

## GUIA RÁPIDA: instalación Eclipse-Tomcat

### **1. Instalación**

#### **Instalación de Java 2 JDK SE**

El paquete JDK puede bajarse de <http://java.sun.com>.

Para instalar, ejecutar **j2sdk-1\_4\_2\_12-windows-i586-p.exe**

#### **Instalación de Jakarta Tomcat**

Tomcat puede bajarse de <http://jakarta.apache.org/tomcat>.

Para instalar, ejecutar **jakarta-tomcat-5.5.9.exe**

Como usamos la versión 1.4 de Java es necesario descargarse el paquete Compaq de Tomcat para que sea compatible con esta versión, **jakarta-tomcat-5.5.9-compat.zip** Y descomprimirlo en el directorio donde se nos ha instalado el Tomcat (reemplazar ficheros coincidentes).

#### **Instalación de Eclipse**

El IDE Eclipse puede bajarse de <http://www.eclipse.org>.

Para instalar, descomprimir eclipse-SDK-3.1.1-win32.zip en un directorio del disco local

#### **Instalación del plugin de Eclipse para Tomcat**

El plugin puede bajarse de <http://www.sysdeo.com/eclipse/tomcatPlugin.html>.

Para instalar, descomprimir **tomcatPluginV31beta.zip** dentro del directorio **/plugins** de Eclipse.

Ejecutar *eclipse.exe*

Entrar al menú **Window/Preferences/Tomcat**.

Marcar nuestra versión correspondiente de Tomcat, "**Tomcat 5.0.x**" y rellenar el campo "**Tomcat Home**" con el directorio donde se ha instalado Tomcat:

*C:\Archivos de programa\Apache Software Foundation\Tomcat 5.5*

## 2. Testear Tomcat

### **Ejecutar Tomcat**

Para iniciar Tomcat, hacemos click en el icono "Start Tomcat" de Eclipse o bien en la pestaña *Tomcat/Start Tomcat*.

Abrimos el navegador web y escribimos en el campo de dirección la url <http://localhost:8080/index.html>

Deberá aparecer la página por defecto de Tomcat, que nos indica que ha sido instalado correctamente.

Para parar Tomcat, hacemos click en el icono "Stop Tomcat" o bien en la pestaña *Tomcat/Stop Tomcat*.

## 3. Crear un proyecto

Para crear un proyecto nuevo, seleccionamos desde Eclipse *File/New/Project/Java* y elegimos **Tomcat Project**.

Escribimos un nombre para el proyecto, por ejemplo "tomcat1". Seleccionamos *Next*.

En la segunda pantalla, nos aseguraremos de que quede activada la casilla "Can update server.xml file". De esta forma Eclipse modificará automáticamente el archivo server.xml de Tomcat, donde se declaran las aplicaciones web que ejecutará Tomcat, con la configuración de nuestra aplicación "tomcat1". Seleccionamos *Finish*.

### **Crear un package**

Creamos un package donde irán todas nuestras clases java. Nos situamos sobre el directorio WEB-INF/src de nuestro proyecto y hacemos click con el botón derecho, seleccionamos *New/Package* y escribimos el nombre del package, por ejemplo, **com.empresa.tomcat1**.

### **Crear una clase**

Sobre el nuevo package **com.empresa.tomcat1** hacemos click con el botón derecho y seleccionamos *New/Class*. Escribimos el nombre de la clase, por ejemplo "TomcatTest". Seleccionamos *Finish*.

Escribimos el código de la clase:

```
package com.empresa.tomcat1;
public class TomcatTest {

    public String welcome() {
        String hello = "Hola Mundo!";
        return hello;
    }
}
```

## Crear un .jsp que utilice la clase de ejemplo

Creamos un archivo index.jsp en el directorio raíz de nuestra aplicación, *workspace\tomcat1*

```
<jsp:useBean id="test" class="com.empresa.tomcat1.TomcatTest"
scope="page"/>
Resultado: <%= test.welcome() %><br>
```

Para probarlo arrancamos Tomcat mediante el icono Start Tomcat, abrimos el navegador web y escribimos

<http://localhost:8080/tomcat1/index.jsp>

## 4. Configurar conexiones a bbdd

A la hora de crear conexiones a la base de datos se debe editar el fichero server.xml de Tomcat, dentro del directorio *Tomcat 5.5\conf\server.xml*

Y añadir el DataSource correspondiente:

```
<!-- Global JNDI resources -->
<GlobalNamingResources>
...
  <Resource name="jdbc/server"
    auth="Container"
    type="javax.sql.DataSource"
    factory="org.apache.commons.dbcp.BasicDataSourceFactory"
    username="SERVER"
    password="SERVER"
    driverClassName="oracle.jdbc.driver.OracleDriver"
    maxActive="5"
    maxIdle="1"
    maxWait="10000"
    url="jdbc:oracle:thin:@server.es:1521:server"
    validationQuery="select 1 from dual"
  />
...
</GlobalNamingResources>
```

y:

```
<Host name="localhost" ...>
...
<Context path="/app" reloadable="true" docBase="D:\workspace\app" workDir="
D:\workspace\app\work">
  <ResourceLink name="jdbc/server"
    global="jdbc/server"
    type="javax.sql.DataSource" />
</Context>
```

```
...  
</Host>
```

Además necesitamos una serie de librerías de Tomcat para permitir el acceso a bdd, en el directorio `/common/lib/`

### **commons-\*.jar**

Nombre	Tamaño	Tipo
classes12.jar	1.169 KB	Executable J
commons-collections.jar	162 KB	Executable J
commons-dbcp-1.1.jar	99 KB	Executable J
commons-el.jar	110 KB	Executable J
commons-lang-2.3.jar	240 KB	Executable J
commons-lang-2.3-javadoc.jar	608 KB	Executable J
commons-lang-2.3-sources.jar	314 KB	Executable J
commons-logging-api.jar	22 KB	Executable J
commons-pool-1.1.jar	39 KB	Executable J
jasper-compiler.jar	375 KB	Executable J
jasper-compiler-jdt.jar	898 KB	Executable J
jasper-runtime.jar	75 KB	Executable J
jsp-api.jar	50 KB	Executable J
naming-factory.jar	36 KB	Executable J
naming-factory-dbcp.jar	151 KB	Executable J
naming-resources.jar	46 KB	Executable J
servlet-api.jar	96 KB	Executable J