



UNJu | Universidad
Nacional de Jujuy



Argentina
programa
4.0

Introduccion



JEE | Evolución a través del tiempo



Platform version	Released	Java SE Support	Important Changes
Jakarta EE 10	2022-09-13	Java SE 17 Java SE 11	Removal of deprecated items in Servlet, Faces, CDI and EJB (Entity Beans and Embeddable Container). CDI-Build Time.
Jakarta EE 9.1	2021-05-25	Java SE 11 Java SE 8	JDK 11 support
Jakarta EE 9	2020-12-08	Java SE 8	API namespace move from <code>javax</code> to <code>jakarta</code>
Jakarta EE 8	2019-09-10	Java SE 8	Full compatibility with Java EE 8
Java EE 8	2017-08-31	Java SE 8	HTTP/2 and CDI based Security
Java EE 7	2013-05-28	Java SE 7	WebSocket , JSON and HTML5 support
Java EE 6	2009-12-10	Java SE 6	CDI managed Beans and REST
Java EE 5	2006-05-11	Java SE 5	Java annotations
J2EE 1.4	2003-11-11	J2SE 1.4	WS-I interoperable web services
J2EE 1.3	2001-09-24	J2SE 1.3	Java connector architecture
J2EE 1.2	1999-12-17	J2SE 1.2	Initial specification release



API Java | Standard Edition + Enterprise Edition



API Java

Sun Microsystems
Oracle (.., Ago 2017)

Eclipse Foundation
(Feb 2018, ..)

Java EE 8

javax.persistence
javax.servlet, javax.*



Jakarta EE 10

jakarta.persistence
jakarta.servlet, jakarta.*



JSP ,
JPA ,
JS TL ,
JAX-RS , ..

Java SE Development Kit 17 LTS

OpenJDK

Oracle,
Eclipse Foundation
(Eclipse Adoptium),
Microsoft, RedHat,
Amazon, ..



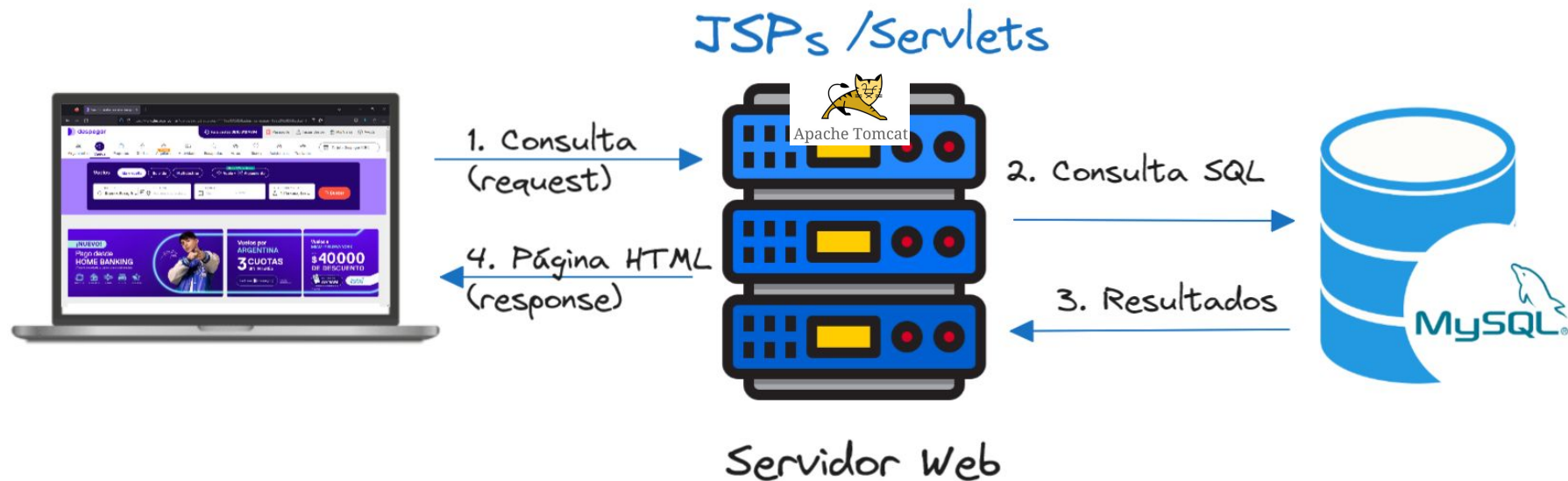
API Java | Documentación oficial



- ❑ Java SE: <https://eclipse.dev/openj9/docs/api-jdk17/>
- ❑ Java EE: <https://docs.oracle.com/javaee/7/api/>
- ❑ Jakarta EE: <https://jakarta.ee/specifications/platform/9/apidocs/>



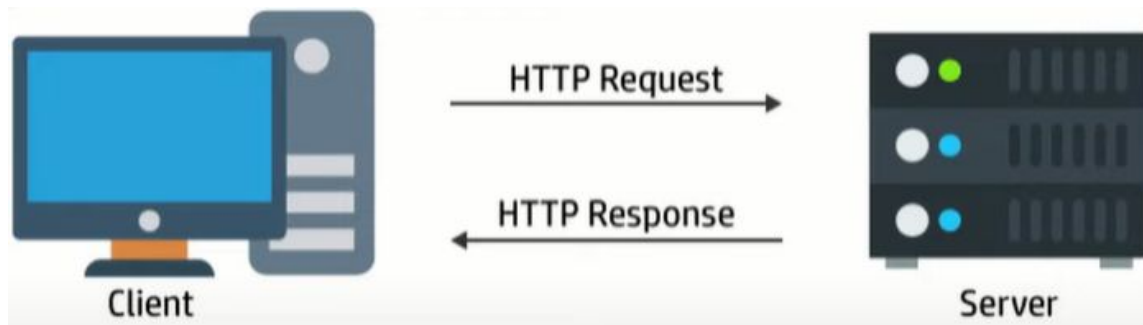
JEE | Aplicación convencional



JSP | ¿Cómo trabaja la web?



- ❑ La “web” se basa en **peticiones** o **consultas** que hace un **cliente** hacia un **servidor web** que se encuentra alojado en algún lugar.
- ❑ Ese **cliente** se comunica con un **servidor**, mediante un protocolo que se llama HTTP o Protocolo de Transferencia de Hipertexto.
- ❑ Mediante este protocolo, el cliente realiza consultas (**Request**) y recibe respuestas (**Responses**).



JSP | Jakarta Server Pages



- ❑ Java **S**erver **P**ages o **P**áginas de **S**ervidor **J**ava.
- ❑ Tecnología Java que sirve para crear aplicaciones web.
- ❑ JSP, requiere de un **servidor web** que sea compatible con contenedores **servlet** como por ejemplo: *Apache Tomcat* o *Glassfish*.
- ❑ Si bien JSP no es una tecnología nueva, tiene la ventaja de la portabilidad de Java y permite comprender los fundamentos de la programación web.



Servidor | Apache Tomcat



The screenshot shows a web browser window with the title "Apache Tomcat® - Apache Tomcat". The address bar displays the URL "https://tomcat.apache.org/download-90.cgi". The main content area contains the following text:

You are currently using **https://dlcdn.apache.org/**. If you encounter a problem with this mirror, please select another mirror. If all mirrors are failing, there are *backup* mirrors (at the end of the mirrors list) that should be available.

Other mirrors:

9.0.74

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

Binary Distributions

- Core:
 - [zip](#) ([pgp](#), [sha512](#))
 - [tar.gz](#) ([pgp](#), [sha512](#))
 - [32-bit Windows zip](#) ([pgp](#), [sha512](#))
 - [64-bit Windows zip](#) ([pgp](#), [sha512](#))
 - [32-bit/64-bit Windows Service Installer](#) ([pgp](#), [sha512](#))
- Full documentation:
 - [tar.gz](#) ([pgp](#), [sha512](#))

A red arrow points to the "64-bit Windows zip" link.



Servidor | Apache Tomcat



File Explorer window showing the directory structure of Apache Tomcat 9.0.74 on a Windows system.

Path: `Este equipo > Disco local (C:) > app > apache-tomcat-9.0.74-windows-x64 > apache-tomcat-9.0.74`

Nombre	Fecha de modificación	Tipo
bin	18/4/2023 20:46	Carpeta de archi...
conf	18/4/2023 21:49	Carpeta de archi...
lib	18/4/2023 20:46	Carpeta de archi...
logs	19/4/2023 12:14	Carpeta de archi...
temp	18/4/2023 20:46	Carpeta de archi...
webapps	18/4/2023 20:46	Carpeta de archi...
work	18/4/2023 21:49	Carpeta de archi...
BUILDING	13/4/2023 08:10	Documento de t...
CONTRIBUTING.md	13/4/2023 08:10	Archivo MD
LICENSE	13/4/2023 08:10	Archivo
NOTICE	13/4/2023 08:10	Archivo
README.md	13/4/2023 08:10	Archivo MD
RELEASE-NOTES	13/4/2023 08:10	Archivo
RUNNING	13/4/2023 08:10	Documento de t...

A red arrow points to the `apache-tomcat-9.0.74` folder in the left sidebar.



JSP | Primer aplicación



```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Página JSP</title>
  </head>
  <body>
    <h1>Hola desde HTML!</h1>
    <% String variable = "Hola desde Java"; %>
    <%= variable %>
  </body>
</html>
```



JSP | Etiquetas



	Etiqueta	Uso/Descripción
Comentarios	<code><%-- --%></code>	Apertura y cierre para realizar comentarios. Por ejemplo: <code><%-- esto es un comentario --%></code>
Directivas	<code><%@ %></code>	Apertura y cierre para directivas/atributos de configuración de JSP. Por ejemplo: <code><%@ page language='java' contentType='text/html' %></code>
Scriptlets	<code><% %></code>	Apertura y cierre para inclusión de sentencias o código Java en general. Esto no es visto/percibido por el usuario. Por ejemplo: <code><% if (numero > numero2) {...} %></code>
Expresiones	<code><%= %></code>	Apertura y cierre para mostrar el resultado de una expresión o contenido de una variable. Lo que se indique aquí será visualizado por el usuario en el apartado de HTML dentro del JSP que se indique. Por ejemplo: <code><%= nombre %></code>
Declaraciones	<code><%! %></code>	Apertura y cierre para hacer uso exclusivo de declaración de variables y métodos de instancia. Que se compartirán entre varios JSP asociados al mismo servlet. Nota: Para declarar variables locales se usar <code><% %></code> .

Varias Líneas

Una línea

Varias Líneas



JSP | Expresiones



```
14 <body>
15     <!-- EXPRESIONES: Java, Matemáticas, booleanas -->
16     Fecha de hoy: <%= new java.util.Date()%> <br>
17     7 + 5 = <%= 7+5%> <br>
18     10 > 100: <%= 10>100%>
19 </body>
```

← → ↻ 🏠 ⓘ http://localhost:8080/PrimerJSP/

Fecha de hoy: Wed Apr 19 20:08:45 ART 2023

7 + 5 = 12

10 > 100: false



JSP | Scriptlets



```
14 <body>
15     <%-- SCRIPTLES --%>
16     <%
17         for (int i=0; i<5; i++){
18             out.println("<br> Vuelta "+i);
19         }
20     %>
21 </body>
```

← → ↻ 🏠 ⓘ http://localhost:8080/PrimerJSP/

Vuelta 0
Vuelta 1
Vuelta 2
Vuelta 3
Vuelta 4



JSP | Declaraciones



```
14 <body>
15     <h1>Ejemplo Declaraciones</h1>
16     <%!
17         private int resultado;
18         public int suma(int a, int b){
19             resultado = a + b;
20             return resultado;
21         }
22         public int resta(int a, int b){
23             resultado = a - b;
24             return resultado;
25         }
26     %>
27     <%=suma(5, 6) %> <br>
28     <%=resta(5, 6) %>
29 </body>
```

← → ↺ 🏠 ⓘ http://localhost:8080/PrimerJSP/

Ejemplo Declaraciones



JSP | Algunos objetos predefinidos

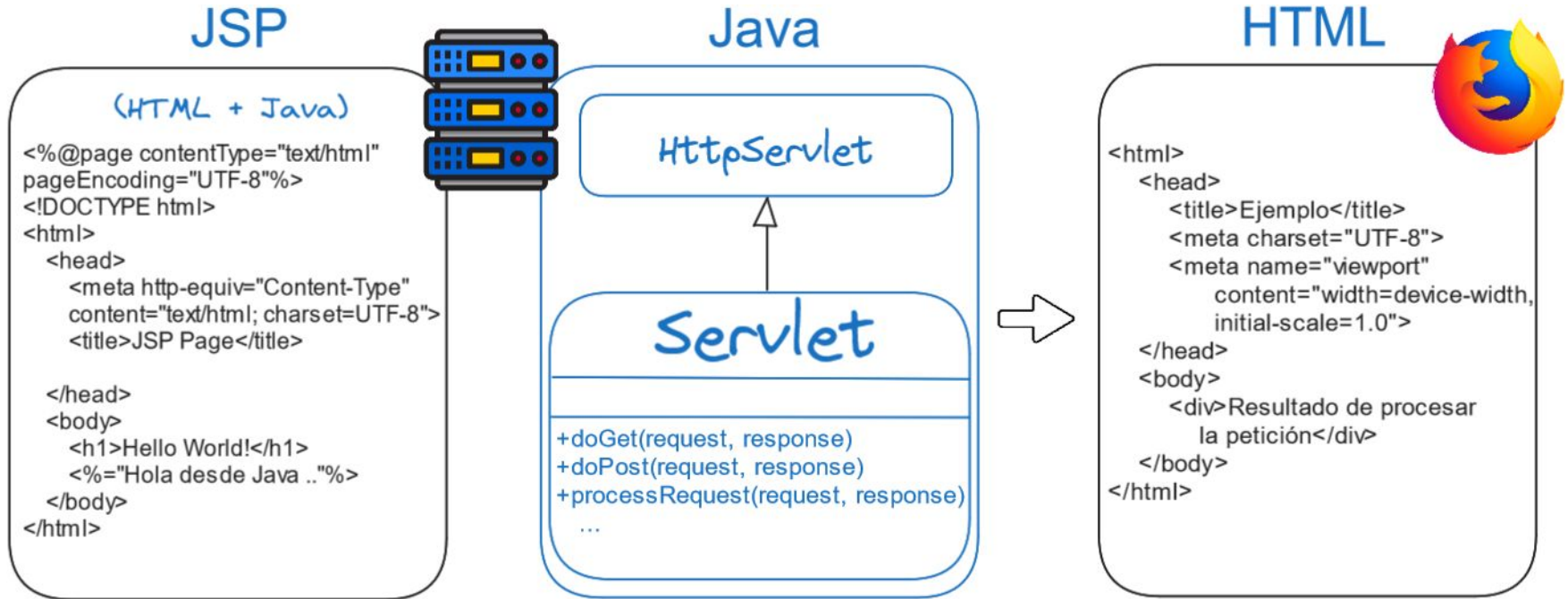


OBJETO	DESCRIPCIÓN
request	Este objeto representa la petición del cliente. Perteneciente a interfaz <code>javax.servlet.HttpServletRequest</code> . Proporciona los datos recibidos vía formulario, GET/POST etc.
response	Este objeto encapsula la respuesta por parte del servidor. Perteneciente a interfaz <code>javax.servlet.HttpServletResponse</code> .
session	Este objeto proporciona una sesión única para cada usuario. Perteneciente a interfaz <code>javax.servlet.http.HttpSession</code> .
out	Este objeto proporciona recursos para escribir contenido en página web. Perteneciente a interfaz <code>javax.servlet.jsp.JspWriter</code> .

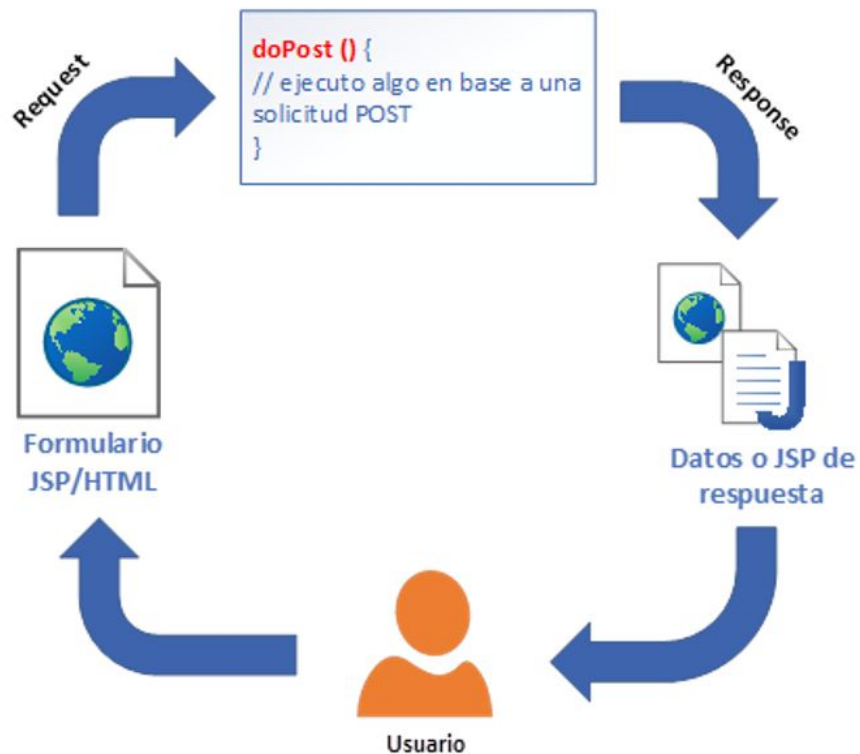
En esta diapositiva debe reemplazarse `javax` por la palabra `jakarta`



Servlets | Diferencia con JSP



Servlets | método doPost()



Servlets | **paquete** jakarta.servlet.*



Un servlet descende de la clase **HttpServlet** y una cuestión que actualmente hay que tener en cuenta es que HttpServlet pertenecía al paquete:

javax.servlet.http.*

Sin embargo, lo más probable es que su IDE le pida el cambio de paquete a

jakarta.servlet.http.*

de la API de ***Jakarta Enterprise Edition***.



Servlets | métodos



Los servlets tienen diferentes métodos que pueden ser utilizados dependiendo del tipo de solicitud que reciban por parte del cliente, sin embargo, los dos más utilizados son el método `doGet()`, para recibir peticiones mediante GET y `doPost()`, para recibir peticiones mediante POST.

- ❑ **`doGet()`**: Es el método encargado de recibir las solicitudes que provienen mediante GET. Generalmente recibe los parámetros desde la URL de la petición y su principal función es la de solicitar datos del servidor y devolverlos al cliente.
- ❑ **`doPost()`**: Es el método encargado de recibir las solicitudes que provienen mediante POST. Los parámetros, objetos o datos pueden provenir en el header o body de una solicitud, a partir del envío (submit) de un formulario HTML desde el JSP. Su principal función es la de obtener datos desde el cliente para generar cambios en el servidor.

