

# ➤ Despliegue de una aplicación JEE

---

**Plan formativo:** Desarrollo de Aplicaciones Full Stack Java Trainee V2.0

# HOJA DE RUTA

¿Cuáles **skill** conforman el programa?



# REPASO CLASE ANTERIOR

En la clase anterior trabajamos :

- ✓ Concepto de MVC
- ✓ Componentes del MVC

# LEARNING PATHWAY

¿Sobre qué temas trabajaremos?

5.6

Start! 🏁

**Despliegue de una  
aplicación JEE**

Tomcat

**Personalizando el  
descriptor**

Personalizando el descriptor de  
despliegue web.xml  
Conociendo el servidor Tomcat  
Despliegue de una aplicación web  
con Tomcat Manager

# OBJETIVOS DE APRENDIZAJE

¿Qué aprenderemos?

- *Aprender a personalizar el descriptor de despliegue*
- *Conocer el servidor Tomcat*
- *Desplegar una aplicación con Tomcat Manager*



# › Despliegue de una aplicación JEE



# Despliegue de una aplicación JEE



Normalmente cuando se trabaja sobre un proyecto web J2EE se hace sobre una herramienta de desarrollo como Eclipse IDE, que para probar la aplicación realiza un despliegue automático para poder hacer pruebas de forma rápida.



Cuando tenemos la aplicación terminada el despliegue se hará sobre un servidor de producción. Este servidor es el que usarán los usuarios de la aplicación Web. Este despliegue normalmente se realiza utilizando un archivo WAR.



# › Personalizando el descriptor de despliegue web-xml





# Personalizando el descriptor de despliegue web-xml



**Algunos parámetros que podemos configurar son:**

- Nombre de la aplicación
- Session timeout
- Encoding de caracteres
- Parámetros de conexión a BD
- Declaración de servlets y mapping de URLs



Es necesario actualizar el descriptor de despliegue para registrar nuestros servlets con el contenedor.



# Personalizando el descriptor de despliegue web-xml



```
*web.xml X
1 <?xml version="1.0" encoding="UTF-8"?>
2 <web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"
3     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4     xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
5         http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd"
6     version="4.0">
7
8     <!-- Configuración del servlet -->
9     <servlet>
10         <servlet-name>ProductServlet</servlet-name>
11         <servlet-class>src.main.ProductController</servlet-class>
12     </servlet>
13
14     <!-- Mapeo del servlet a una URL -->
15     <servlet-mapping>
16         <servlet-name>ProductServlet</servlet-name>
17         <url-pattern>/product</url-pattern>
18     </servlet-mapping>
19
20     <!-- Otras configuraciones, si las hubiera -->
21
22 </web-app>
```



Insertamos nuestros descriptores de despliegue de servlets entre las etiquetas `<web-app>` y `</web-app>` del fichero:



# Personalizando el descriptor de despliegue web-xml





De esta manera, hemos creado nuestra primera aplicación Web que contiene un sólo servlet. Ahora deberíamos poder desplegar esta aplicación en cualquier contenedor servlet compatible. De esta forma sabremos cómo trabajar con ellas y cómo desplegar nuestras aplicaciones Web en un modo de desarrollo.






# ¿Que es Despliegue (Deployment) de una aplicación?

Normalmente cuando se trabaja sobre un proyecto web J2EE se hace sobre una herramienta de desarrollo como eclipse y para probar la aplicación la herramienta suele hacer un despliegue automático para poder hacer pruebas de forma rápida.



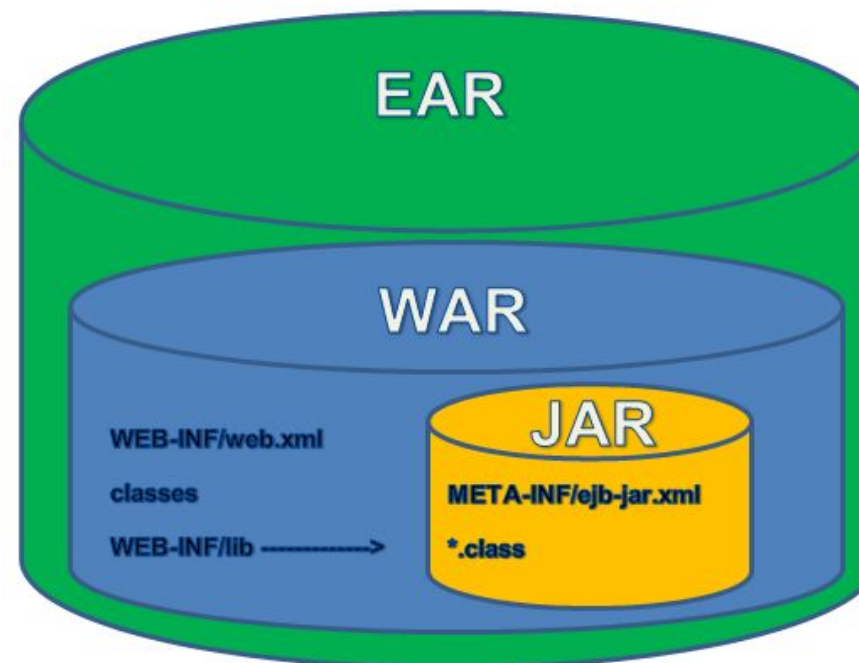
Cuando tenemos la aplicación terminada el despliegue se hará sobre un servidor de producción. Este servidor es el que usarán los usuarios de la aplicación Web. Este despliegue normalmente se realiza utilizando un archivo WAR.





# Crear War

Un fichero WAR simplemente es un fichero JAR con la extensión cambiada para reflejar su propósito diferente. Una buena razón para utilizar archivos \*.war es que normalmente los equipos de desarrollo de aplicaciones y los de instalación están conformados por distintas personas. Enviar un solo archivo para desplegar es más sencillo y presta menos confusiones que enviar varios.





# Crear War

Eclipse JEE permite generar el archivo WAR de la aplicación seleccionando el proyecto, **click en botón derecho - Export -WAR file.**

Para crear un WAR para nuestra aplicación de manera manual , vamos al directorio raíz que contiene nuestra aplicación Web y tecleamos el siguiente comando:

**jar cv0f myWebApp.war .**

Observa el punto obligatorio que hay al final de la línea; le dice al programa jar que archive el directorio actual.



El comando jar anterior creará un fichero WAR llamado myWebApp.war.

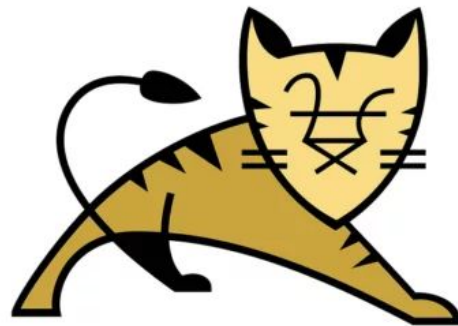


# › Conociendo el servidor Tomcat



# Conociendo el servidor Tomcat

Apache Tomcat es un contenedor web open source que implementa las especificaciones de Java EE para aplicaciones web. Nos permite desplegar y ejecutar aplicaciones desarrolladas con Java.



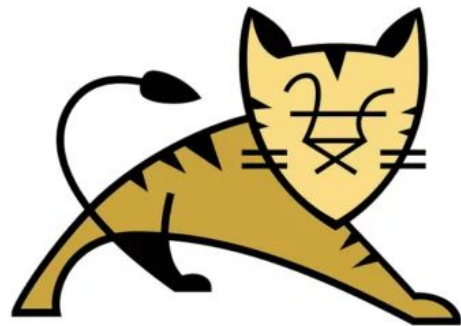
## Apache Tomcat





# Conociendo el servidor Tomcat

Apache Tomcat (o, sencillamente, Tomcat) es un contenedor de servlets que se puede usar para compilar y ejecutar aplicaciones web realizadas en Java. Implementa y da soporte tanto a servlets como a páginas JSP (Java Server Pages) o Java Sockets.



## Apache Tomcat



# Conociendo el servidor Tomcat



## Características principales:

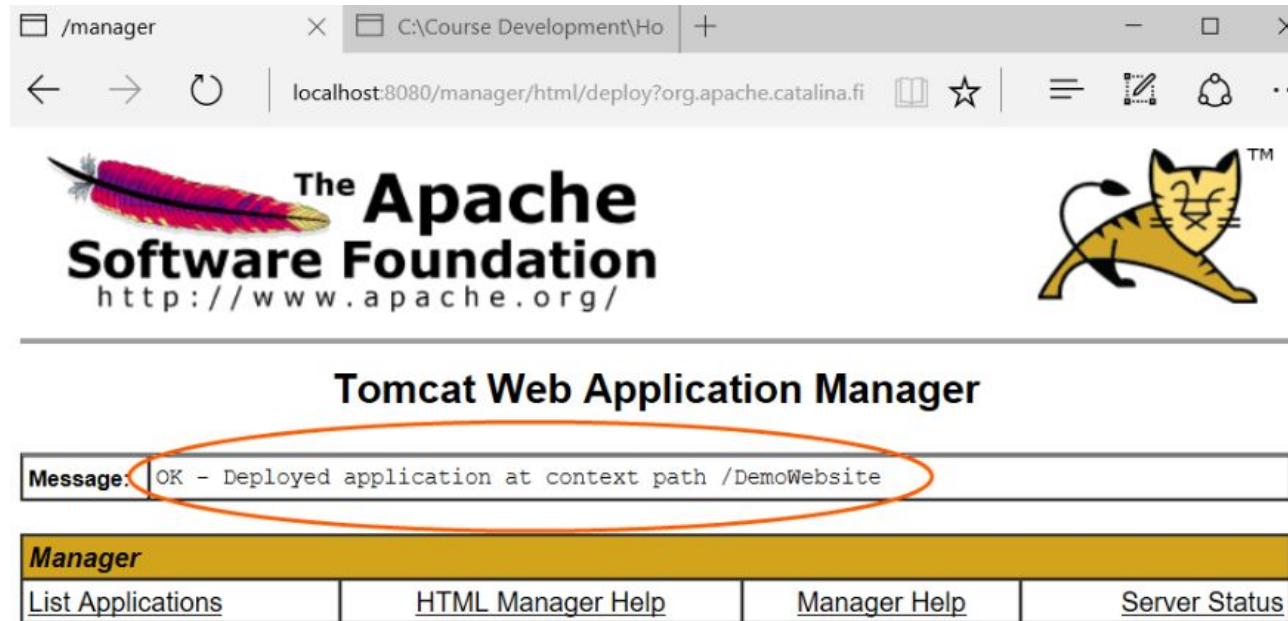
- Soporte para servlets y JSPs
- Contenedor web ligero y eficiente
- Uso extendido en aplicaciones Java
- Código abierto y gratuito
- Fácil configuración e integración
- Administración vía interfaz web



Tomcat nos provee todo lo necesario para desplegar y servir nuestra aplicación Java EE en un entorno productivo.

# ➤ Despliegue de una aplicación web con Tomcat Manager

# Despliegue de una aplicación web con Tomcat Manager



El Tomcat Manager es una aplicación web que viene incluida con el servidor Apache Tomcat. Permite realizar tareas de administración, como desplegar, detener, reiniciar o eliminar aplicaciones web de manera remota.

# Despliegue de una aplicación web con Tomcat Manager

## - Iniciar sesión:

Para acceder al Tomcat Manager, necesitas proporcionar credenciales de administrador. Estos datos se establecen en el archivo tomcat-users.xml, que se encuentra en la carpeta conf de tu instalación de Tomcat.

Asegúrate de tener un usuario con el rol manager-gui definido en ese archivo. Por ejemplo:

```
<tomcat-users>
  <user username="admin" password="password" roles="manager-gui"/>
</tomcat-users>
```

Reemplaza "admin" con tu nombre de usuario y "password" con tu contraseña.

# Despliegue de una aplicación web con Tomcat Manager

## - Acceder al Tomcat Manager:

Abre un navegador web y ve a la URL del Tomcat Manager. Por lo general, lo encuentras en la ruta: <http://localhost:8080/manager>.

Elegimos la opción superior derecha de Manager y nos solicitará un usuario y una clave para acceder al manager , introducimos nuestras credenciales y accedemos al manager en donde tenemos las aplicaciones por defecto.

Después de iniciar sesión, verás una lista de aplicaciones web desplegadas en el servidor Tomcat. Verás sus nombres de contexto y estados (activo, detenido, etc.).

localhost:8080/manager/html




## Gestor de Aplicaciones Web de Tomcat

Mensaje: OK

**Gestor**

[Listar Aplicaciones](#)
[Ayuda HTML de Gestor](#)
[Ayuda de Gestor](#)
[Estado de Servidor](#)

**Aplicaciones**

Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Ninguno especificado	Welcome to Tomcat	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/docs	Ninguno especificado	Tomcat Documentation	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/examples	Ninguno especificado	Servlet and JSP Examples	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/manager	Ninguno especificado	Tomcat Manager Application	true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos

**Desplegar**

Desplegar directorio o archivo WAR localizado en servidor

Trayectoria de Contexto (opcional):   
 URL de archivo de Configuración XML:   
 URL de WAR o Directorio:




# Despliegue de una aplicación web con Tomcat Manager



Una vez tenemos esto, **en la parte inferior de la pantalla podemos desplegar una aplicación nueva con un war** (web1.war) y aparecerá en la lista.



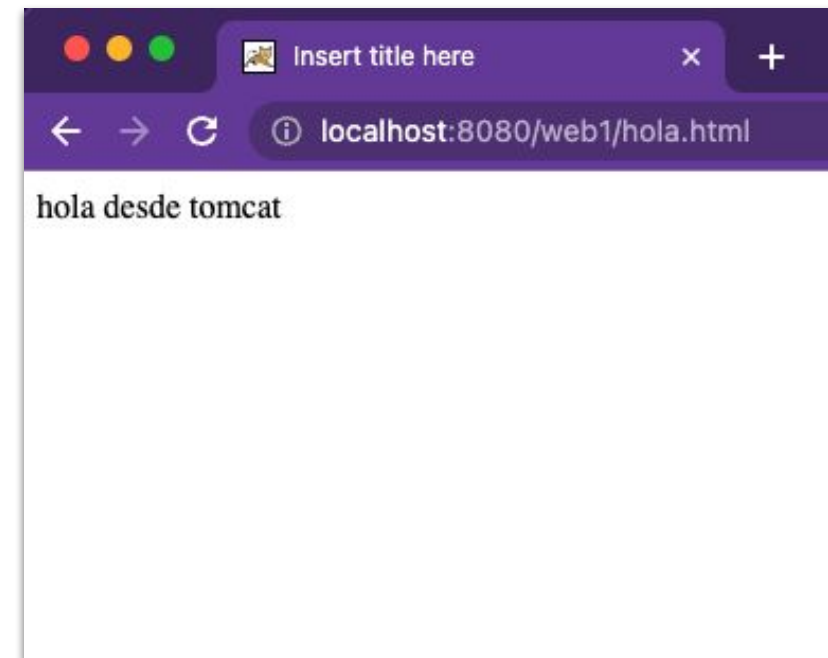
Haz click en el botón "**Choose File**" (o similar) para seleccionar el archivo WAR de tu aplicación que desees desplegar en Tomcat. Luego, haz clic en el botón "**Deploy**". Tomcat procesará el archivo WAR y desplegará la aplicación.





# Despliegue de una aplicación web con Tomcat Manager

Una vez estamos dentro del Tomcat Manager y la aplicación aparece en la lista, nos queda acceder a ella y cargar un ejemplo de **hola.html** que nos dirá que la aplicación funciona dentro de Tomcat.





# Despliegue de una aplicación web con Tomcat Manager



En la página del Tomcat Manager, puedes hacer clic en el nombre de una aplicación para acceder a información más detallada, como los registros (logs) de la aplicación y otras opciones de configuración.



Es importante tener en cuenta que el acceso al Tomcat Manager debe estar restringido a usuarios autorizados, ya que es una herramienta poderosa que permite realizar cambios en el servidor.



# Evaluación Integradora ✨

¿Listos para un nuevo desafío? En esta clase comenzamos a construir nuestro...

## Trabajo Integrador del Módulo 💪

Iremos completándolo progresivamente clase a clase.





# **Ejercicio N° 1**

# Deployando



# Deployando

## Manos a la obra: 🙌

Vas a realizar el despliegue de una aplicación básica en nuestro IDE Eclipse y utilizando Tomcat Manager.

## Consigna: 📝

- 1- Crear las clases necesarias para interactuar con UserDAO, AccountDAO y CurrencyDAO .
- 2- Crear una nueva BD Wallet e insertar registros.
- 3- Modificar el nombre de al menos dos usuarios.

**Tiempo** 🕒: 30 minutos



○

# ¿Alguna consulta?

+



# RESUMEN

¿Qué logramos en esta clase?

- ✓ **Comprender el concepto de *Despliegue de aplicación***
- ✓ **Aprender a utilizar *Tomcat y Tomcat Manager***



# #WorkingTime

Continuemos ejercitando

**¡Antes de cerrar la clase!** Te invitamos a: 🙌 🙌 🙌

1. Repasar nuevamente la grabación de esta clase
2. Revisar el material compartido en la plataforma de Moodle (lo que se vio en clase y algún ejercicio adicional)
  - a. *Lectura Modulo 5, Lección 6: páginas 1 - 5*
3. Traer al próximo encuentro, todas tus dudas y consultas para verlas antes de iniciar nuevo tema.



# ¡Muchas Gracias!

Nos vemos en la próxima clase 🙌



Momento: ✚

# Time-out!

🕒 5 min.

