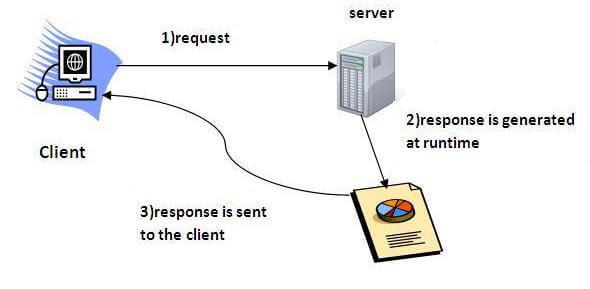
**Servlet** technology is used to create a web application (resides at server side and generates a dynamic web page).

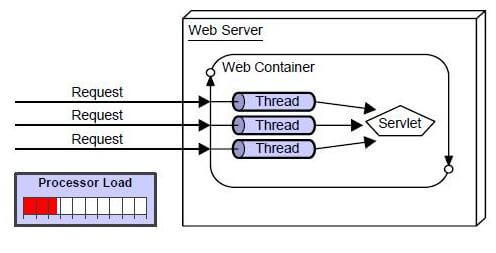


What is a Servlet?

Servlet can be described in many ways, depending on the context.

* Servlet is a technology which is used to create a web application.
* Servlet is an API that provides many interfaces and classes including documentation.
* Servlet is an interface that must be implemented for creating any Servlet.
* Servlet is a class that extends the capabilities of the servers and responds to the incoming requests. It can respond to any requests.
* Servlet is a web component that is deployed on the server to create a dynamic web page.

### Advantages of Servlet



There are many advantages of Servlet over CGI. The web container creates threads for handling the multiple requests to the Servlet. Threads have many benefits over the Processes such as they share a common memory area, lightweight, cost of communication between the threads are low. The advantages of Servlet are as follows:

1. **Better performance:** because it creates a thread for each request, not process.
2. **Portability:** because it uses Java language.
3. **Robust:** JVM manages Servlets, so we don't need to worry about the memory leak, garbage collection, etc.
4. **Secure:** because it uses java language.

|  |  |
| --- | --- |
| **Static Website** |  |
| **Static Website** | **Dynamic Website** |
| Prebuilt content is same every time the page is loaded. | Content is generated quickly and changes regularly. |
| It uses the **HTML**code for developing a website. | It uses the server side languages such as **PHP,SERVLET, JSP, and ASP.NET**etc. for developing a website. |
| It sends exactly the same response for every request. | It may generate different HTML for each of the request. |
| The content is only changed when someone publishes and updates the file (sends it to the web server). | The page contains "server-side" code which allows the server to generate the unique content when the page is loaded. |
| Flexibility is the main advantage of static website. | Content Management System (CMS) is the main advantage of dynamic website. |

# HTTP (Hyper Text Transfer Protocol)

The Hypertext Transfer Protocol (HTTP) is application-level protocol for collaborative, distributed, hypermedia information systems. It is the data communication protocol used to establish communication between client and server.

HTTP is TCP/IP based communication protocol, which is used to deliver the data like image files, query results, HTML files etc on the World Wide Web (WWW) with the default port is TCP 80. It provides the standardized way for computers to communicate with each other.



**The Basic Features of HTTP (Hyper Text Transfer Protocol):**

There are three fundamental features that make the HTTP a simple and powerful protocol used for communication:

* **HTTP is media independent:** It specifies that any type of media content can be sent by HTTP as long as both the server and the client can handle the data content.
* **HTTP is connectionless:** It is a connectionless approach in which HTTP client i.e., a browser initiates the HTTP request and after the request is sent the client disconnects from server and waits for the response.
* **HTTP is stateless:** The client and server are aware of each other during a current request only. Afterwards, both of them forget each other. Due to the stateless nature of protocol, neither the client nor the server can retain the information about different request across the web pages.

HTTP Requests

The request sent by the computer to a web server, contains all sorts of potentially interesting information; it is known as HTTP requests.

The HTTP client sends the request to the server in the form of request message which includes following information:

* The Request-line
* The analysis of source IP address, proxy and port
* The analysis of destination IP address, protocol, port and host
* The Requested URI (Uniform Resource Identifier)
* The Request method and Content
* The User-Agent header
* The Connection control header
* The Cache control header
* The HTTP request method indicates the method to be performed on the resource identified by the **Requested URI (Uniform Resource Identifier)**. This method is case-sensitive and should be used in uppercase.
* The HTTP request methods are:

|  |  |
| --- | --- |
| **HTTP Request** | **Description** |
| **GET** | Asks to get the resource at the requested URL. |
| **POST** | Asks the server to accept the body info attached. It is like GET request with extra info sent with the request. |
| **HEAD** | Asks for only the header part of whatever a GET would return. Just like GET but with no body. |
| **TRACE** | Asks for the loopback of the request message, for testing or troubleshooting. |
| **PUT** | Says to put the enclosed info (the body) at the requested URL. |
| **DELETE** | Says to delete the resource at the requested URL. |
| **OPTIONS** | Asks for a list of the HTTP methods to which the thing at the request URL can respond |

## Get vs. Post

There are many differences between the Get and Post request. Let's see these differences:

|  |  |
| --- | --- |
| **GET** | **POST** |
| 1) In case of Get request, only **limited amount of data**can be sent because data is sent in header. | In case of post request, **large amount of data**can be sent because data is sent in body. |
| 2) Get request is **not secured**because data is exposed in URL bar. | Post request is **secured**because data is not exposed in URL bar. |
| 3) Get request **can be bookmarked.** | Post request **cannot be bookmarked.** |
| 4) Get request is **idempotent**. It means second request will be ignored until response of first request is delivered | Post request is **non-idempotent.** |
| 5) Get request is **more efficient**and used more than Post. | Post request is **less efficient**and used less than get. |

GET and POST

Two common methods for the request-response between a server and client are:

* **GET**- It requests the data from a specified resource
* **POST**- It submits the processed data to a specified resource

Anatomy of Get Request

The query string (name/value pairs) is sent inside the URL of a GET request:

1. GET/RegisterDao.jsp?name1=value1&name2=value2

As we know that data is sent in request header in case of get request. It is the default request type. Let's see what information is sent to the server.



* It remains in the browser history
* It can be bookmarked
* It can be cached
* It have length restrictions
* It should never be used when dealing with sensitive data
* It should only be used for retrieving the data

Anatomy of Post Request

The query string (name/value pairs) is sent in HTTP message body for a POST request:

1. POST/RegisterDao.jsp HTTP/1.1
2. Host: www. javatpoint.com
3. name1=value1&name2=value2

As we know, in case of post request original data is sent in message body. Let's see how information is passed to the server in case of post request.



Some other features of POST requests are:

* This requests cannot be bookmarked
* This requests have no restrictions on length of data
* This requests are never cached
* This requests do not retain in the browser history

**Servlet Container States**

The servlet container is the part of web server which can be run in a separate process. We can classify the servlet container states in three types:

* **Standalone:** It is typical Java-based servers in which the servlet container and the web servers are the integral part of a single program. For example:- Tomcat running by itself
* **In-process:** It is separated from the web server, because a different program runs within the address space of the main server as a plug-in. For example:- Tomcat running inside the JBoss.
* **Out-of-process:** The web server and servlet container are different programs which are run in a different process. For performing the communications between them, web server uses the plug-in provided by the servlet container.

**The Servlet Container performs many operations that are given below:**

* Life Cycle Management
* Multithreaded support
* Object Pooling
* Security etc.