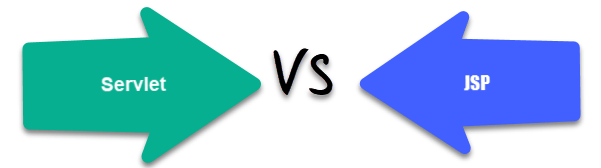
**Servlet vs JSP – Difference Between Them**

**Key Difference between Servlet and JSP**

* Servlet can accept all protocol requests, including HTTP, while JSP can only accept HTTP requests.
* In MVC architecture, servlet works as a controller, while JSP works as a view for displaying output.
* Servlet should be used when there is more data processing involved, whereas JSP is generally used when there is less involvement of data processing.
* Servlets run faster than JSP, on the other hand, JSP runs slower than servlet as it takes time to compile the program and convert it into servlets.
* You can override the service() method in servlet, but in JSP, you can’t override the service() method.
* In Servlet, you have to implement both business logic and presentation logic in a single file. Whereas in JSP, business logic is split from presentation logic using JavaBeans.

Difference between Servlet and JSP

**What is Servlet?**

**Servlet** is a Java technology that is managed by a container called a servlet engine. It generates dynamic content and interacts with the client through Request and Response. Servlet extends the functionality of a web server. Though servlets can respond to many types of requests, they generally implement web containers for hosting any website on web server. Therefore, it is qualified as a server-side servlet web API.

**What is JSP?**

**JSP** is a collection of technologies developed by Sun Microsystems. It is used to develop web pages by inserting Java code into the HTML pages by making special JSP tags. It can consist of either HTML or XML (combination of both is also possible) with [JSP Actions](https://www.guru99.com/jsp-action-tags.html) and commands. The full form of JSP is Java Server Pages.

**Difference between Servlet vs JSP**

Here are the main differences between Servlet and JSP:

|  |  |
| --- | --- |
| **Servlet** | **JSP** |
| Servlets run faster than JSP. | JSP runs slower than servlet as it takes time to compile the program and convert into servlets. |
| It is hard to write code in servlet. | It’s easier to code in JSP compared to servlets. |
| In MVC architecture, servlet works as a controller. | In MVC architecture, JSP works as a view for displaying output. |
| It should be use when there is more data processing involved. | JSP is generally used when there is no involvement of much data processing. |
| There is no custom tag writing facility in servlets. | You can easily build custom tags that can directly call Java beans. |
| Servlet is a java code. | JSP is a HTML-based code. |
| It can accept all protocol requests, including HTTP. | It can only accept HTTP requests. |
| You can override the service() method. | In JSP, you can’t override the service() method. |
| In Servlet, by default, session management is not enabled, user has to enable it explicitly. | In JSP, session management is automatically enabled. |
| In Servlet, you have to implement both business logic and presentation logic in the single file. | In JSP, business logic is split from presentation logic using JavaBeans. |
| Modification in Servlet file is a time consuming due to reloading, recompiling, and restarting the server. | JSP modification is fast, as you just need to click one refresh button. |

**Why use Servlet?**

Here are the reasons for using Servlet:

* The performance is much better.
* Servlet is used when you do not like to create a separate process to handle each and every request of client.
* Developers can use servlets when they have to take advantage of all the features of Java.
* Java class libraries which are available to a servlet can communicate with applets, databases, or other software via RMI and sockets mechanisms.

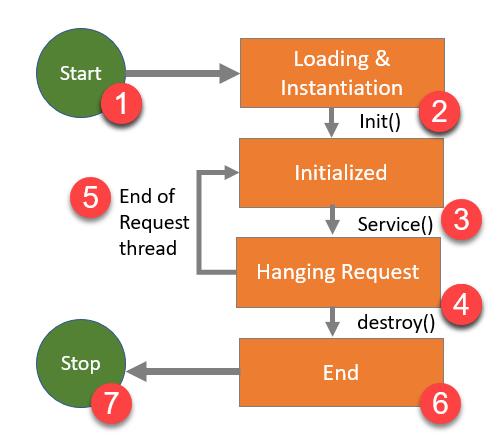
**Why use JSP?**

Here are the reasons of using JSP:

* In Java server pages JSP, the execution is much faster compared to other dynamic languages.
* It is much better than Common Gateway Interface (CGI).
* Java server pages (JSP)are always compiled before its processed by the server as it reduces the effort of the server to create process.
* Java server pages are built over Servlets API. Hence, it has access to all Java APIs, JNDI, JDBC EJB, and other components of java.
* JSP is an important part of Java EE (Enterprise Edition), which is a platform for enterprise-level applications.

**Servlet Life Cycle**

Servlet life cycle is depicted in the below diagram:

Servlet Life Cycle

**1) Start:**Execution of servlet begins.

**2) Loading & instantiation void init():** It is called when servlet is first loaded. This method lets you initialize servlet.

**3) Initialized** **void service()**: The purpose of this method is to serve a request. You can call it as many times as you like.

**4) Handling request and destroying servlet:** Java application must be first determined what code is needed to execute the request URL to provide a response. To destroy servlet Void destroy method is used at the end of servlet life cycle.

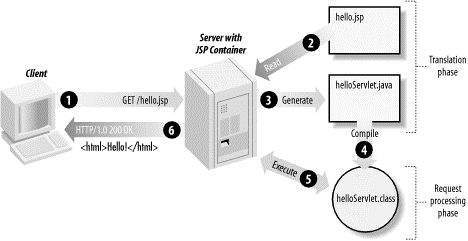
**5) End of Request Thread:**When service() finishes its task, either the thread ends or returns to the thread pool that is managed by servlet contaier.

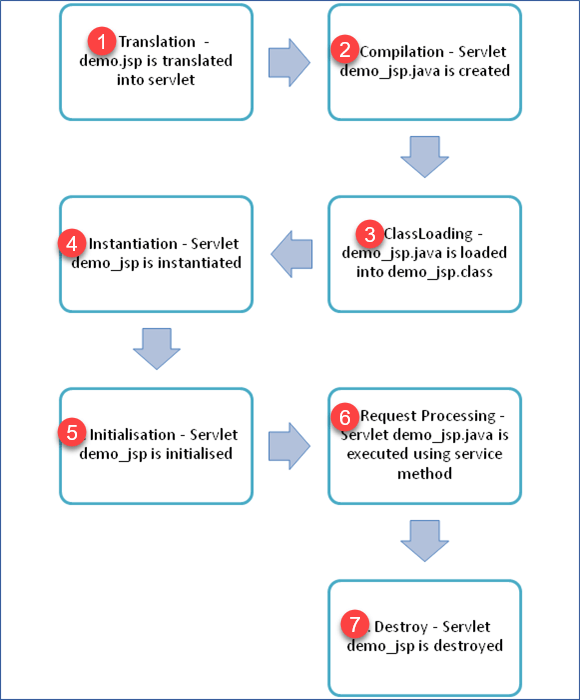
**6) End:**Servlet lifecycle finishes.

**7) Stop:** Servlet stop executing.

**JSP Life Cycle**

JSP life cycle is depicted in the diagram below:





JSP Life Cycle

1. Translation of JSP page into a servlet.
2. Compilation of JSP page(Compilation of JSP page into \_jsp.java)
3. Classloading (\_jsp.java is converted to class file \_jsp.class)
4. Instantiation(Object of generated servlet is created)
5. Initialisation(\_jspinit() method is invoked by container)
6. Request Processing(\_jspservice() method is invoked by the container)
7. Destroy (\_jspDestroy() method invoked by the container)

**Advantages of Servlet**

Here are the advantages of using Servlet:

* Servlet loads only one copy of it into JVM (Java Virtual Machine).
* It saves time to respond to the first request which increases response time.
* It uses standard API that is supported by numerous web servers.
* Programming languages like platform independence and ease of development.
* You can access the large set of APIs that are available for the Java platform.
* The web container makes threads for handling more than one request to the servlet.
* Easy coordination between two or more servlet to make web applications.
* Servlet containers support many other features like sessions, resource management, persistence, security, etc.
* Servlet usage doesn’t constrain by the web browser.

**Disadvantages of Servlet**

Here are the disadvantages for using Servlet:

* One servlet is loaded into JVM. It does matter numbers of requests.
* When there is a request, there is a thread, not a process.
* Servlet is persistent until it destroys.
* Designing in a servlet is difficult and slows down the application.
* You need a JRE(Java Runtime Environment) on the server to run servlets.
* For non-java developers, servlet is not suitable as they required to have a broad knowledge of Java servlet.
* HTML code is mixed up with Java code therefore, changes done in one code can affect another code.
* Writing HTML code in servlet programming is very difficult. It also makes servlet looks bulky.
* In servlet programming, if you want to use implicit objects, you need to write some additional code in order to access them.
* Developers must take care of exception handling because servlet programming is not thread-safe by default.

**Advantages of JSP**

Here are the advantages of using JSP:

* It is very much convenient to modify the regular HTML.
* We can write the servlet code into the JSP.
* It is only intended for simple inclusions which can use form data and make connections.
* You can easily connect with JSP with the MySQL database.
* The performance and scalability of JSP are good because JSP allows embedding of dynamic elements in HTML pages.
* JSP is built on Java technology hence it is platform-independent and not depending on any operating systems.
* JSP includes the feature of multithreading of java.
* We can also make use of exception handling of java into JSP.
* It enables to separate presentation layer with the business logic layer in the web application.
* It is easy for developers to show as well as process the information.

**Disadvantages of JSP**

Here are the disadvantages for using JSP:

* It is hard to trace JSP pages error because JSP pages are translated to servlet.
* As JSP output is HTML, it is not rich in features.
* It is very hard to debug or trace errors because JSP pages are first translated into servlets before the compilation process.
* Database connectivity is not easy.
* JSP pages require more disk space and time to hold JSP pages as they are compiled on the server.