

Desarrollo web en Java

Configura paso a paso todas las herramientas para crear proyectos web

Jorge Armando Arrambide Hernández

Todos los derechos reservados

Contenido

[Agradecimientos](#)

[Créditos](#)

[Acerca del autor](#)

[Propósito](#)

[Requerimientos](#)

[Capítulo 1. Conceptos](#)

[Capítulo 2. Frameworks](#)

[Capítulo 3. Base de datos](#)

[Capítulo 4. Servidor de aplicaciones](#)

[Capítulo 5. Herramientas](#)

[Capítulo 6. Descarga del kit de desarrollo](#)

[JDK \(Java Development Kit\)](#)

[JRE \(Java Runtime Environment\)](#)

[Tomcat 8.0](#)

[Eclipse](#)

[Oracle Express Edition](#)

[SQL Developer](#)

[Maven](#)

[Capítulo 7. Instalación del kit de desarrollo](#)

[Instalar JDK y JRE](#)

[Configurar variables de entorno para el JDK](#)

[Instalar Apache Tomcat](#)

[Instalar Eclipse](#)

[Instalar Oracle XE](#)

[Instalar SQL Developer](#)

[Capítulo 8. Oracle SQL Developer](#)

[Creando una conexión](#)

[Crear usuario](#)

[Crear nueva conexión con el nuevo usuario](#)

[Cambiar el puerto de OracleXE](#)

[Crear una tabla](#)

[Capítulo 9. Crear espacio de trabajo](#)

[Capítulo 10. Configurar Apache Tomcat](#)

[Capítulo 11. Crear un proyecto web](#)

[Capítulo 12. Estructura de un proyecto web](#)

[Capítulo 13. Un vistazo a Java Server Faces](#)

[Librerías de JSF](#)

[Configurar proyecto](#)

[Archivo web.xml](#)

[Archivo LoginView.java](#)

[Archivo index.xhtml](#)

[Archivo inicio.xhtml](#)

[Desplegar el proyecto](#)

[Capítulo 14. Conexión a base de datos](#)

[Apéndice](#)

[A.- Plug-in Jboss](#)

[B.- Configurar editor de XHTML](#)

[C.- Driver Oracle y configuración JNDI para el datasource](#)

[Configurar JNDI para el datasource](#)

[D.- Instalar y configurar Apache Maven](#)

[E.- Conociendo el entorno de desarrollo Eclipse](#)

Agradecimientos

*Dedico este esfuerzo a mi esposa
Norma, a mi hija Astrid y a mi hijo
Santiago.*

Créditos

Imagen de portada

Ing. Federico Treviño Hernández

Revisores

Ing. Emmanuel Eliud Castillo Avalos

M.E.S. Norma Jeaneth Treviño Hernández

Acerca del autor

Lic. Jorge Arrambide, egresado de la Facultad de Ciencias Físico-Matemáticas, es desarrollador Java Senior con más de 11 años en desarrollo web, además imparte cursos de desarrollo web en java.

Ha participado en diferentes proyectos para diversos giros de empresas: mensajería, financiera, comercio electrónico (e-commerce), gobierno, entre otras; con diferentes arquitecturas de desarrollo y frameworks.

Propósito

Este libro está dirigido a todas aquellas personas que quieran o necesiten incursionar en el desarrollo de sistemas web. Principalmente personas con muy pocos conocimientos o que ya desarrollan en otra tecnología.

Aclaro que no es un libro de programación, ya que solo se aborda el tema para preparar el ambiente en el cual se empezará a desarrollar, es decir, instalación y configuración de las herramientas: Java, Eclipse, Tomcat, Oracle XE, SQL Developer.

Contiene además un pequeño proyecto web de ejemplo, en el cual usaremos el framework de Java Server Faces y Oracle como base de datos, para introducirte en el desarrollo web.

Mencionaré algunas terminologías que se usan al momento de estar desarrollando, así como tips que he aprendido a lo largo de mi trayectoria.

Este libro será de mucha utilidad a aquellas personas que estén en busca de un trabajo o para aquellas que desean adquirir el conocimiento en dicho tema. Pretendo compartir un poco de mi conocimiento y experiencia.

Los capítulos son cortos y concisos, con lo cual pretendo no pierdas el ritmo y el paso de la configuración. Si tienes alguna duda no dudes en contactarme al siguiente correo:
jorge@bettik.com, y como **asunto**: *Libro Desarrollo Web en Java*.

Requerimientos

Los requerimientos que presento aquí son los mínimos, pero mientras más reciente sea tu computadora, tendrás un mejor tiempo de respuesta.

- Computadora con Windows
- 2 GB RAM
- El procesador puede ser cualquiera
- 3 GB de espacio en disco duro
- WinRAR o WinZip
- Navegador web

Capítulo 1. Conceptos

A continuación, presento algunos conceptos relevantes que pueden ser de ayuda conforme avances en el mundo de la programación:

API. - Leerás este término en diferentes ámbitos del desarrollo, no solo en proyectos web. Son métodos o funciones proporcionados por librerías o servicios web. Por ejemplo, Google Maps tiene una API para usar sus funciones de JavaScript en nuestras páginas HTML un ejemplo de ello: incrustar mapas en dichas páginas.

Java Standard Edition (JSE). - Es el conjunto de librerías *mínimas* necesarias para crear programas de escritorio o en consola de java.

Java Enterprise Edition (JEE). - Son las librerías de JSE más las librerías para crear proyectos escalables empresariales. Contiene las APIs para crear proyectos web, EJBs, Web Services y otros componentes. Estas librerías las puedes encontrar en los servidores de aplicaciones como: Tomcat, Wildfly, Jetty, Weblogic, etc.

Framework. - Es un conjunto de librerías, las cuales tienen como propósito hacer más con menos código, para agilizar el desarrollo. Hay frameworks para java, javascript, css, etc.

Archivos .JAR. - Son las librerías de java que necesita un proyecto java de escritorio o web para funcionar. Para que te des una idea, son como carpetas comprimidas en donde están archivos .class.

Servidor de aplicaciones. - Esta herramienta nos sirve para desplegar (hacer funcionar) el proyecto web, EJBs, WebServices, etc.

Base de datos. - Básicamente se usan para almacenar y consultar información, para ello existen diferentes tipos de base de datos, entre ellas las más usadas son las relacionales, como por ejemplo, MySQL y Oracle.

Proyecto web. - Hay dos tipos de proyectos web que se pueden crear en el entorno de desarrollo de Eclipse:

- Dinámico (es el que nos interesa)
- Estático

Un *proyecto web dinámico*, nos sirve para manejar formularios (páginas con cajas de texto, combos, carga de archivos, etc.), para poder almacenar información, editarla y/o consultarla.

Un *proyecto web estático*, solo sirve para mostrar información, pero no podemos manipular datos, es decir, no hay formularios. Es solo un sitio web con información e imágenes estática.

Despliegue (Deploy). - Se refiere a “montar” el archivo .war (el proyecto web compilado y empaquetado), en un servidor de aplicaciones (Tomcat, Jboss, etc.), para poder acceder vía el navegador al proyecto.

Capítulo 2. Frameworks

Los frameworks son librerías (en nuestro caso archivos .jar) que nos sirven para agilizar la programación, conforme pasa el tiempo aparecen nuevos frameworks que nos facilitan hacer más con menos código.

Algunos frameworks de java para proyectos web son los siguientes:

- Spring
- Java Server Faces
- Hibernate
- Struts
- Struts 2
- JPA

Y de seguro te preguntarás ¿cuál elegir?, lo único que te puedo decir en mi experiencia es que elijas de los más recientes (últimos 3 ó 4 años) y que tenga mucha comunidad en internet. Por ejemplo yo actualmente uso JSF 2.2 con Spring 4.2.2, ya que ambos tienen ya sus años de desarrollo y hay mucha documentación.

Por lo regular cuando entras a laborar en alguna empresa, ya hay un encargado de crear la arquitectura y de elegir el framework. Por lo que al principio no es necesario que sepas a detalle; lo principal es que conozcas los elementos necesarios para desarrollar.

Cuando vas empezando es difícil tomar una decisión de qué framework usar, por lo regular te basas en recomendaciones, pero por alguno tienes que empezar, acumular experiencia en dicho framework y luego experimentar con otros. Yo inicie con Struts y estaba aferrado (y con miedo a intentar con otro), pero conforme fui trabajando en diferentes proyectos vi que hay otros que tienen más ventajas. En mi caso pasé a usar Spring MVC y posteriormente Java Server Faces.

Capítulo 3. Base de datos

Existen diversas “marcas” de base de datos y en años recientes han surgido diversos “tipos” de base de datos. Me refiero a “marcas” como Mysql, Oracle, DB2, PostgreSQL, y me refiero a “tipos” como las relacionales y no relacionales (NoSQL), por presentar un ejemplo.

Para no confundir al lector, las más usadas en el ámbito laboral (por lo menos en México) son Oracle y Mysql que son de tipo relacional, de las cuales nos enfocaremos en la de Oracle.

En este libro no entrará en detalles de configuración y optimización de base datos, simplemente realizaremos algunos queries sencillos para comprobar la comunicación entre la base de datos y un proyecto web.

Si quieras iniciar un proyecto personal, puedes empezar con MySQL sin problemas. Conozco empresas que tienen cientos de transacciones por minuto y operan solo con esta base de datos. Además de no requerir licencia de pago para usarla. En un dado caso de que quieras soporte de dicha herramienta, en ese momento tendrás que pagar.

Capítulo 4. Servidor de aplicaciones

Al principio puede confundir el término “servidor de aplicaciones” con “servidor” (que es una máquina física que tiene un Sistema Operativo).

Un servidor de aplicaciones, es un componente o herramienta en donde se despliega un proyecto web, servicios web, etc.

A su vez uno o varios servidores de aplicaciones están instalados en un servidor.

Existen diferentes tipos de servidores de aplicaciones, algunos gratuitos y otros de pago.
¿Cuál es el mejor?, ¿Puedo usar el que quiera? ó ¿hay alguna restricción?

Por lo regular cuando empiezas a hacer tus primeros proyectos, uno de los servidores de aplicaciones (por popular) que puedes usar es el *Apache Tomcat*.

Listo algunos otros que conozco:

- Apache Tomcat
- GlassFish
- Jboss
- Weblogic
- Websphere
- Wildfly

Es importante que, conforme empieces a trabajar en diversos proyectos, conozcas algunos de ellos. Para que sepas en donde encontrar sus archivos de configuración.

Todos sirven para lo mismo: **desplegar aplicaciones web**, pero verás que los servidores que requieren pago de licencia tienen características adicionales, las cuales son de utilidad para empresas que quieren un grado de robustez alto en su sistema.

En todos servidores de aplicaciones existen archivos de configuración (xml) en donde se pueden modificar los números de puertos del servidor, así como características de seguridad (SSL, que viene siendo el candadito verde que se pinta en los navegadores web, para identificar un sitio seguro), configuración de URLs, etc.

Además, en el servidor de aplicaciones puedes configurar la conexión a la base de datos (que puede ser mediante un pool de conexiones) de la o las aplicaciones que tengas desplegadas en dicho servidor de aplicaciones.

Para los que van empezando solo es necesario que sepan los siguientes puntos básicos de un servidor de aplicaciones:

- Desplegar un proyecto web (es decir, echar a funcionar el proyecto)
- Configurar la conexión a la base de datos (Cómo se conectará nuestro código de java a nuestra base de datos)
- Conocer el archivo server.xml (en el caso de Tomcat; en otros servidores es un archivo con nombre similar), y algunas de las configuraciones que vienen allí.

Por lo regular hay personas que se dedican a “*tunear*” y mantener dichos servidores de aplicaciones en óptimas condiciones de acuerdo a la cantidad de usuarios que usen la aplicación que se haya desarrollado, añadiendo más memoria RAM o uso de procesador.

Capítulo 5. Herramientas

A continuación, presento las herramientas que se necesitan para desarrollar en java, en el siguiente capítulo veremos de donde se descargan y como instalarlas:

JDK, Java Development Kit, es el encargado de compilar las clases .java en .class. Actualmente no es tan necesario instalarlo, ya que el IDE de Eclipse tiene uno ya preinstalado. Pero de todas formas lo instalaremos más adelante.

JRE, Java Runtime Environment, sirve para poder ejecutar el IDE Eclipse y el servidor de aplicaciones Tomcat, también es el que se encarga de leer/procesar los archivos .class.

Eclipse, uno de los IDE (*Integrated Development Environment*) más populares, en esta herramienta se realiza toda la programación de java.

Tomcat, uno de los servidores de aplicaciones gratuitos más populares, en esta herramienta se “coloca” el proyecto web desarrollado para ser desplegado.

Oracle XE (express edition), un motor de base de datos muy usado en la mayoría de las empresas. la versión gratuita de Oracle es la *XE*.

Oracle Sql Developer, un IDE que nos sirve para administrar la base de datos, puedes hacer entre otras cosas: crear usuarios, esquemas, tablas, procedimientos, querys, etc.

Maven, esta herramienta es opcional y sirve para crear proyectos y administrar sus librerías necesarias para el desarrollo.

Capítulo 6. Descarga del kit de desarrollo

En este capítulo veremos el tema de las descargas en el cual detallo los pasos con capturas de pantalla, te recomiendo ampliamente seguir los pasos, ya que en ocasiones en la página web hay varias opciones de descarga, lo que también te evitará que te estanques en algún paso más adelante.

Nota: Es necesario registrarse en el sitio de Oracle, para poder descargar su software y librerías.

JDK (Java Development Kit)

El JDK es lo que sirve para compilar los archivos .java y generar los archivos .class.

Para descargar el Java SE Development Kit (conocido también como JDK) lo puedes hacer de la siguiente manera (te comento que este instalador también contiene el JRE con la misma versión del JDK, por lo que te ahorras la descarga por separado).

Paso 1: Buscar el JDK SE 8

The screenshot shows a Google search results page for the query "jdk". The search bar at the top contains "jdk". Below the search bar are navigation links for "Todos", "Imágenes", "Videos", "Noticias", "Aplicaciones", "Más", and "Herramientas de búsqueda". A status message indicates "Cerca de 14,900,000 resultados (0.34 segundos)". The first result is a link to "Java SE - Downloads | Oracle Technology Network | Oracle" with the URL "www.oracle.com/technetwork/es/java/javase/downloads/". A red arrow points to the link "Java SE Development Kit 8 ...". To the right of the result, there is a snippet of text: "Java SE Development Kit 8 ... Download JDK 8, a development environment for building ...". Another result listed is "Java SE Development Kit 7 ...". To the right of this result, there is a snippet: "Java EE Java EE is developed using the Java Community Process, with ...". At the bottom of the results, there is a link "Más resultados de oracle.com »".

Paso 2: Dar clic en el radio “Accept License Agreement” y descargar según el sistema operativo, en el caso de Windows, elegir “Windows x64” si tienes un procesador de 64 bits, si tienes un procesador de 32 bits, elige “Windows x86”.

Overview Downloads Documentation Community Technologies Training

Java SE Development Kit 8 Downloads

Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building applications, applets, and components using the Java programming language.

The JDK includes tools useful for developing and testing programs written in the Java programming language and running on the Java platform.

See also:

- Java Developer Newsletter: From your Oracle account, select Subscriptions, expand Technology, and subscribe to Java.
- Java Developer Day hands-on workshops (free) and other events
- Java Magazine

JDK 8u101 Checksum
JDK 8u102 Checksum

Java SE Development Kit 8u101

You must accept the Oracle Binary Code License Agreement for Java SE to download this software.

Accept License Agreement Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.77 MB	jdk-8u101-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	74.72 MB	jdk-8u101-linux-arm64-vfp-hflt.tar.gz
Linux x86	160.28 MB	jdk-8u101-linux-i586.rpm
Linux x86	174.96 MB	jdk-8u101-linux-i586.tar.gz
Linux x64	158.27 MB	jdk-8u101-linux-x64.rpm
Linux x64	172.95 MB	jdk-8u101-linux-x64.tar.gz
Mac OS X	227.36 MB	jdk-8u101-macosx-x64.dmg
Solaris SPARC 64-bit	139.66 MB	jdk-8u101-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	98.96 MB	jdk-8u101-solaris-sparcv9.tar.gz
Solaris x64	140.33 MB	jdk-8u101-solaris-x64.tar.Z
Solaris x64	96.78 MB	jdk-8u101-solaris-x64.tar.gz
Windows x86	188.32 MB	jdk-8u101-windows-i586.exe
Windows x64	193.68 MB	jdk-8u101-windows-x64.exe

JRE (Java Runtime Environment)

El JRE, explicado de una manera sencilla, es lo que hace que los archivos .class se puedan ejecutar.

La descarga de este componente es opcional, ya que en la descarga anterior el JDK ya lo incluye. En caso de que quieras obtener por separado el JRE o alguna versión en específico lo puedes hacer de la siguiente manera:

Paso 1: Buscar JRE

A screenshot of a Google search results page. The search term 'jre' is entered in the search bar. Below the search bar, there are navigation links: Todos, Imágenes, Videos, Maps, Aplicaciones, Más ▾, and Herramientas de búsqueda. A red arrow points to the first search result, which is titled 'Java SE Runtime Environment 8 - Downloads - Oracle'. The result includes a link to the Oracle website (www.oracle.com/technetwork/es/java/javase/jre7-downloads-1880261.html) and a brief description: 'If you want to run Java programs, but not develop them, download the Java Runtime Environment, or JRE™. If you want to develop applications for Java, ...'.

Paso 2: Seleccionar la opción Accept License Agreement y luego descargar **Windows x64 Offline**. (Offline para tener el instalador localmente)

A screenshot of the Java SE Runtime Environment 8u101 download page. At the top, there are tabs: Overview, Downloads, Documentation, Community, Technologies, and Training. The 'Downloads' tab is selected. Below the tabs, the heading 'Java SE Runtime Environment 8 Downloads' is displayed. A note says: 'Do you want to run Java™ programs, or do you want to develop Java programs? If you want to run Java programs, but not develop them, download the Java Runtime Environment, or JRE™.' Another note below it says: 'If you want to develop applications for Java, download the Java Development Kit, or JDK™. The JDK includes the JRE, so you do not have to download both separately.' There are two 'Checksum' links: 'JRE 8u101 Checksum' and 'JRE 8u102 Checksum'. A modal dialog box titled 'Java SE Runtime Environment 8u101' contains the text: 'You must accept the Oracle Binary Code License Agreement for Java SE to download this software.' and 'Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.' A red arrow points to the 'Windows x64 Offline' row in the download table. The table has columns: Product / File Description, File Size, and Download. The rows are:

Product / File Description	File Size	Download
Linux x86	54.79 MB	jre-8u101-linux-i586.rpm
Linux x86	70.58 MB	jre-8u101-linux-i586.tar.gz
Linux x64	52.68 MB	jre-8u101-linux-x64.rpm
Linux x64	68.49 MB	jre-8u101-linux-x64.tar.gz
Mac OS X	55.99 MB	jre-8u101-macosx-x64.tar.gz
Mac OS X	64.32 MB	jre-8u101-macosx-x64.dmg
Solaris SPARC 64-bit	52 MB	jre-8u101-solaris-sparcv9.tar.gz
Solaris x64	49.85 MB	jre-8u101-solaris-x64.tar.gz
Windows x86 Online	0.71 MB	jre-8u101-windows-i586-iftw.exe
Windows x86 Offline	52.63 MB	jre-8u101-windows-i586.exe
Windows x86	59.42 MB	jre-8u101-windows-i586.tar.gz
Windows x64 Offline	59.17 MB	jre-8u101-windows-x64.exe
Windows x64	62.77 MB	jre-8u101-windows-x64.tar.gz

Tomcat 8.0

Tomcat es el servidor de aplicaciones gratuito usado por muchos desarrolladores, tiene las características mínimas necesarias para desplegar un proyecto web.

Paso 1: Buscar tomcat 8.0 en internet:

Cerca de 9,040,000 resultados (0.43 segundos)

Apache Tomcat® - Apache Tomcat 8 Software Downloads
<https://tomcat.apache.org/download-80.cgi> ▾ Traducir esta página
Welcome to the Apache Tomcat® 8.x software download page. This page provides download links for obtaining the latest versions of Tomcat 8.x software, ...

Apache Tomcat® - Welcome!
tomcat.apache.org/ ▾ Traducir esta página
The Apache Tomcat® software is an open source implementation of the Java Servlet, JavaServer Pages, Java ... 2016-07-02 Tomcat Native 1.2.8 Released.

Paso 2: Descargar el archivo 64-bit Windows.zip (dependiendo del procesador de tu computadora)

Nota: Descargar la versión 8.0.XX, ya que la versión 8.5.X requiere la versión de IDE de Eclipse más reciente.

Misc

- Who We Are
- Heritage
- Apache Home
- Resources
- Contact
- Legal
- Sponsorship
- Thanks

Source Code Distributions

- [tar.gz \(pgp, md5, sha1\)](#)
- [zip \(pgp, md5, sha1\)](#)

8.0.37

Please see the [README](#) file for packaging information. It explains what every

Binary Distributions

- Core:
 - [zip \(pgp, md5, sha1\)](#)
 - [tar.gz \(pgp, md5, sha1\)](#)
 - [32-bit Windows zip \(pgp, md5, sha1\)](#)
 - [64-bit Windows zip \(pgp, md5, sha1\)](#)
 - [32-bit/64-bit Windows Service Installer \(pgp, md5, sha1\)](#)
- Full documentation:
 - [tar.gz \(pgp, md5, sha1\)](#)

Eclipse

Uno de los IDE más populares, destaca principalmente por la ligereza, fácil instalación y que tiene muchos Plugins de terceros que le dan más robustez.

Al momento de escribir esta guía, la versión que descargamos será **Eclipse Neon**.

Paso 1: Buscamos la palabra **eclipse** en google, y damos clic en el enlace que muestro a continuación:

Cerca de 192,000,000 resultados (0.65 segundos)

[Eclipse Downloads](https://www.eclipse.org/downloads/)

<https://www.eclipse.org/downloads/> ▾ Traducir esta página
Download Eclipse Technology that is right for you. SAP ... Install your favorite Eclipse packages.
Eclipse Che is a developer workspace server and cloud IDE.
Eclipse Downloads · Getting Started with Eclipse · Projects Gateway · Eclipse Help

[Eclipse - The Eclipse Foundation open source community website.](https://eclipse.org/)
<https://eclipse.org/> ▾ Traducir esta página
A project aiming to provide a universal toolset for development. Open Source IDE mostly provided in

Paso 2: Damos clic en el link “**Download Packages**”

Download Eclipse Technology that is right for you

Tool Platforms

 Get Eclipse **Neon**

Install your favorite Eclipse packages.

[DOWNLOAD 64 BIT](#)

[Download Packages](#)

 Eclipse Che

Eclipse Che is a developer workspace server and cloud IDE

Paso 3: En la siguiente pantalla buscamos Eclipse IDE for Java EE Developers y descargamos la versión según la arquitectura de nuestro equipo de cómputo, en la mayoría de los casos será 64 bits.

Tip: Del lado derecho de la imagen, se pueden ver versiones anteriores de Eclipse (Mars, Luna, Kepler, etc.), en caso de que lo requieras, los puedes descargar.

Eclipse IDE for Java EE Developers

Eclipse IDE for Java Developers

Eclipse IDE for C/C++ Developers

Eclipse for Android Developers

Windows
32 bit | 64 bit

RELATED LINKS

- Compare & Combine Packages
- New and Noteworthy
- Install Guide
- Documentation
- Updating Eclipse
- Forums

MORE DOWNLOADS

- Other builds
- Eclipse Neon (4.6)
- Eclipse Mars (4.5)
- Eclipse Luna (4.4)
- Eclipse Kepler (4.3)
- Eclipse Juno (4.2)
- Older Versions

Paso 4: Para los despistados, dejo la siguiente imagen:

eclipse

GETTING STARTED MEMBERS PROJECTS MORE ▾

HOME / DOWNLOADS / ECLIPSE DOWNLOADS - SELECT A MIRROR

All downloads are provided under the terms and conditions of the [Eclipse Foundation Software User Agreement](#) unless otherwise specified.

DOWNLOAD

Download from: Canada - University of Waterloo Computer Science Club (http)

File: [eclipse-jee-neon-3-win32-x86_64.zip](#) SHA-512

>> Select Another Mirror

RAPID
THE VISUAL
BEST GUI E
NEW HIBERNA
WEB • MOBILE • DE
DOWNLOAD
OTHER OPTIONS

Oracle Express Edition

Lo siguiente que descargamos será el motor de la base de datos, en este caso usaremos *Oracle XE*, por ser uno de los más usados en las empresas.

Al momento de escribir este libro está disponible la *versión 11g*, de uso gratuito. (Por si no coincide la versión que hayas encontrado)

Nota: Al momento de escribir esta guía, pide registrarse en el sitio, para poder descargar el instalador.

Paso 1: Buscamos *Oracle XE*

Cerca de 446,000 resultados (0.80 segundos)

[Oracle Database Express Edition 11g Release 2](#)

www.oracle.com › ... › Database Express Edition ▾ Traducir esta página
Oracle Database Express Edition 11g Release 2 (Oracle Database XE) is an entry-level, small-footprint RDBMS based on the Oracle Database 11g Release 2 ...

[Oracle Database Express Edition 11g Release 2 Downloads](#)
www.oracle.com › ... › Database Express Edition ▾ Traducir esta página

Paso 2: Seleccionamos “Accept License Agreement” y luego descargamos *Oracle Database Express Edition 11g Release 2 for Windows x64* (según el procesador de tu PC):

Oracle Technology Network > Database > Database Technology Index > Database Express Edition > Downloads

Database 12c
Database In-Memory
Multitenant
Options
Application Development
Big Data Appliance
Data Warehousing & Big Data
Database Appliance
Database Cloud
Exadata Database Machine
High Availability
Manageability

Overview Downloads Documentation Community Learn More

Oracle Database Express Edition 11g Release 2

June 4, 2014

You must accept the OTN License Agreement for Oracle Database Express Edition 11g Release 2 to download this software.

Accept License Agreement | Decline License Agreement

[Oracle Database Express Edition 11g Release 2 for Windows x64](#)
- Unzip the download and run the DISK1/setup.exe
[Oracle Database Express Edition 11g Release 2 for Windows x32](#)
- Unzip the download and run the DISK1/setup.exe
[Oracle Database Express Edition 11g Release 2 for Linux x64](#)
- Unzip the download and the RPM file can be installed as normal

SQL Developer

Oracle cuenta además con un IDE de uso gratuito, mediante el cual podremos realizar consultas, insertar, actualizar y eliminar registros de la base de datos, que es el Oracle SQL Developer, a continuación, los pasos para descargarlo:

Paso 1: Buscar *sql developer*

Google search results for "sql developer". The first result is for Oracle SQL Developer, which is highlighted with a red arrow. The result includes a link to "Downloads".

Cerca de 5,470,000 resultados (0.44 segundos)

Oracle SQL Developer
www.oracle.com/.../developer.../sql-developer/.../index.html?... ▾ Traducir esta |
Oracle SQL Developer to query, update monitor the Oracle Database. Use for SQL and F
Provides reporting, unit testing and tuning.
Oracle SQL Developer · New Features · Develop · SQL Developer

Oracle SQL Developer Downloads
www.oracle.com › Developer Tools › SQL Developer ▾ Traducir esta página
SQL Developer Downloads. License Agreement. You must accept the OTN ...

Paso 2: Descargar el instalador. Lo descargamos **sin el JDK 8**, ya que previamente lo descargamos manualmente.

SQL Developer Downloads

License Agreement

Thank you for accepting the OTN License Agreement; you may now download this software. To learn more about our development tools, join every Wednesday for a free [Webinar on Database Development Tools](#).

SQL Developer 4.1.3

Version 4.1.3.20.78, Updated December 22, 2015

[Bugs Fixed](#), [Release Notes](#), [New Features](#), [Documentation](#)

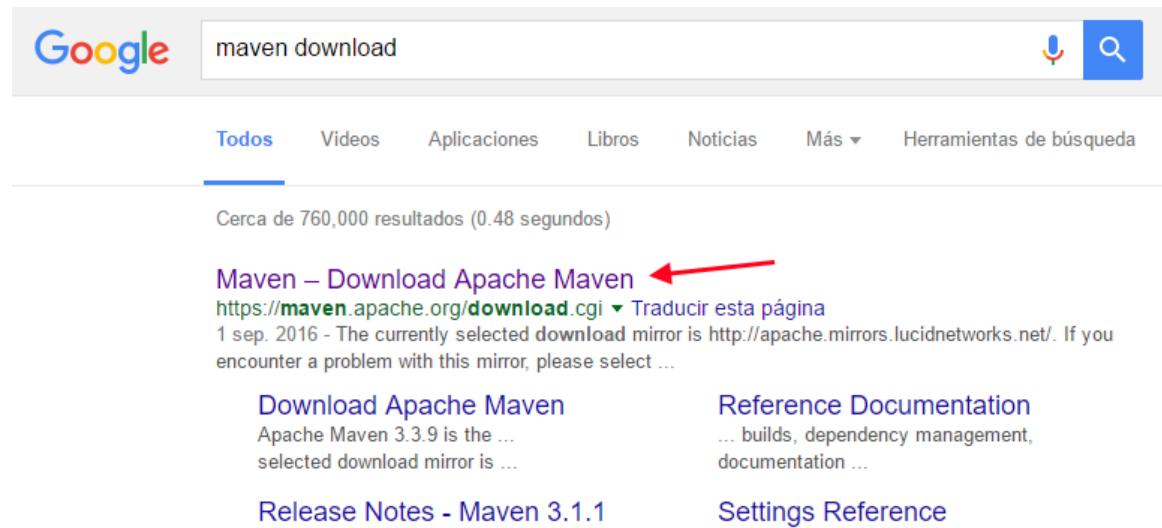
Windows 64-bit with JDK 8 included
[Installation Notes](#) 381 MB [Download](#)

Windows 32-bit/64-bit
[Installation Notes](#), [JDK 8 or above required](#) 314 MB [Download](#)

Maven

Por último descargamos Maven, esta herramienta nos servirá para descargar las librerías (.jar) necesarias para un proyecto.

Paso 1: Buscamos maven



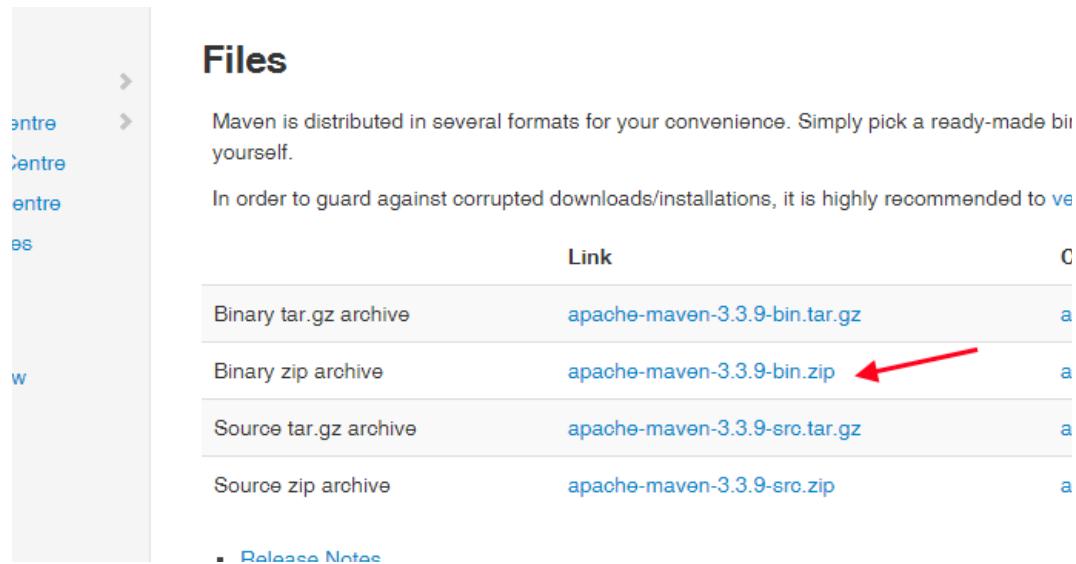
Cerca de 760,000 resultados (0.48 segundos)

[Maven – Download Apache Maven](https://maven.apache.org/download.cgi) 

<https://maven.apache.org/download.cgi> Traducir esta página
1 sep. 2016 - The currently selected download mirror is http://apache.mirrors.lucidnetworks.net/. If you encounter a problem with this mirror, please select ...

Download Apache Maven Apache Maven 3.3.9 is the ... selected download mirror is ...	Reference Documentation ... builds, dependency management, documentation ...
Release Notes - Maven 3.1.1	Settings Reference

Paso 2: Seleccionamos la segunda opción, ZIP con archivos binarios (*apache-maven-3.x.x-bin.zip*):



Files

Maven is distributed in several formats for your convenience. Simply pick a ready-made binary yourself.

In order to guard against corrupted downloads/installations, it is highly recommended to verify the checksums.

	Link	Checksum
Binary tar.gz archive	apache-maven-3.3.9-bin.tar.gz	a
Binary zip archive	apache-maven-3.3.9-bin.zip	a
Source tar.gz archive	apache-maven-3.3.9-src.tar.gz	a
Source zip archive	apache-maven-3.3.9-src.zip	a

[Release Notes](#)

Capítulo 7. Instalación del kit de desarrollo

Ya que tenemos todo el software descargado es momento de realizar la instalación.

Al principio te puede parecer tedioso pero solo se hace una vez y con la práctica se hace más sencillo.

IMPORTANTE: Es necesario que sigas los pasos y leas el texto que hay en ellos, ya que en ocasiones mencionó pasos que están los Apéndices de este libro.

Nota: *Eclipse, Apache Maven y Tomcat*, se recomienda que se instalen directamente C:/. Por ejemplo: C:\eclipse, C:\Apache-Maven. Principalmente para simplificar la ruta de los mismos y cuando necesitemos accesarlos sea más fácil.

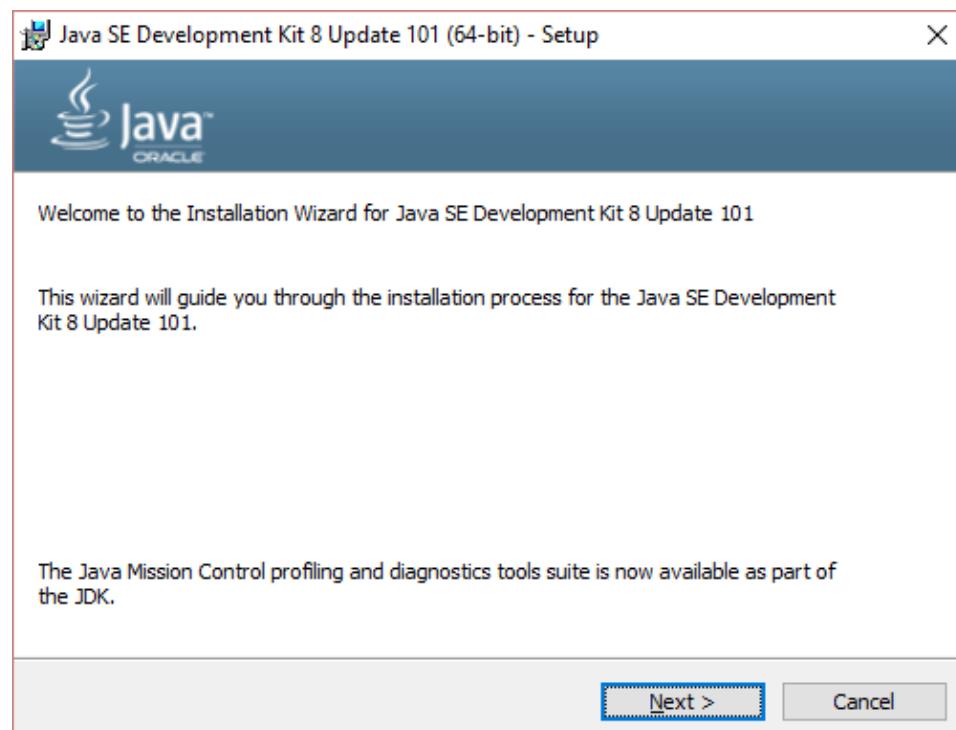
TIP: Te recomiendo crear accesos directos de: *Eclipse* y de *SQL Developer*, ya que no los crea de forma automática.

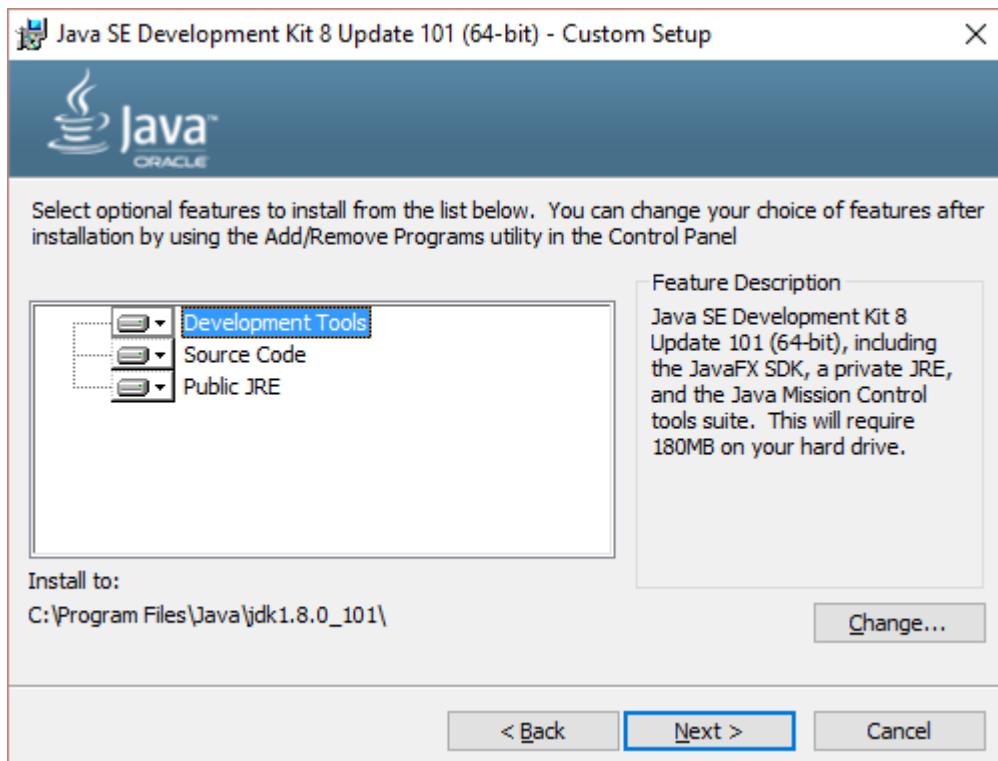
Instalar JDK y JRE

La instalación es muy sencilla solo hay que dar clic en *Siguiente*, en cada paso de la instalación.

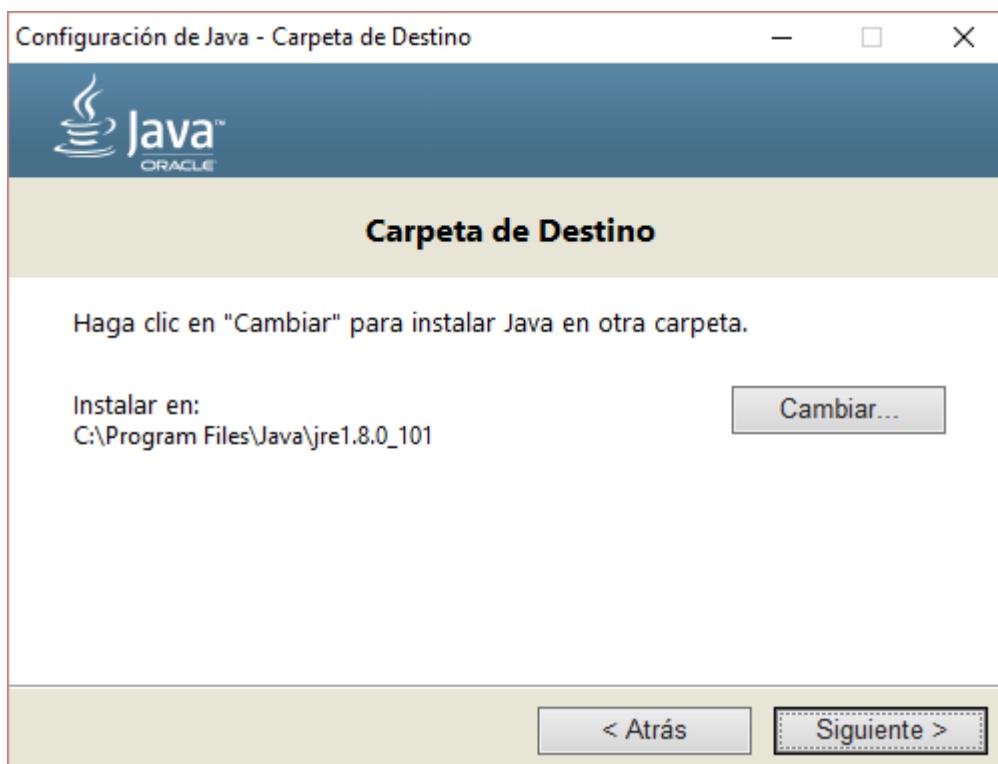
Nota: Antes de terminar verás que también se instala el JRE, que viene incluido en el JDK.

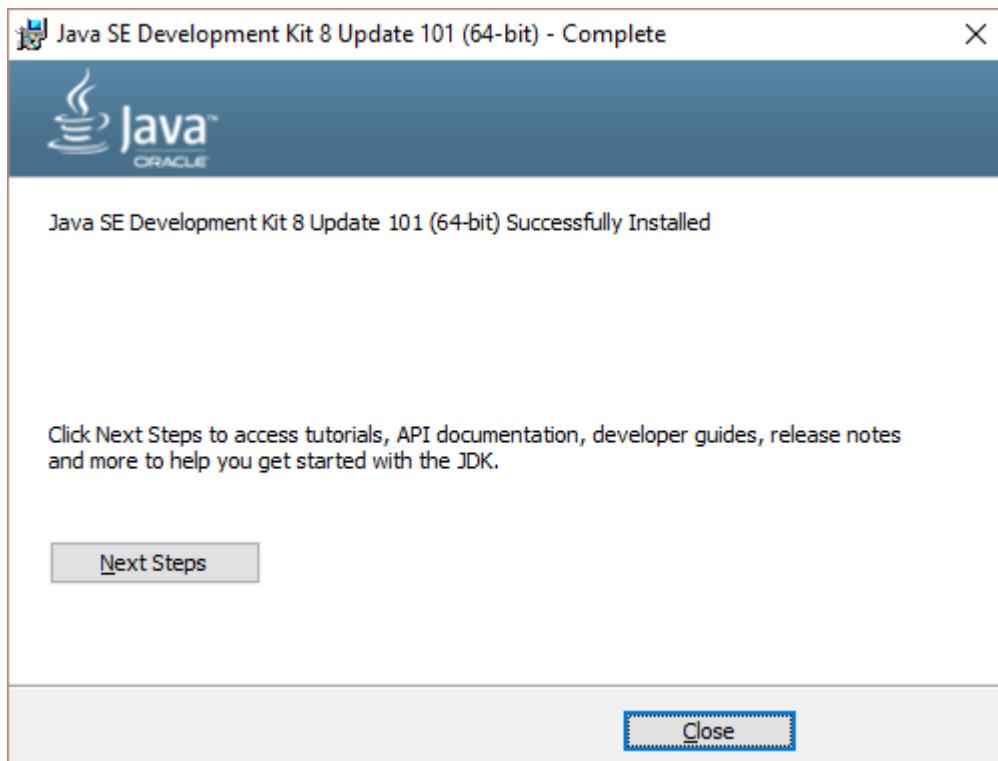
Paso 1: Ejecuta el Java SE Development Kit





Esta ventana es para instalar el JRE que trae el instalador, que coincide con la versión del JDK, lo dejamos como está y clic en *Siguiente*:





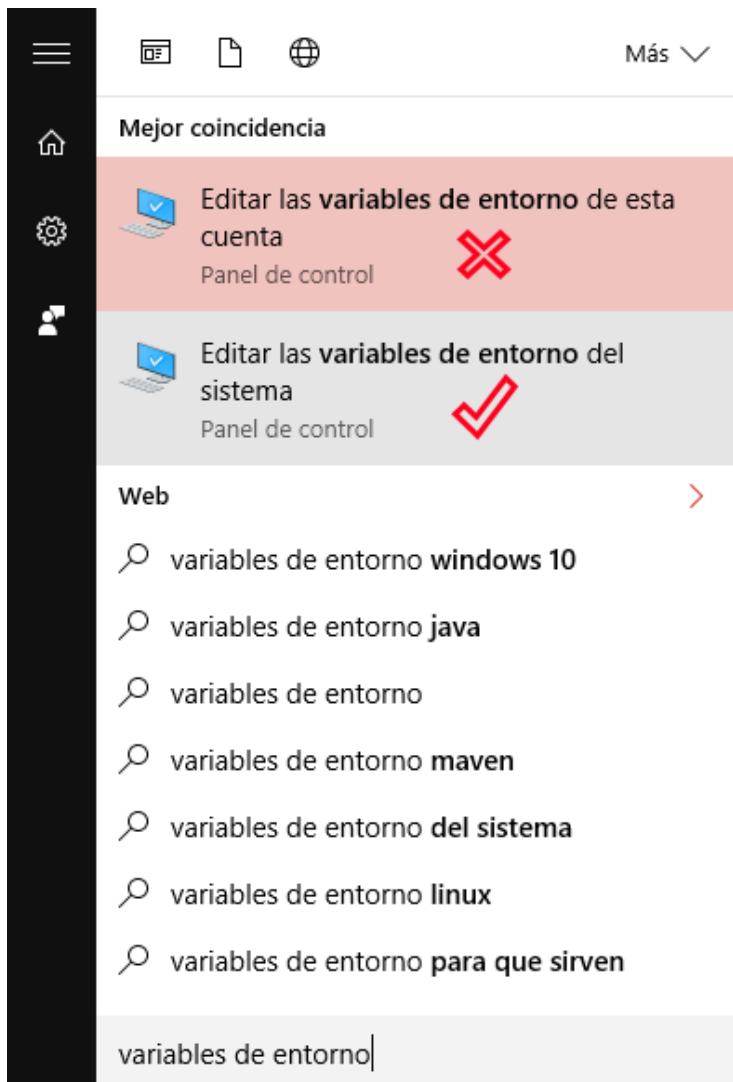
Configurar variables de entorno para el JDK

Configurar las variables de entorno sirve para poder ejecutar los comandos de java desde cualquier carpeta; si no se realiza esta configuración, tienes que poner tu código fuente (archivos .java) directamente en la **carpeta bin** de Java.

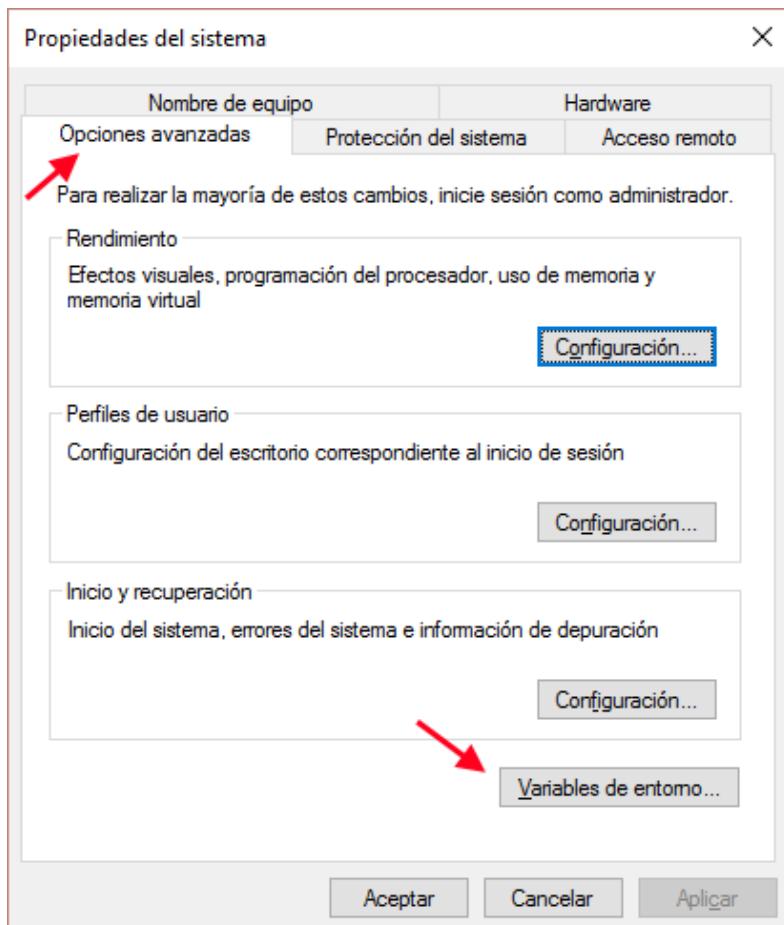
Las variables de entorno se configuran de la misma manera en todos los sistemas operativos con Windows, en esta ocasión se realizará con Windows 10. Puedes seguir estos pasos si tienes Windows 8.

En caso de que tengas Windows 7 o vista, te recomiendo busques en internet “*Configurar variables de entorno*” para que puedas ver interfaz gráfica de las ventanas.

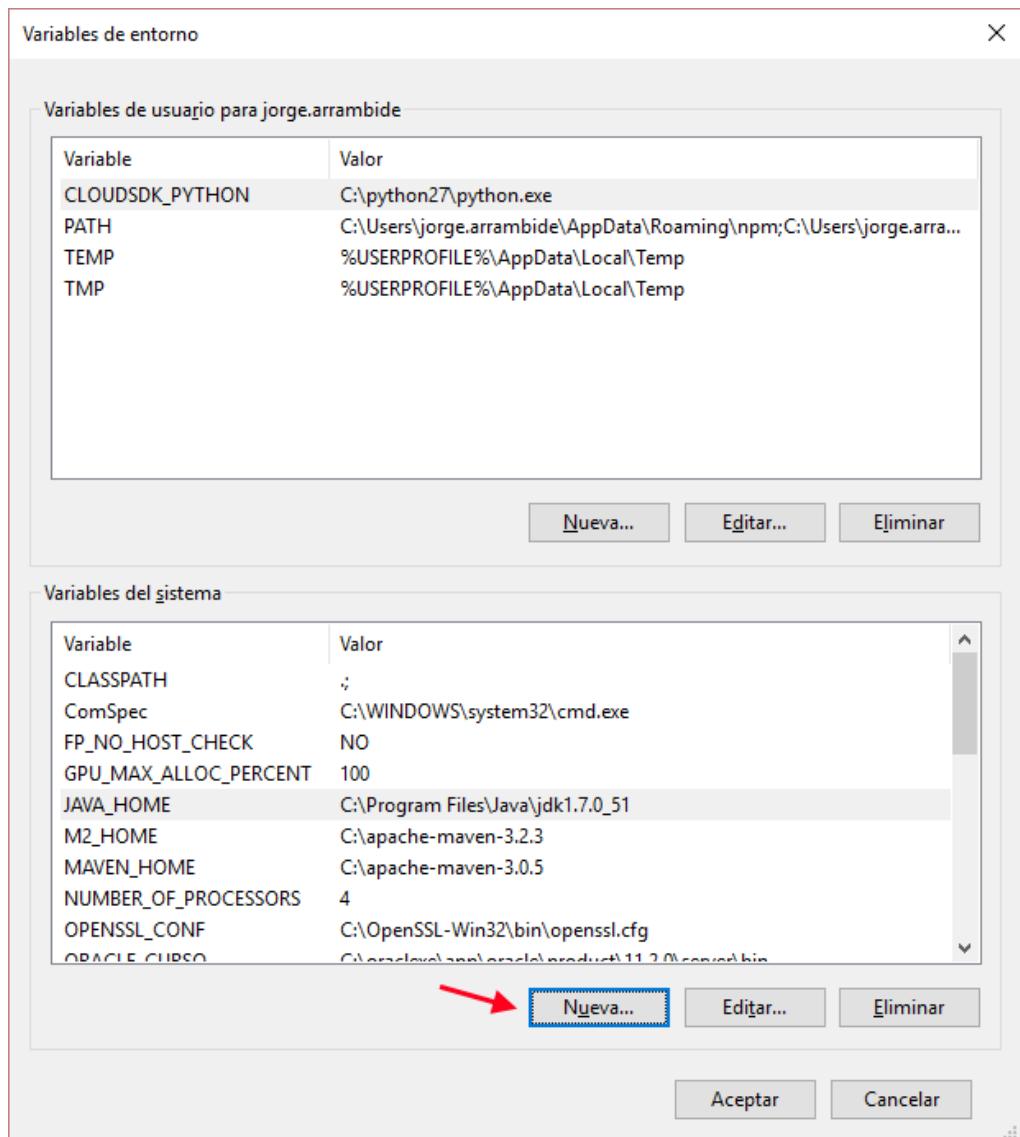
Paso 1: Vamos al botón de inicio de Windows, y buscamos “**variables de entorno**”, en algunas ocasiones puede mostrar más de una opción como se muestra en la siguiente imagen:



Paso 2: Debemos seleccionar la que dice “**Editar variables de entorno del sistema**”. Veremos la siguiente ventana, dar clic en el botón “*Variables de entorno*”.

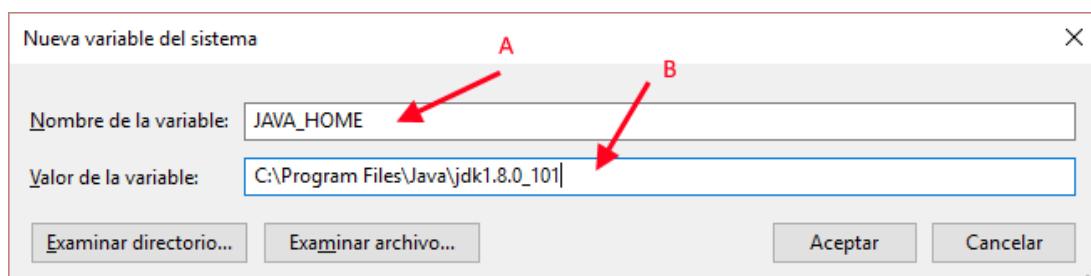


Paso 3: Dar clic en el botón *Nueva*, en la sección **Variables del sistema** (cuadro de abajo)



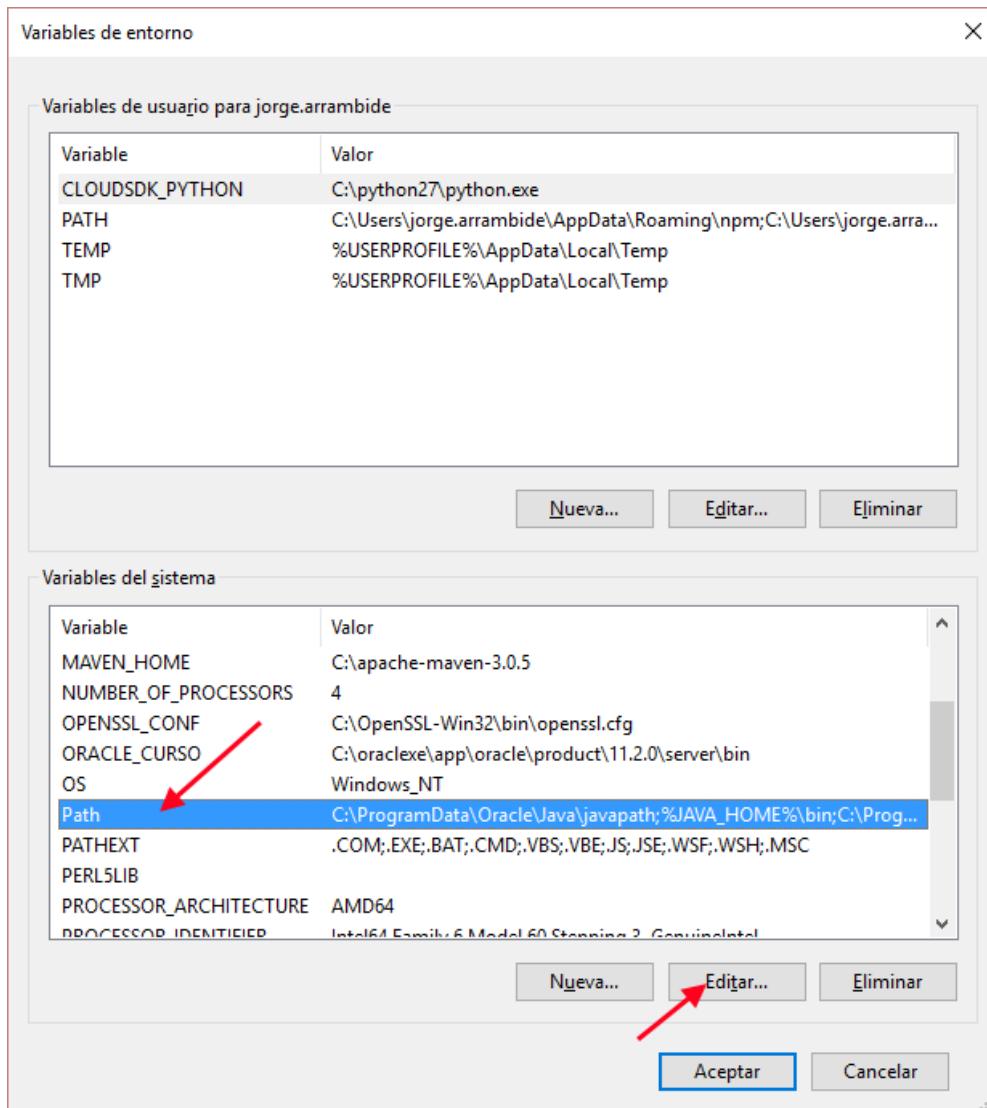
Paso 4: Introducimos lo siguiente, en el punto A irá el “alias” y en el B debes de poner la ruta en donde se instaló Java.

Tip: Te recomiendo copiar la ruta desde la carpeta de Windows, para evitar problemas de error de dedo.

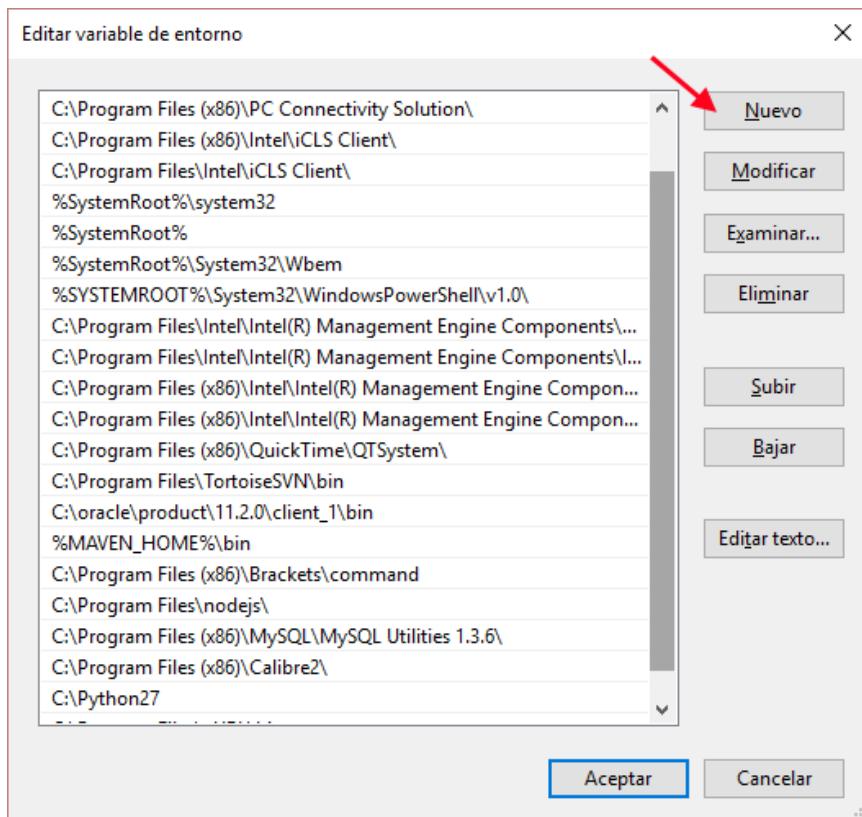


Paso 5: Una vez creada la variable JAVA_HOME, hay que ponerla en la variable Path, como lo muestro en la siguiente imagen.

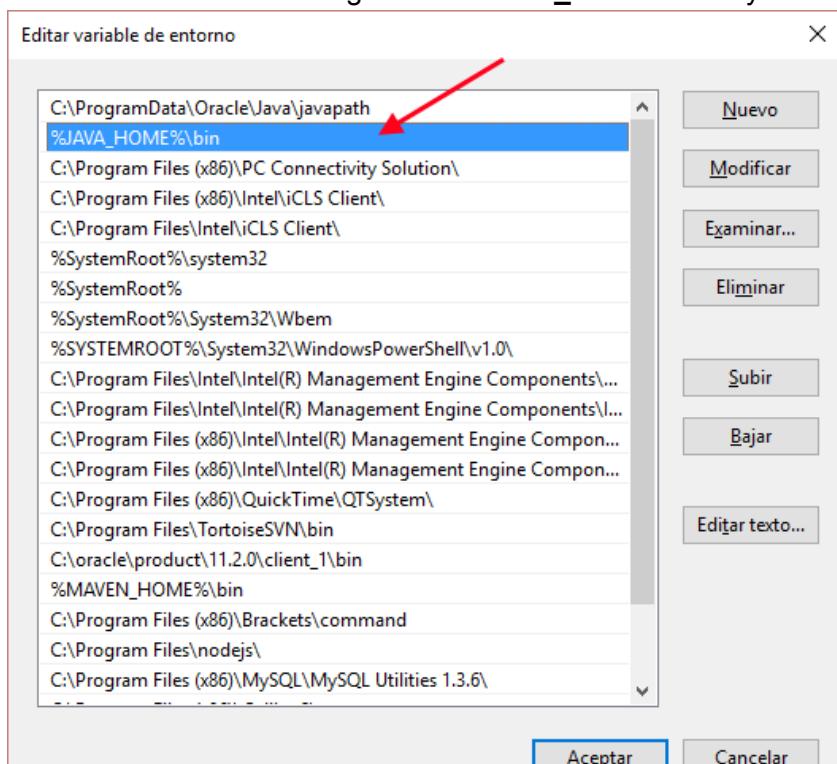
Seleccionamos la variable Path, y luego damos clic en el botón Editar.



Paso 6: En la siguiente ventana damos clic en el botón Nuevo:



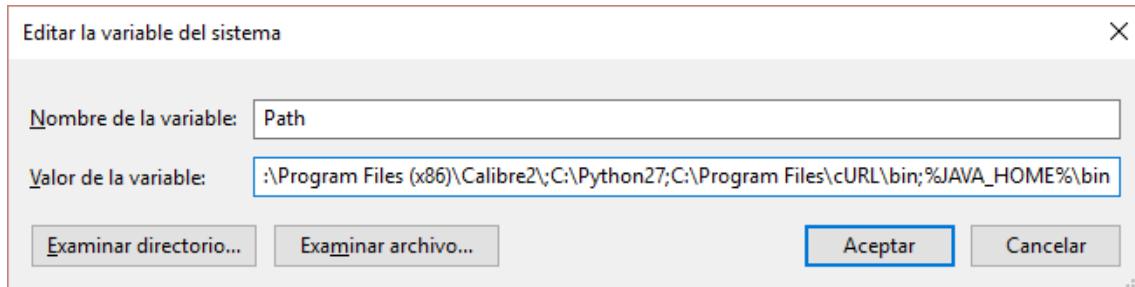
Paso 7: Introducimos lo siguiente `%JAVA_HOME%\bin` y damos clic en Aceptar.



Tip: Si tienes una versión inferior a Windows 10, verás una ventana como la siguiente, en lugar de la ventana del **PASO 7**. De ser así solo coloca al final lo siguiente:

`;%JAVA_HOME%\bin`

Tip: No olvides el punto y coma que está al inicio.



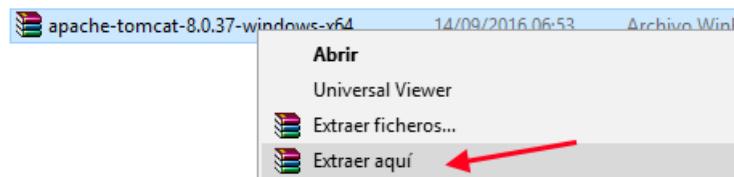
Instalar Apache Tomcat

La instalación de Apache Tomcat es relativamente sencilla, solo hay que descomprimir el archivo ZIP, de preferencia en C: directamente, como se muestra a continuación:

Tip: Si Windows no te deja descomprimir en C: por cuestiones de permisos de administrador, descomprímelo en donde si te deje hacerlo (Ej. Mis Documentos), copia la carpeta descomprimida y pegala en C.

Extra Tip: En ocasiones, Windows no deja descomprimir directamente en C, por lo que te recomiendo descomprimir en otra parte y luego Copiar en C.

Paso 1: Dar clic derecho, extraer aquí (en caso de que tengas WinRAR)



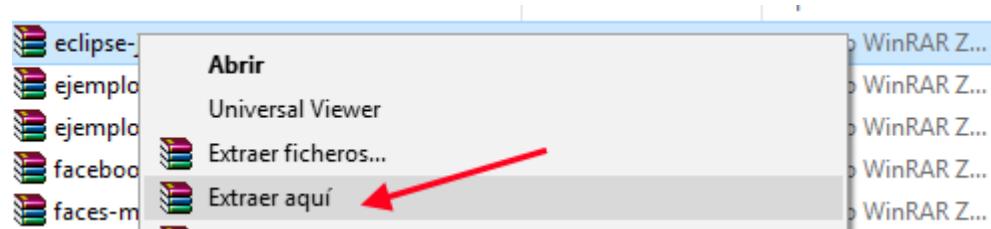
Paso 2: Asegurar que esté en C:

Nombre	Fecha de modifica...	Tipo
bin	14/09/2016 06:57 ...	Carpeta de archivos
conf	14/09/2016 06:57 ...	Carpeta de archivos
lib	14/09/2016 06:57 ...	Carpeta de archivos
logs	01/09/2016 11:02 a...	Carpeta de archivos
temp	14/09/2016 06:57 ...	Carpeta de archivos
webapps	14/09/2016 06:57 ...	Carpeta de archivos
work	01/09/2016 11:02 a...	Carpeta de archivos
LICENSE	01/09/2016 11:06 a...	Archivo
NOTICE	01/09/2016 11:06 a...	Archivo

Instalar Eclipse

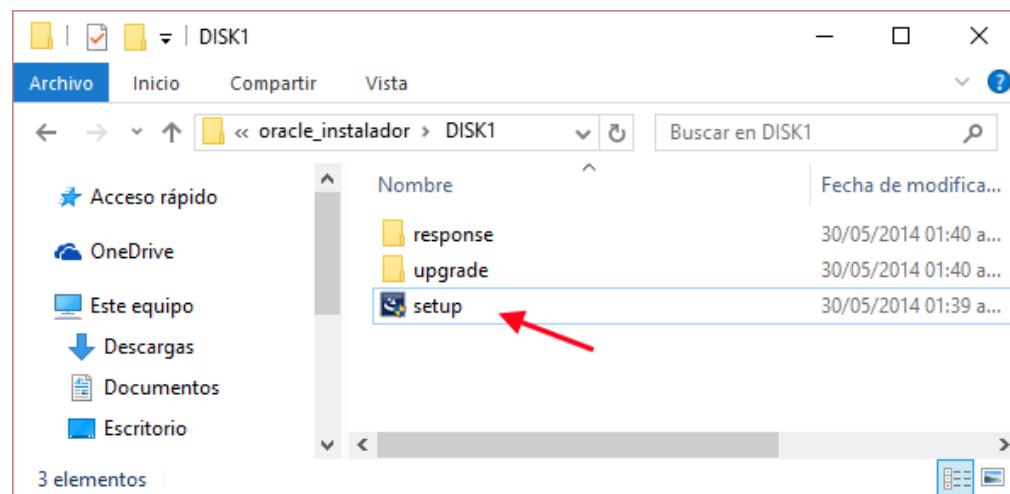
La instalación de Eclipse es la más sencilla ya que no es una instalación convencional, solo hay que descomprimir el archivo .ZIP y listo.

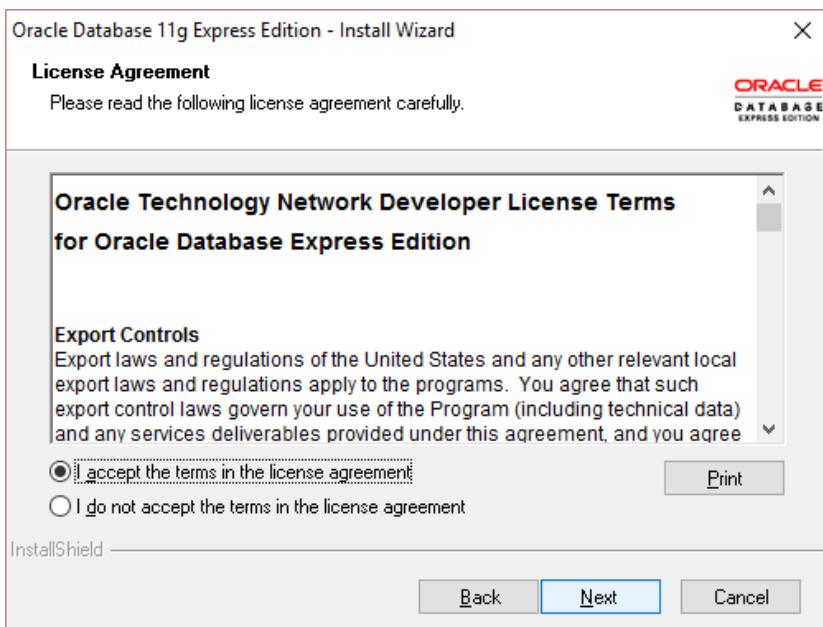
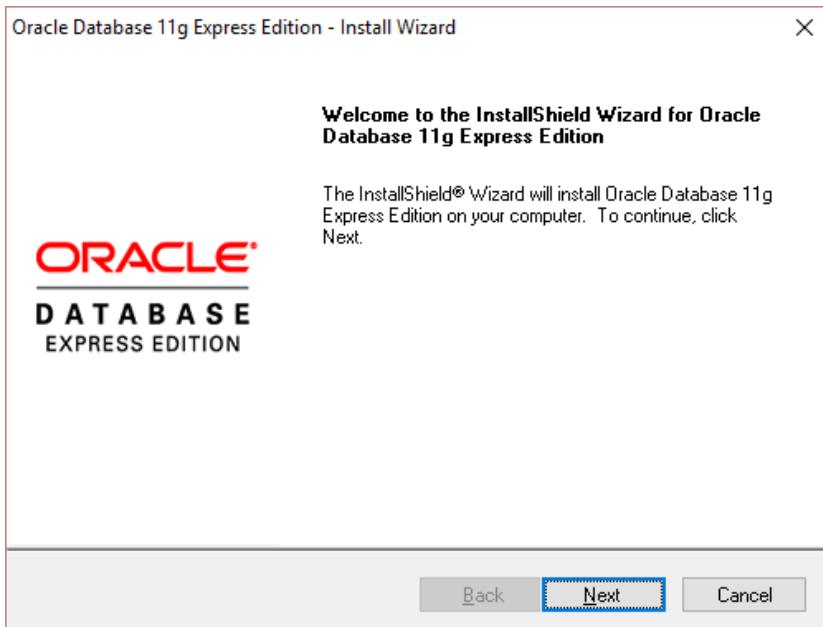
Tip: Una vez descomprimido, colocarlo en C.

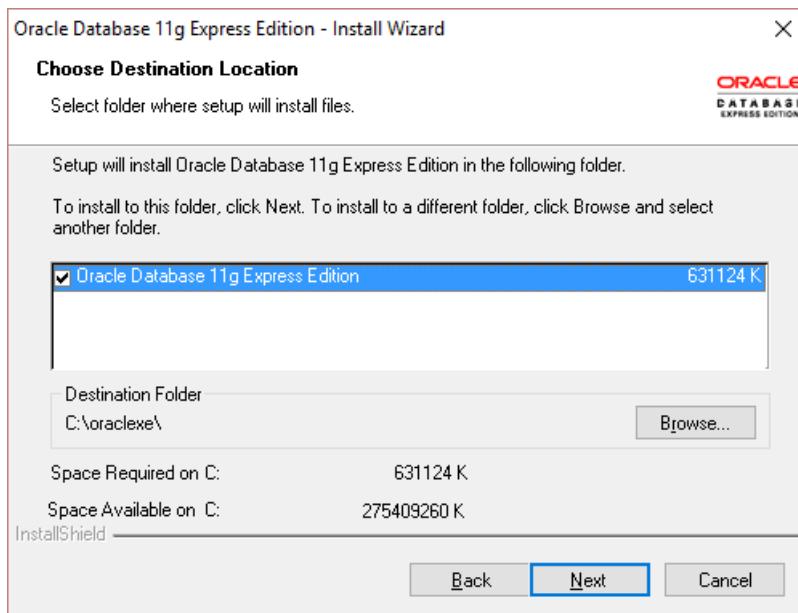


Instalar OracleXE

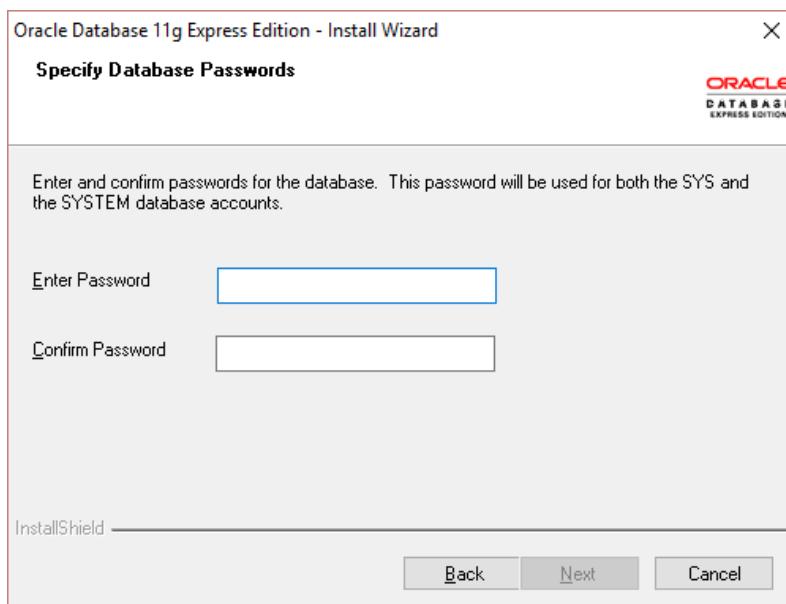
La instalación del motor de la base de datos, también es sencilla y se realiza mediante un asistente. No detallaré los pasos, pero pondré las imágenes como referencia.

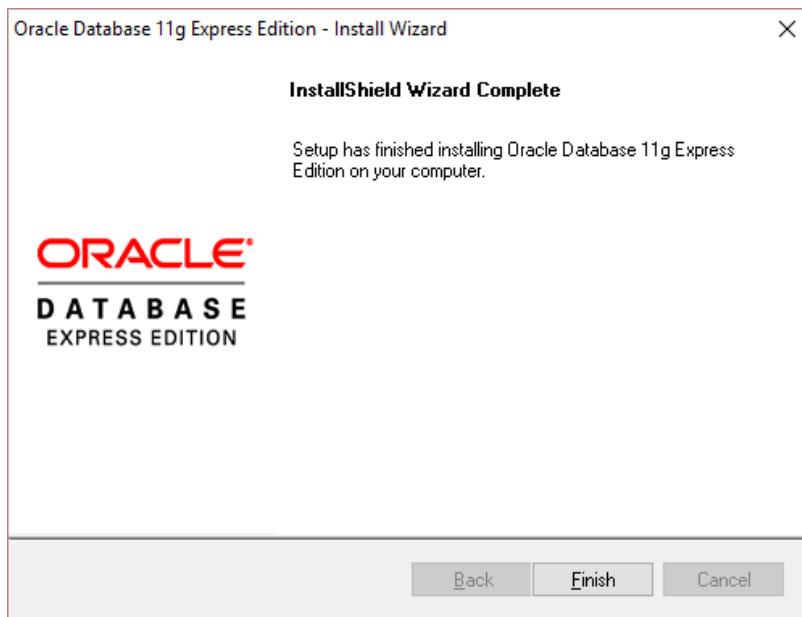






Tip: Introducimos una contraseña sencilla de recordar y mucho cuidado con mayúsculas y minúsculas, por ejemplo: **admin**





Listo, con eso hemos instalado el motor de la base de datos de Oracle. Lo siguiente será instalar el **IDE SQL Developer** para realizar la manipulación de datos mediante querys.

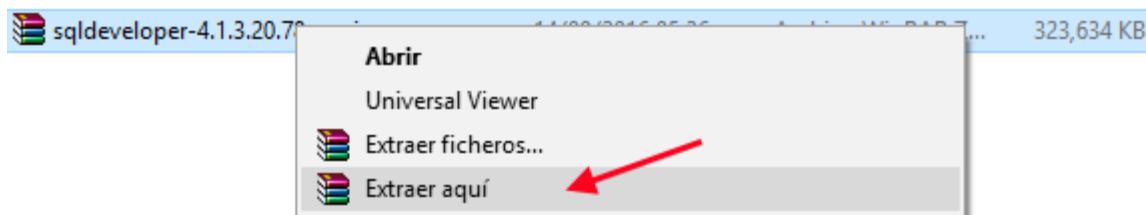
Tip: Por lo regular la contraseña SIEMPRE se olvida, así que guárdala en un lugar que siempre tengas accesible, por ejemplo, enviartela a tu correo o en un Google Keep.

Instalar SQL Developer

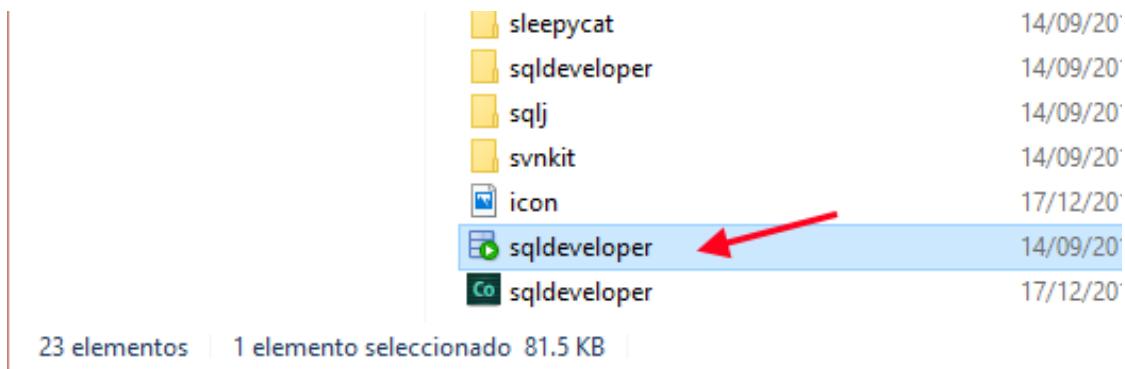
Para instalar el SQL Developer, **es requisito tener el JDK instalado** (ya que está hecho en java).

Mientras más reciente sea el SQL Developer, necesitará una versión más reciente del JDK (al momento de instalarlo te notificará mediante un mensaje de error, en caso de que no cumpla dicho requisito).

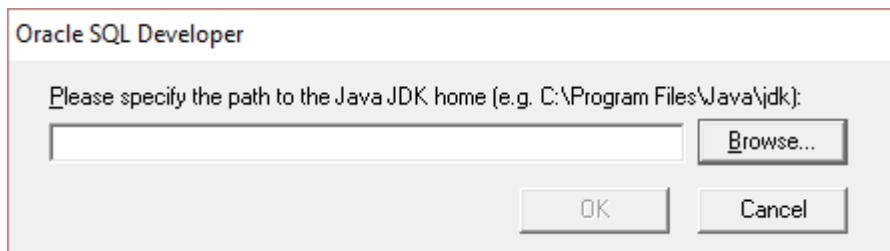
Paso 1: La instalación solo requiere descomprimir el archivo:



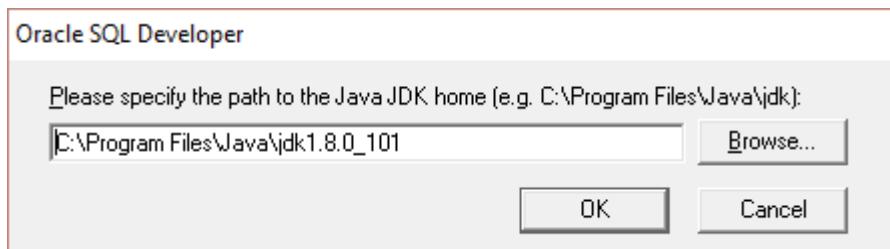
Paso 2: Luego buscamos el archivo SQL Developer y lo ejecutamos:



Paso 3: La primera vez te mostrará la siguiente ventana, solicitando la ruta del JDK; en caso de que no te lo solicite, quiere decir que descargaste un SQL Developer que ya lo tiene. (Por no seguir mis pasos)



Especificamos la ruta en donde instalamos el JDK en los pasos anteriores:



Tip: Te recomiendo copiar y pegar la ruta, desde la carpeta en donde esté instalado.

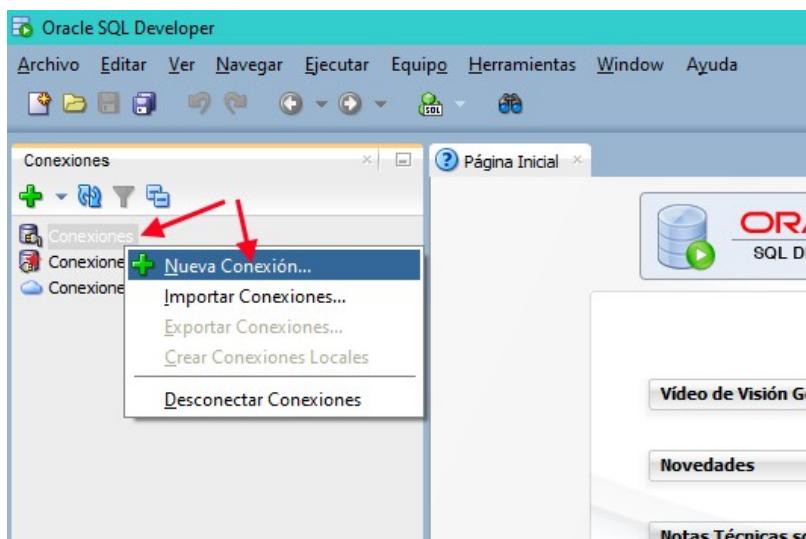
Capítulo 8. Oracle SQL Developer

Con esta herramienta nos conectaremos a la base de datos Oracle XE, además nos sirve para crear nuevos usuarios, tablas, secuencias, procedimientos, etc. También nos permite ejecutar querys y hacer debug de los procedimientos y funciones.

Creando una conexión

A continuación, crearemos una conexión con el usuario por default: SYSTEM, para luego crear un usuario personalizado.

Paso 1: Abrir Oracle SQL Developer y crear nueva conexión.



Paso 2:

Daremos un nombre a la conexión.

Introducimos el usuario SYSTEM (Oracle tiene por default SYSTEM y SYS)

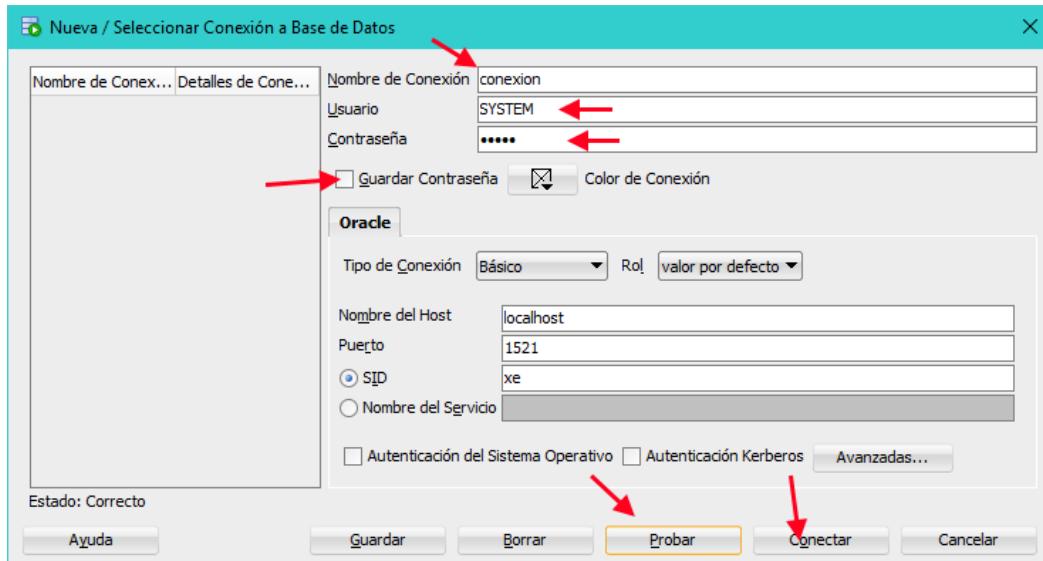
Proporcionamos la contraseña (la que proporcionaste en la instalación de ORACLE XE)

En los parámetros de Nombre del Host, Puerto y SID, revisar que tenga los mismos que se muestra en la imagen.

Damos clic en Probar, y nos debe de mostrar el mensaje: **Estado Correcto**. De lo contrario algo está mal capturado.

Si todo está bien, da clic en **Conectar**.

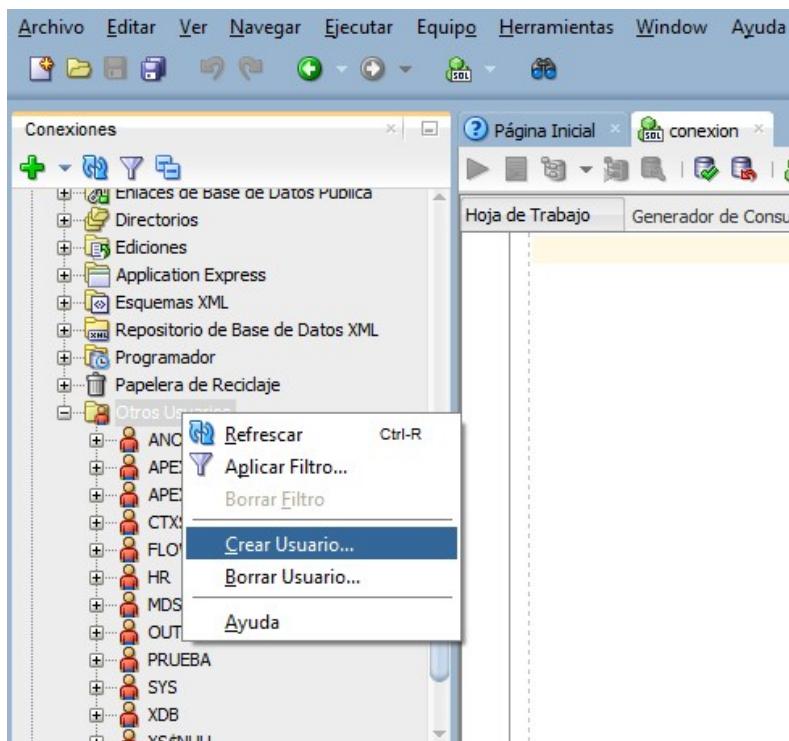
Tip: Mucho ojo con mayusculas y minusculas en la contraseña.



Crear usuario

Ahora que ya estamos conectados crearemos un usuario de base datos para el proyecto en JSF que veremos más adelante. Presento los pasos a continuación:

Paso 1: Ir al elemento *Otros Usuarios*, dar clic derecho y *Crear Usuario*:



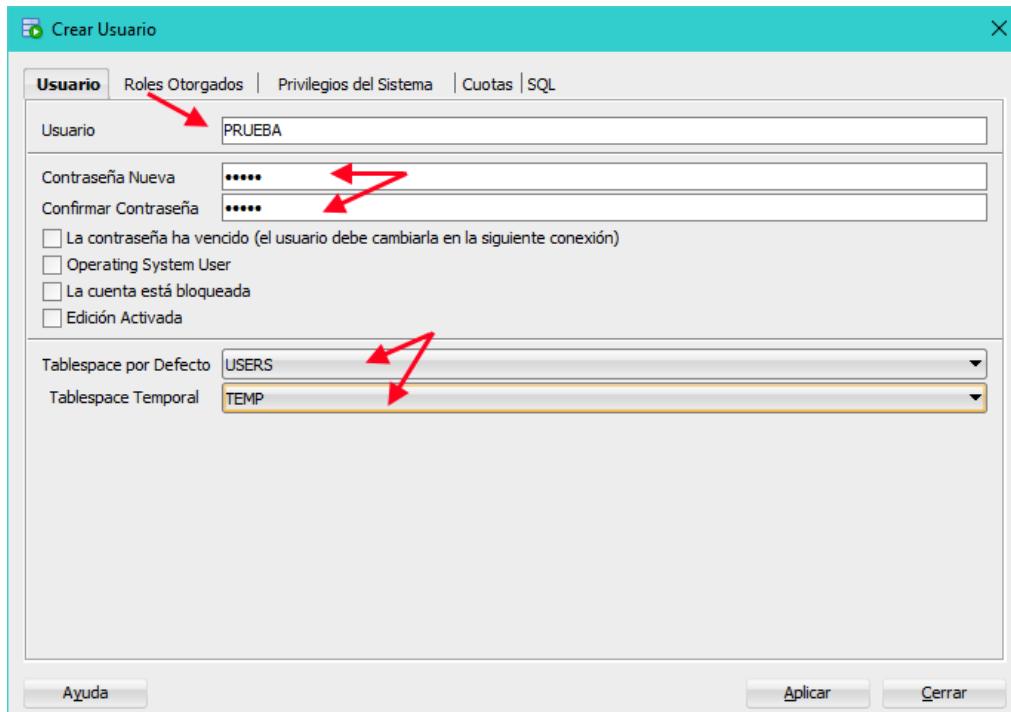
Paso 2:

Colocar en usuario la palabra PRUEBA

Crear una contraseña para este usuario (puede ser la misma: **admin**)

Seleccionar en Tablespace por defecto y Tablespace Temporal lo que se muestra en la imagen.

Nota: ¡Aún no des clic en Aplicar!, pasaremos a la pestaña Roles Otorgados

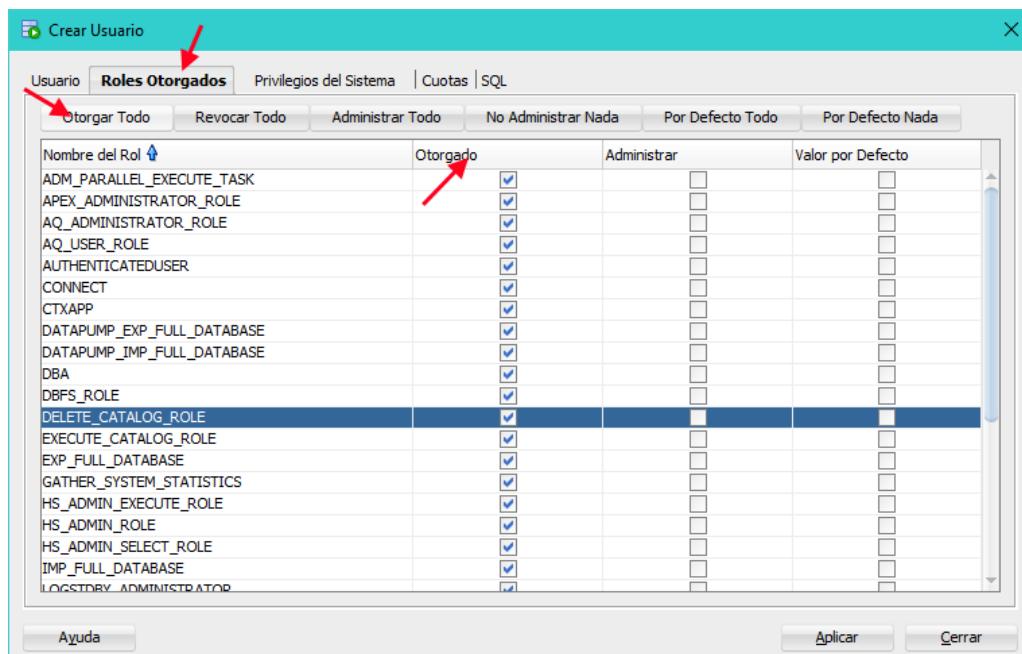


Paso 3:

Ir a la pestaña de Roles Otorgados

Seleccionar el botón Otorgar Todo

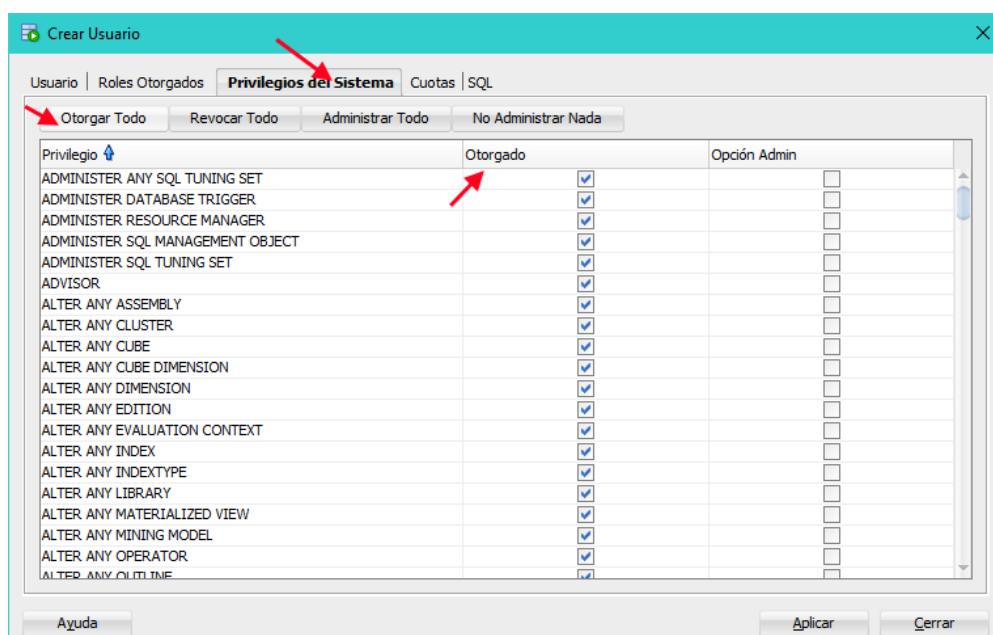
Luego iremos a la pestaña Privilegios del Sistema



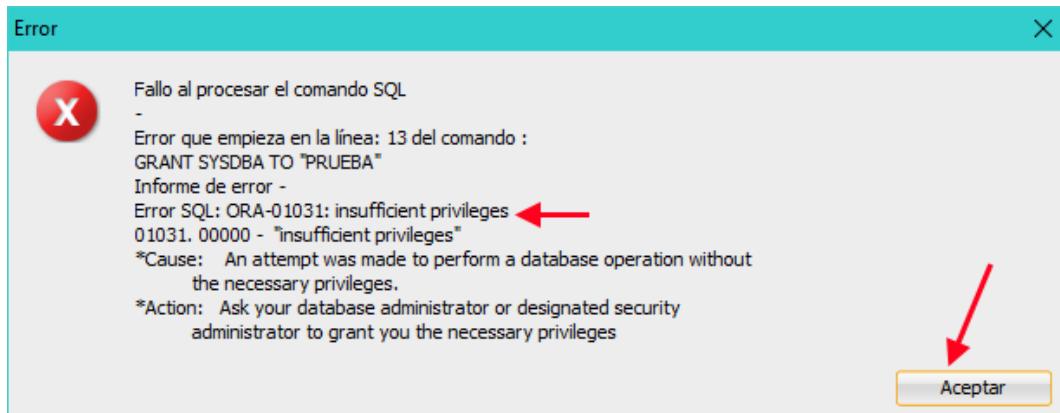
Paso 4:

En la pestaña de Privilegios del Sistema, daremos clic en el botón Otorgar Todo.

Nota: En esta pestaña hay ciertos privilegios, que el usuario con el que estamos conectados (SYSTEM), no puede asignar al crear nuevos usuarios, pero como son bastantes lo haremos de una forma “práctica”.



Al dar clic en Aplicar, se mostrará la siguiente ventana de error, lo cual solo nos indica lo que puse en la nota del paso anterior. Con esto el usuario PRUEBA se creó, pero sin algunos privilegios.



Sin embargo, si volvemos a revisar los privilegios del usuario creado, veremos que tiene solo algunos, que para nuestro caso son los mínimos necesarios, a continuación, sólo señalo algunos que no se le pudo otorgar:

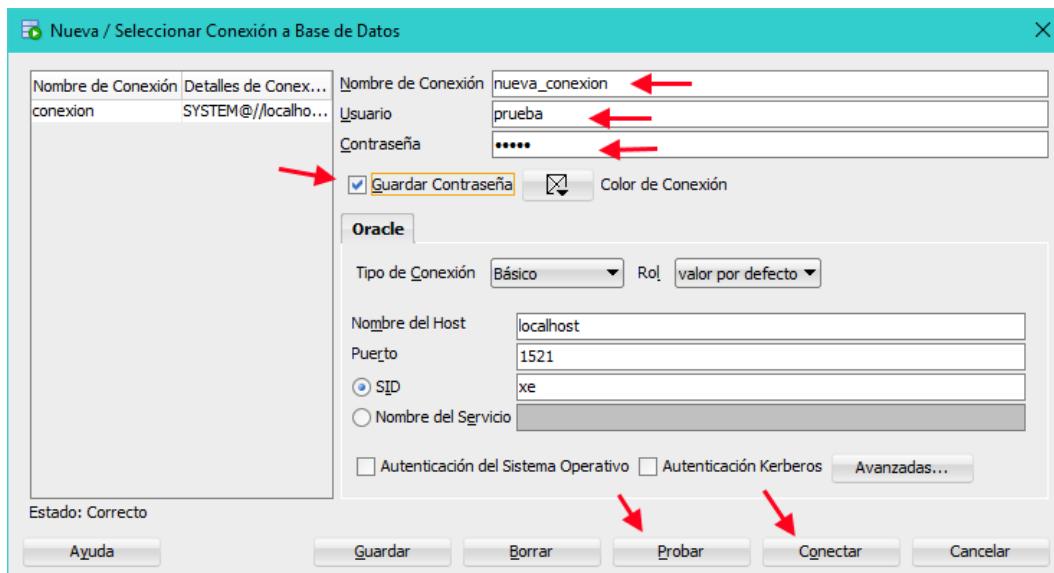
The screenshot shows the 'Editar Usuario' (Edit User) window with the 'Privilegios del Sistema' (System Privileges) tab selected. The interface includes tabs for 'Usuario', 'Roles Otorgados', 'Privilegios del Sistema' (highlighted in blue), 'Cuotas' (Quotas), and 'SQL'. Below the tabs are buttons for 'Otorgar Todo' (Grant All), 'Revocar Todo' (Revoke All), 'Administrar Todo' (Manage All), and 'No Administrar Nada' (Do Not Manage Anything). The main area displays a table of system privileges:

Privilegio	Otorgado	Opción Admin
ADMINISTER ANY SQL TUNING SET ←	<input type="checkbox"/>	<input type="checkbox"/>
ADMINISTER DATABASE TRIGGER	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ADMINISTER RESOURCE MANAGER ←	<input type="checkbox"/>	<input type="checkbox"/>
ADMINISTER SQL MANAGEMENT OBJECT ←	<input type="checkbox"/>	<input type="checkbox"/>
ADMINISTER SQL TUNING SET	<input type="checkbox"/>	<input type="checkbox"/>
ADVISOR	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ALTER ANY ASSEMBLY	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ALTER ANY CLUSTER	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ALTER ANY CUBE	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ALTER ANY CUBE DIMENSION	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ALTER ANY DIMENSION	<input type="checkbox"/>	<input type="checkbox"/>
ALTER ANY EDITION	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ALTER ANY EVALUATION CONTEXT	<input type="checkbox"/>	<input type="checkbox"/>
ALTER ANY INDEX	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ALTER ANY INDEXTYPE	<input type="checkbox"/>	<input type="checkbox"/>
ALTER ANY LIBRARY	<input type="checkbox"/>	<input type="checkbox"/>
ALTER ANY MATERIALIZED VIEW	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ALTER ANY MINING MODEL	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ALTER ANY OPERATOR	<input type="checkbox"/>	<input type="checkbox"/>
ALTER ANY OLAP TIME	<input checked="" type="checkbox"/>	<input type="checkbox"/>

At the bottom of the window are buttons for 'Ayuda' (Help), 'Aplicar' (Apply), and 'Cerrar' (Close).

Crear nueva conexión con el nuevo usuario

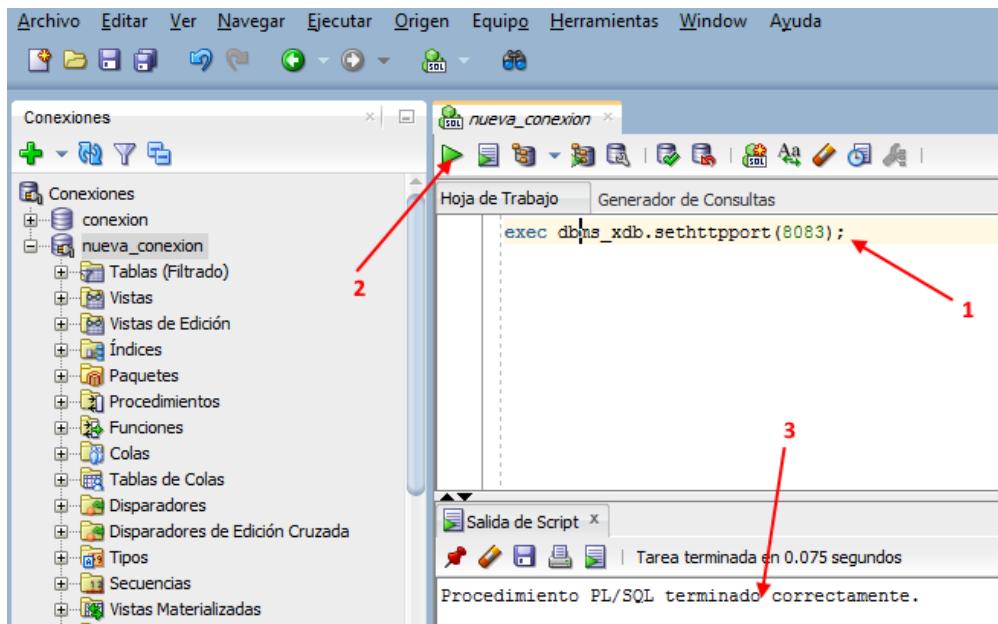
Ahora crearemos una nueva conexión con el usuario PRUEBA, para verificar la contraseña asignada y crear una tabla en la base de datos.



Cambiar el puerto de OracleXE

Este paso es necesario ya que la herramienta Tomcat también ocupa el puerto 8080, entonces lo que haremos será cambiar el puerto de OracleXE a un puerto diferente, en este caso el 8083 (puede ser cualquier otro puerto disponible).

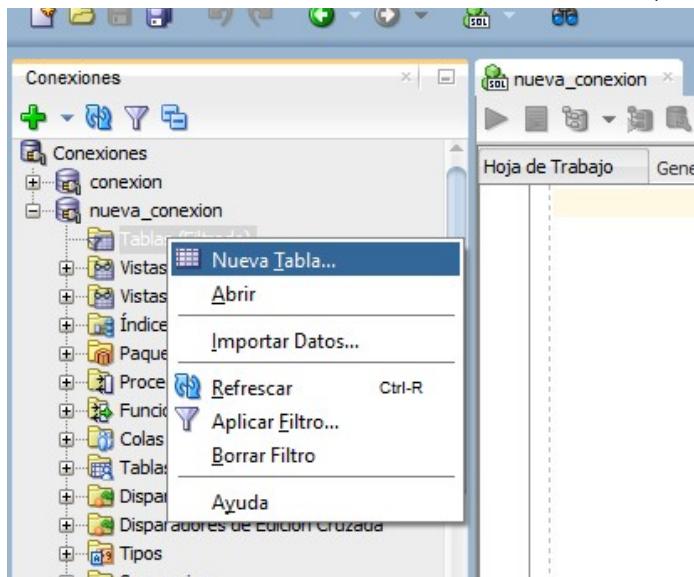
Ejecutamos un procedimiento de Oracle: `exec dbms_xdb.sethttpport(8083);`



Crear una tabla

Si ya tenemos nuestra nueva conexión, procedemos a crear una tabla:

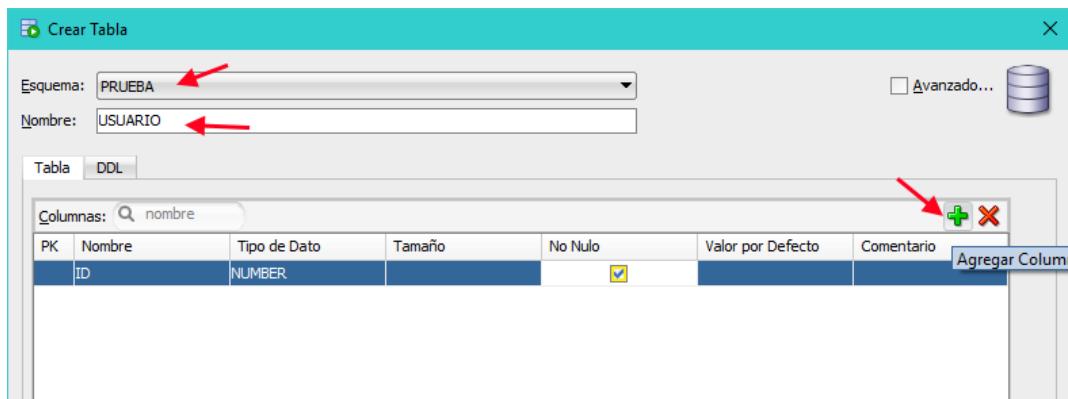
Paso 1: Damos *clic derecho* en la sección Tablas, luego “**Nueva Tabla**”



Paso 2:

El nombre de la tabla será **USUARIO**

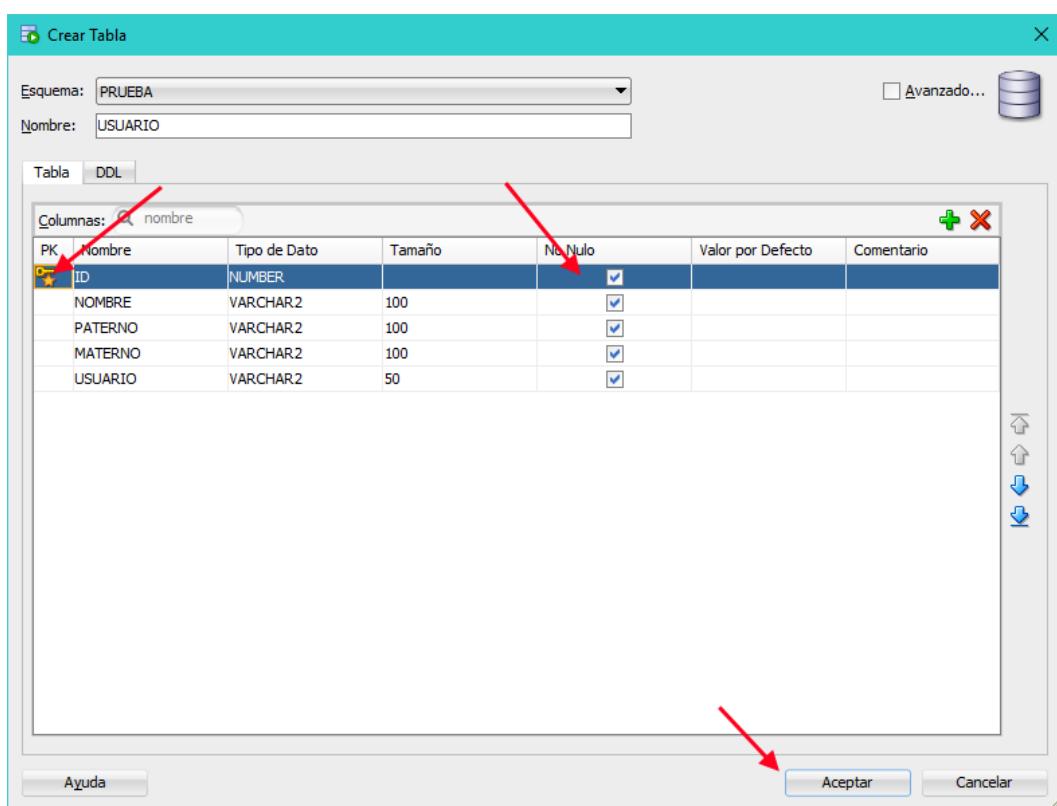
Con el símbolo de +, agregaremos columnas



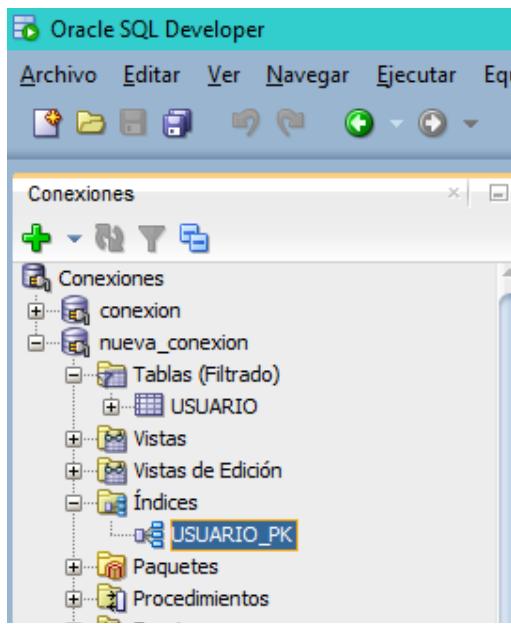
Paso 3:

Crear las columnas: ID, NOMBRE, PATERNO, MATERNO y la columna USUARIO.

- 1.- Seleccionamos la columna ID y damos clic en donde dice PK. Con esto hacemos esta columna la llave primaria.
- 2.- Todas las columnas las pondremos como No nulo
- 3.- Damos clic en Aceptar

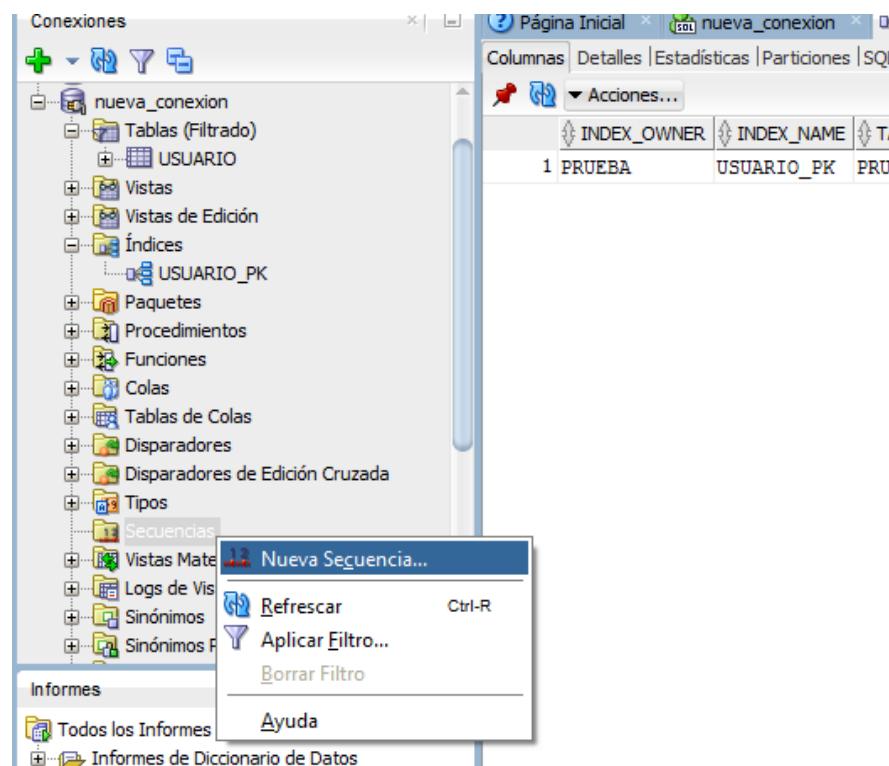


Podrás ver en la siguiente imagen que se generó el índice de la llave primaria (**USUARIO_PK**) de la columna ID:



Paso 4:

Crearemos una secuencia (*este componente lleva el control de la secuencia, es decir, el siguiente número a insertar para la columna primaria ID*) para la tabla USUARIO, damos clic derecho sobre la sección Secuencia, luego Nueva Secuencia:



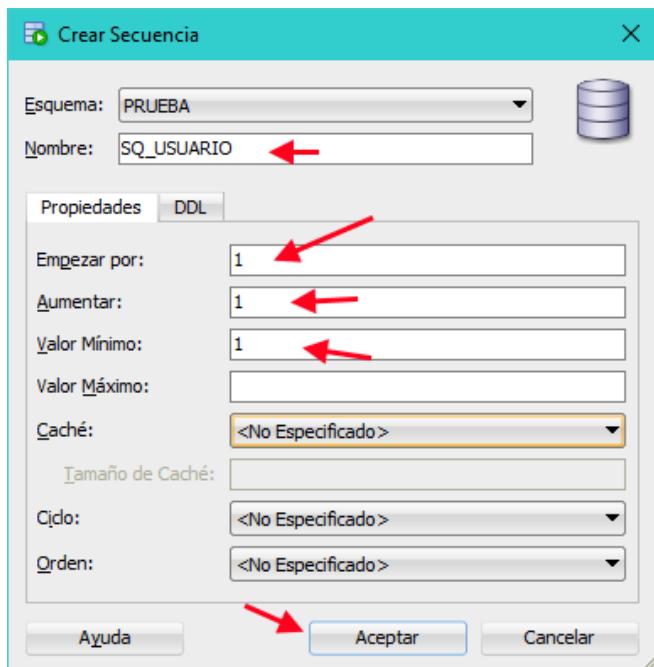
Paso 5:

Se nos mostrará la siguiente ventana, en donde daremos nombre a la secuencia.

SQ_USUARIO

Inicializamos los campos en 1 que marcó en la imagen

Damos clic en Aceptar



Realizamos una consulta de prueba, y veremos que en la parte de abajo en la sección Resultado de la consulta nos muestra las columnas sin datos, lo cual es correcto ya que no hemos insertado información.

The screenshot shows the MySQL Workbench interface. In the SQL editor, the query 'select * from prueba.USUARIO;' is entered. Below it, the 'Resultado de la Consulta' (Result Set) tab shows an empty table with columns ID, NOMBRE, PATERNO, MATERNO, and USUARIO.

ID	NOMBRE	PATERNO	MATERNO	USUARIO

Capítulo 9. Crear espacio de trabajo

El espacio de trabajo o workspace es en donde tendremos tantos proyectos como necesitemos.

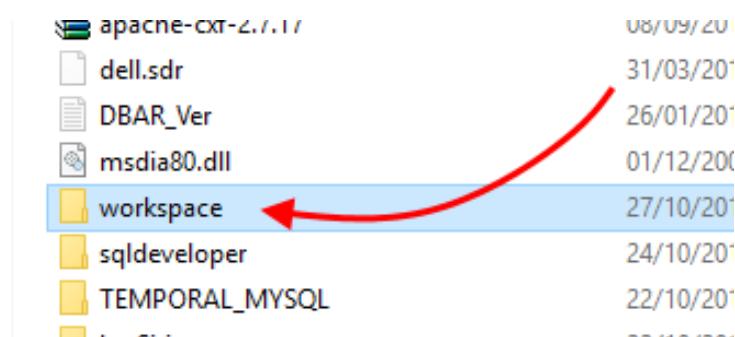
En caso de que llegues a tener muchos, puedes crear otro workspace (Ejemplo: C:\workspace_2) y allí crear más proyectos.

Es importante mencionar que no puedes ver proyectos de un workspace_1 desde otro workspace_2.

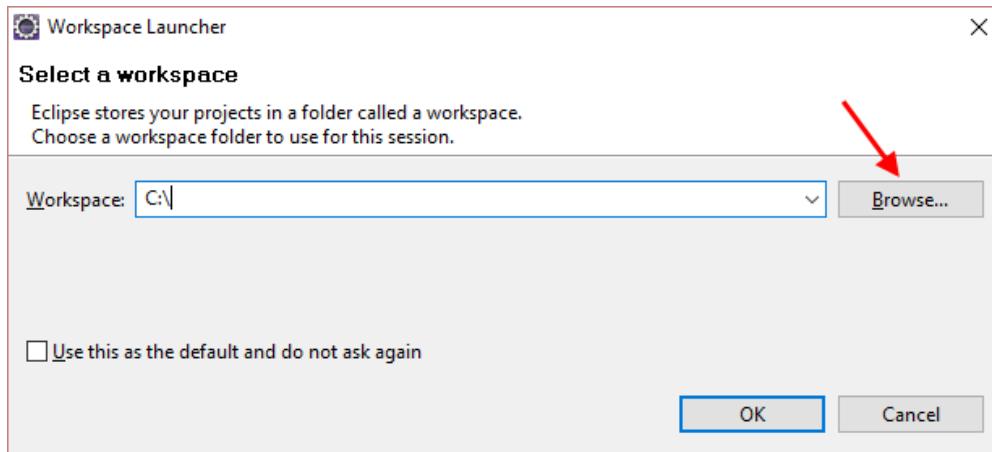
Tip: Si quieres conocer un poco más acerca de los botones y espacios más usados del entorno Eclipse, puedes ir al [Apéndice E](#)

Lo que sí puedes hacer es abrir dos Eclipse, y cada uno con su respectivo workspace, al mismo tiempo. A continuación los pasos:

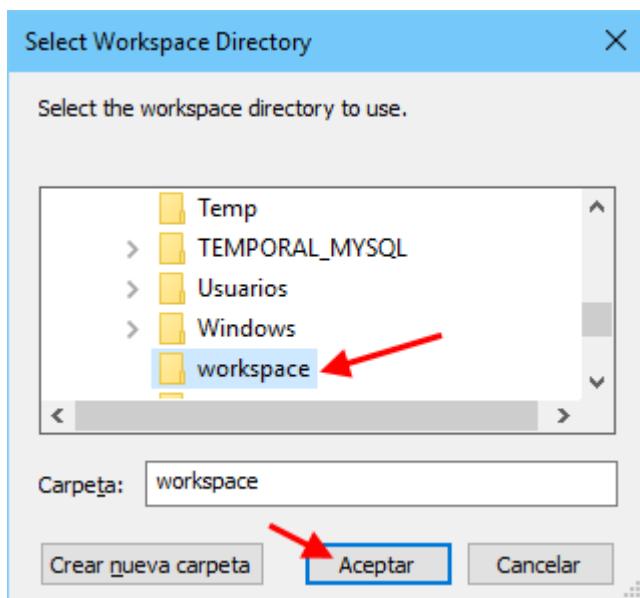
Paso 1: Primero que nada crearemos una carpeta en C:\, llamada **workspace**. Recomiendo crearla en raíz de cualquier unidad que tengas disponible.



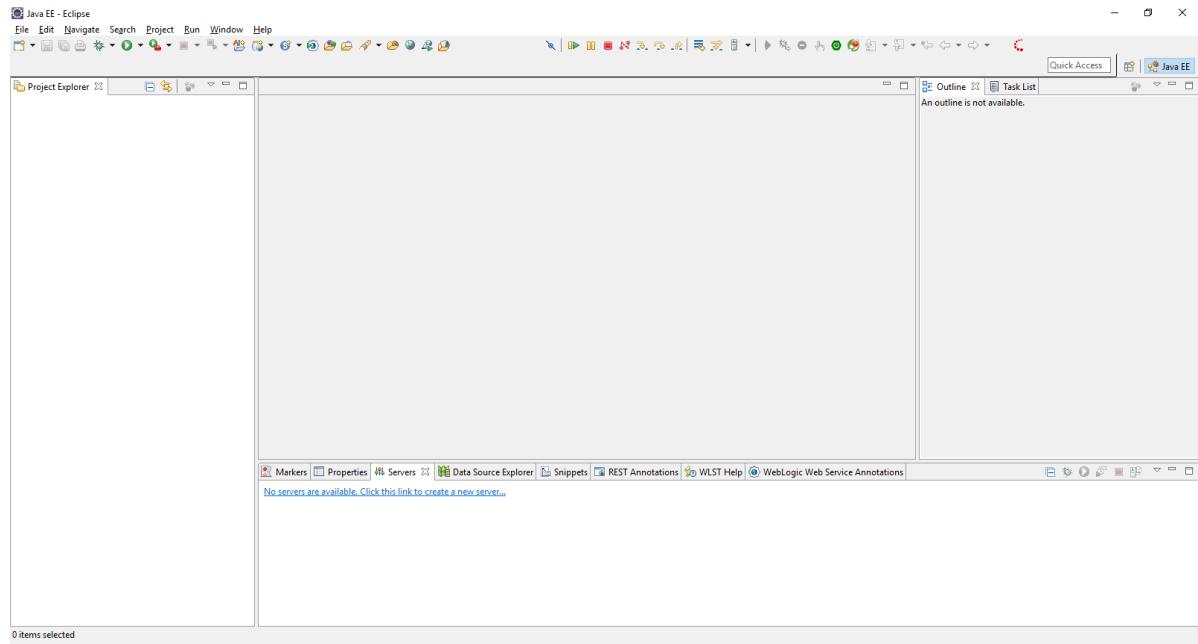
Paso 2: Ejecutamos el archivo **eclipse.exe** (localizado en C:\eclipse), nos saldrá una ventana como la siguiente:



Paso 3: Damos clic en **Browse**, y buscamos la carpeta workspace, la seleccionamos y damos clic en **OK**.



Al seleccionar el espacio de trabajo, se te mostrará algo como esto:



Capítulo 10. Configurar Apache Tomcat

Como ya lo he mencionado al inicio del libro, esta herramienta nos sirve para desplegar nuestro proyecto web y poder usarlo en el navegador web.

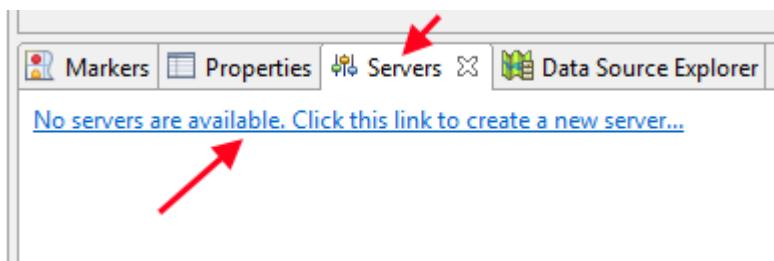
Nota: En el [Apéndice C](#), se detalla la configuración a la base de datos.

Hay dos formas de usar Apache Tomcat:

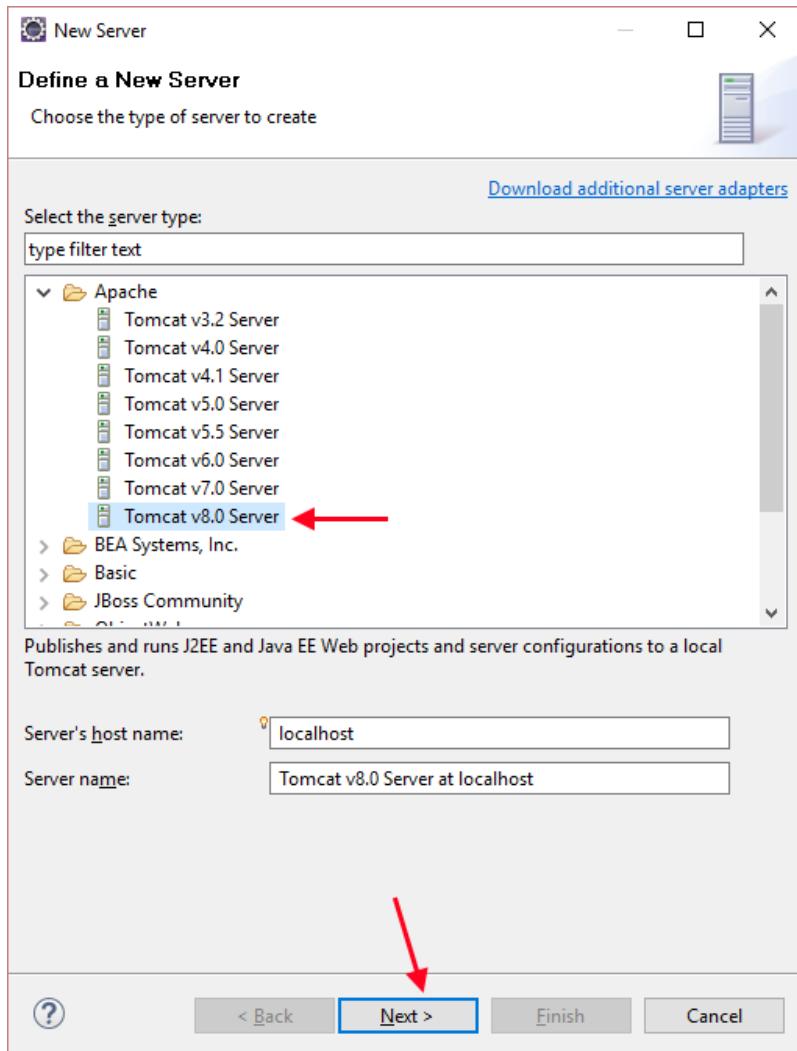
1. Poner nuestro proyecto web (archivo .war) en la carpeta *webapps* que está en donde tengamos instalado tomcat, y luego “iniciar” Tomcat con el archivo *startup.bat* que está en la carpeta *bin*. (Para uso de ambiente productivo)
2. Desde Eclipse, para propósito de desarrollo esta es la mejor opción, ya que te crea un “copia” ligera de *Apache Tomcat*, y puedes tener tantas como necesites, cada una con diferente configuración. (Para uso de ambiente de desarrollo)

Lo que nos interesa el segundo punto, con lo cual detallo los pasos a continuación:

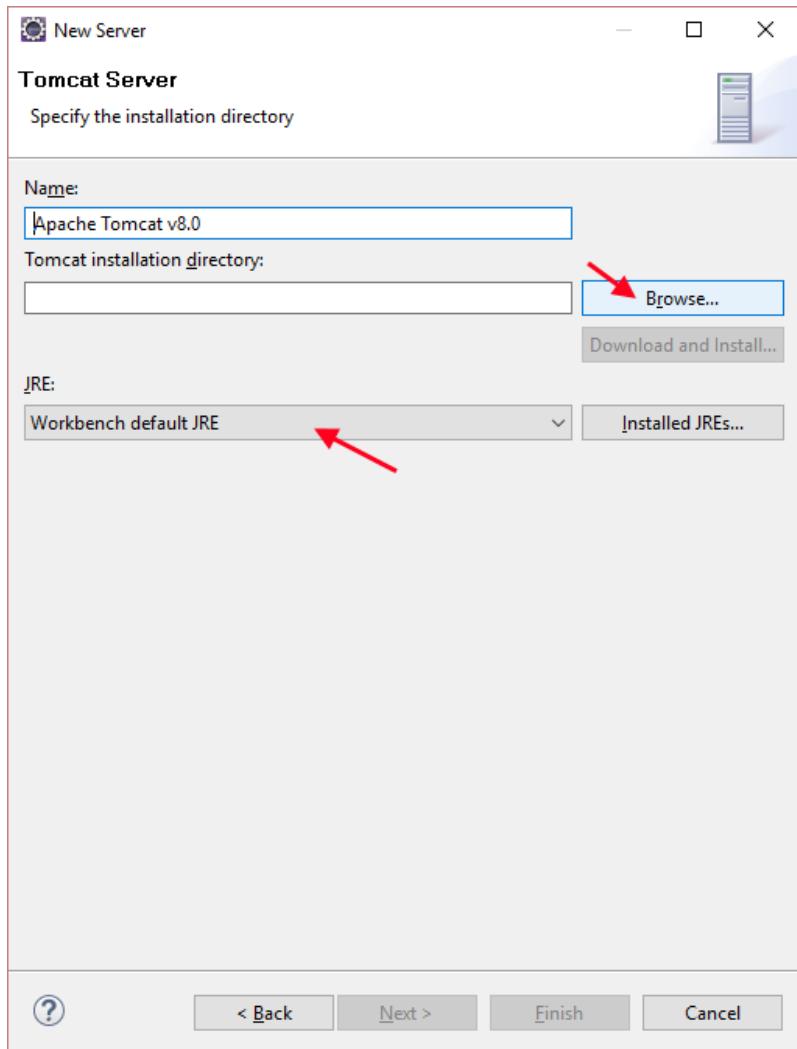
Paso 1: Ir a la parte inferior en donde esta una pestaña que se llama **Servers**, damos clic en las **letras azules** para crear un nuevo servidor:



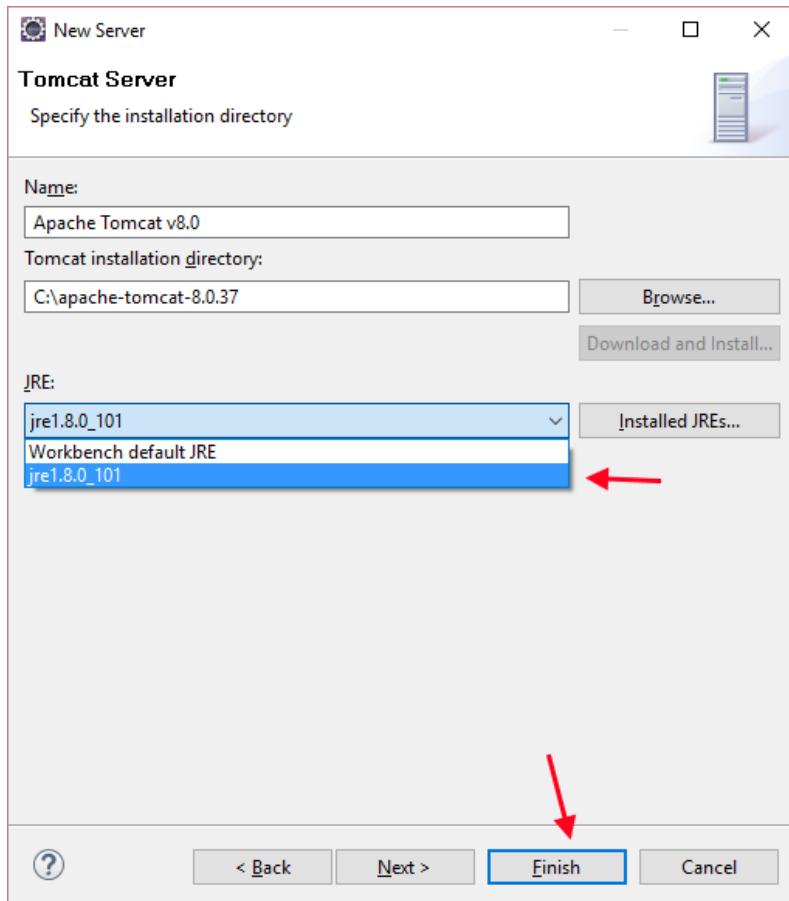
Paso 2: Se nos muestra la siguiente ventana, en donde seleccionaremos la versión de Tomcat que descargamos e instalamos en pasos previos y damos clic en Next:



Paso 3: En esta ventana podemos asignar un nombre (alias) al servidor o lo puedes dejar como aparece (Apache Tomcat v8.0), damos clic en **Browse** para que se abra el explorador de archivos.



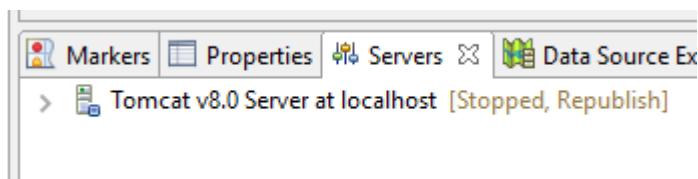
Paso 4: Seleccionamos la ruta donde instalamos Tomcat en nuestro equipo y en el combo de JRE elegimos la versión de JRE con que funcionará el servidor de aplicaciones y damos clic en Finish.



Una vez creada la instancia de Tomcat, ahora verás en la **parte izquierda**, en la pestaña de **Project Explorer** se mostrará una carpeta llamada Servers, en dicha carpeta podrás encontrar los archivos server.xml, context.xml y otros archivos de configuración.



En la parte inferior de Eclipse, en la pestaña de **Servers**, podrás ver que se agregó el servidor de aplicaciones, esta pestaña es muy importante ya que la usaremos para “*iniciar/arrancar*” y detener el proyecto web:



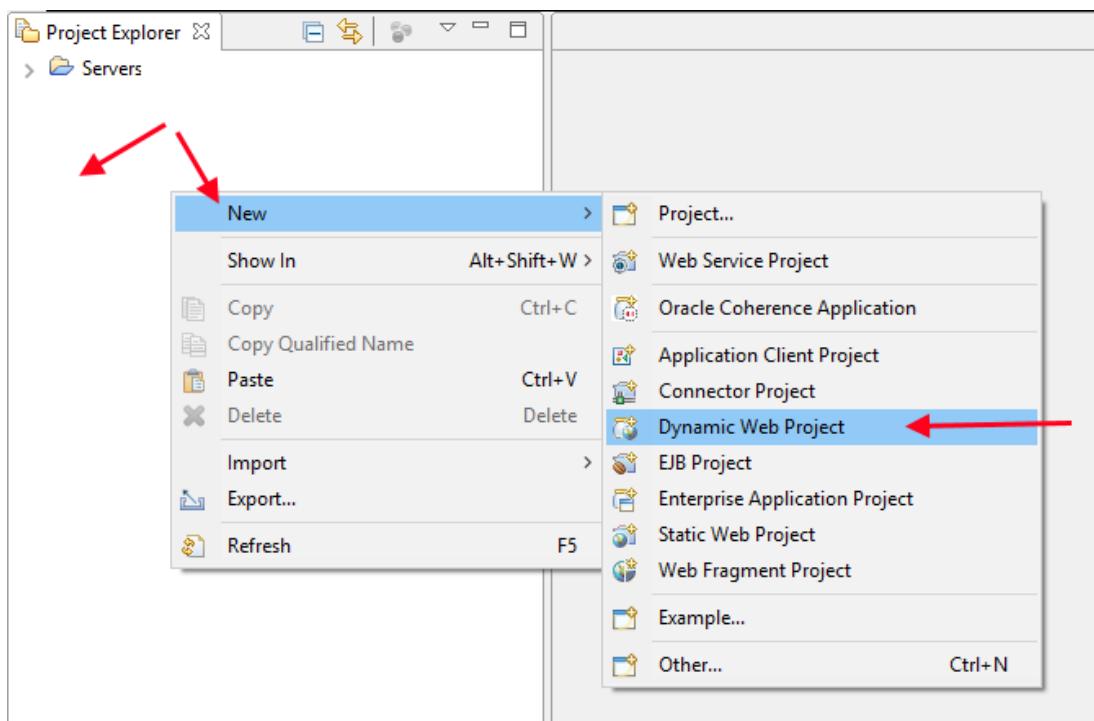
Nota: Es importante que realices el [Apéndice C](#) antes de continuar, para configurar la conexión a la base de datos.

Capítulo 11. Crear un proyecto web

Los siguientes pasos te servirán para crear un proyecto web desde cero. La mayoría se crean así, en los proyectos con desarrolladores más experimentados, los crean con Maven u otra herramienta similar. Al final de cuentas hacen lo mismo.

Antes de crear el proyecto es importante que instales plugin **Jboss para Eclipse**, para eso debes ir al [Apéndice A](#) y también debes configurar **Maven**, en el [Apéndice D](#) lo encontrarás.

Paso 1: Ya que tenemos nuestro espacio de trabajo (carpeta workspace), lo siguiente es **crear un proyecto web dinámico**, para ello damos **clic derecho** en cualquier lugar del espacio blanco de la pestaña de Project Explorer, seleccionamos New y posteriormente **Dynamic Web Project**:



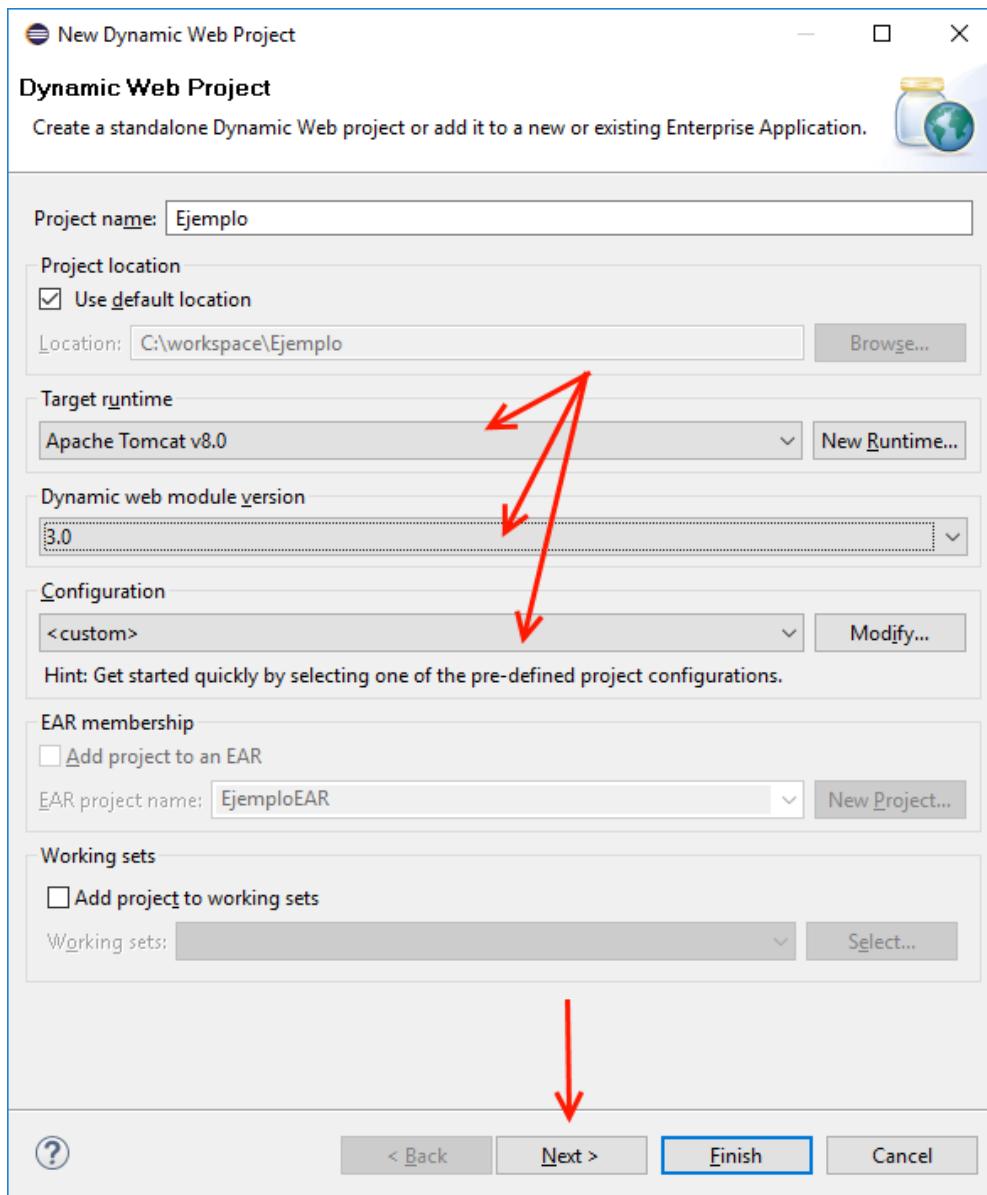
Paso 2: En la siguiente ventana le damos un nombre al proyecto, en este caso Ejemplo, además de configurar los siguientes atributos:

Target runtime, seleccionar la instancia del tomcat que creamos en el paso anterior.

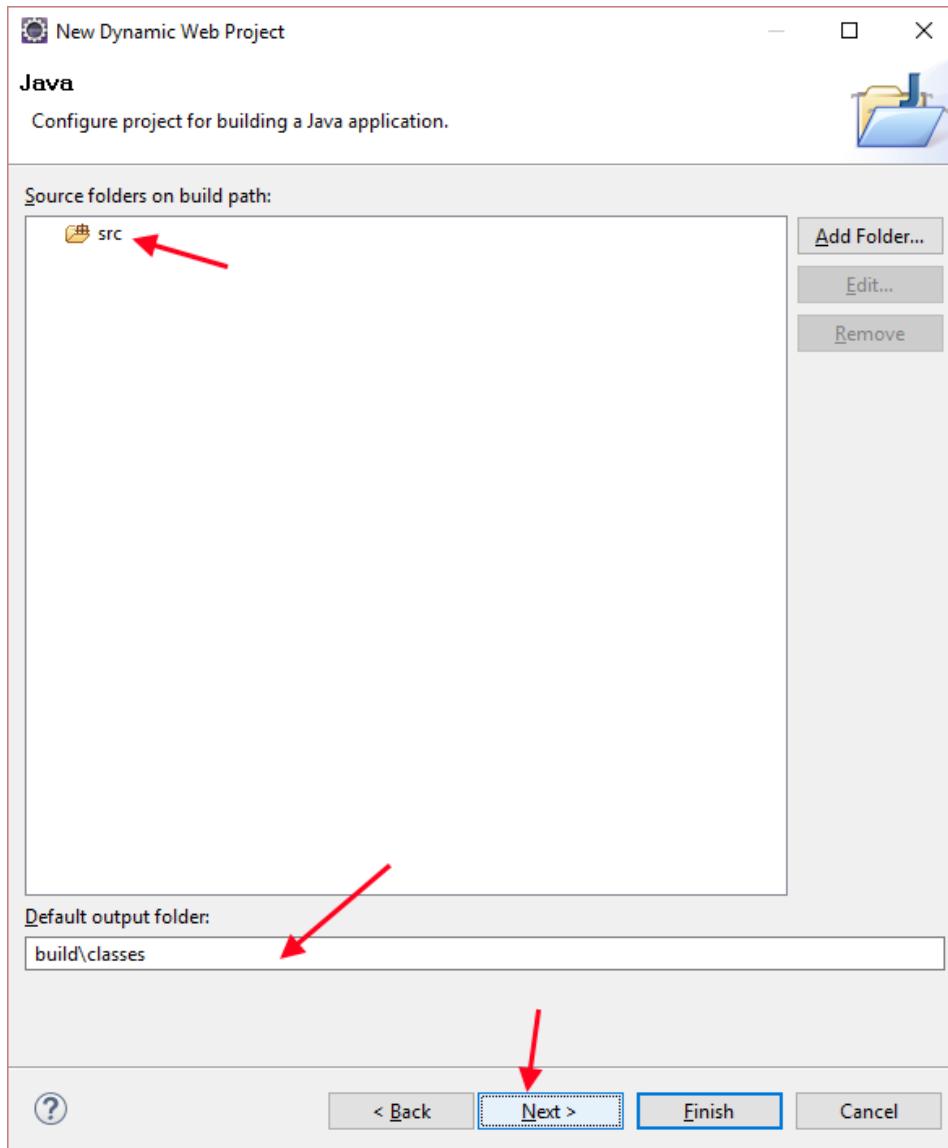
Dynamic web module version, seleccionar la versión 3.0

Configuration, seleccionar custom.

Dar clic en **Next**



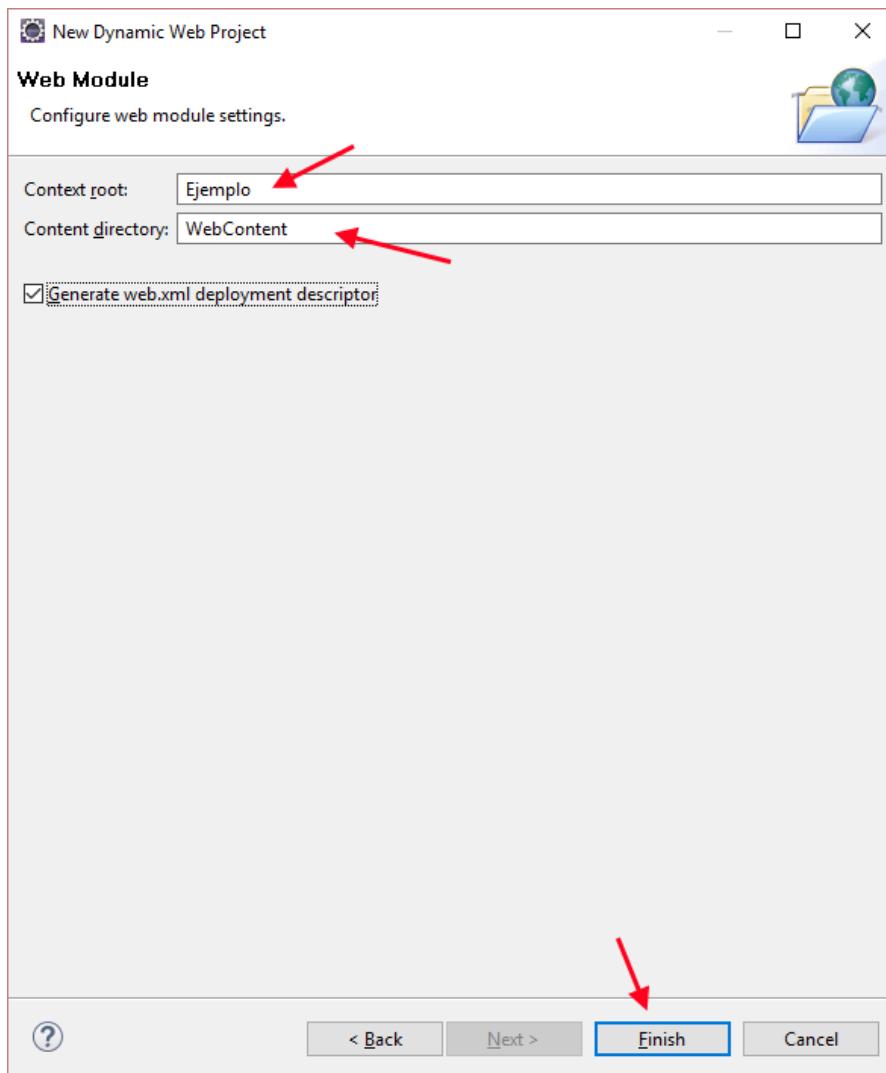
Paso 3: En esta ventana se nos presenta la carpeta (src) en donde Eclipse buscará las clases java, para compilar y colocarlas en la carpeta build/classes. No es necesario hacer modificaciones, lo dejamos como lo presenta y damos clic en **Next**.



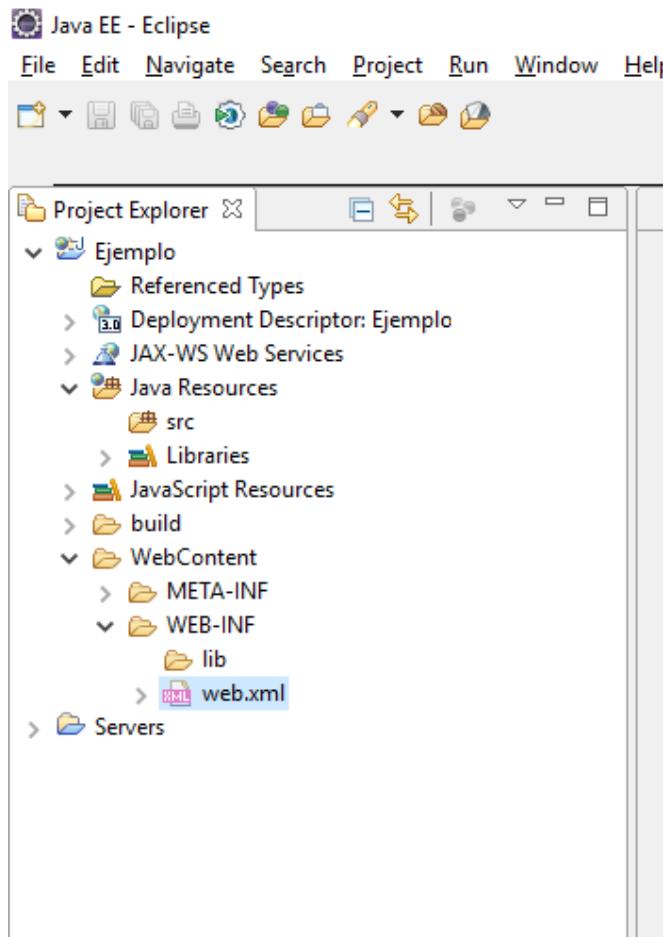
Paso 4: En la siguiente ventana, se muestra el **nombre del proyecto** que se mostrará en la URL del navegador cuando inicies el proyecto (Ej. localhost:8080/**Ejemplo**).

La carpeta WebContent en donde se encuentra el contenido como los jsp, xhtml, javascript, WEB-INF, etc.

Damos clic en **Finish**.



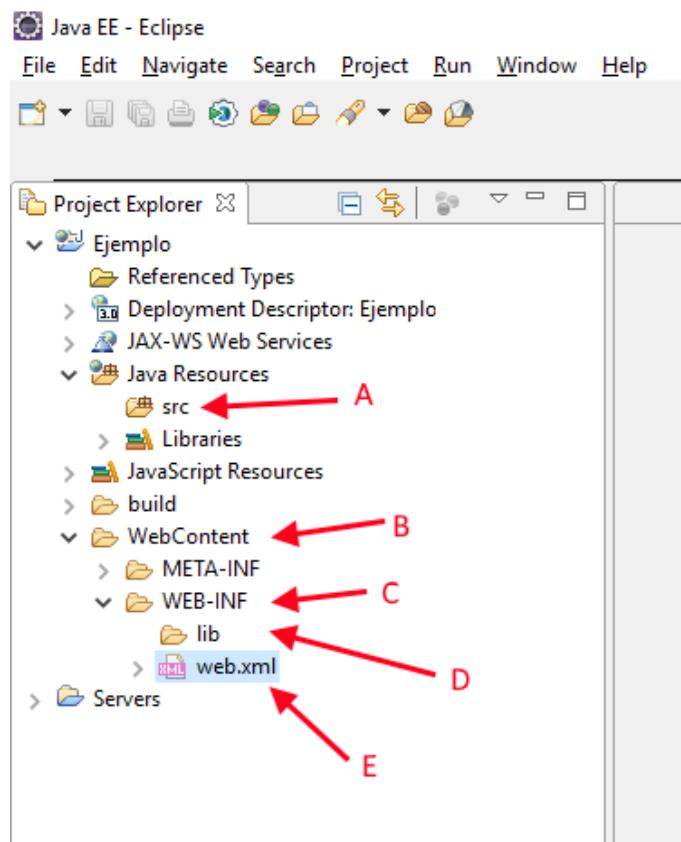
Deberás ver algo como lo siguiente:



Y con esto terminamos de crear el “cascarón” de un proyecto web, ahora lo siguiente es usar algún framework para desarrollar, pero primero explicaré brevemente la estructura del proyecto.

Capítulo 12. Estructura de un proyecto web

En la siguiente imagen se muestra el proyecto web que acabamos de crear, en la mayoría de los casos, se presenta así:



Hago hincapié en que la mayoría se muestra así, ya que hay otras maneras de crear proyectos web, por ejemplo, con Maven, en cual la estructura de las carpetas varía, pero es muy similar.

De la anterior imagen que presento, remarco las siguientes secciones, que más interesan al momento del desarrollo:

- **src**: aquí se colocan las clases de *java* y en ocasiones archivos con extensión *.properties*, y que por lo general se agrupan en paquetes (que son básicamente carpetas)
- **WebContent**: en esta carpeta se ponen los archivos CSS, JS, XHTML o JSP. Los archivos XHTML y JSP es donde va el código HTML. Dichos archivos contienen etiquetas propias del framework que sirven para manipular el HTML mostrado.
Ejemplo:

```
> <body> <!-- Elementos HTML -->
>   <h:form> <!-- Elementos de JSF -->
>     <table>
>       <tr>
>         <td>
>           <h:outputText value="Usuario:" />
>         </td>
>         <td>
>           <h:inputText value="#{loginView.usuario}" />
>         </td>
>       </tr>
>       <tr>
>         <td>
>           <h:outputText value="Contraseña:" />
>         </td>
>         <td>
>           <h:inputText value="#{loginView.contraseña}" />
>         </td>
>       </tr>
>     </table>
>   </h:form>
> </body>
```

- **WEB-INF:** en esta carpeta se colocan los archivos de configuración (**por lo general archivos con extensión .xml**), como lo es el web.xml, en caso de usar frameworks, aquí puedes encontrar sus respectivos archivos XML de configuración. Esta carpeta no puede ser accedida desde la url del navegador.
- **lib:** en esta carpeta se colocan los archivos .jar que se vayan a usar en el proyecto, puede ser desde frameworks hasta librerías con un propósito específico (por ejemplo, se puede poner archivos .jar para la carga y descarga de archivos). Los elementos de esta carpeta no pueden ser accedidos desde la url del navegador web.
- **Archivo web.xml:** en este archivo se coloca la mayoría de la configuración, aquí por ejemplo, le indicas que framework se usará, la página de bienvenida, filtros, etc.

Capítulo 13. Un vistazo a Java Server Faces

Usaremos el framework de Java Server Faces (JSF) como un primer vistazo a esta tecnología.

Librerías de JSF

Las librerías o archivos .jar los descargamos de un sitio que contiene casi todas las librerías necesarias para desarrollo en java: www.mvnrepository.com

A continuación, muestro los pasos para descargar:

Paso 1: Buscar **JSF 2.2**, que al momento de escribir este libro es la más reciente y damos clic en los links que señalo abajo:

The screenshot shows the Maven Repository search interface. A red box highlights the search bar containing 'jsf 2.2'. Below the search bar, the text 'Found 1097 results' is displayed. Underneath, there are sorting options: 'Sort: relevance | popular | newest'. Two results are listed, each with a red arrow pointing to it from the text above:

- 1. Oracle's Implementation Of The JSF 2.2 Specification API.**
com.sun.faces » jsf-api
This is the master POM file for Oracle's Implementation of the JSF 2.2 Specification.
Last Release on Nov 14, 2016
- 2. Oracle's Implementation Of The JSF 2.2 Specification.**
com.sun.faces » jsf-impl
This is the master POM file for Oracle's Implementation of the JSF 2.2 Specification.
Last Release on Nov 14, 2016

Each result entry includes a small 'Sun ORACLE' logo, the package name, the artifact ID, a brief description, and the last release date. To the right of each entry are usage statistics ('319 usages' and 'GPL | CDDL') and a link to the repository page.

Paso 2: El primer link es el de **JSF-API**, descargamos la versión **2.2.14** que señalo en la imagen.

Home » com.sun.faces » jsf-api

Oracle's Implementation Of The JSF 2.2 Specification API.

This is the master POM file for Oracle's Implementation of the JSF 2.2 Specification.

License	CDDL GPL
Tags	jsf api
Used By	319 artifacts

Version	Repository	Usages
2.2.14	Central	4
2.2.13	Central	16



Y posteriormente la librería **JSF-IMPL** que también indico en la siguiente imagen, con esto tenemos las librerías necesarias para usar Java Server Faces.

Nota: En el caso de **JSF-IMPL**, tiene varios repositorios, elegir **Central**.

Oracle's Implementation Of The JSF 2.2 Specification.

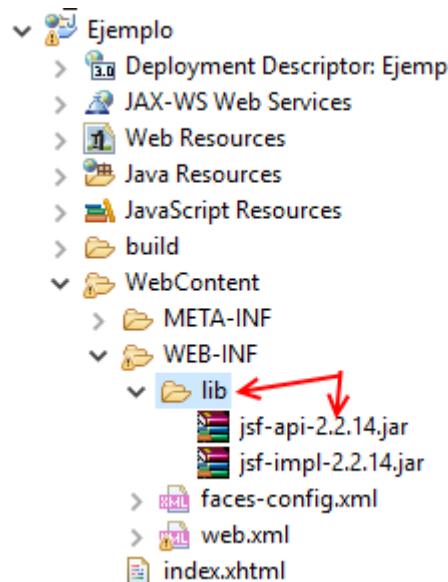
This is the master POM file for Oracle's Implementation of the JSF 2.2 Specification.

License	CDDL GPL
Tags	jsf
Used By	165 artifacts

Version	Repository	Usages	Date
2.2.14	Central	1	(Nov, 2016)
2.2.13.SP4	JBoss 3rd-party	0	(Jun, 2017)
2.2.13.SP3	JBoss 3rd-party	3	(Jul, 2016)
2.2.13.SP2	JBoss 3rd-party	0	(Jul, 2016)

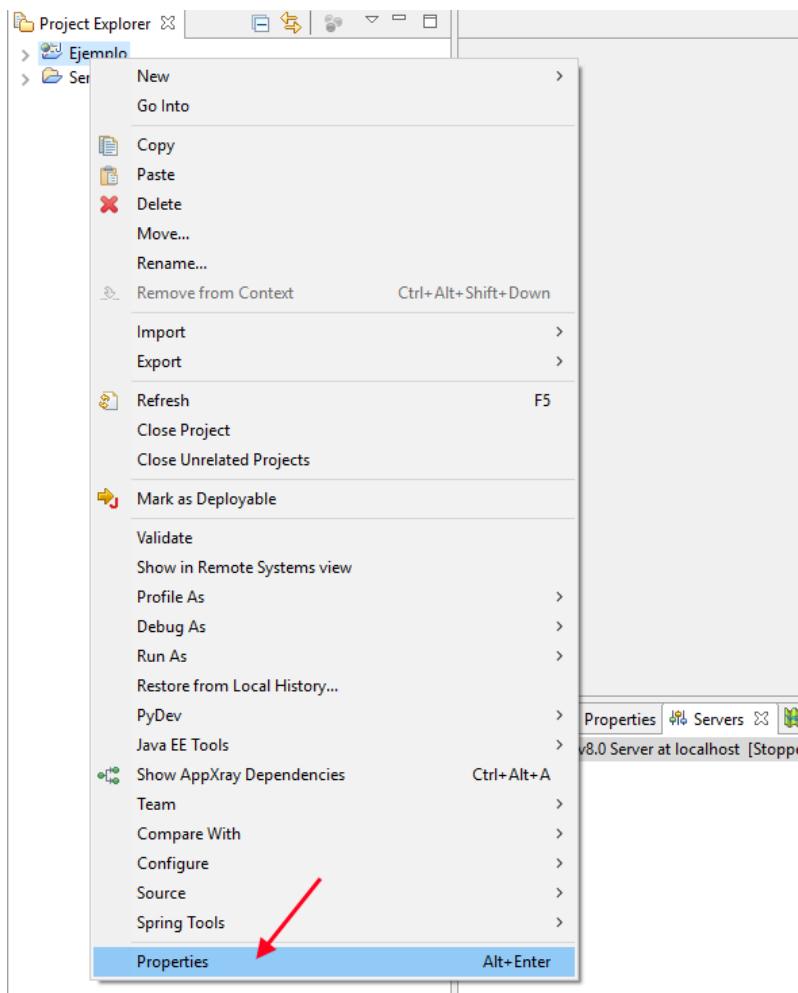


Paso 3: Una vez descargados los archivos .jar los colocamos la carpeta **lib**.



Configurar proyecto

Paso 1: Lo primero que haremos será dar clic derecho **sobre el proyecto**, y en el menú que se despliega ir a **properties**, como se muestra en la siguiente imagen:



Paso 2: Luego en la sección de la izquierda buscar **Project Facets**, y del lado derecho seleccionar Java Server Faces **versión 2.2**, con esto en la parte de abajo se muestra el mensaje **Further configuration required**, damos clic en dicho mensaje.

Properties for Ejemplo

type filter text

Resource
AppXray
Builders
CDI (Context and Dependency Injection)
Deployment Assembly
FreeMarker Context
Hibernate Settings
Java Build Path
Java Code Style
Java Compiler
Java Editor
Javadoc Location
JavaScript
JAX-RS
JBoss Tools Knowledge Base
JSP Fragment
Project Archives
Project Facets A
Project References
Run/Debug Settings
Seam Settings
Server
Service Policies
Targeted Runtimes
Task Repository
Task Tags
Validation
Web Content Settings
Web Page Editor
Web Project Settings
WikiText
XDoclet

Project Facets

Configuration: <custom>

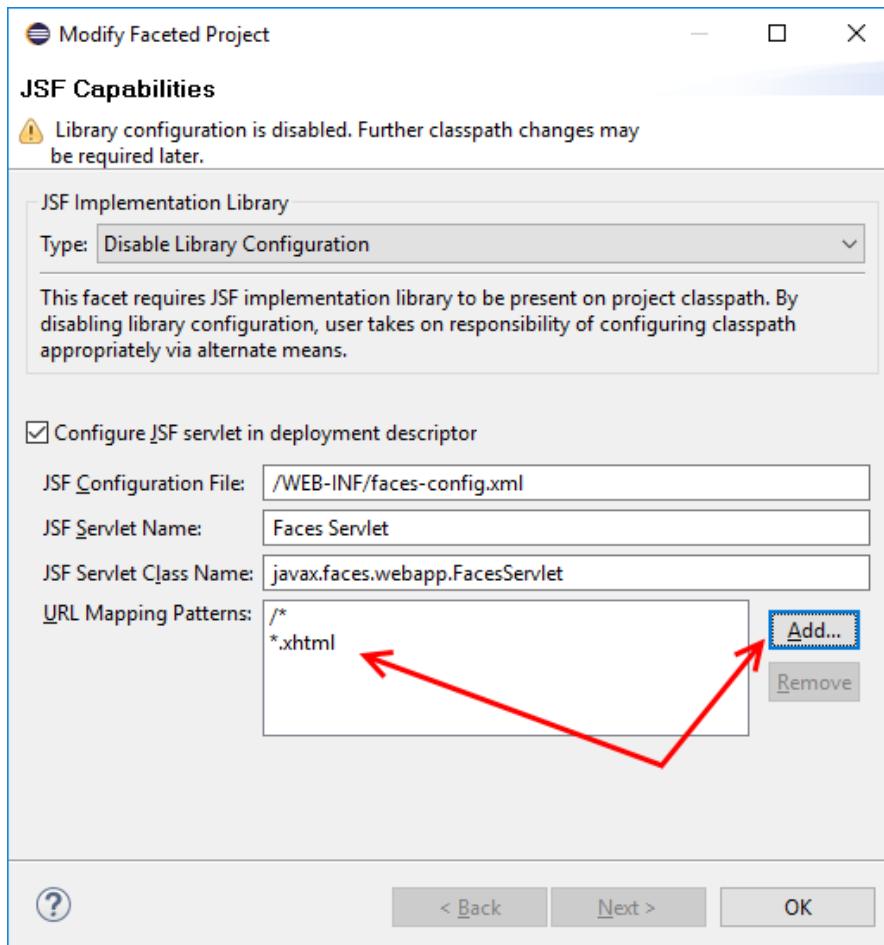
Project Facet	Version
Axis2 Web Services	1.1
CDI (Contexts and Dependency Injection)	1.0
CXF 2.x Web Services	3.0
Dynamic Web Module	1.8
Java	5.0
Java Annotation Processing Support	1.0
JavaScript	2.2
JavaServer Faces B	1.1
JAX-RS (REST Web Services)	2.2
JAXB	1.0
JBoss Maven Integration	3.0
JBoss Portlets	2.1
JBoss Web Services Core	1.2
JPA	1.2
JSTL	1.3
Oracle Coherence	2.3
Seam	2.0
Struts	1.3
Trinidad	1.2.3
WebDoclet (XDoclet)	1.0

Details Runtime JavaServer F... C

Enables the project Requires the following Dynamic We...

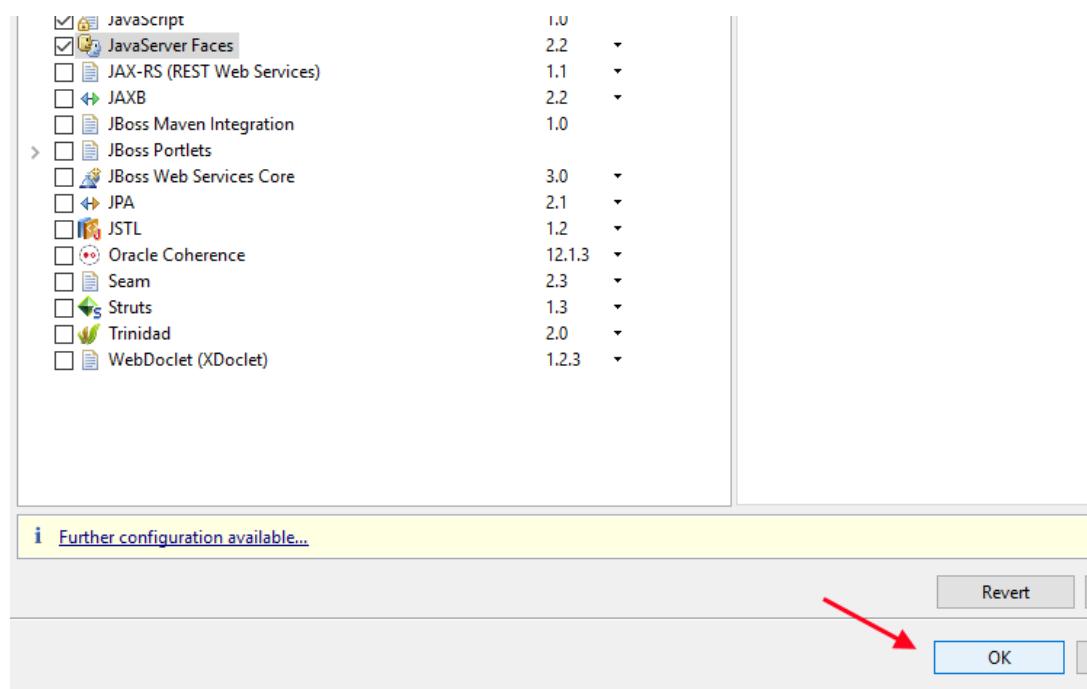
Further configuration required... D

Paso 3: Se nos muestra la siguiente pantalla, en donde modificaremos los valores de los elementos seleccionados y finalmente damos clic en **OK**.



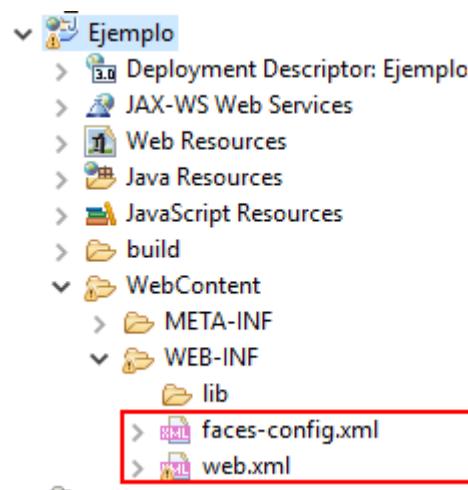
Nota: Si dejamos **/faces/** tendremos que crear una carpeta llamada **faces** a nivel de WebContent y allí poner todas las páginas **XHTML**; y si lo cambiamos por el comodín **/*** y ***.xhtml**, ya los podemos poner en cualquier carpeta dentro de WebContent.

Una vez que se cierra la ventana anterior, damos clic en **OK** en la ventana de Project Facets, revisa la siguiente imagen:



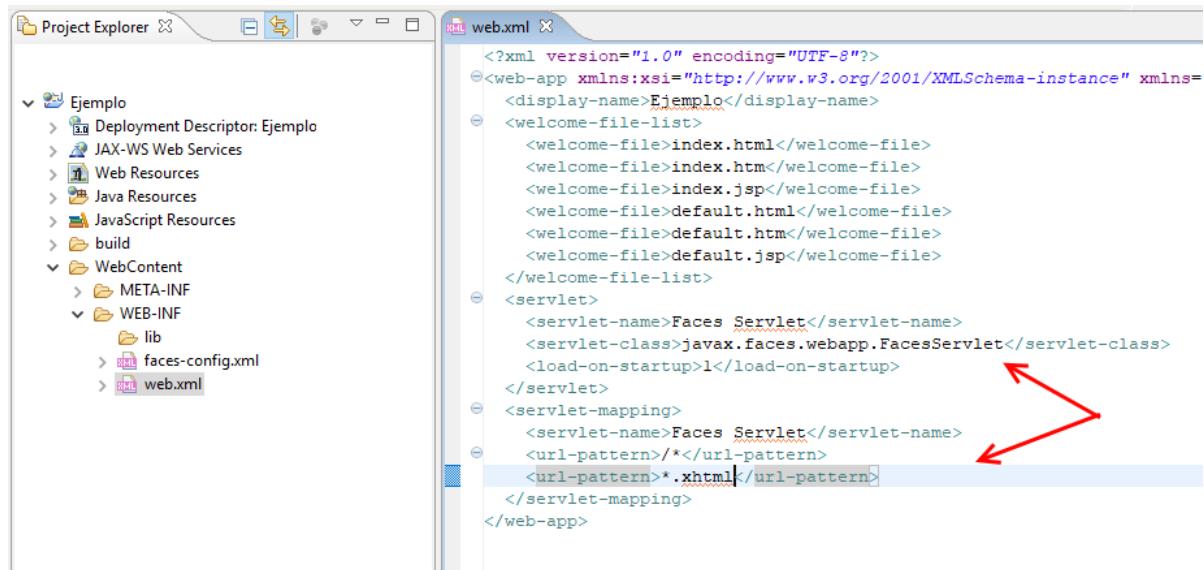
Con lo anterior hemos configurado nuestro proyecto web para usar Java Server Faces.

En la siguiente imagen podrás ver que se creó un nuevo archivo **faces-config.xml**, en el cual puedes configurar algunas cosas propias de Java Server Faces, como por ejemplo el archivo para la internacionalización de la aplicación (es decir, para cambiar el idioma de las etiquetas que se muestran en el HTML)



Puedes abrir el archivo **web.xml** y verás que se crearon nuevas propiedades, por ejemplo, la etiqueta de **welcome-file**, podrás ver que están marcadas de color amarillo ya que dichos archivos no existen.

Además, se agregaron otras propiedades en el archivo web.xml, como *Faces Servlet*, con el cual le indicamos al proyecto que usaremos el framework JSF.



```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns=
<display-name>Ejemplo</display-name>
<welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.htm</welcome-file>
    <welcome-file>default.jsp</welcome-file>
</welcome-file-list>
<servlet>
    <servlet-name>Faces Servlet</servlet-name>
    <servlet-class>javax.faces.webapp.FacesServlet</servlet-class>
    <load-on-startup>1</load-on-startup>
</servlet>
<servlet-mapping>
    <servlet-name>Faces Servlet</servlet-name>
    <url-pattern>/*</url-pattern>
    <url-pattern>*.xhtml</url-pattern>
</servlet-mapping>
</web-app>
```

Archivo web.xml

Lo que haremos en el archivo web.xml será dejar una sola etiqueta de todas las welcome-file creadas:

```
<welcome-file>index.xhtml</welcome-file>
```

Con esto indicamos que cuando tecleamos en el navegador lo siguiente:

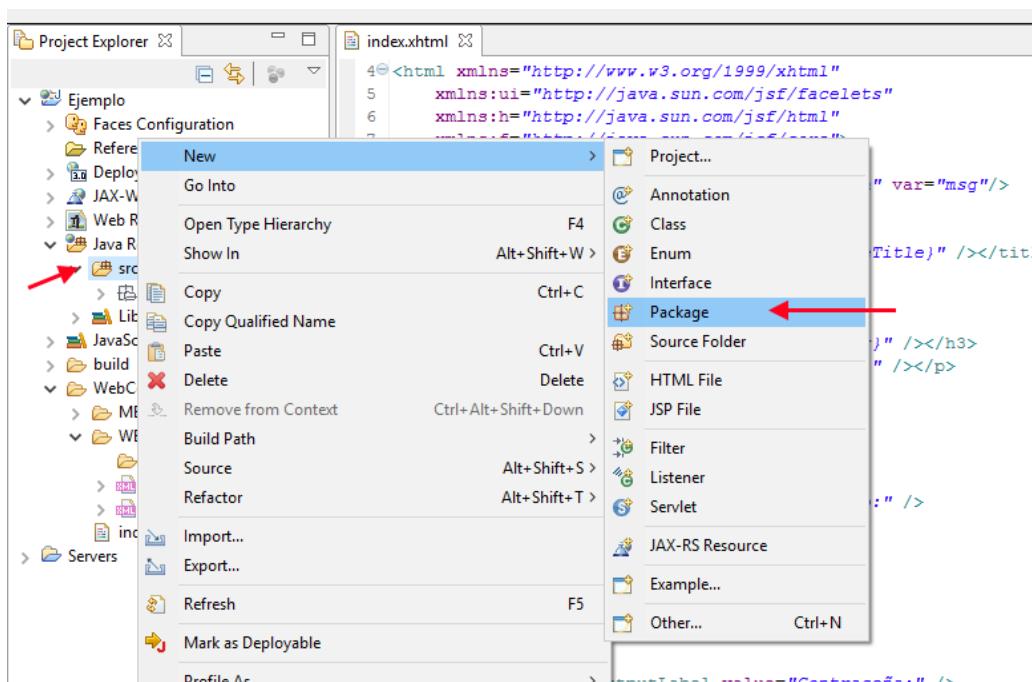
localhost:8080/Ejemplo

El proyecto buscará el archivo **index.xhtml** para mostrarlo en el navegador, ya que está definido como la página inicial.

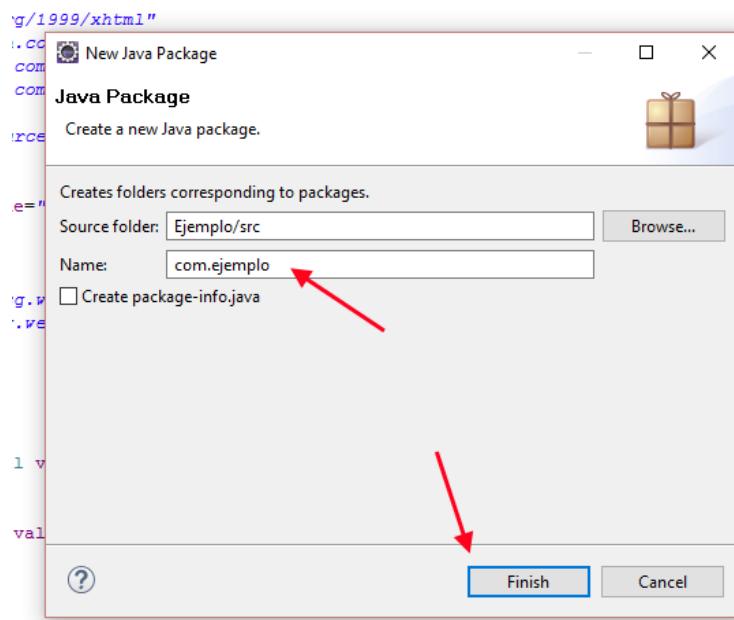
Archivo LoginView.java

Crearemos este archivo para controlar el inicio de sesión de un usuario, pero primero debemos de crear un paquete (carpetas), para ordenar en donde poner qué clase.

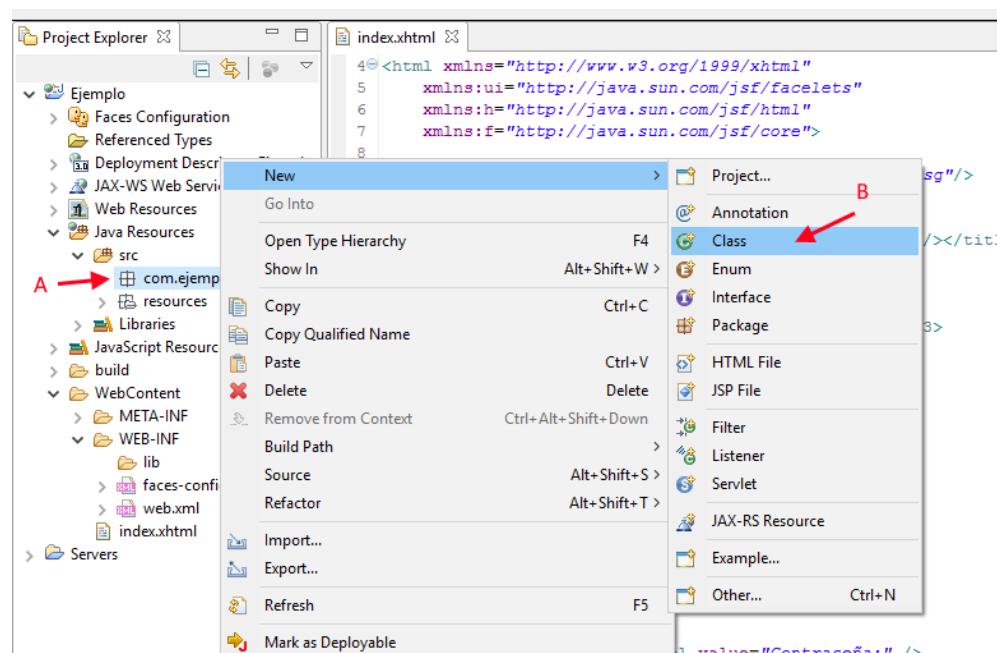
Damos clic derecho sobre **src**, seleccionamos New y luego Package:



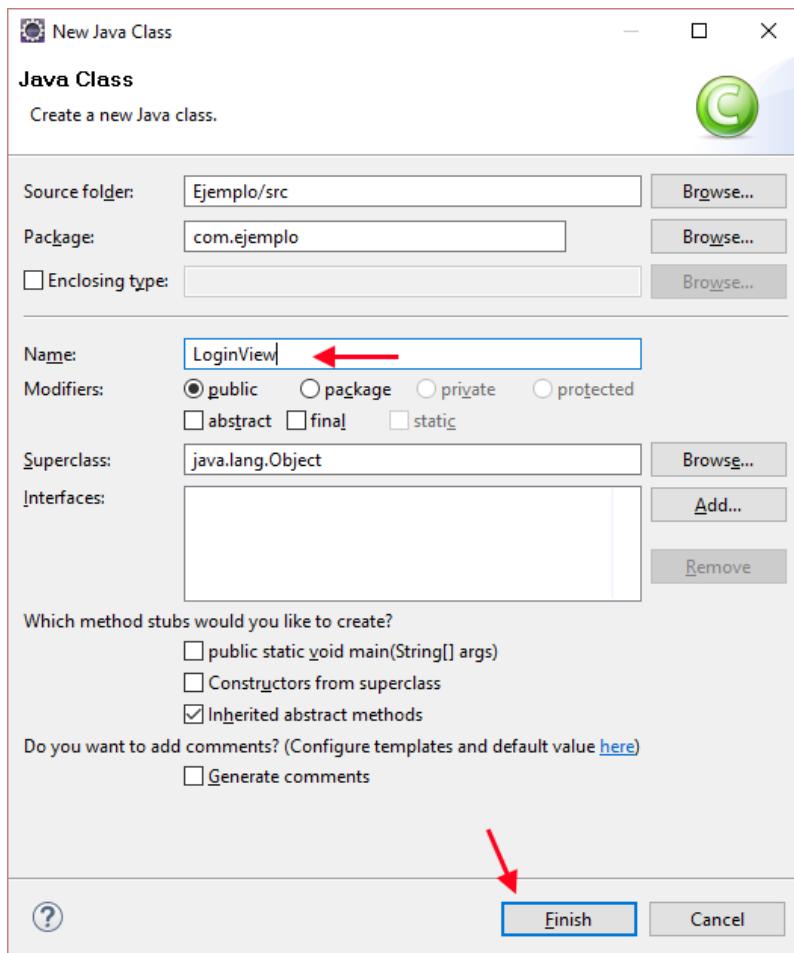
Creamos un paquete con el siguiente nombre: **com.ejemplo**



Luego crearemos una clase (archivo .java), de la siguiente manera:



Colocamos el nombre de LoginView y damos clic en Finish:



Al abrir dicha clase agregaremos lo siguiente:

- `@ManagedBean` y `@SessionScoped`
- Crear atributos `usuario` y `password`

Tip: Revisa muy bien la parte de las importaciones (B), ya que en ocasiones existe más de una clase con el mismo nombre, pero que pertenece a diferente librería (.jar)

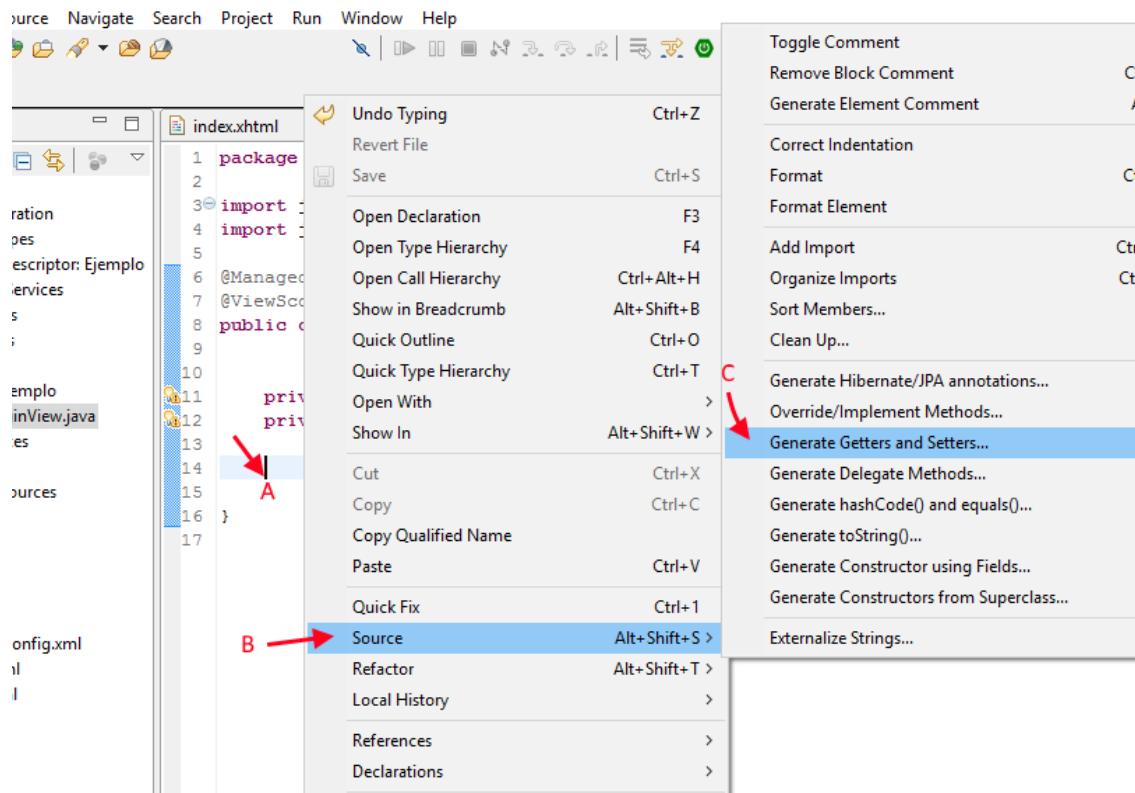
The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows the project structure under "Ejemplo".
- Code Editor:** Displays `LoginView.java` with the following code:

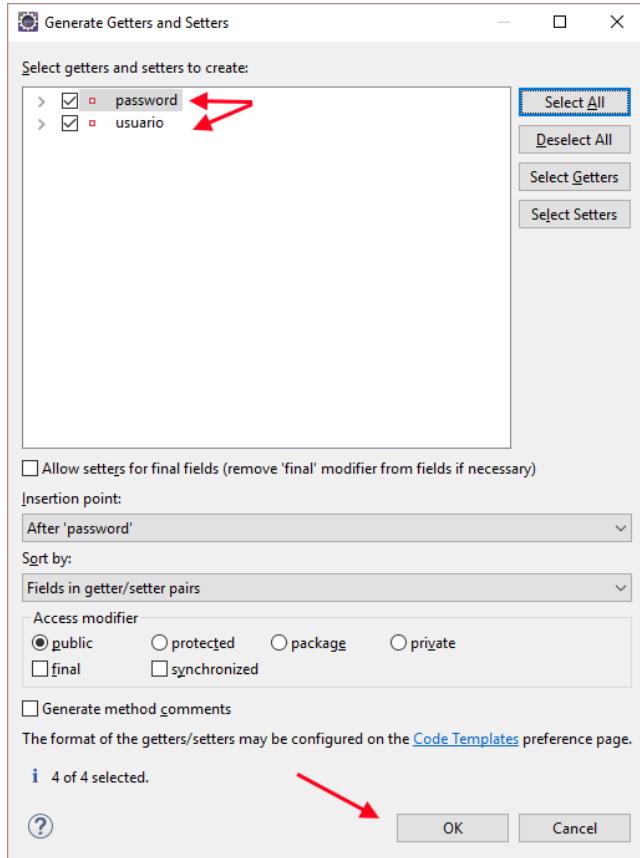
```
1 package com.ejemplo;
2
3 import javax.faces.bean.ManagedBean; ← B
4 import javax.faces.bean.SessionScoped; ← B
5
6 @ManagedBean ← A
7 @SessionScoped ← A
8 public class LoginView {
9
10
11     private String usuario; ← C
12     private String password; ← C
13
14
15
16 }
17
```
- Annotations:** Red arrows labeled A, B, and C point to the annotations and field declarations.

En esa misma clase, daremos clic derecho en cualquier parte (A), luego seleccionamos *Source* y por ultimo *Generate Getter and Setters*.

Con esto se generan los métodos con prefijo **getXXX** y **setXXX**, por cada atributo declarado en la clase.



Seleccionamos los atributos de los cuales generamos los métodos *Getters and Setters*:



Ahora veremos la clase de la siguiente manera:

```
1 package com.ejemplo;
2
3 import javax.faces.bean.ManagedBean; B
4 import javax.faces.bean.SessionScoped;
5
6 @ManagedBean A
7 @SessionScoped
8 public class LoginView {
9
10
11     private String usuario;
12     private String password;
13
14
15     public String getUsuario() {
16         return usuario;
17     }
18     public void setUsuario(String usuario) {
19         this.usuario = usuario;
20     }
21     public String getPassword() {
22         return password;
23     }
24     public void setPassword(String password) {
25         this.password = password;
26     }
27
28
29
30 }
```

Lo siguiente será crear un método **inicializa** que inicializa los atributos de la página (xhtml), dicho método le colocamos la anotación @Postconstruct:

```
1 package com.ejemplo;
2
3 import javax.annotation.PostConstruct;
4 import javax.faces.bean.ManagedBean;
5 import javax.faces.bean.SessionScoped;
6
7 @ManagedBean
8 @SessionScoped
9 public class LoginView {
10
11
12     private String usuario;
13     private String password;
14
15     @PostConstruct ← A
16     public void inicializa(){
17         this.usuario = "";
18         this.password = ""; ← B
19     }
20
21
22     public String getUsuario() {
23         return usuario;
24     }
25     public void setUsuario(String usuario) {
26         this.usuario = usuario;
27     }
28     public String getPassword() {
29         return password;
30     }
31     public void setPassword(String password) {
32         this.password = password;
33     }
34
35
36
37 }
```

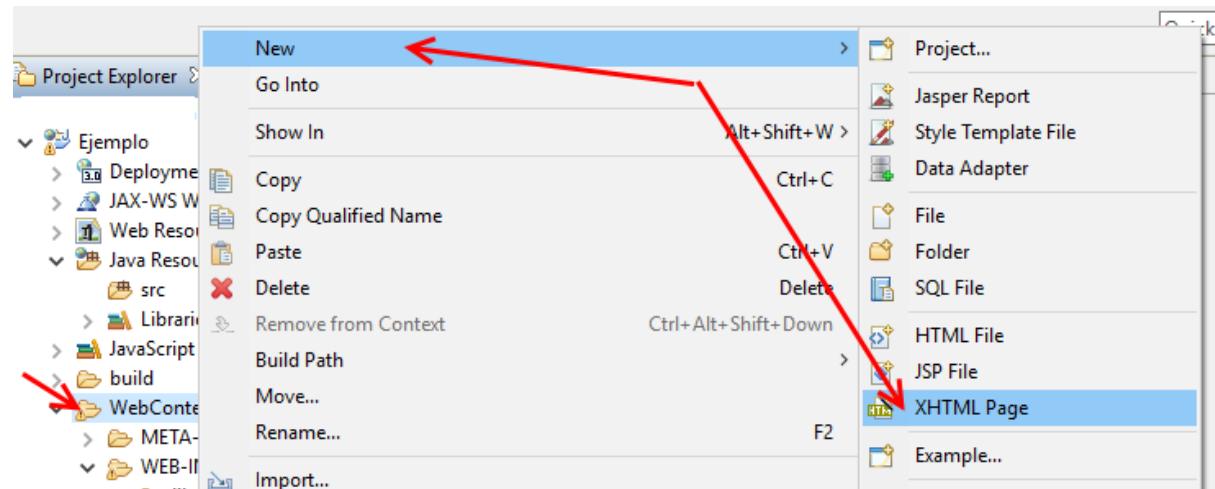
Luego agregaremos el método **botonLogin** en el cual declaramos la página a la que será dirigido cuando de clic en el login:

```
1 package com.ejemplo;
2
3 import javax.annotation.PostConstruct;
4 import javax.faces.bean.ManagedBean;
5 import javax.faces.bean.SessionScoped;
6
7
8 @ManagedBean
9 @SessionScoped
10 public class LoginView {
11
12
13     private String usuario;
14     private String password;
15
16     @PostConstruct
17     public void inicializa(){
18         this.usuario = "";
19         this.password = "";
20     }
21
22     public String botonLogin(){ A
23
24         String paginaInicio = "inicio.xhtml"; B
25
26         //Aquí va lógica para validar usuario y contraseña
27
28         return paginaInicio;
29
30     }
31 }
```

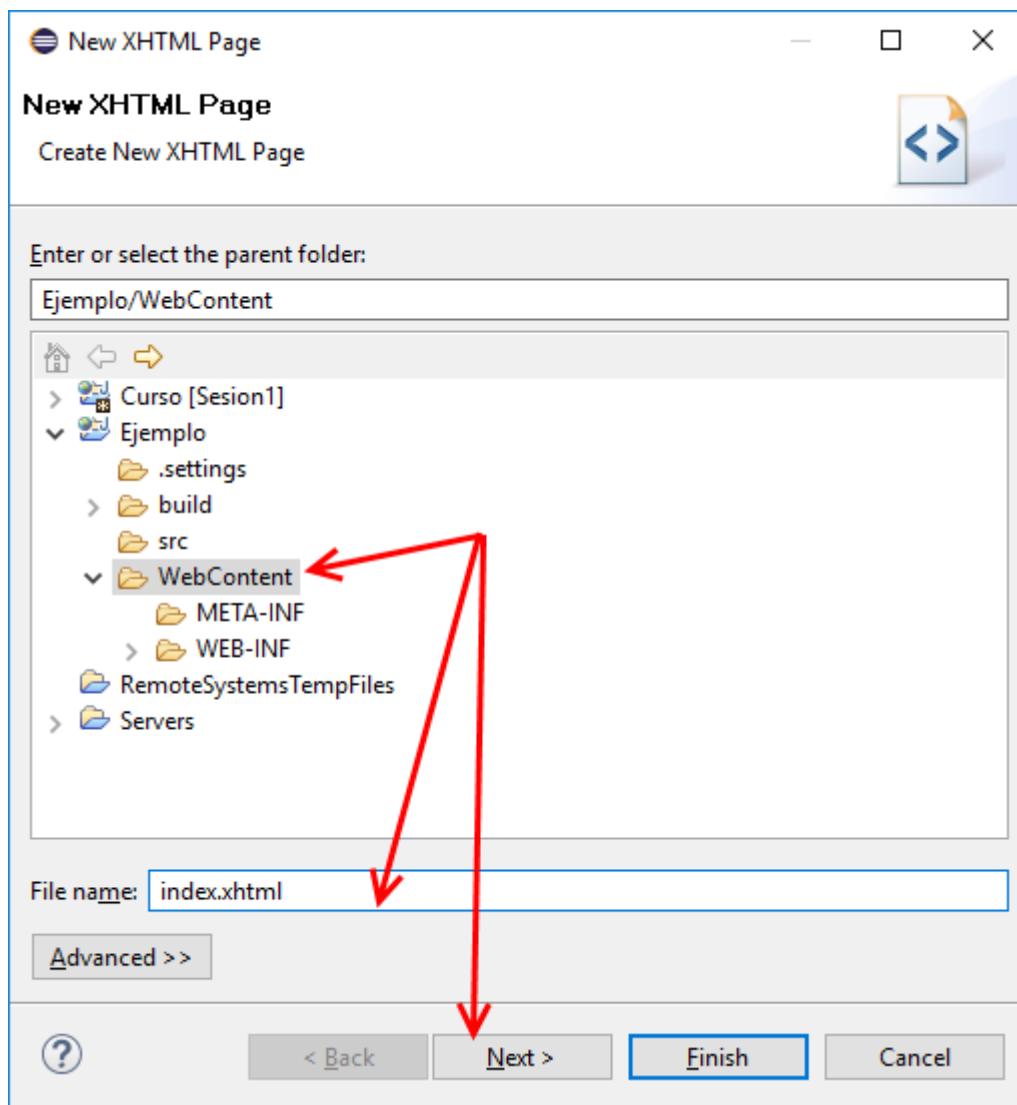
Archivo index.xhtml

Crearemos el archivo **index.xhtml**, a continuación, muestro los pasos:

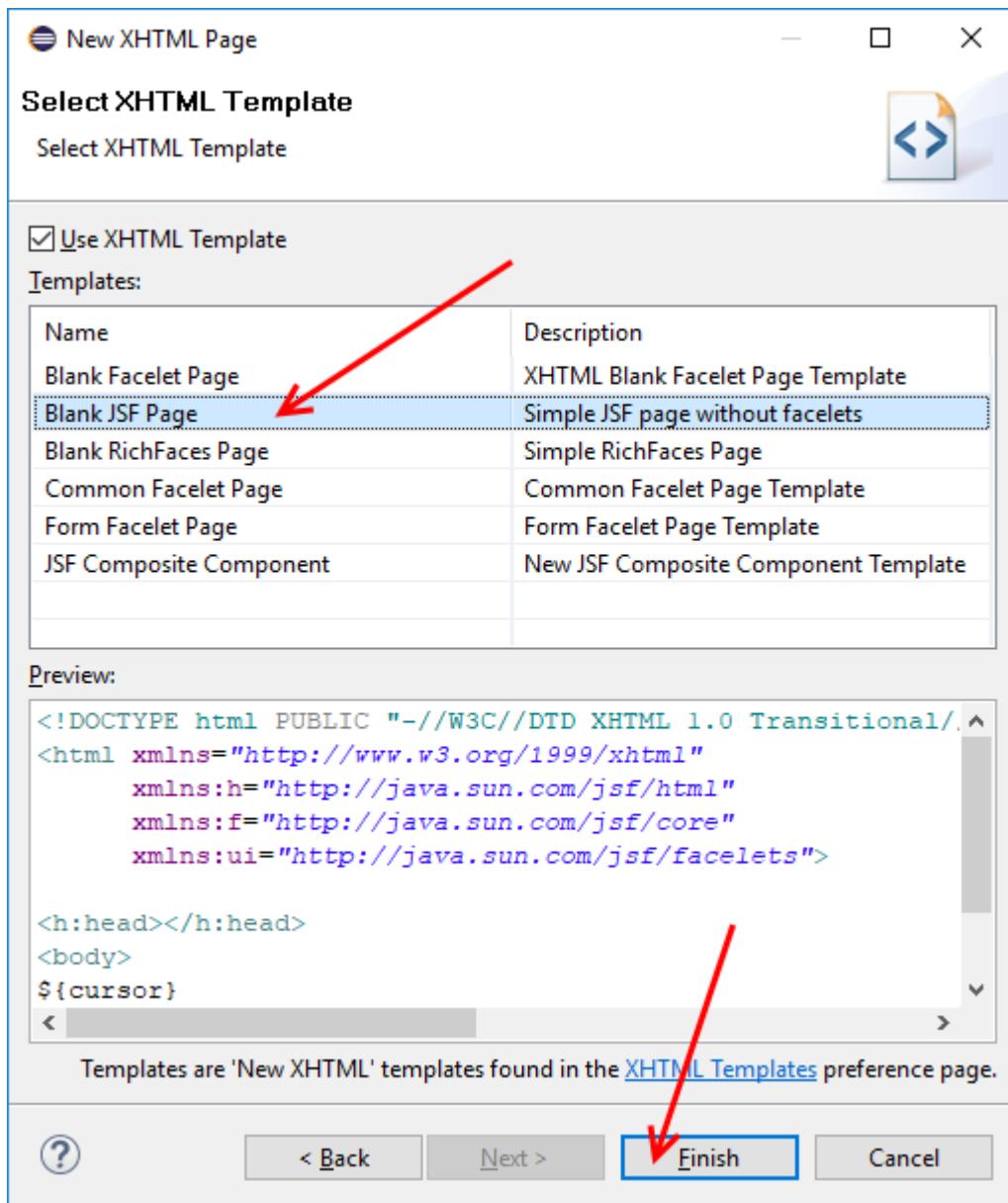
Paso 1: Da clic derecho sobre la carpeta **WebContent**:



Paso 2: Seleccionar de nuevo **WebContent**, nombrar el archivo como **index.xhtml** y dar clic en **Siguiente:**



Paso 3: Seleccionar **Blank JSF Page** y dar clic en **Finalizar**:



Nota: Para que Eclipse solo muestre el código del archivo xhtml y no muestre la pantalla dividida de código y vista previa, te recomiendo hacer los pasos del [Apéndice B](#).

Tip: Revisa la imagen que está después de esta, en ella muestro cómo autocompletar código.

Paso 4: Lo siguiente será introducir el siguiente TODO código que muestro en la imagen, señalo con flechas las etiquetas form ya que son importantes:

```
index.xhtml <-->
<h:head>
    <title>
        <h:outputText value="Titulo" />
    </title>
</h:head>
<body> | 
    <h:form> ←
        <table>
            <tr>
                <td>
                    <h:outputText value="Usuario:" />
                </td>
                <td>
                    <h:inputText value="#{loginView.usuario}" />
                </td>
            </tr>
            <tr>
                <td>
                    <h:outputText value="Contraseña:" />
                </td>
                <td>
                    <h:inputSecret value="#{loginView.password}" />
                </td>
            </tr>
            <tr>
                <td colspan="2">
                    <h:commandButton action="#{loginView.botonLogin()}" value="Iniciar" />
                </td>
            </tr>
        </table>
    </h:form> ←
</body>
```

En la siguiente imagen solo muestro el autocompletado de código, muy útil para no tener errores de sintaxis:

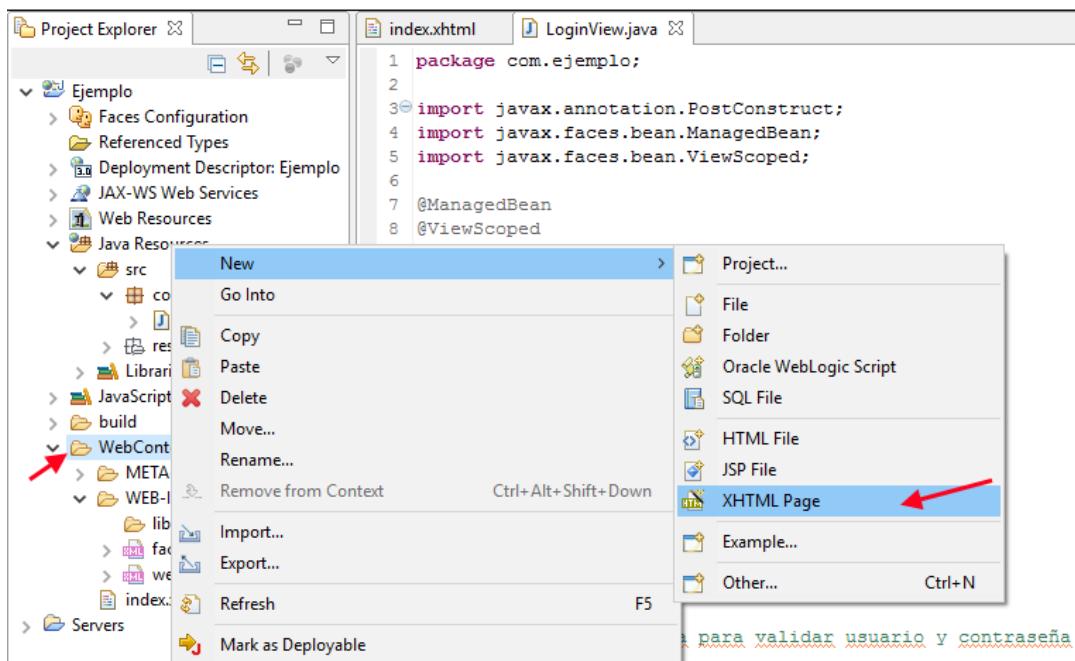
The screenshot shows a code editor with Java code for a login form. A tooltip window titled '# Three row template block - Three Row Composite' is displayed, listing 'loginView : LoginView'. The tooltip has three red arrows pointing to it from the text: 'B' points to the window title, 'C' points to the class name 'loginView', and 'A' points to the text 'ctrl + barra de espacio' at the bottom of the tooltip.

```
18 19 <h:form>
20     <table>
21         <tr>
22             <td>
23                 <h:outputLabel value="Usu"
24             </td>
25             <td>
26                 <h:inputText value="#{logi
27             </td>
28         </tr>
29         <tr>
30             <td>
31                 <h:outputLabel value="Con
32             </td>
33             <td>
34                 <h:inputText value="logi" />
35             </td>
36         </tr>
37     </table>
38 </h:form>
39
```

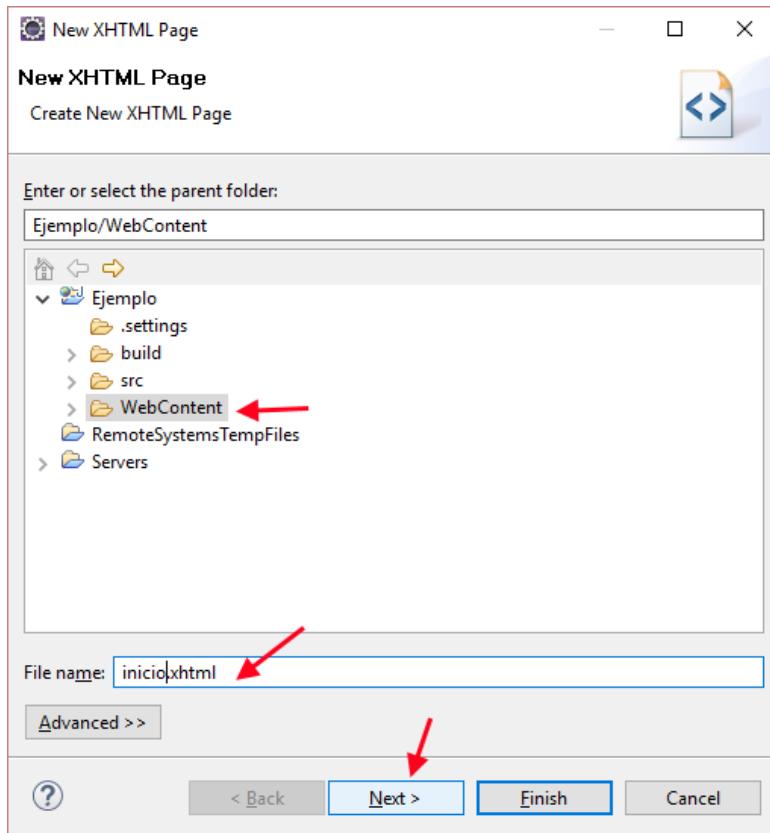
Tip: Si en el archivo .xhtml visualizas una *palabra que está subrayada de color amarillo*, quiere decir que dicha atributo no existe en la clase LoginView.java

Archivo inicio.xhtml

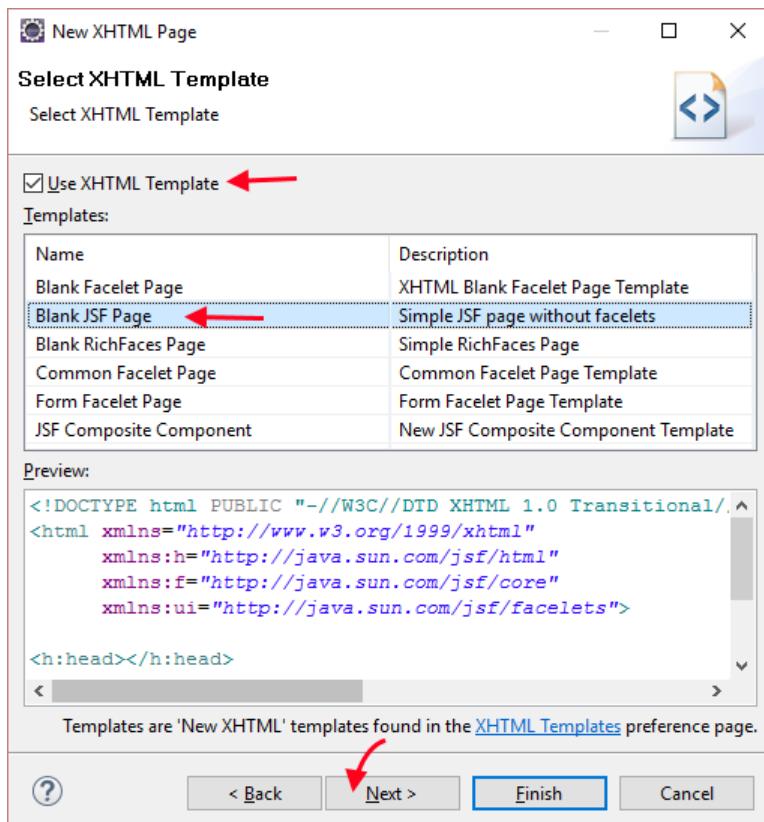
Crearemos un archivo XHTML con el nombre de **inicio.xhtml**, éste archivo será invocado por el método que tiene asignado el botón **Iniciar** del archivo **index.xhtml**, a continuación muestro los pasos en imágenes:



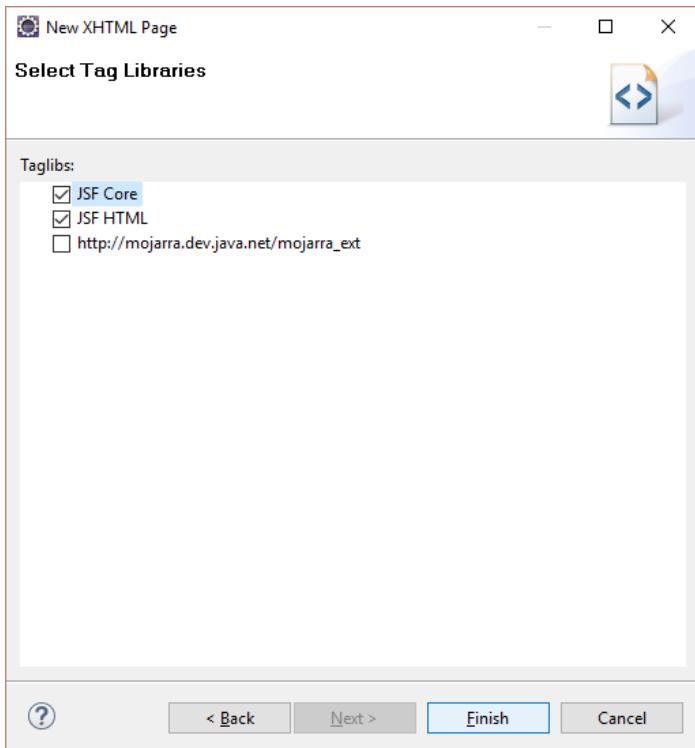
Seleccionamos la carpeta WebContent, nombramos el archivo como inicio.xhtml y clic en *Next*.



Seleccionamos “Use XHTML Template” y clic en *Next*.

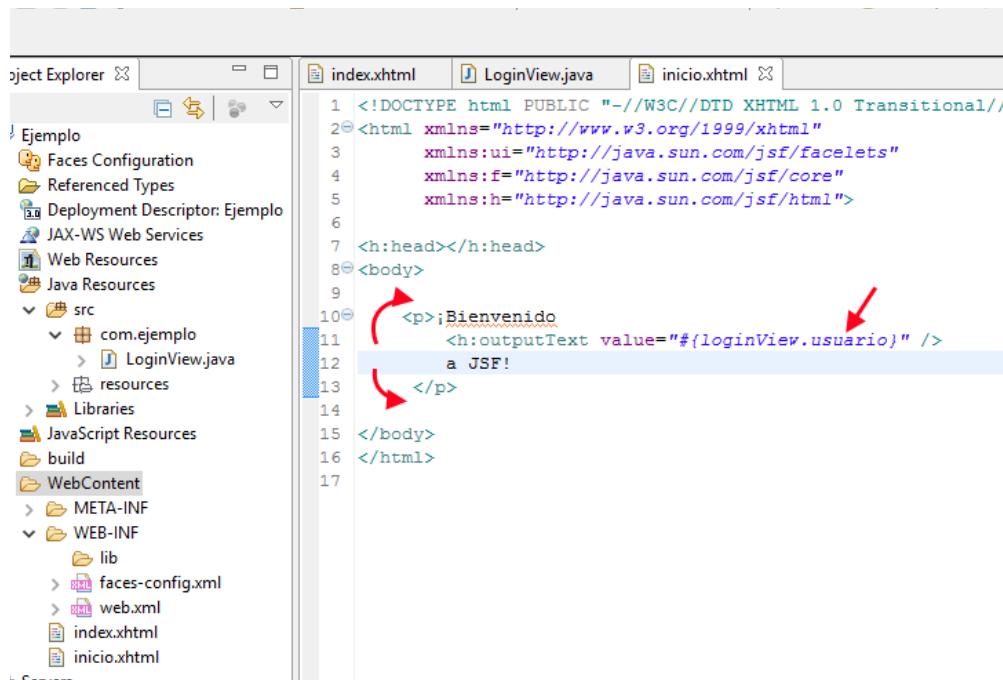


Seleccionamos JSF Core y JSF HTML y clic en *Finish*.



Se creará el siguiente XHTML, en el cual agregaremos esto:

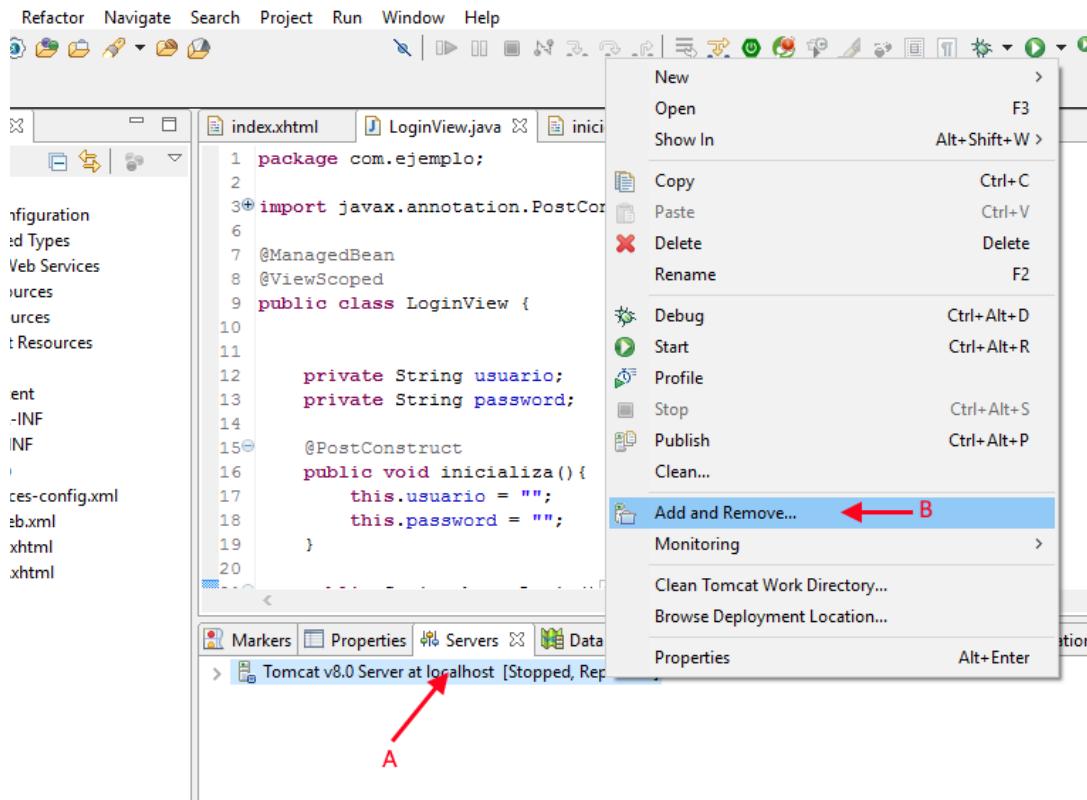
```
<p>¡Bienvenido <h:outputText value="#{loginView.usuario}" />  
a JSF!</p>
```



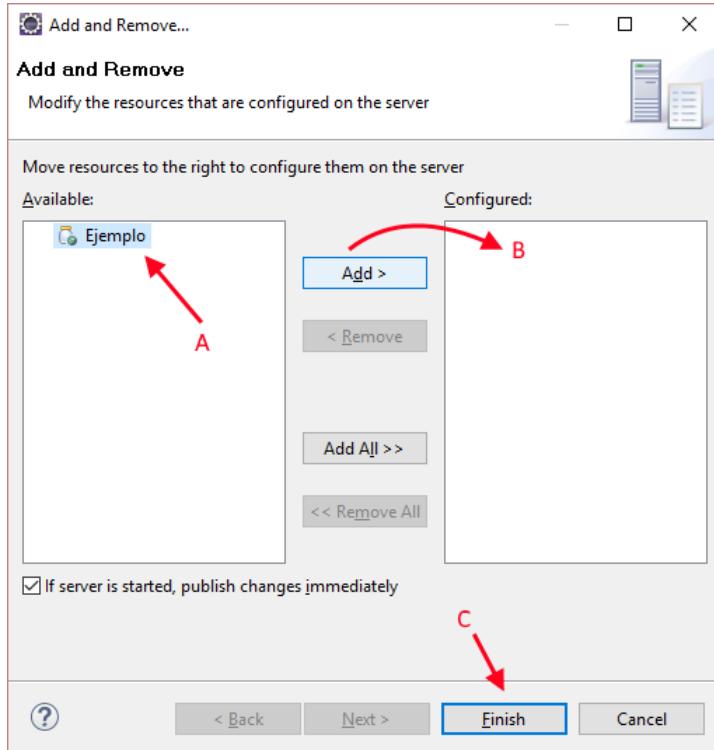
Desplegar el proyecto

Si has llegado a este paso, lo siguiente es echar a andar el proyecto, para esto realizaremos los siguientes pasos:

Dar **clic derecho** sobre el servidor de aplicaciones Tomcat, luego en **Add and Remove**.

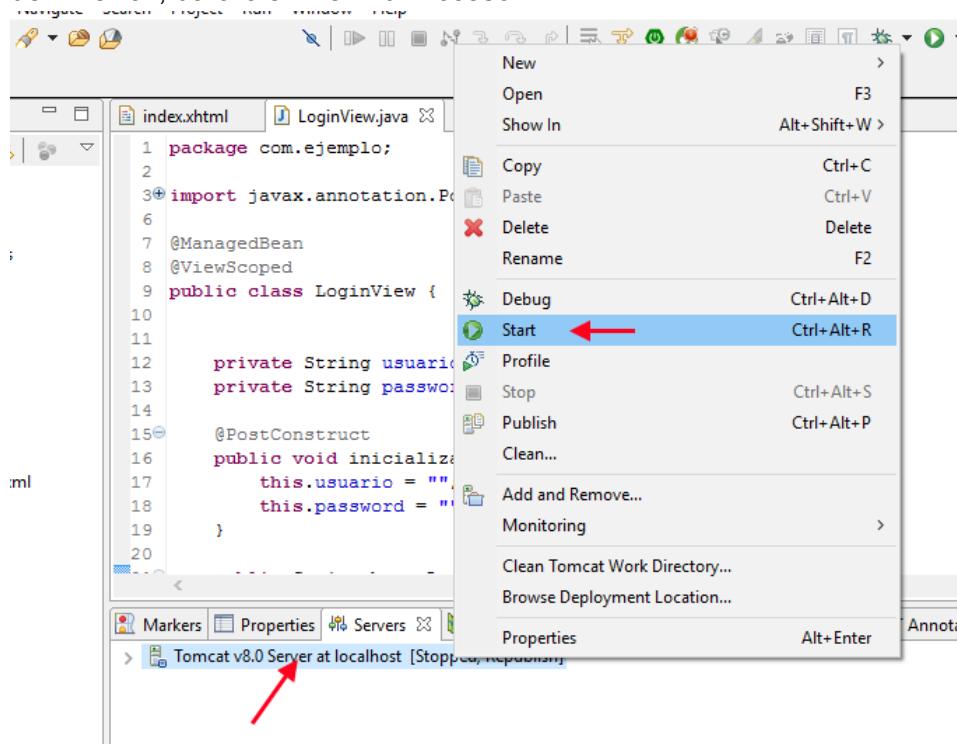


En la siguiente ventana agregamos el proyecto Ejemplo al servidor de aplicaciones:



Luego de dar clic en Finish, en la siguiente imagen volvemos a dar **clic derecho** sobre Tomcat y luego dar clic en **Start**:

Nota: La primera vez que inicias (Start) Tomcat, Windows te mostrará un mensaje de alerta del Firewall, da clic en Permitir Acceso.



En la pestaña de **Console**, podrás ver como se muestran logs conforme se despliega el proyecto, cuando veas el mensaje de Server startup in XXX ms, quiere decir que ya está listo:

```
17     this.usuario = "";
18     this.password = "";
19 }
20
Tomcat v8.0 Server at localhost [Apache Tomcat] C:\Program Files\Java\jre1.8.0_101\bin\javaw.exe (21/10/2016 17:59:46)
INFORMACIÓN: Inicializando Mojarra 2.2.0 ( 20130502-2118 https://svn.java.net/svn
oct 21, 2016 5:59:48 PM com.sun.faces.spi.InjectionProviderFactory createInstance
INFORMACIÓN: JSF1048: hay presentes anotaciones PostConstruct/PreDestroy. Los mé
oct 21, 2016 5:59:49 PM com.sun.faces.config.ConfigureListener contextInitialized
INFORMACIÓN: Inicializando Mojarra 2.2.0 ( 20130502-2118 https://svn.java.net/svn
oct 21, 2016 5:59:49 PM com.sun.faces.spi.InjectionProviderFactory createInstance
INFORMACIÓN: JSF1048: hay presentes anotaciones PostConstruct/PreDestroy. Los mé
oct 21, 2016 5:59:49 PM com.sun.faces.mgbean.BeanManager addBean
ADVERTENCIA: JSF1074: ya se ha registrado el bean administrado denominado 'loginV
oct 21, 2016 5:59:49 PM org.apache.coyote.AbstractProtocol start
INFORMACIÓN: Starting ProtocolHandler ["http-nio-8080"]
oct 21, 2016 5:59:49 PM org.apache.coyote.AbstractProtocol start
INFORMACIÓN: Starting ProtocolHandler ["ajp-nio-8009"]
oct 21, 2016 5:59:49 PM org.apache.catalina.startup.Catalina start
INFORMACIÓN: Server startup in 1616 ms
```

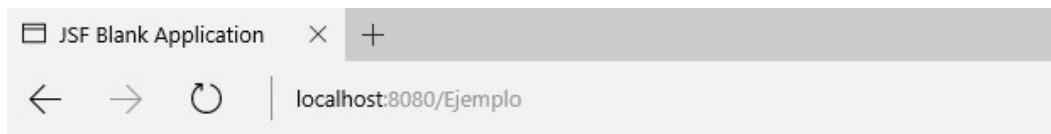
Tip: Si al iniciar Tomcat te marca un error de puerto 8080 ocupado, regresa al [siguiente apartado](#).

Ahora abre el navegador web de tu preferencia y teclea la siguiente url:

localhost:8080/Ejemplo

Nota: si no te presenta la siguiente página quiere decir que te saltaste algún por lo que te recomiendo regresar al [siguiente apartado](#).

Verás la siguiente pantalla:



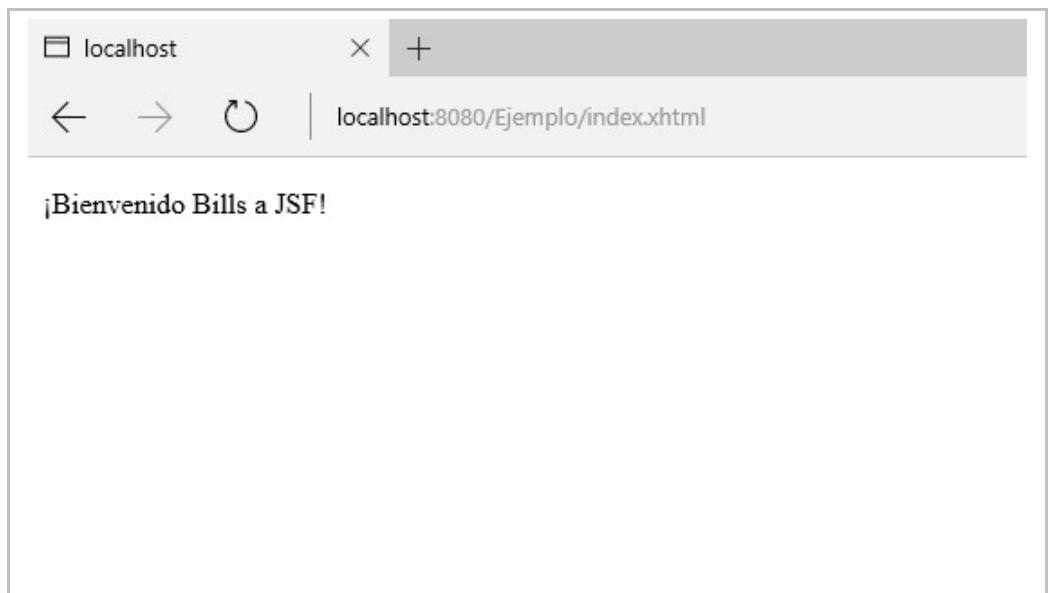
Welcome!

This is a JSF blank application. You can find the application.properties file with this message if you have a look around.

Usuario:

Contraseña:

Cuando das clic en el botón de Iniciar, debes de ser redirigido a la siguiente pantalla:



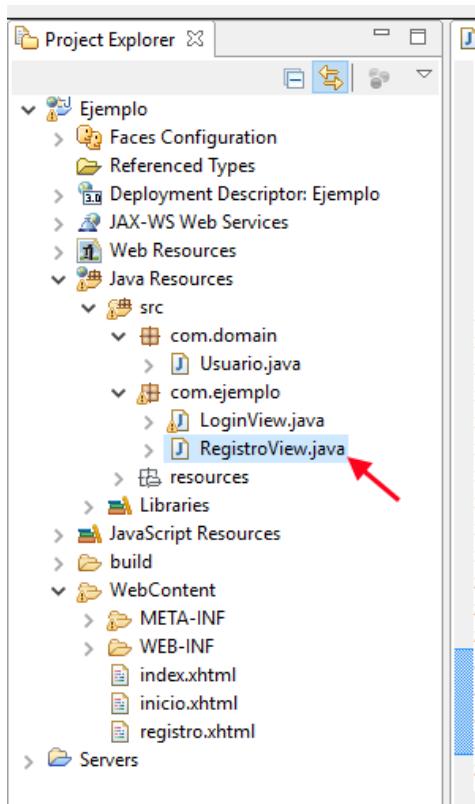
Capítulo 14. Conexión a base de datos

Para esto usaremos el proyecto web con JSF que hemos venido trabajando, en el cual agregaremos los siguientes archivos:

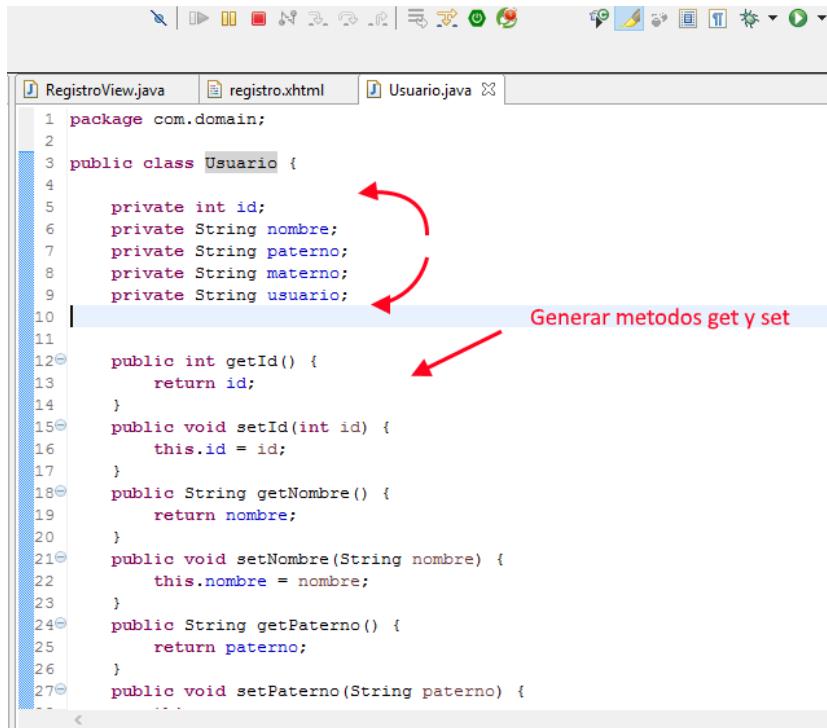
- registro.xhtml
- RegistroView.java
- Usuario.java

Usaremos la tabla usuario creada en la sección [Oracle SQL Developer](#)

El proyecto quedaría de la siguiente manera:



Así quedaría la clase Usuario.java, recuerda que para generar los métodos get y set de cada atributo es mediante **clic derecho** dentro de la clase, luego **Source** y finalmente **Generate Getters and Setters**.



```
1 package com.domain;
2
3 public class Usuario {
4
5     private int id;
6     private String nombre;
7     private String paterno;
8     private String materno;
9     private String usuario;
10
11    public int getId() {
12        return id;
13    }
14    public void setId(int id) {
15        this.id = id;
16    }
17    public String getNombre() {
18        return nombre;
19    }
20    public void setNombre(String nombre) {
21        this.nombre = nombre;
22    }
23    public String getPaterno() {
24        return paterno;
25    }
26    public void setPaterno(String paterno) {
27        ...
28    }
29}
```

La clase RegistroView.java quedará de la siguiente manera:

```
gistroView.java ✘ | registro.xhtml | Usuario.java
@ManagedBean
@ViewScoped
public class RegistroView {

    private Usuario usuario;

    @PostConstruct
    public void inicializa(){
        this.usuario = new Usuario();
    }

    public void registrar() throws NamingException, SQLException{
        //Inicia: Configuracion de conexion a base de datos
        Context ctx = new InitialContext();
        A Context envctx = (Context) ctx.lookup("java:/comp/env");
        DataSource dataSource = (DataSource) envctx.lookup("jdbc/myoracle");
        Connection con = dataSource.getConnection();

        B if(con==null){
            throw new SQLException("No se pudo obtener la conexion a la base de datos.");
        }
        //Termina: Configuracion de conexion a base de datos

        PreparedStatement ps;

        C String sql = "INSERT INTO usuario"
            + "(id, nombre, paterno, materno, usuario) VALUES(sq_usuario.nextval,?, ?, ?, ?)";

        ps= con.prepareStatement(sql);

        //Poner en el mismo orden que las columnas en el insert
        D ps.setString(1, usuario.getNombre());
        ps.setString(2, usuario.getPaterno());
        ps.setString(3, usuario.getMaterno());
        ps.setString(4, usuario.getUsuario());
        ps.executeUpdate(); //Ejecuta el insert
    }
}
```

Nota: Es importante que generes los métodos get y set del atributo usuario, que no se ven en la imagen.

Tip: Dejo la siguiente imagen para que verifiquen las clases importadas, en ocasiones hay clases que tienen el mismo nombre, pero que pertenecen a diferentes librerías:

```
1 package com.ejemplo;
2
3 import java.sql.Connection;
4 import java.sql.PreparedStatement;
5 import java.sql.SQLException;
6
7 import javax.annotation.PostConstruct;
8 import javax.faces.bean.ManagedBean;
9 import javax.faces.bean.ViewScoped;
10 import javax.naming.Context;
11 import javax.naming.InitialContext;
12 import javax.naming.NamingException;
13 import javax.sql.DataSource;
14 import com.domain.Usuario;
15
16 @ManagedBean
17 @ViewScoped
18 public class RegistroView {
```

Y finalmente la página registro.xhtml sería de la siguiente manera:

```
<h:form>
    <table>
        <tr>
            <td>
                <h:outputLabel value="Nombre:" />
            </td>
            <td>
                <h:inputText value="#{registroView.usuario.nombre}" />
            </td>
        </tr>
        <tr>
            <td>
                <h:outputLabel value="Paterno:" />
            </td>
            <td>
                <h:inputText value="#{registroView.usuario.paterno}" />
            </td>
        </tr>
        <tr>
            <td>
                <h:outputLabel value="Materno:" />
            </td>
            <td>
                <h:inputText value="#{registroView.usuario.materno}" />
            </td>
        </tr>
        <tr>
            <td>
                <h:outputLabel value="Usuario:" />
            </td>
            <td>
                <h:inputText value="#{registroView.usuario.usuario}" />
            </td>
        </tr>
        <tr>
            <td colspan="2">
                <h:commandButton action="#{registroView.registrar()}" value="Registrar" />
            </td>
        </tr>
    </table>
</h:form>
```

Por último haremos una **modificación** al archivo **inicio.xhtml**, le agregaremos un botón que nos redirige a la pantalla de de **registro.xhtml**:

```
RegistroView.java  registro.xhtml  Usuario.java  inicio.xhtml 
!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:f="http://java.sun.com/jsf/core"
      xmlns:h="http://java.sun.com/jsf/html">

    <h:head></h:head>
    <body>

        <p>Bienvenido<br/>
           a JSF!
        </p>

        <h:form>

            <table>
                <tr>
                    <td>
                        <h:commandButton action="registro.xhtml" value="Ir a la pagina de registro" />
                    </td>
                </tr>
            </table>

        </h:form>
    </body>

```

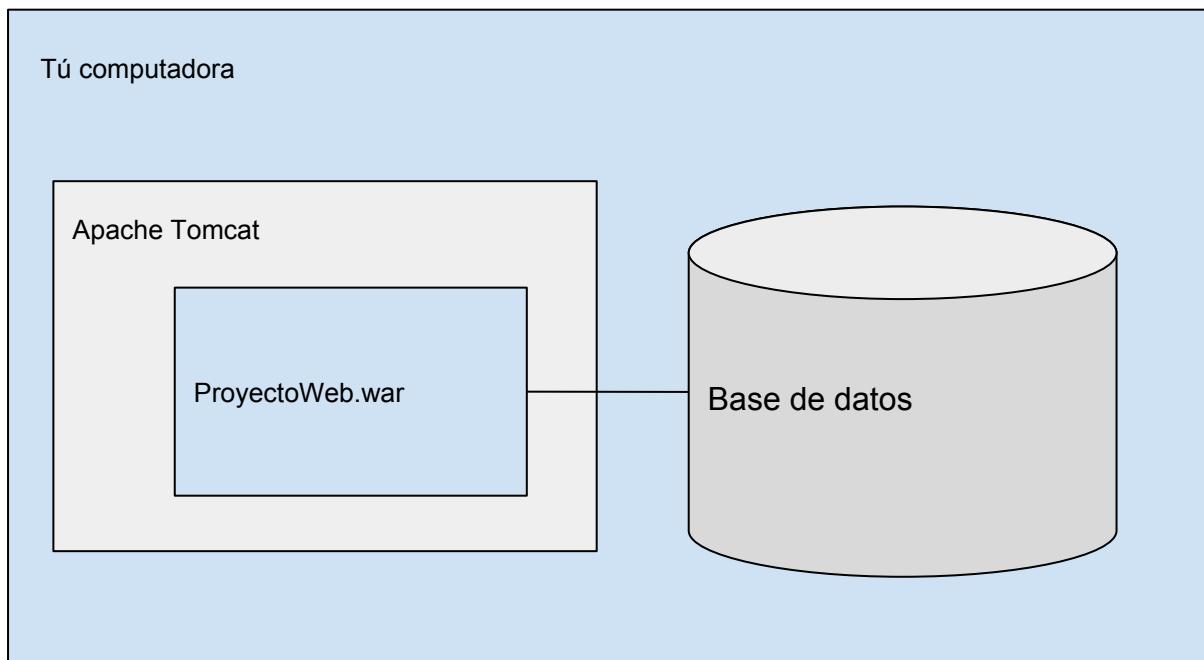
Ahora solo es necesario que inicies el proyecto introduzcas usuario y contraseña, luego en la pantalla de inicio.xhtml deberás de ver el *botón que te redirige a la pantalla de registro*, la cual debe de verse así:

Nombre:	Jorge
Paterno:	Arrambide
Materno:	Hernandez
Usuario:	jorge_

Registrar

Nota: Si hay algún error en la conexión a la base de datos revisa de nuevo el [Apéndice C](#)

A continuación, dejo una imagen sobre como quedaría funcionando tu proyecto web:



¡Felicitaciones!, ahora ya tienes el conocimiento para crear y desplegar una aplicación web. Este es tan solo un paso más. Aún falta camino por recorrer, ya que aún faltan temas por ver, como por ejemplo, las sesiones, filtros, plantillas, pool de conexiones, entre otras cosas. Lo importante es seguir avanzando. Pendiente porque pienso publicar otros libros, entre ellos un proyecto web para usar en ambientes productivos. ;)

Apéndice

A.- Plug-in Jboss

El entorno de desarrollo de Eclipse puede personalizarse con plugins que están disponibles en la sección de *Eclipse Marketplace*, mediante el menú *Help*. Puedes instalar tantos como necesites.

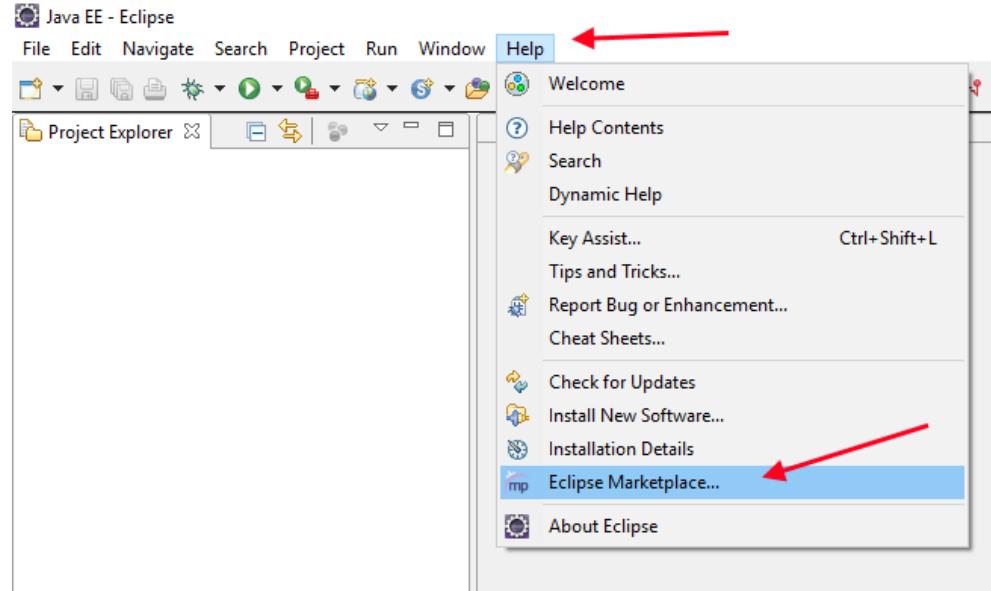
Tip: Recuerdas que Eclipse no se instala, solo se descomprime el archivo .ZIP que descargaste y se ejecuta. Pues bien, estos plug-in se instalan solo en el eclipse que descomprimimos, es decir, si descomprimes el eclipse en dos carpetas diferentes, los plug-ins que descargas en uno, no estarán disponibles en el otro.

A continuación, se mostrará como instalar el plug-in de JBoss, **un plugin muy necesario para la finalidad de este libro**, el cual le da la capacidad a Eclipse de agregar funciones adicionales, como por ejemplo, poder agregar un servidor de aplicaciones de JBoss, poder crear aplicaciones con algunas APIs propias de Jboss, entre otras cosas.

Lo que nos interesa, es que con este plug-in podamos habilitar la asistencia de código (assistant code) en los archivos XHTML, esto es, autocompletar al momento de estar escribiendo el código en dichos archivos.

A continuación detallo los pasos para instalar el plugin:

Paso 1: Vamos al menú *Help*, luego a *Eclipse Marketplace...*



Paso 2: Se mostrará la siguiente ventana, en la cual introduciremos la palabra **jboss**, en la caja de texto del buscador y damos clic en la **lupa** para iniciar la búsqueda:

Eclipse Marketplace

Select solutions to install. Press Finish to proceed with installation.
Press the information button to see a detailed overview and a link to more information.

Search Recent Popular Installed September Newsletter (IoT)

Find: **jboss**  All Markets

Featured

JRebel for Eclipse 6.5.1

 **JRebel** **Promoted** - JRebel is a productivity tool that allows developers to rebuild, restart, and redeploy cycle common in... [more info](#)
by ZeroTurnaround, Commercial
[J2EE](#) [eclipse](#) [java](#) [ee](#) [tools](#) [productivity](#) [eclipse](#) [ide](#) [software](#) [tools](#) [Dev](#) [ninja](#) [tool](#) [stop](#) [redeploying](#) [forever](#) [avoid](#) [redeploys](#) [jrebel](#) [javarebel](#) [plugins](#) [热部署](#)
★ 108  Installs: 233K (4,760 last month)

Paso 3: En los resultados se mostrarán dos plug-ins que tiene Jboss, de esos elegiremos instalar el que tiene el icono azul (**JBoss Tools 4.2.3 Final**). Dependiendo en qué momento estés realizando esto, puede ser que haya una versión más reciente, lo cual no importa para este ejercicio.

Damos clic en **Install**:

Search Recent Popular Installed September Newsletter (IoT)

Find: **jboss**  All Markets 

★ 0  Installs: 1.93K (45 last month) **Install**

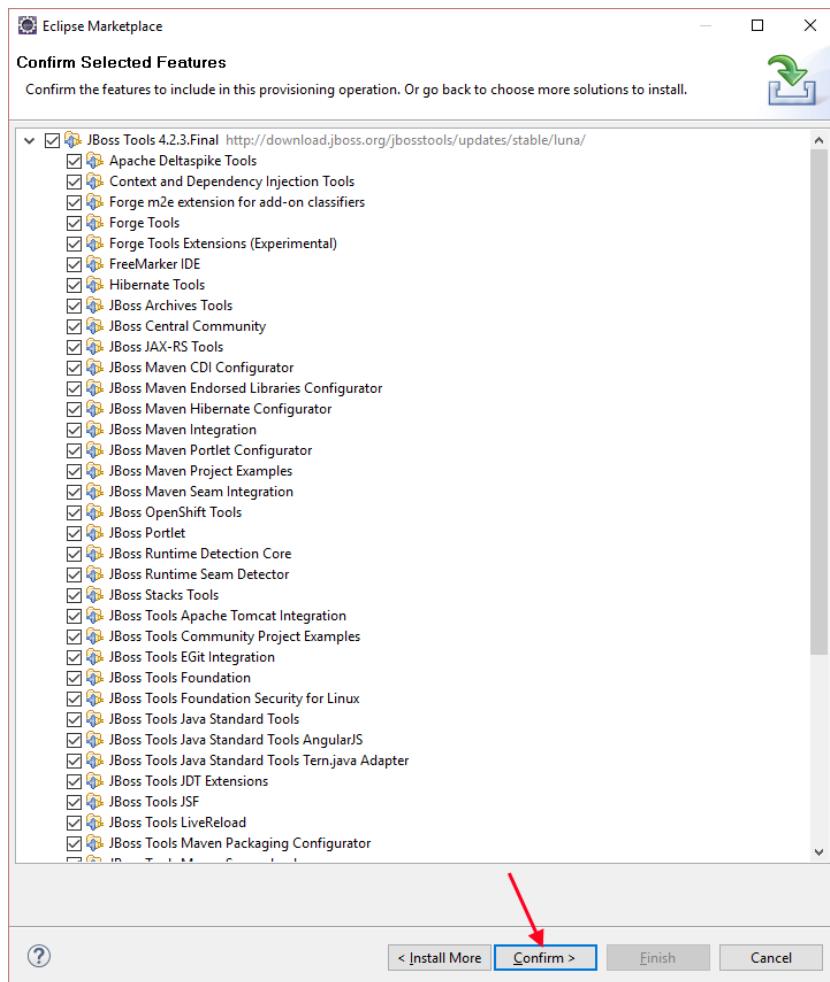
JBoss Tools 4.2.3.Final 

JBoss Tools is an umbrella project for a set of Eclipse plugins that includes support for JBoss and related technologies, such as Hibernate, JBoss AS, CDI,... [more info](#)
by Red Hat, Inc., EPL
[jbosstools](#) [maven](#) [hibernate](#) [Mobile](#) [JSF](#) [seam](#) [cdi](#) [visual](#) [editor](#) [jboss](#) [richfaces](#) [jbds](#) [openshift](#) [feedhenry](#) [freemarker](#) [fileExtension](#) [htm](#) [fileExtension](#) [html](#) [fileExtension](#) [xhtm](#) [fileExtension](#) [xhtml](#) [fileExtension](#) [ftl](#) [fileExtension](#) [ftlx](#) [fileExtension](#) [ftlh](#) [fileExtension](#) [jsp](#) [fileExtension](#) [jspx](#) [fileExtension](#) [jsf](#) [fileExtension](#) [jsfx](#) [fileExtension](#) [hbm](#) [fileExtension](#) [js](#) [fileExtension](#) [json](#) [fileExtension](#) [java](#)
★ 92  Installs: 448K (18,815 last month) **Install**

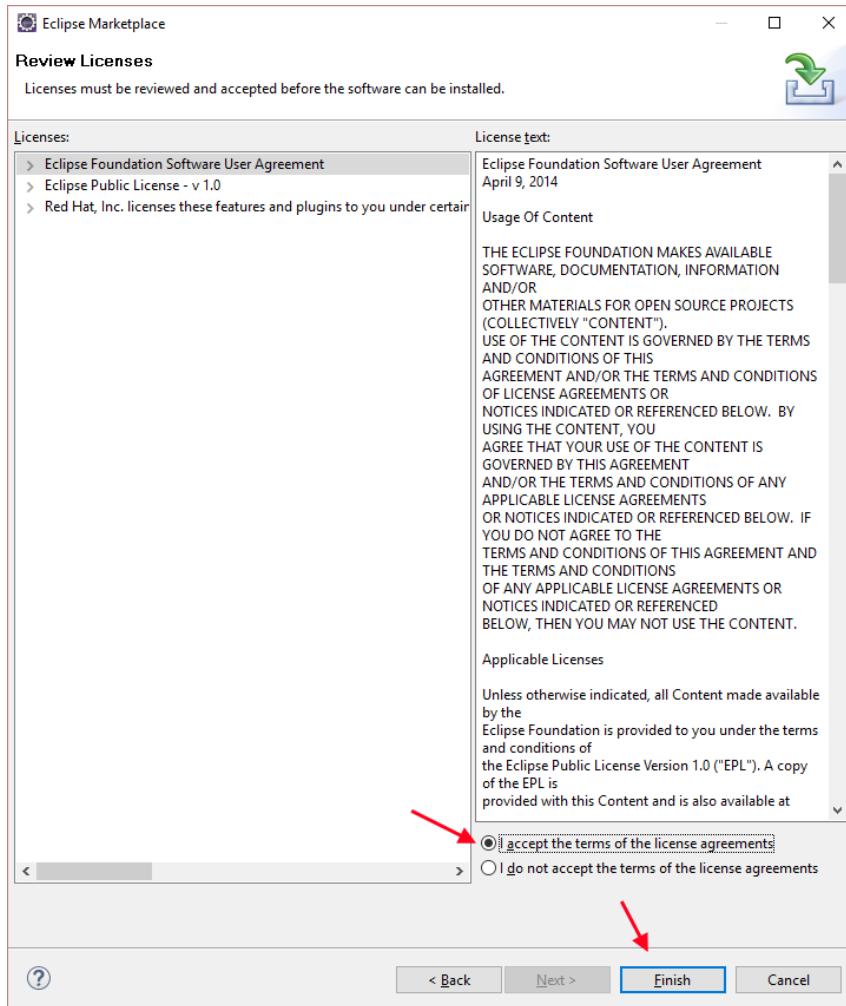
Red Hat JBoss Developer Studio 8.1.0.GA 

Single Development Tool, Tailored for Extreme Productivity. JBoss® Developer Studio provides superior support for your entire development lifecycle. It includes a... [more info](#)

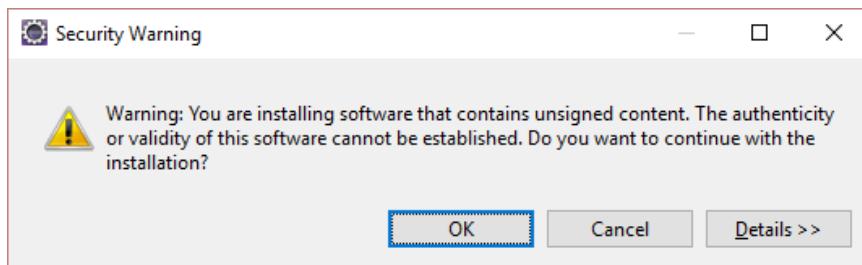
Paso 4: Se nos mostrará la siguiente ventana en donde podemos ver todas las funcionalidades que se le agregarán al entorno de Eclipse, damos clic en **Confirmar**.



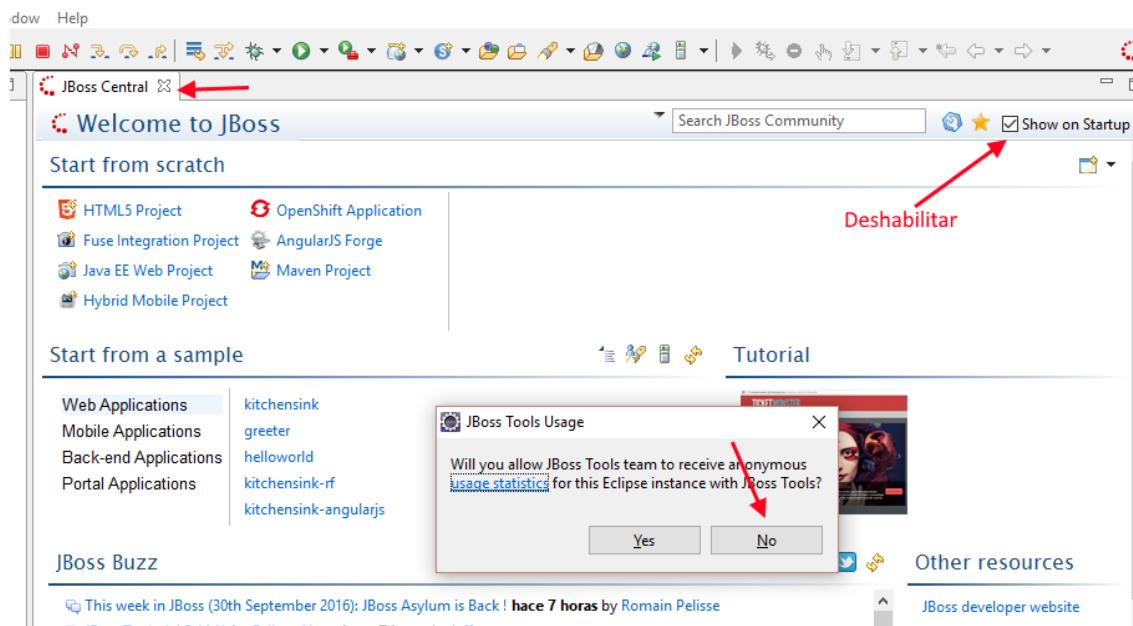
En la siguiente ventana aceptamos los términos y damos clic en **Finalizar**.



A la mitad de la instalación, se muestra la siguiente ventana, damos clic en **OK**, para que continúe la instalación:



Cuando termina la instalación, en el entorno de Eclipse se mostrará lo siguiente:



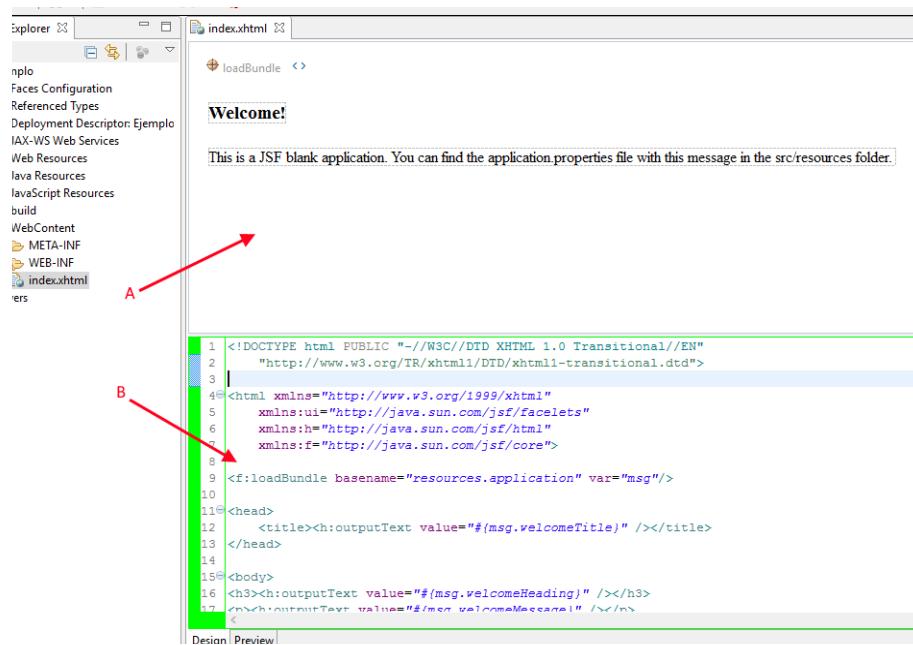
En la ventana **Jboss Tools Usage**, que sirve para enviar reportes a la empresa Red Hat (creador de este plug-in) acerca de problemas que pueda ocasionar dicho plug-in, es tu decisión Aceptar o negar dichos reportes, al final es anónimo.

En la parte superior derecha **desmarcamos** la casilla de **Show on Startup**, y cerramos esta pestaña, para que no la muestre cada vez que abramos Eclipse.

B.- Configurar editor de XHTML

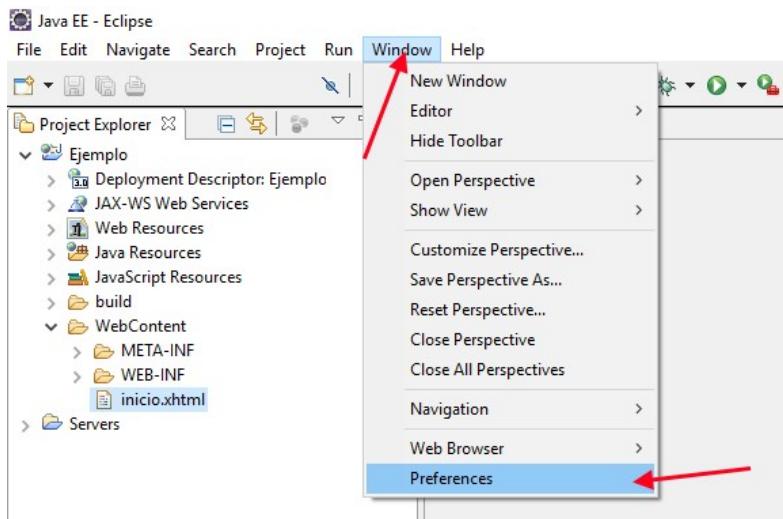
En este apéndice, veremos como abrir los archivos XHTML con un editor diferente al que viene por defecto, cuando damos doble clic sobre el archivo.

Si das doble clic en algún archivo XHTML (*después de haber instalado el plug-in de Jboss en el Apéndice A*) veras que lo abre con un editor de Jboss el cual muestra una pantalla dividida en: código y vista previa.



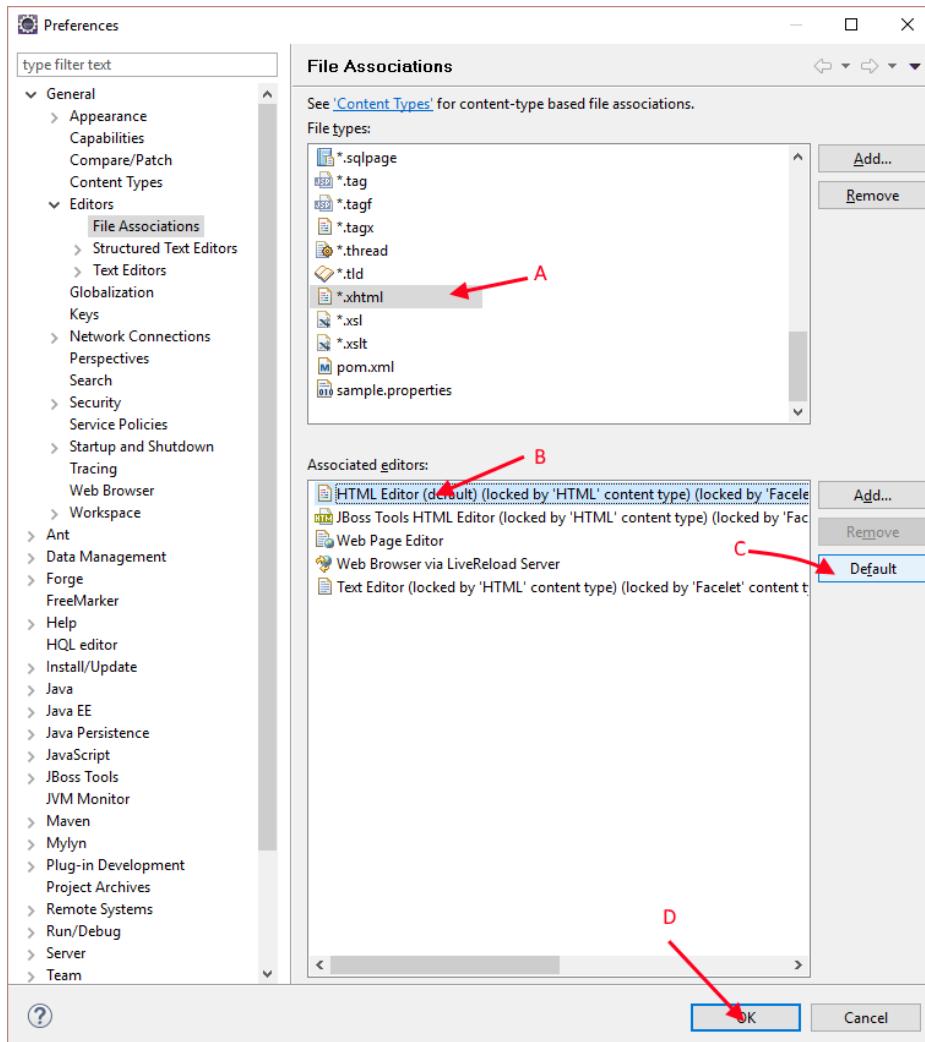
Esto no sirve de mucho y más que ayudar, disminuye la visualización de código, entonces veremos cómo cambiar este editor que tiene por defecto.

Para eso vamos al menú, seleccionamos **Window**, luego **Preferences**.



En esta ventana vamos a la sección General, Editors, File Associations. En el apartado de File Associations, buscamos *.xhtml, y en la parte de abajo Associated editors, seleccionamos:

HTML Editor, y damos clic en el botón **Default**. Y listo, ahora cuando des doble clic en un archivo XHTML solo mostrará el código, sin la vista previa.



C.- Driver Oracle y configuración JNDI para el datasource

El driver (librería **ojdbc6.jar**) de Oracle es necesario para la conexión a la base de datos, solo se encuentra disponible desde su sitio en Internet, no está disponible en el repositorio de Maven, por lo que tienes que descargarla directamente, como muestro a continuación.

Paso 1: Buscamos en Internet **Oracle Driver Jdbc** y seleccionamos la url que muestro en la imagen.

Google Oracle Driver jdbc

Todos Videos Imágenes Noticias Maps Más ▾ Herramientas de búsqueda

Cerca de 448,000 resultados (0.52 segundos)

JDBC/UCP Download Page - Oracle
www.oracle.com › Database › Database Features ▾ Traducir esta página
Older JDBC Driver Downloads - 10g (Unsupported). Oracle Database 10g Release 2 (10.2.0.5),
(10.2.0.4), (10.2.0.3), (10.2.0.2), (10.2.0.1.0) drivers ...

Oracle Database 11g Release 2 JDBC Driver Downloads
www.oracle.com/technetwork/apps.../jdbc-112010-090769.html ▾ Traducir esta página
JDBC Thin for All Platforms. Download ojdbc policy (10,591 bytes) - Sample security policy file for
Oracle Database JDBC drivers. Download JavaDoc ...

Paso 2: Aceptamos los términos y luego damos clic en la liga para descargar el archivo **ojdbc6.jar**

Technology Network > Applications

Pre-Built Integrations
Business Applications
Technology
Policy Automation
Oracle
Instruction and Engineering
Commerce
Communications

Oracle Database 11g Release 2 JDBC Drivers
You must accept the OTN License Agreement to download this software.
 Accept License Agreement | Decline License Agreement

Oracle Database 11g Release 2 (11.2.0.4) JDBC Drivers

SimpleFAN
simplefan.jar (20,365 bytes) - (SHA1 Checksum:
307a7e203d7e141964158d181ca849d512d7e710)
Classes for subscribing to RAC events via ONS; simplefan policy and javadoc

JDBC Thin for All Platforms
ojdbc.policy (10,591 bytes) - Sample security policy file for Oracle Database JDBC
drivers
JavaDoc (6,415,512 bytes)
README

ojdbc6.jar (2,739,670 bytes) - (SHA1 Checksum:
a483a046eee2f404d864a6ff5b09dc0e1be3fe6c)
Certified with JDK 8, JDK 7 and JDK 6: It contains the JDBC driver classes except
classes for NLS support in Oracle Object and Collection types.
ojdbc6_g.jar (4,494,956 bytes) - (SHA1 Checksum:
bf50af31967911af63058a6e1e5249c2dae34823)

Paso 3: Una vez descargado, **copiamos el archivo ojdbc5.jar** en la **carpeta lib** de Apache Tomcat, a continuación, muestro la imagen con la ruta (tú tienes que colocar en donde hayas descomprimido Tomcat):

Nombre	Fecha de modifica...	Tipo
annotations-api	01/09/2016 11:06 a...	Archivo Wi
catalina	01/09/2016 11:06 a...	Archivo Wi
catalina-ant	01/09/2016 11:06 a...	Archivo Wi
catalina-ha	01/09/2016 11:06 a...	Archivo Wi
catalina-storeconfig	01/09/2016 11:06 a...	Archivo Wi
catalina-tribes	01/09/2016 11:06 a...	Archivo Wi
ecj-4.5	01/09/2016 11:06 a...	Archivo Wi
el-api	01/09/2016 11:06 a...	Archivo Wi
jasper	01/09/2016 11:06 a...	Archivo Wi
jasper-el	01/09/2016 11:06 a...	Archivo Wi
jsp-api	01/09/2016 11:06 a...	Archivo Wi
ojdbc6	21/10/2016 06:26 ...	Archivo Wi
servlet-api	01/09/2016 11:06 a...	Archivo Wi

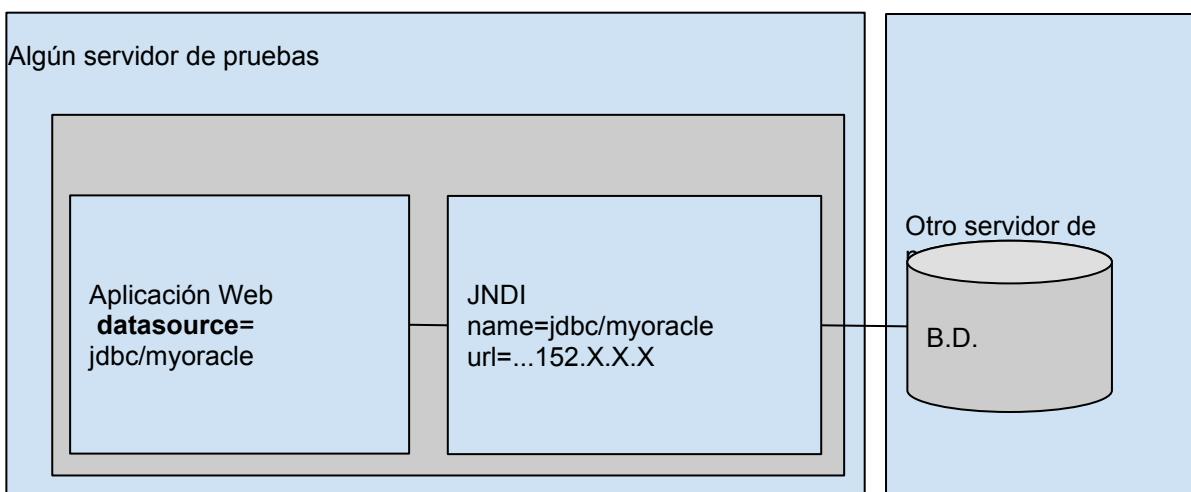
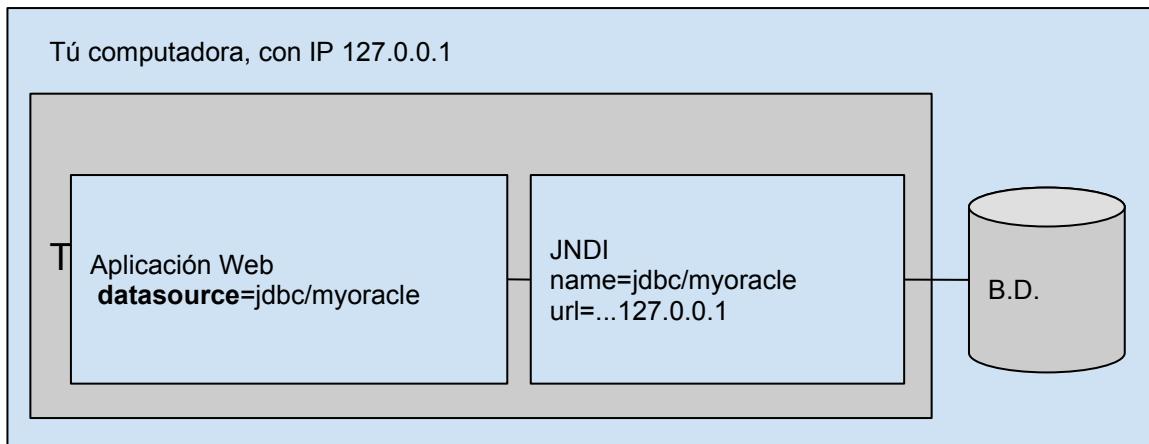
Hasta el paso anterior (paso 3) terminamos de instalar el driver para la conexión a la base de datos. Ahora lo que sigue será configurar un archivo **context.xml**, para crear un JNDI.

Configurar JNDI para el datasource

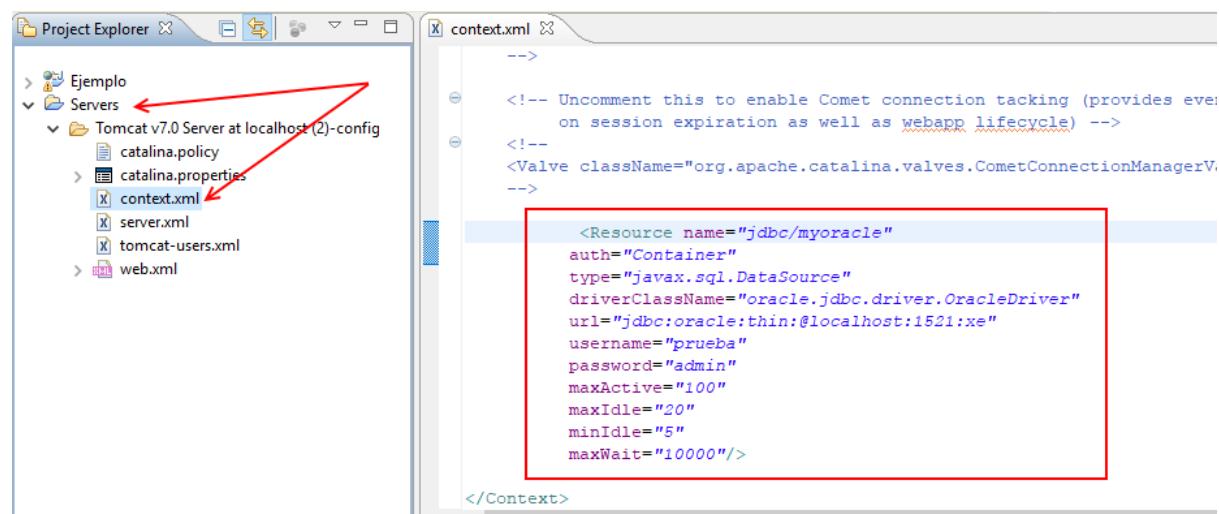
La siguiente configuración solo se hace una vez, cada vez que crees una instancia (copia) del servidor de aplicaciones Tomcat.

El JNDI es útil cuando se tienen servidores de aplicaciones (Tomcat) y una base de datos para cada ambiente: desarrollo, pruebas y producción, es decir, la dirección IP de la base de datos puede ser distinta para cada ambiente, por lo que es conveniente tener un nombre en común para el datasource.

Como puedes ver en la siguiente imagen, el JNDI se configura a nivel de Tomcat, y se hace referencia al nombre del JNDI desde la Aplicación Web (datasource). Es decir, cada vez que crees (.war) de la aplicación web, ésta tendrá el mismo nombre de JNDI, ya dependiendo del Tomcat en donde se despliegue la aplicación, se conectará a la base de datos indicada en la url del JNDI.



Para configurar el JNDI, necesitamos acceder al archivo **context.xml**, el cual se encuentra en nuestra instancia local en Eclipse, en el **Project Explorer** del lado izquierdo puedes ver una carpeta llamada **Servers**.



El bloque de código a introducir es el siguiente:

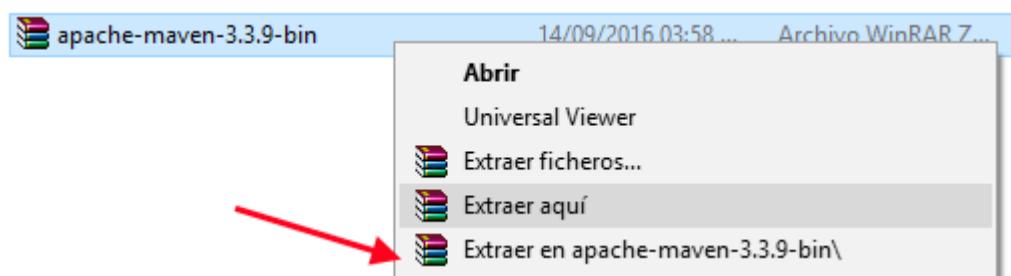
```
<Resource name="jdbc/myoracle" auth="Container"
  type="javax.sql.DataSource" driverClassName="oracle.jdbc.OracleDriver"
  url="jdbc:oracle:thin:@127.0.0.1:1521:XE"
  username="prueba" password="admin" maxTotal="20" maxIdle="10"
  maxWaitMillis="-1"/>
```

Nota: El atributo **username** y **password** deben de ser los usados para conectarse a la base de datos de Oracle ([Capítulo 8](#)).

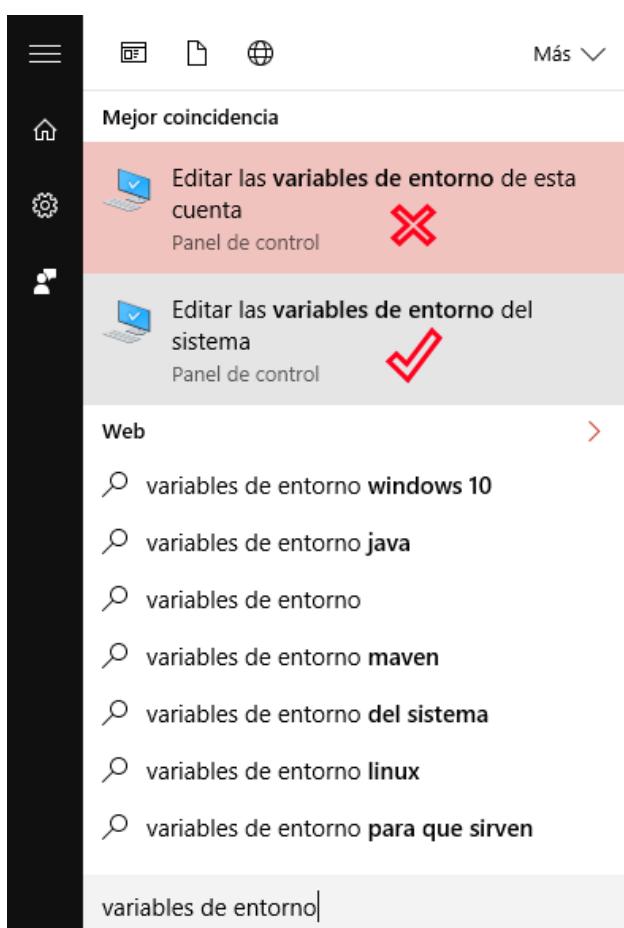
D.- Instalar y configurar Apache Maven

Apache Maven, que sólo lo usaremos para descargar librerías (archivos .jar), también consiste en descomprimir el archivo.

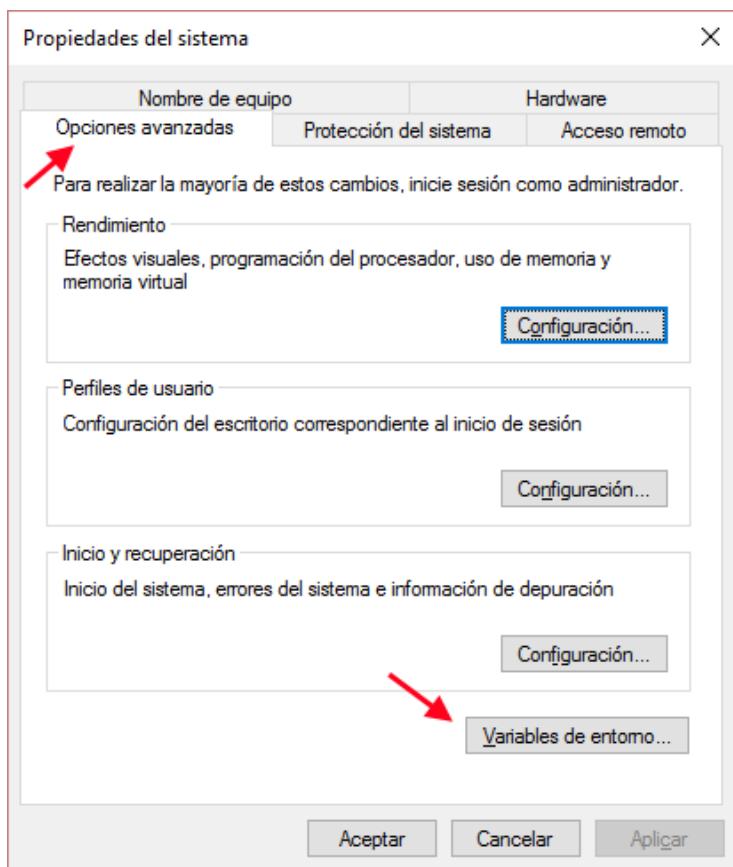
Paso 1: Descomprimir el archivo en raíz, de preferencia C.



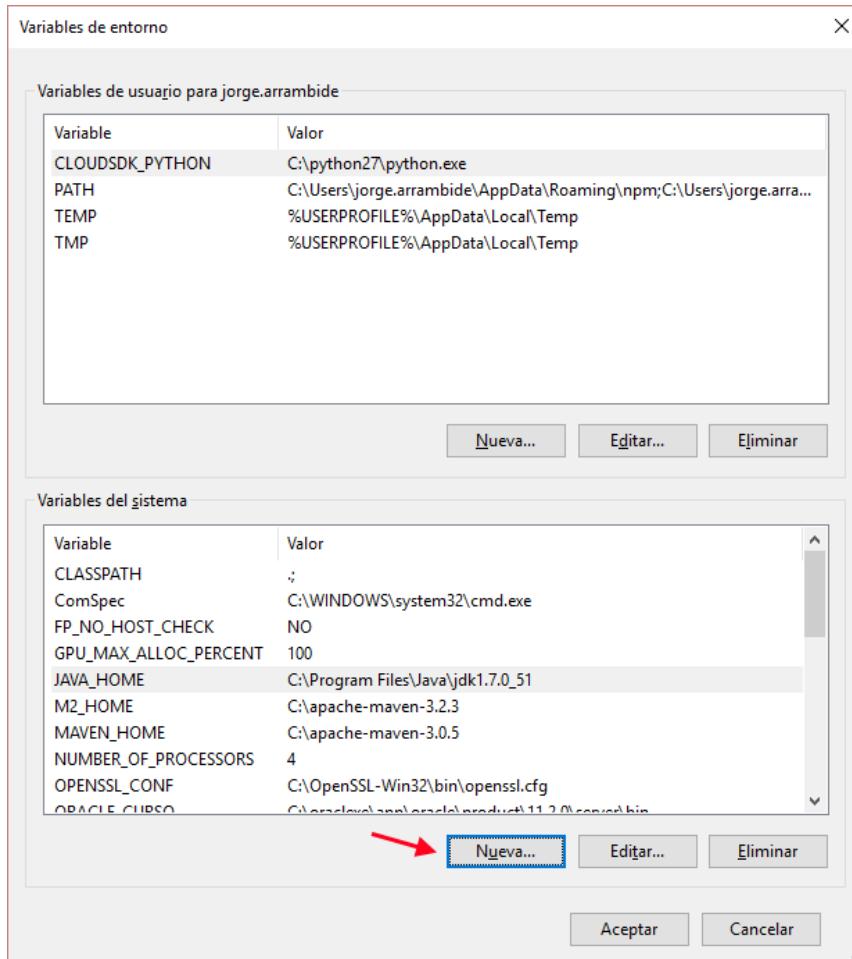
Paso 2: Modificar las variables de entorno del sistema.



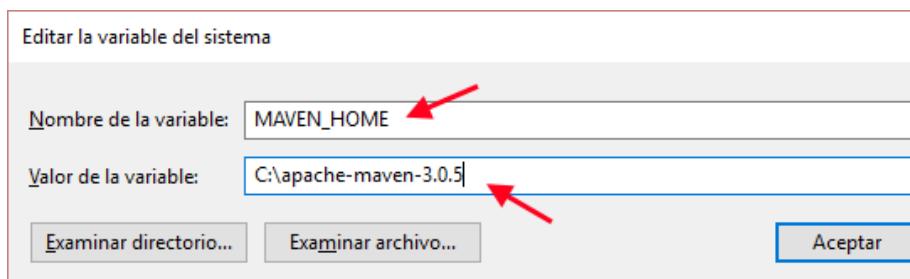
Paso 3: En la siguiente ventana posicionararse en la pestaña **Opciones avanzadas**, luego dar clic en **Variables de entorno**.



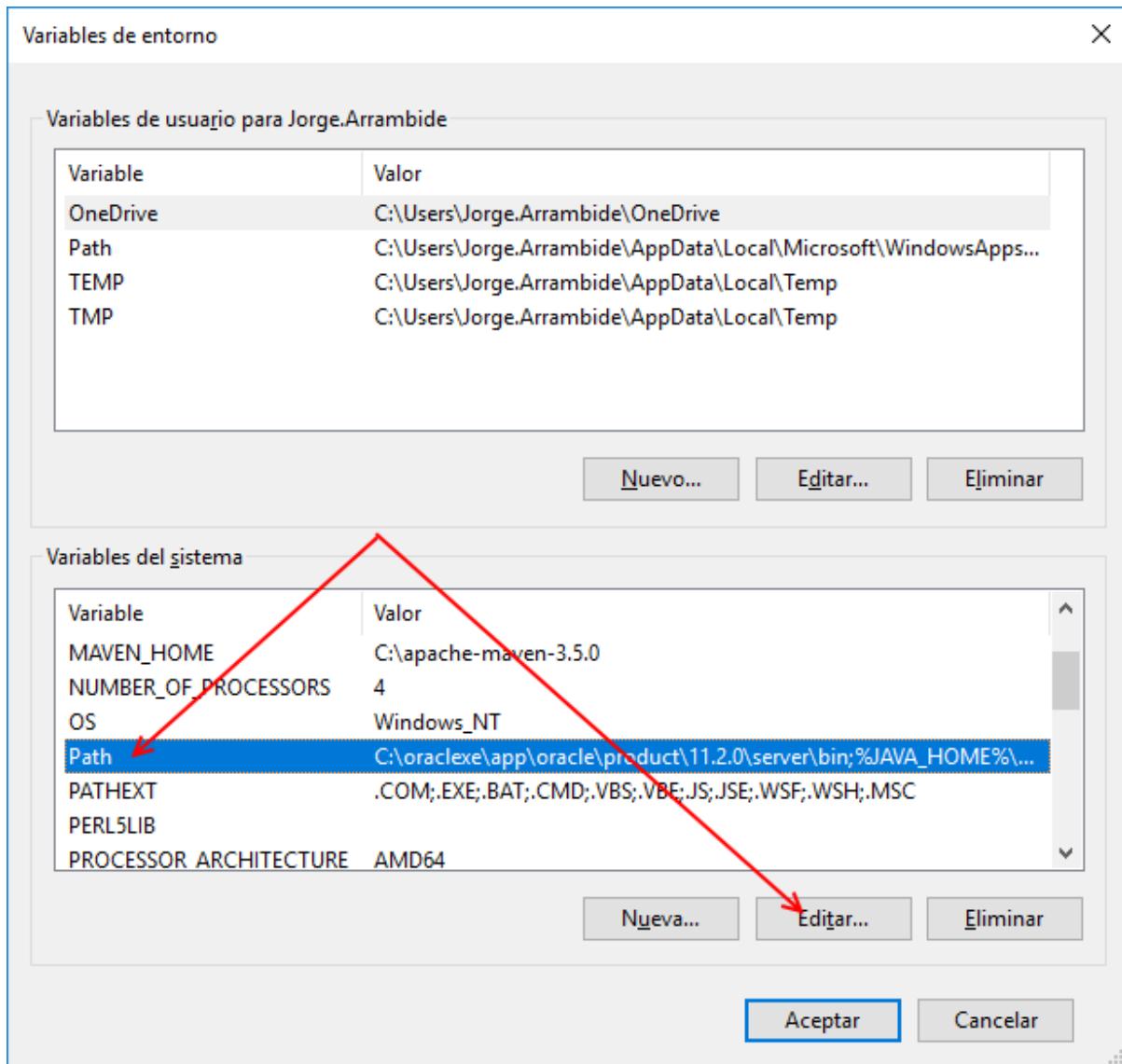
Paso 4: En la siguiente ventana dar clic en el botón **Nueva**, del cuadro **Variables del sistema**, como muestro en la imagen:



Paso 5: En la siguiente ventana colocamos como nombre de variable: **MAVEN_HOME** y en valor: **C:\'ruta de instalación'**. Finalmente damos clic en **Aceptar**.



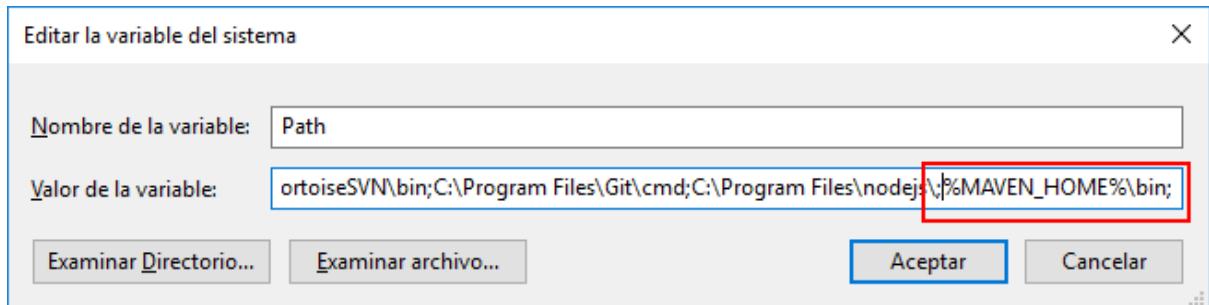
Paso 6: Colocamos esta nueva variable (**MAVEN_HOME**) en **Path**, como se muestra en la siguiente imagen:



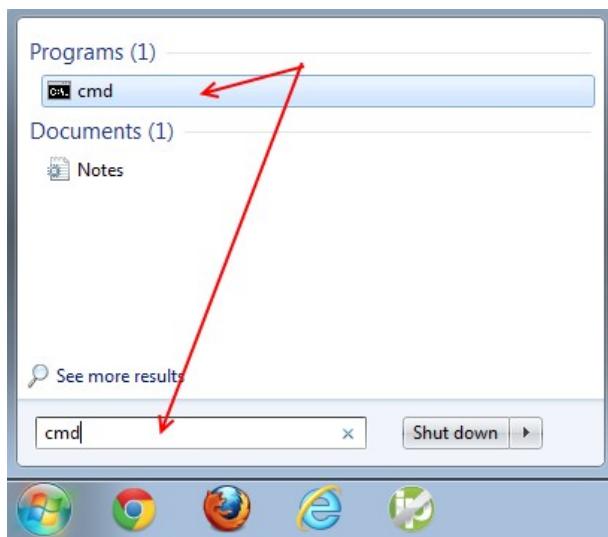
Paso 7: En la caja de texto Valor de la variable, colocamos al final lo siguiente y guarda los cambios:

`;%MAVEN_HOME%\bin;`

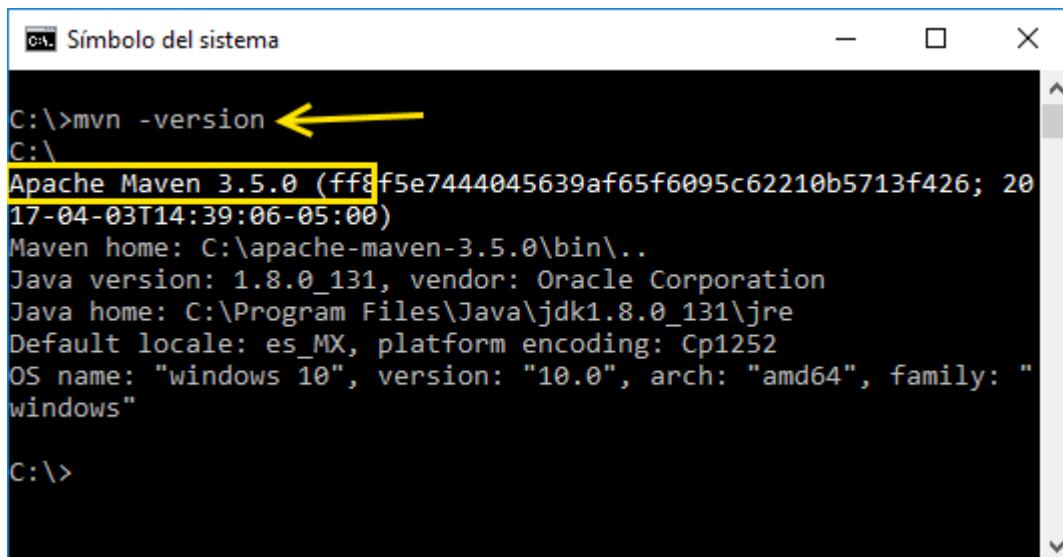
Nota: No olvides el símbolo de “punto y coma” al inicio y al final.



Paso 8: Verificar que lo hayas configurado correctamente, para ello vamos a la tecla Inicio de Windows y buscamos **CMD**. Muestro dos imágenes que corresponden a Windows 10 y a Windows 7.



Paso 9: Tecleamos la siguiente instrucción: **mvn -version**, en donde debemos de poder ver la versión de maven: 3.5.0 o la que hayas descargado. Si algo salió mal revisa los pasos anteriores (mayúsculas, minúsculas,error al teclear, etc.)



```
C:\>mvn -version ←
C:\_
Apache Maven 3.5.0 (ff8f5e7444045639af65f6095c62210b5713f426; 20
17-04-03T14:39:06-05:00)
Maven home: C:\apache-maven-3.5.0\bin\..
Java version: 1.8.0_131, vendor: Oracle Corporation
Java home: C:\Program Files\Java\jdk1.8.0_131\jre
Default locale: es_MX, platform encoding: Cp1252
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"

C:\>
```

E.- Conociendo el entorno de desarrollo Eclipse

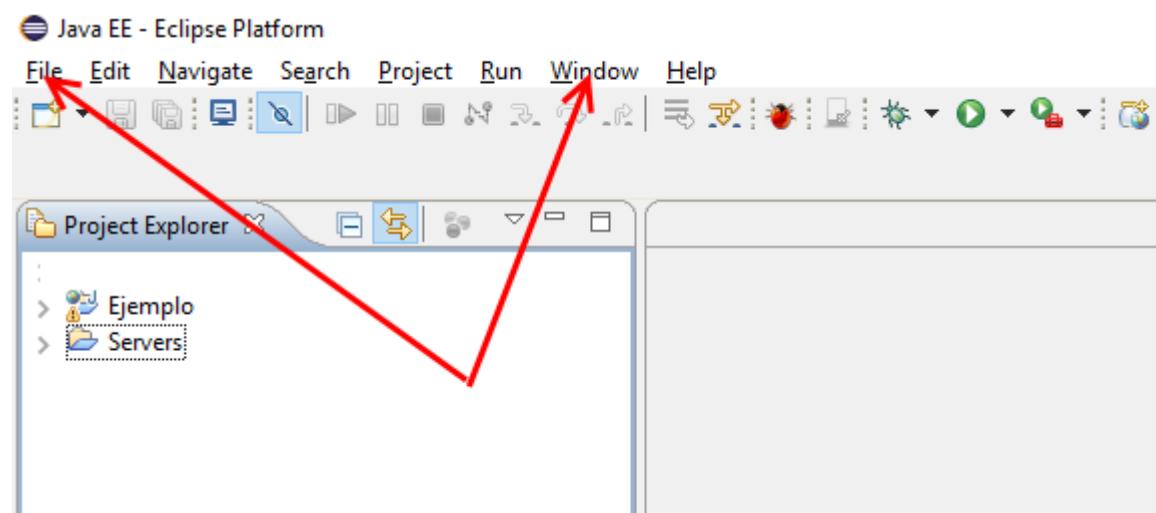
A continuación te mencionaré los aspectos básicos más usados en este IDE.

Comenzamos en la parte superior izquierda:

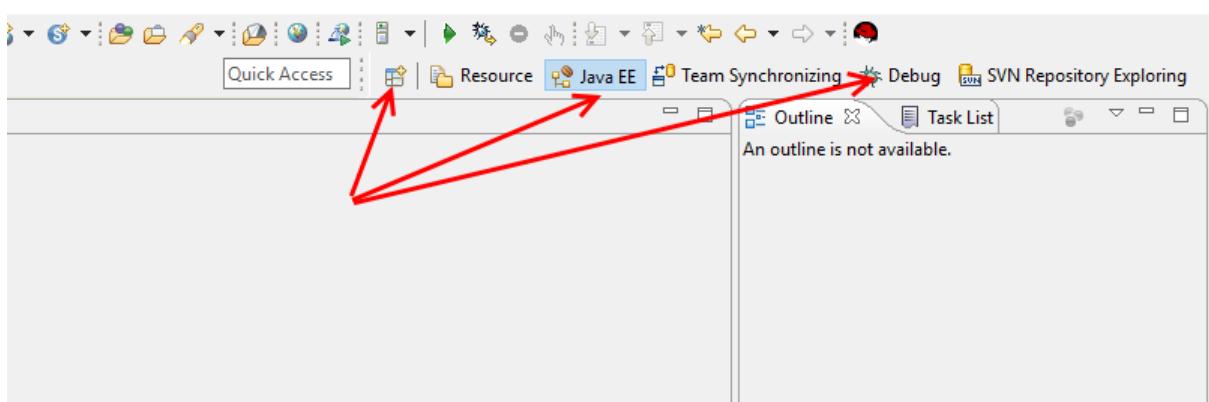
En la barra de menú lo que más usarás será File y Window

En **File** está lo que normalmente encontramos como lo es: archivos recientes, cambiar de workspace.

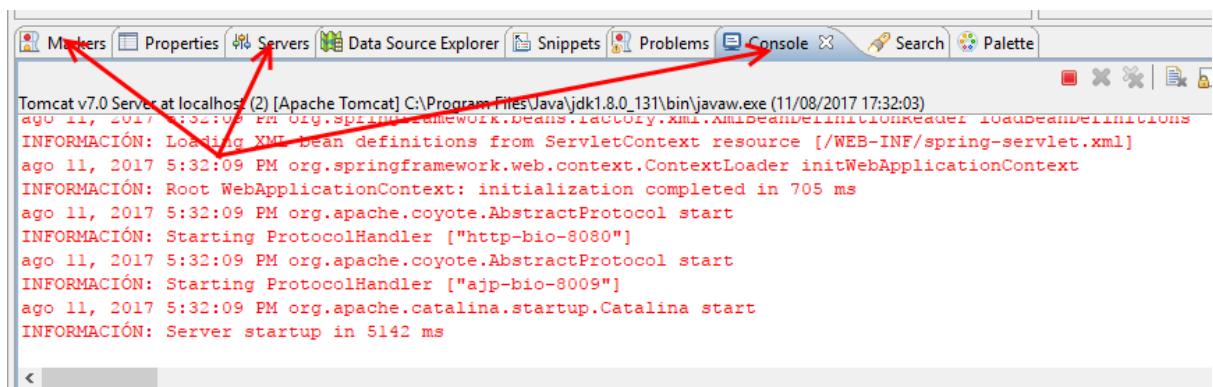
En **Window**, encontraremos Preferences, Show View, Open Perspective, entre otros. Mencione estos tres porque los considero los más usados, en caso de que quieras cambiar alguna preferencia como estilo de letra o fondo (Preferences); mostrar alguna vista diferente a Java EE, como por ejemplo cambiar a vista de Debug (Show Perspective); o abrir alguna pestaña que hayas cerrado por accidente o bien que en alguna Perspectiva quieras mostrar alguna pestaña que no tiene por default (Show View).



En el lado superior derecho lo que más usarás es el icono que señalo en la imagen, que basicamente hace que cambies de Perspectiva (Java EE, Debug, etc).



En la parte de abajo están varias pestañas, de las cuales las más usadas son: Markers, Servers y Console.



En **Markers** encontrarás alertas que eclipse encontró a uno o varios proyectos. Dependiendo del tipo de alerta (Error o Warning) lo puedes atender o ignorar. Por lo general puedes ignorar los warning.

En **Servers**, encontrarás los servidores de aplicaciones que hayas configurado, puedes tener tantos como necesites.

En la pestaña **Console** (en caso de que no la veas, se muestra cuando “despliegas” el proyecto web en Tomcat), encontrarás mensajes de diferentes tipos llamados Logs. Estos logs nos muestran información, precauciones o errores al momento de desplegar el proyecto web, en el servidor de aplicaciones. Esta pestaña es la más usada, ya que allí consultamos los errores.