# Servlets

Los servlets son programas que se ejecutan en un servidor web o servidor de aplicaciones y actúan como una capa intermedia entre una petición web, y bases de datos o aplicaciones alojadas en el servidor.

Usando servlets, puedes acceder a los datos enviados por el usuario a través de formularios, presentar registros desde una base de datos y crear páginas web dinámicas.

Los servlets comúnmente cumplen el mismo propósito de los programas que usan CGI (Common Gateway Inteface). Pero los servlets ofrecen un número de ventajas en comparación con el CGI:

* El performance de la aplicación es mejor
* Los servlets se ejecutan dentro del espacio de un servidor web. No es necesario crear un proceso separado para manejar cada petición del cliente.
* Los servlets son independientes de la plataforma, pues están escritos en java.
* El manejador de seguridad de Java obliga a una serie de restricciones para proteger los recursos de la máquina servidor. Es decir, los servlets son confiables.
* El conjunto completo de las clases de la API de Java están disponibles para ser usados en los servlets.

# Tareas de los Servlets

Los servlets ejecutan las siguientes tareas principales:

* Leen los datos explicitos enviados por el cliente (navegadores). Esto incluye formularios HTML en una página web o puede también ser una petición de un applet.
* Leer los datos implícitos enviados por el cliente. Esto incluye cookies, tipos de medios y esquemas de compresión.
* Procesa los datos y genera los resultados. Este proceso puede implicar comunicarse con una base de datos, ejecutar una llamada RMI o CORBA, invocar un servicio web o computar la respuesta directamente.
* Envía los datos expicitos (el documento) a los clientes. Este documento puede ser enviado en una variedad de formatos, incluyendo texto (HTML o XML), binario (GIF, imágenes), Excel, etc.
* Envía los datos implícitos de la respuesta. Esto incluye decir a los navegadores u otros clientes el tipo de documento que recibirá, establecer cookie, etc.

# Paquetes de los Servlets

Los servlets de Java son clases ejecutadas por un servidor web que contiene un interprete que soporta la especificación de Java Servlet.

Los servlets pueden ser creados utilizando los paquetes javax.servlet y javax.servlet.http, que son una parte estándar de Java Enterprise Edition (JEE), una versión extendida del conjunto de clases de Java que permiten soportar el desarrollo de proyectos a gran escala.

# Ciclo de vida de un Servlet

El ciclo de vida de un servlet puede definirse como el proceso completo desde su creación hasta su destrucción.

Los siguientes son los caminos seguidos por el servlet:

* El Servet es inicializado invocando el método init()
* El servlet invoca el método service() que procesa la petición del cliente
* El servlet es finalizado invocando el método destroy()
* Finalmente, el servlet es destruido por el recolector de basura de la JVM

## El método init()

El método init está diseñado para ser invocado solo una vez. Es invocado cuando el servlet es creado, y no se vuelve a invocar en cada petición del usuario.

Un servlet es normalmente creado al momento de que un usuario accede a la URL que corresponde al servlet, pero puedes también especificar que el servlet debe ser creado cuando el servidor arranca.

## El método service()

El método service es el método principal que ejecuta las tareas del servlet. El contener de servlets invoca el método service para manejar las peticiones del cliente y escribir la respuesta de regreso al uuario.

Cada vez que el servidor recibe una petición para un servlet, crea un nuevo hilo e invoca el método service. El método se encarga de identificar el tipo de petición HTTP e invoca el método correspondiente.

## El método doGet()

Una petición GET es el resultado de una petición normal a través de una url o de un formulario HTML que no ha especificado un método y es manejada por el método doGet()

## El método doPost()

Una petición POST es el resultado de una petición a través de un formulario HTML que específicamente utiliza el método POST y debe ser manejada con el método doPost()

## El método destroy()

El método destroy es invocado una sola vez, al final del ciclo de vida de un servlet. Este método proporciona al servlet la oportunidad de cerrar conexiones a base de datos, escribir cookies, o cualquier otra tarea de limpieza y liberación de los recursos utilizados por el servlet.