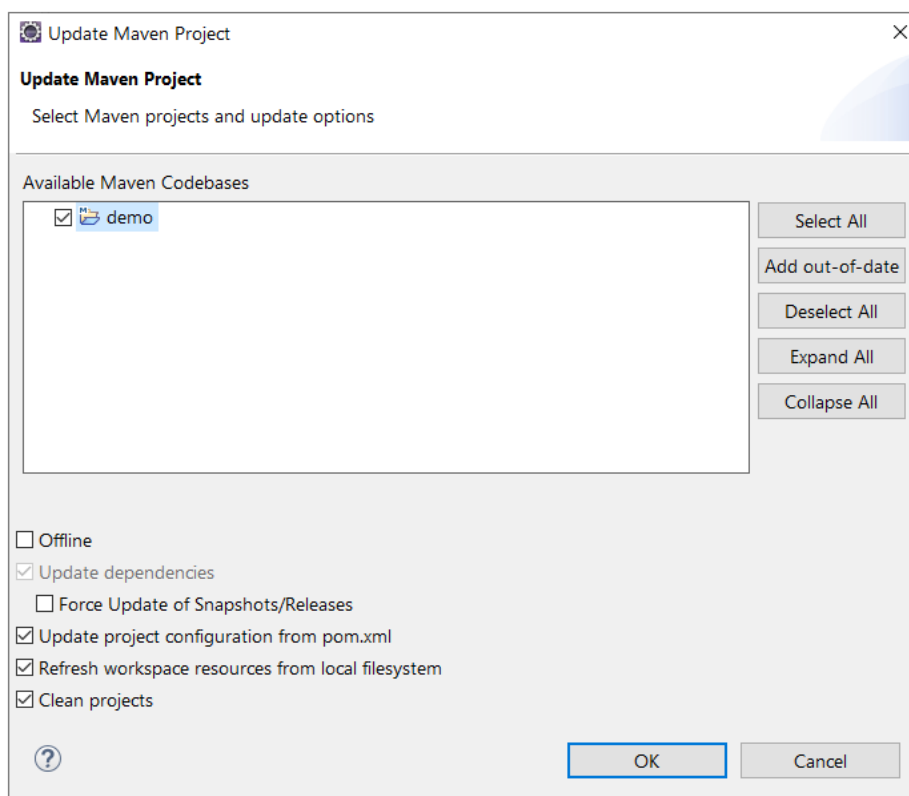
A continuación, debe agregar las siguientes dependencias al archivo pom.xml:

```
<dependency>
  <groupId>javax.servlet</groupId>
  <artifactId>javax.servlet-api</artifactId>
  <version>3.1.0</version>
</dependency>
<dependency>
  <groupId>javax.servlet</groupId>
  <artifactId>jstl</artifactId>
  <version>1.2</version>
</dependency>
<dependency>
  <groupId>javax.servlet.jsp</groupId>
  <artifactId>javax.servlet.jsp-api</artifactId>
  <version>2.3.3</version>
</dependency>
```

Luego, debes grabar el archivo



Finalmente, debes actualizar tu proyecto Maven, después de hacer click con el botón derecho sobre el nombre de tu proyecto, ejecuta la secuencia Maven -> Update Project

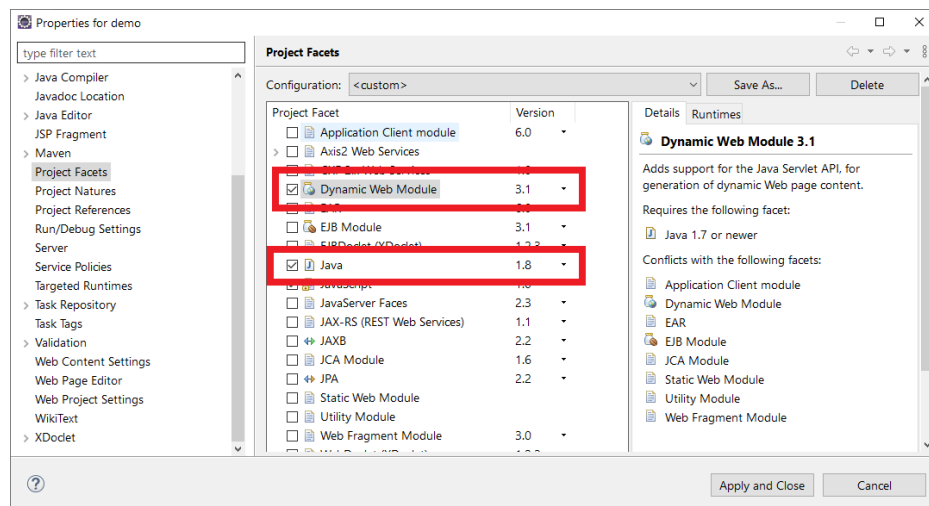


En esta nueva ventana haces click en el botón **Ok**.

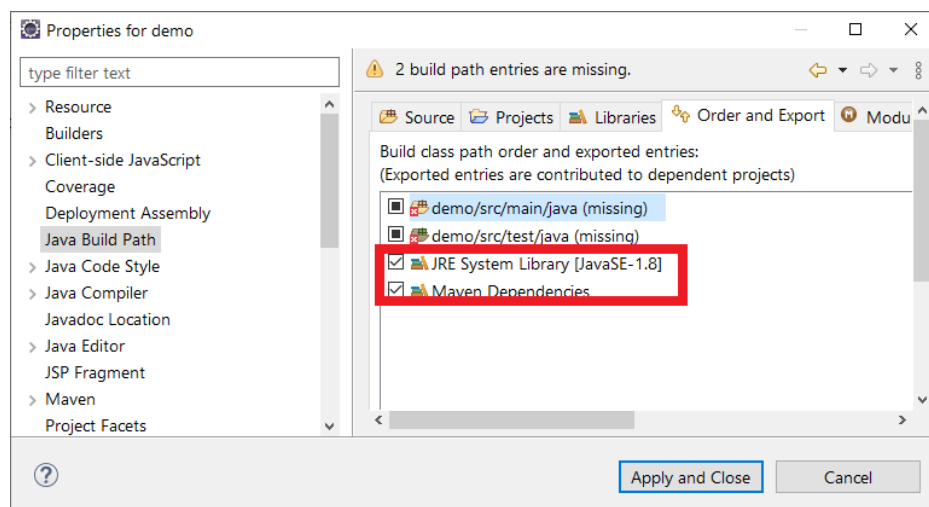


CONFIGURAR SERVLET 3.1

1. Lo primero que debes hacer es abrir la venta de preferencias utilizando la siguiente secuencia: Window -> Preferences.
2. En esta nueva ventana debes dirigirte a **Project Facets**, asegurarte que la versión de Java sea la 1.8 y que la versión del Servlets sea 3.1. Tal como lo puedes apreciar en la siguiente imagen:



3. Ahora debes dirigirte a **Java Build Path**, luego debes habilitar las 2 opciones que aparecen dentro del recuadro rojo en la siguiente imagen:



4. Finalmente, debes hacer click en **Apply and Close**.



ARCHIVO web.xml

El contenido de este archivo debe quedar de la siguiente manera:

```
<web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
    http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd"
  version="3.1">
  <display-name>Archetype Created Web Application</display-name>
</web-app>
```



CREACIÓN DE UN SERVLET

1. En primer lugar debes crear el paquete: **demo.servlets**
2. En segundo lugar, procede a crear el servlet: **DemoServlet.java**
3. A continuación, tienes el código del servlet.

```
package demo.servlets;

import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

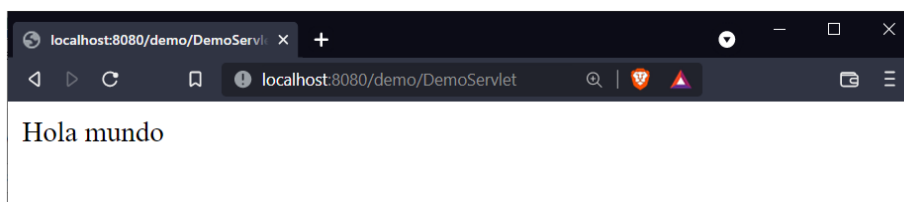
@WebServlet("/DemoServlet")
public class DemoServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    protected void service(HttpServletRequest request, HttpServletResponse
        response) throws ServletException, IOException {

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("Hola mundo");

    }
}
```

4. Ahora ya puedes ejecutar tu servlet.





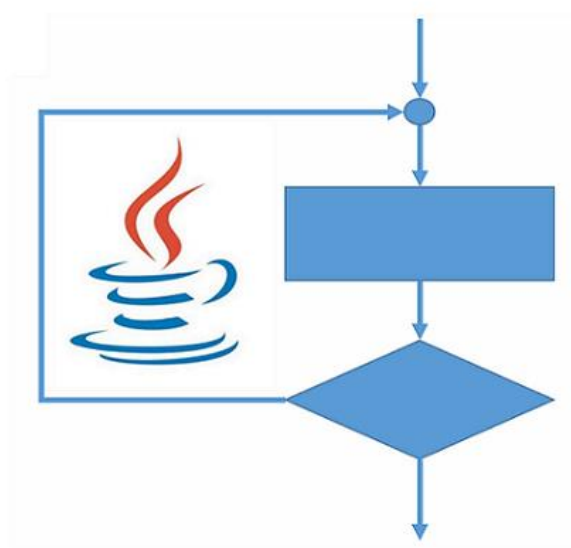
CURSOS VIRTUALES

Cupones

En esta URL se publican cupones de descuento:

<http://gcoronelc.github.io>

Fundamentos de Programación con Java



Tener bases sólidas de programación muchas veces no es fácil, creo que es principalmente por que en algún momento de tu aprendizaje mezclas la entrada de datos con el proceso de los mismos, o mezclas el proceso con la salida o reporte, esto te lleva a utilizar malas prácticas de programación que luego te serán muy difíciles de superar.

En este curso aprenderás las mejores prácticas de programación para que te inicies con éxito en este competitivo mundo del desarrollo de software.

URL del Curso: <https://n9.cl/gcoronelc-java-fund>

Avance del curso: <https://n9.cl/gcoronelc-fp-avance>

Cupones de descuento: <http://gcoronelc.github.io>



Java Orientado a Objetos



CURSO PROFESIONAL DE JAVA ORIENTADO A OBJETOS

Eric Gustavo Coronel Castillo

www.desarrollasoftware.com

I N S T R U C T O R

En este curso aprenderás a crear software aplicando la Orientación a Objetos, la programación en capas, el uso de patrones de software y Swing.

Cada tema está desarrollado con ejemplos que demuestran los conceptos teóricos y finalizan con un proyecto aplicativo.

URL del Curso: <https://bit.ly/2B3ixUW>

Avance del curso: <https://bit.ly/2RYGXIt>

Cupones de descuento: <http://gcoronelc.github.io>



Programación con Java JDBC



PROGRAMACIÓN DE BASE DE DATOS ORACLE CON JAVA JDBC

Eric Gustavo Coronel Castillo

www.desarrollasoftware.com

I N S T R U C T O R

En este curso aprenderás a programar bases de datos Oracle con JDBC utilizando los objetos Statement, PreparedStatement, CallableStatement y a programar transacciones correctamente teniendo en cuenta su rendimiento y concurrencia.

Al final del curso se integra todo lo desarrollado en una aplicación de escritorio.

URL del Curso: <https://bit.ly/31apy0O>

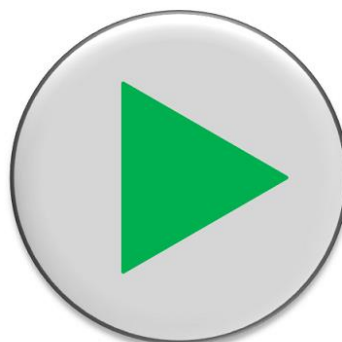
Avance del curso: <https://bit.ly/2vatZOT>

Cupones de descuento: <http://gcoronelc.github.io>



Programación con Oracle PL/SQL

ORACLE PL/SQL



En este curso aprenderás a programar las bases de datos ORACLE con PL/SQL, de esta manera estarás aprovechando las ventajas que brinda este motor de base de datos y mejorarás el rendimiento de tus consultas, transacciones y la concurrencia.

Los procedimientos almacenados que desarrolles con PL/SQL se pueden ejecutarlos de Java, C#, PHP y otros lenguajes de programación.

URL del Curso: <https://bit.ly/2YZjfxT>

Avance del curso: <https://bit.ly/3bcigYb>

Cupones de descuento: <http://gcoronelc.github.io>