DAW Práctica 3.4: Despliegue de una aplicación con Apache Tomcat

En esta actividad vamos a realizar un despliegue de una aplicación básica con Apache Tomcat.

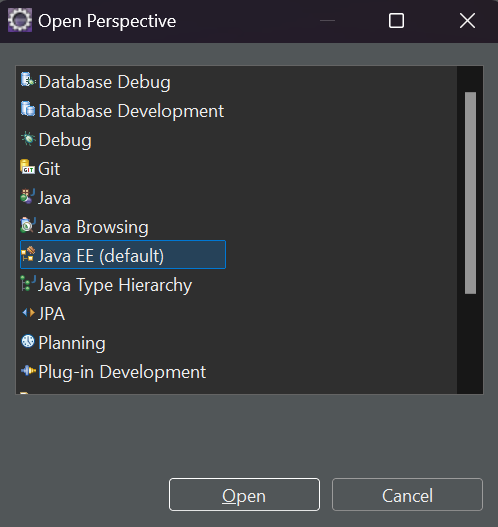
**Procedimiento:**

Instrucciones:

1. Si no lo tienes en tu equipo descarga el IDE Eclipse escogiendo la opción

“Eclipse IDE for Enterprise Java and Web Developers”.

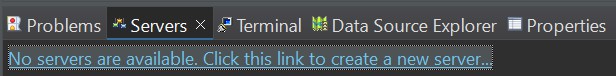
1. Integraremos Eclipse con Tomcat, para ello:
   1. Por defecto aparece la perspectiva de Java EE (herramientas para el desarrollo de aplicaciones). En caso de no tenerla debemos entrar en el menú en Windows->Perspective->Open Perspective->Other y buscamos Java EE.



Interfaz de usuario gráfica, Texto, Aplicación

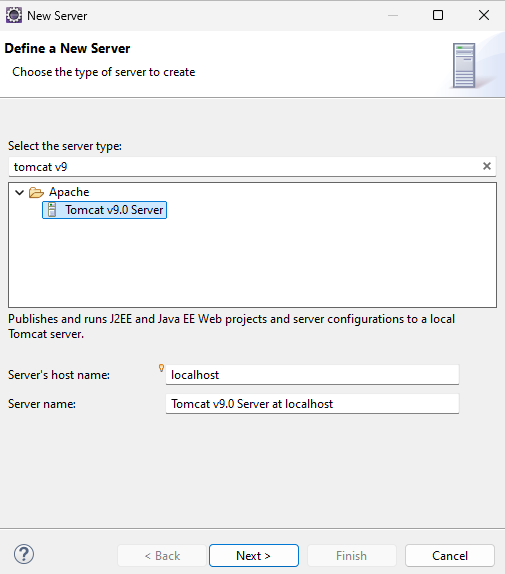
Descripción generada automáticamente

* 1. En la parte inferior, en el menú **Window->Show View** haz click en la opción Servers.



* 1. Añadir un servidor nuevo. Vamos a escoger Apache Tomcat 9.

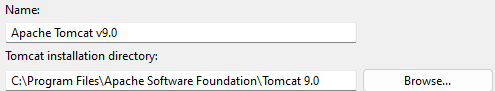
**Observa los servidores de aplicación que nos permite instalar**.



Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

* 1. Especifica el nombre que quieres dar al servidor, selecciona el directorio de instalación de Tomcat 9 (C:\Program Files\Apache Software Foundation\Tomcat 9.0) y haz click en Finalizar.



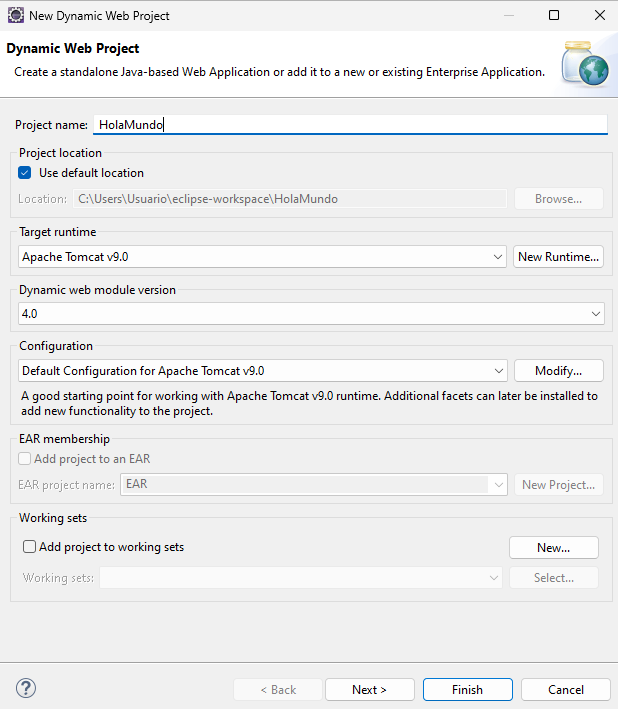
Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

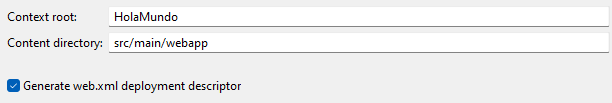
1. Realizaremos una aplicación en eclipse para su posterior despliegue con Tomcat.
   1. Para ello, abrimos Eclipse y creamos un nuevo Dyanamic Web Project llamado HolaMundo.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

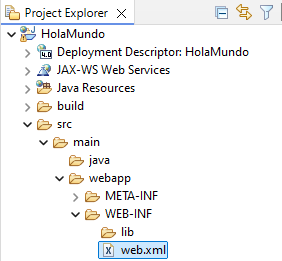


Haz click en siguiente hasta que aparezca la opción “**Generate web.xml deployment descriptor**”. Marca la opción para generar un descriptor web.xml.



Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

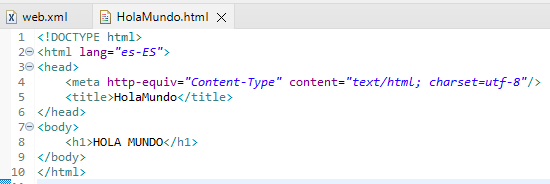
Descripción generada automáticamente

* 1. ****Observa la estructura de carpetas del proyecto. Verás que dentro de la carpeta WEB-INF ha creado el archivo web.xml. **¿Cuál es la función de este archivo?**

También se puede acceder al contenido desde el desplegable

“Deployment Descriptor: HolaMundo”.

* 1. **META-INF** contiene el MANIFEST.MF mapea las clases de ficheros JAR existentes en otros proyectos pertenecientes al mismo Enterprise Application Project.
  2. En **Java Resources** irán los Servlets y los ficheros .java.
  3. Genera un archivo dentro de src->main->webapp que se llame HolaMundo.html (Botón derecho, New, HTML File) con un contenido similar al siguiente:



Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

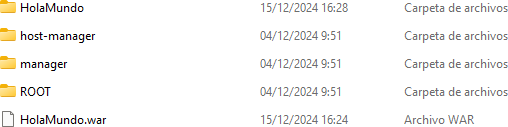
* 1. Ahora exporta el proyecto como archivo WAR. Para ello, vamos a File

-> Export y en el desplegable Web seleccionamos WAR file, el proyecto que deseamos exportar y el destino. **¿Qué es un archivo WAR?**

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* 1. Copia el archivo WAR en *RutaTomcat*\webapps.
  2. Observa que al poco de copiar el archivo WAR se crea la carpeta HolaMundo en la misma carpeta de webapps. Navega por la carpeta y comprueba que el contenido es el mismo que tenías en el proyecto de Eclipse.



Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

* 1. Comprueba que la web ha sido desplegada automáticamente. Para ello intenta acceder a <http://localhost:8080/HolaMundo/HolaMundo.html>

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1. Repetiremos el proceso utilizando un servlet.
   1. En el mismo proyecto nos colocamos sobre el directorio “Java Resources\src”. Hacemos click con el botón derecho, click en New y seleccionamos Servlet.
   2. Crearemos uno nuevo llamado HolaMundoServlet.

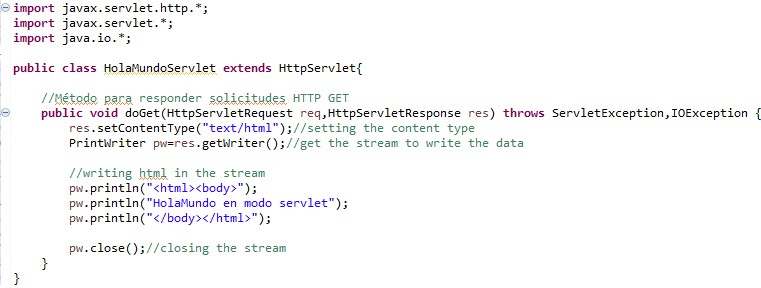
Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

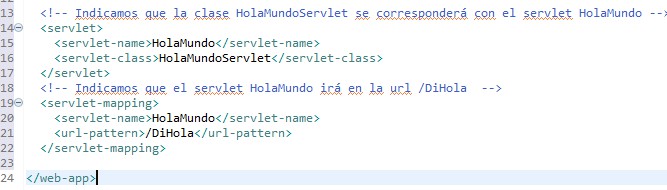
* 1. Observa que ha creado un archivo .java con una plantilla de Servlet que nos podría servir para crear uno nuevo.



* 1. Sin embargo, vamos a modificar el Servlet para que contenga el siguiente código:



* 1. Modifica el archivo web.xml añadiendo el siguiente contenido dentro del elemento **<web-app>:**

****

* 1. Para cada servlet incluido en la aplicación habrá que crear un elemento similar.
  2. A continuación, exportamos el proyecto de nuevo a un archivo .WAR y repetimos el proceso visto en puntos anteriores relacionados con el archivo WAR. Comprobamos que ha funcionado accediendo desde un navegador a la ruta que habéis configurado.

1. Repite el proceso generando tu un propio servlet. Puedes fijarte en propuestas de internet.