**Introduction to Java Servlets**

Last Updated : 05 Mar, 2024

Today, we all are aware of the need to create **dynamic web pages** i.e. the ones that can change the site contents according to the time or can generate the content according to the request received from the client. If you like coding in Java, then you will be happy to know that using Java there also exists a way to generate dynamic web pages and that way is **Java Servlet**. But before we move forward with our topic let’s first understand the need for server-side extensions.

**What is Java Servlet?**

Java Servlets are the Java programs that run on the Java-enabled web server or application server. They are used to handle the request obtained from the web browser, process the request, produce the response, and then send a response back to the web browser.

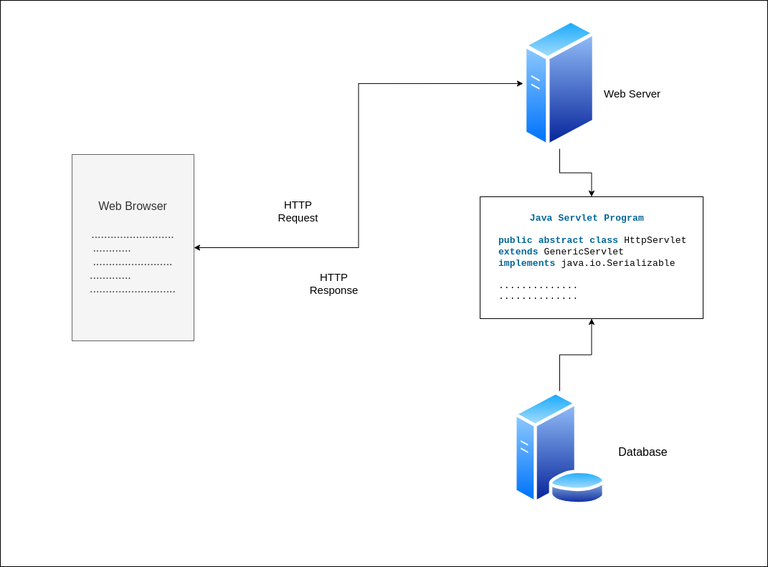
**Properties of Java Servlet**

The properties of Servlets are as follows:

* Servlets work on the server side.
* Servlets are capable of handling complex requests obtained from the web browser/client.

**Java Servlets Architecture**

Servlet Architecture can be depicted from the image itself as provided below as follows:



**Execution of Java Servlets**

Execution of Servlets basicallyinvolves Six basic steps:

1. The Clients send the request to the Web Server.
2. The Web Server receives the request.
3. The Web Server passes the request to the corresponding servlet.
4. The Servlet processes the request and generates the response in the form of output.
5. The Servlet sends the response back to the tomcat web server
6. The Web Server sends the response back to the client and the client browser displays it on the screen.

Now let us do discuss eccentric point that why do we need For Server-Side extensions?

**Need of Server-Side Extensions**

The **Server-Side Extensions** are nothing but the technologies that are used to create dynamic Web pages. Actually, to provide the facility of dynamic Web pages, Web pages need a container or Web server. To meet this requirement, independent Web server providers offer some proprietary solutions in the form of **APIs (**Application Programming Interface).   
These **APIs** allow us to build programs that can run with a Web server. In this case, **Java Servlet** is also one of the component APIs of **Java Platform Enterprise Edition (nowdays known as – ‘Jakarta EE’)** which sets standards for creating dynamic Web applications in Java.

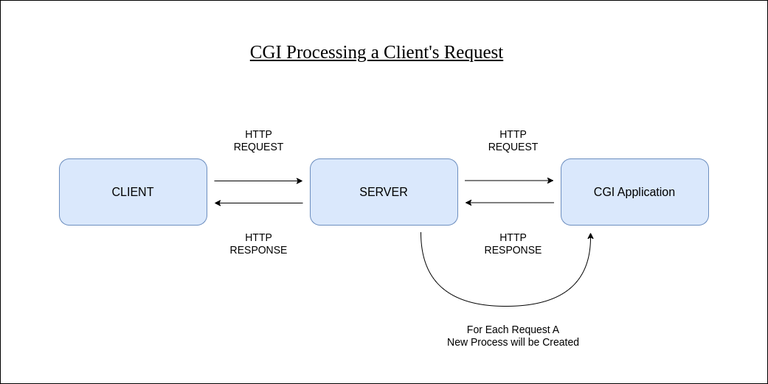
Before learning about something, it’s important to know the need for that something, it’s not like that this is the only technology available for creating Dynamic Web Pages. The Servlet technology is similar to other Web server extensions such as **Common Gateway Interface (**CGI) scripts and **Hypertext Preprocessor** (PHP). However, Java Servlets are more acceptable since they solve the limitations of **CGI** such as low performance and low degree scalability.

**What is CGI(Common Gateway Interface)?**

**CGI** is actually an external application that is written by using any of the programming languages like **C** or **C++** and this is responsible for processing client requests and generating dynamic content.

In CGI application, when a client makes a request to access dynamic Web pages, the Web server performs the following operations:

* It first locates the requested web page *i.e* the required CGI application using URL.
* It then creates a new process to service the client’s request.
* Invokes the CGI application within the process and passes the request information to the application.
* Collects the response from the CGI application.
* Destroys the process, prepares the HTTP response, and sends it to the client.



So, in **CGI** server has to create and destroy the process for every request. It’s easy to understand that this approach is applicable for handling few clients but as the number of clients increases, the workload on the server increases and so the time is taken to process requests increases.

**Difference between Java Servlets and CGI**

| **Servlet** | **CGI (Common Gateway Interface)** |
| --- | --- |
| Servlets are portable and efficient. | CGI is not portable. |
| In Servlets, sharing data is possible. | In CGI, sharing data is not possible. |
| Servlets can directly communicate with the webserver. | CGI cannot directly communicate with the webserver. |
| Servlets are less expensive than CGI. | CGI is more expensive than Servlets. |
| Servlets can handle the cookies. | CGI cannot handle the cookies. |

**Servlets APIs**

Servlets are built from two packages:

* javax.servlet(Basic)
* javax.servlet.http(Advance)

Various classes and interfaces present in these packages are:

javax.servlet.\*;

-------------------

Interface:

Servlet (init() , service() , destroy() lifecycle methods)

ServletRequest

ServletResponse

ServletConfig , RequestDispatcher

class:

GenericServlet (implementation of Servlet interface)

javax.servlet.http.\*;

----------------------

class:

HttpServlet (extends GenericServlet)

doGet() , doPost()

Interface:

HttpServletRequest

HttpServletResponse

Servlet interface ( init, service , destroy )

|

GenericServlet class

| init() {..} , service() { ..} , destroy() { .. }

HttpServlet class

Class HelloServlet extends HttpServlet{

}