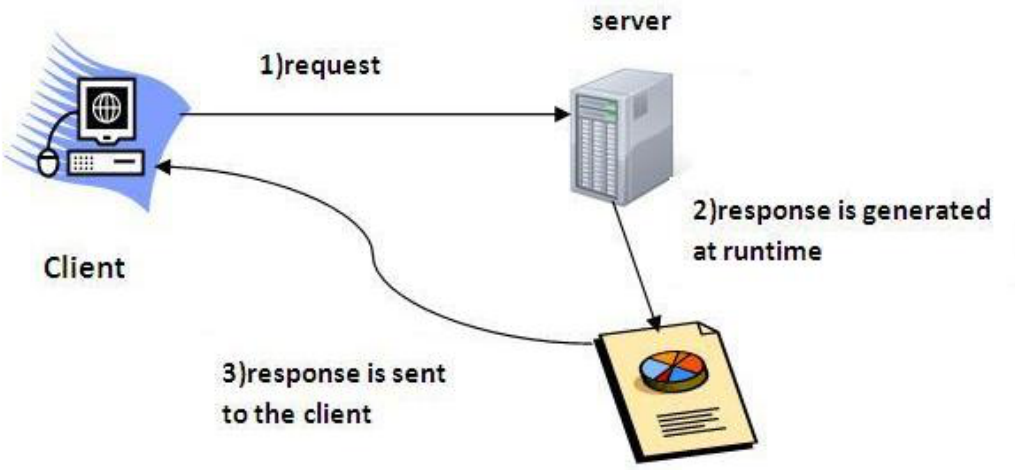
Java Servlets

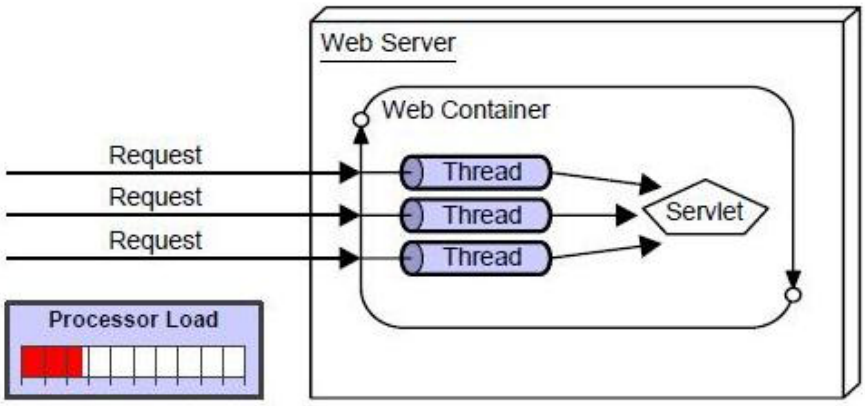
A Java servlet is a Java program that extends the capabilities of a server. Although servlets can respond to any types of requests, they most commonly implement applications hosted on Web servers. Such Web servlets are the Java counterpart to other dynamic Web content technologies such as PHP and ASP.NET.

Servlet technology is used to create web applications (resides at server side and generates dynamic web pages).

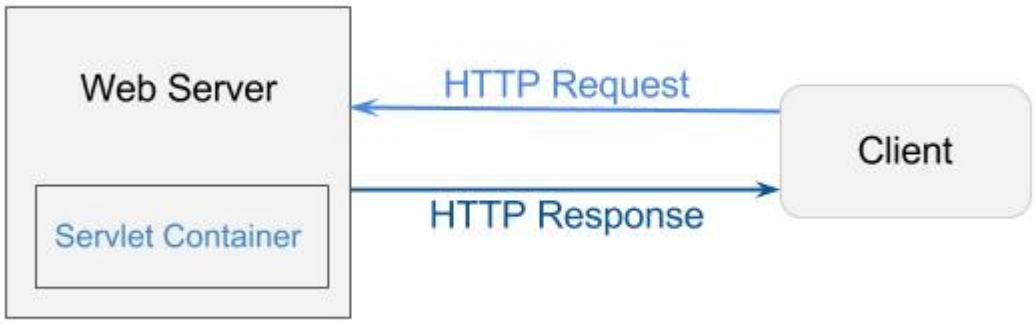
There are many interfaces and classes in the servlet API such as Servlet, GenericServlet, HttpServlet, ServletRequest, ServletResponse etc.



There are many advantages of servlets. The web container creates threads for handling the multiple requests to the servlet.

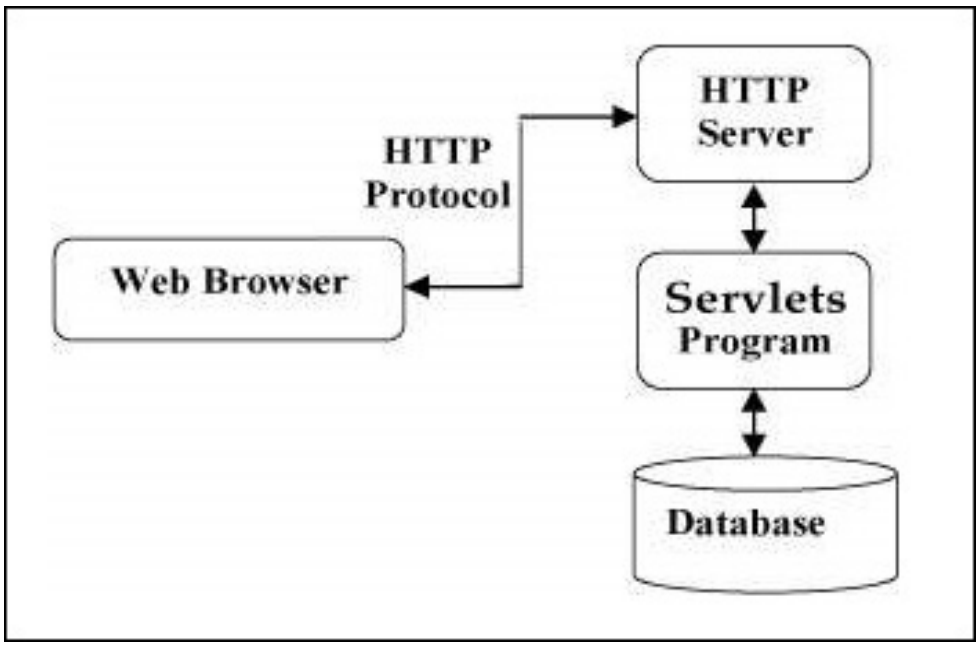


The servlet container is used in java to dynamically generate the web pages on the server side. The container is the part of a web server that interacts with the servlet for handling the dynamic web pages from the client.



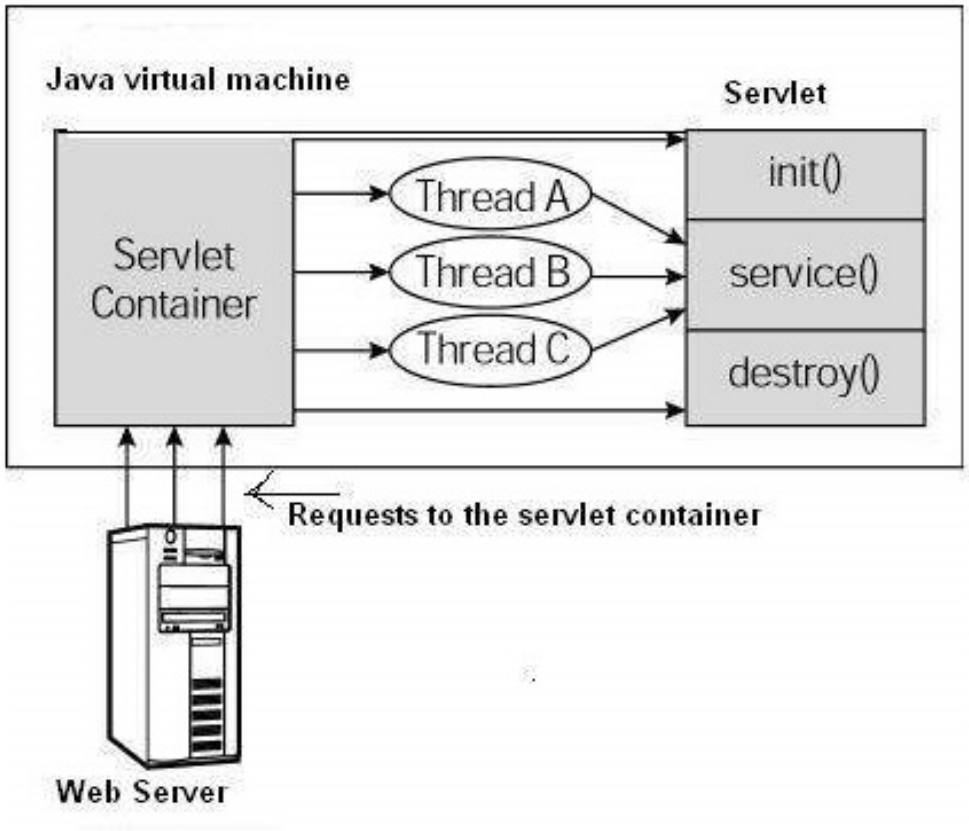
Servlets act as a middle layer between a request coming from a Web browser or other HTTP client and databases or applications on the HTTP server.

Using Servlets, you can collect input from users through web page forms, present records from a database or another source, and create web pages dynamically.



A servlet life cycle can be defined as the entire process from its creation till the destruction. The following are the paths followed by a servlet:

- The servlet is initialized by call the init() method
- The servlet calls service() method to process a client's request
- The servlet is terminated by calling the destroy() method.



The init() method is called only once. It is called only when the servlet is created, and not called for any user requests afterwards. The servlet is normally created when a user first invokes a URL corresponding to the servlet, but you can also specify that the servlet be loaded when the server is first started. When a user invokes a servlet, a single instance of each servlet gets created, with each user request resulting in a new thread that is handed off to doGet() or doPost() as appropriate. The init() method simply creates or lads some data that will be used throughout the life of the servlet.

The service() method is the main method to perform the actual task. The servlet container calls the service() method to handle requests coming from the client's browser and to write the formatted response back to the client. Each time the server receives a request for a servlet, the server spawns a new thread and calls the service. The service() method checks the HTTP request type (GET, POST, PUT, DELETE, etc) and calls doGet(), doPost(), doPut(), doDelete() methods as appropriate.

The destroy() method is called only once at the end of the life cycle of a servlet. This method gives your servlet a chance to close database connections, halt background threads, write cookie lists or hit counts to disk, and perform other such cleanup activities. After the destroy() method is called, the servlet object is marked for garbage collection.