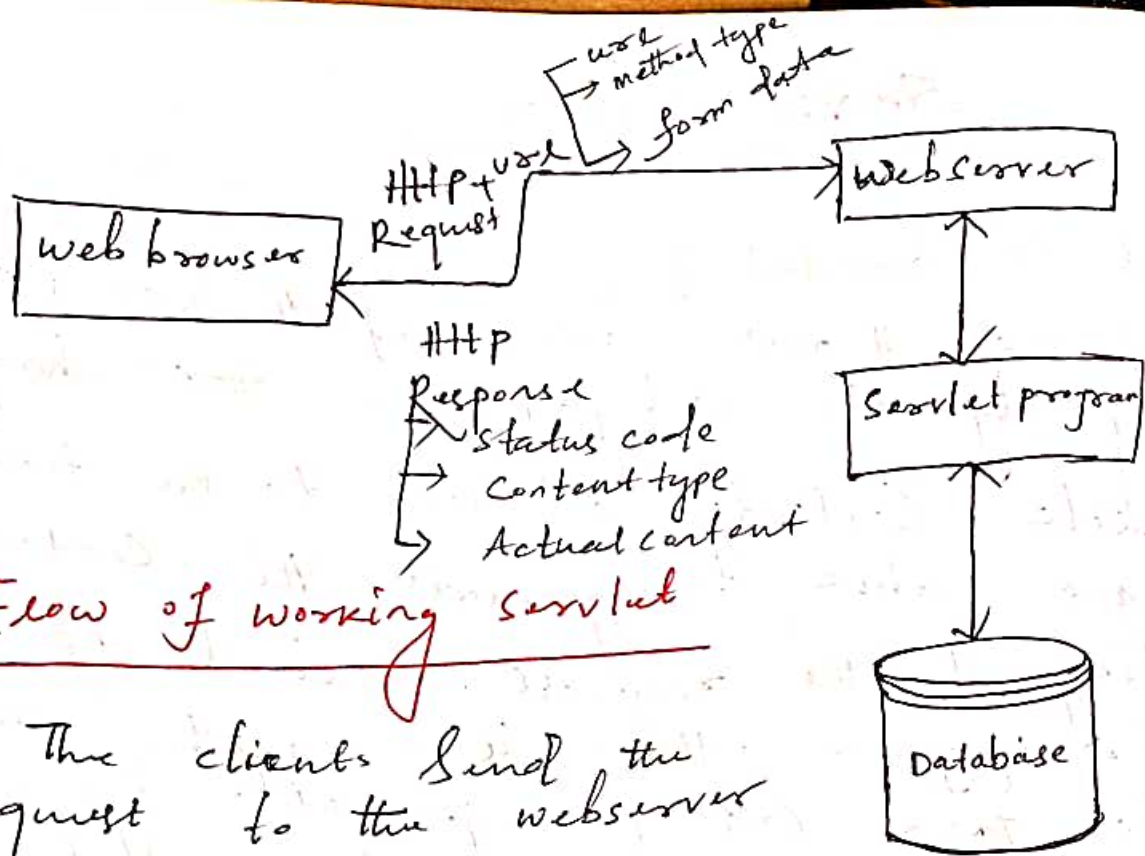


Servlet Question

① What is Servlet?

Today we all are aware of the need of creating dynamic web pages i.e., the change the site contents according to the time or are able to generate the contents according to the request received by the client. In Java there is another way to do Java coding & generate dynamic web pages and this way is Java Servlet.

- For developing any web application we need a server.
- Servlet works on the server-side.
- Server is nothing but a computer, server is hosted in a cloud.
- Servlets are capable of handling complex requests obtained from the webserver.
- To reach a URL we are going to take server.
- Servlet is a piece of Java program which runs on the server.
- Servlet is also a interface.



Flow of working servlet

- 1- The clients send the request to the webserver
- 2- The web server receives the request.
- 3- The webserver 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 client and the client browser displays it on the screen.

why we use servlet

To develop a dynamic web page & to provide the facility of dynamic web pages, web pages need a container or web server.

Advantages of Servlet

- Servlet is faster than CGI as it does not involve the creation of a new process for every new request received.
- Servlets, written in Java, & it is platform-independent.
- Servlet does not run in a separate process for creating a new process.
- This also saves the memory and allows a Servlet to easily manage the client state.

② what is web application?

- web application is a application software that is provided by a third party & that runs on a web browser. & stored on a remote server, and can be accessed from any web browser with any device.
- it is delivered to users with any of the world wide web active network.

③ what is the difference between application server and web application?

- A server is a central repository where information and computer programs are stored and accessed by the programs within the network.
- webserver and Application server are kinds of the server which employed to deliver sites.

web server

- A web server accepts and stores request from clients for static content (like, HTML pages, files, images, and videos) from a website.
- web servers expose business logic to the clients, which generates handle HTTP requests & response.
- serves only web-based applications.
- also support for multithreading.

Application Server

- An application server exposes business logic to the clients, which generates dynamic content.
- It is a software framework that transforms data to provide the specialized functionality offered by a business service or application.
- application server use several protocols.
- Multithreading to support multiple requests in a parallel.

④ ^{abstract} what is generic Servlet? → class

- GenericServlet implements the Servlet interface and provide an implementation for all its method except the service() method, it is an abstract method.
- GenericServlet class defines a protocol-independent (Http-less) servlet.
- while building a website or application, we want Http protocol, in that case, we must extend HttpServlet instead of GenericServlet.
- For making a class Servlet by extending GenericServlet is very easy because we have to provide implementation only for the service() method.
- GenericServlet class is in javax.servlet package. (javax.servlet.GenericServlet).

ex:-

```
class Person extends GenericServlet {  
    public void service(interface ServletRequest req,  
                        ServletResponse resp) throws  
                        ServletException, IOException.  
}
```

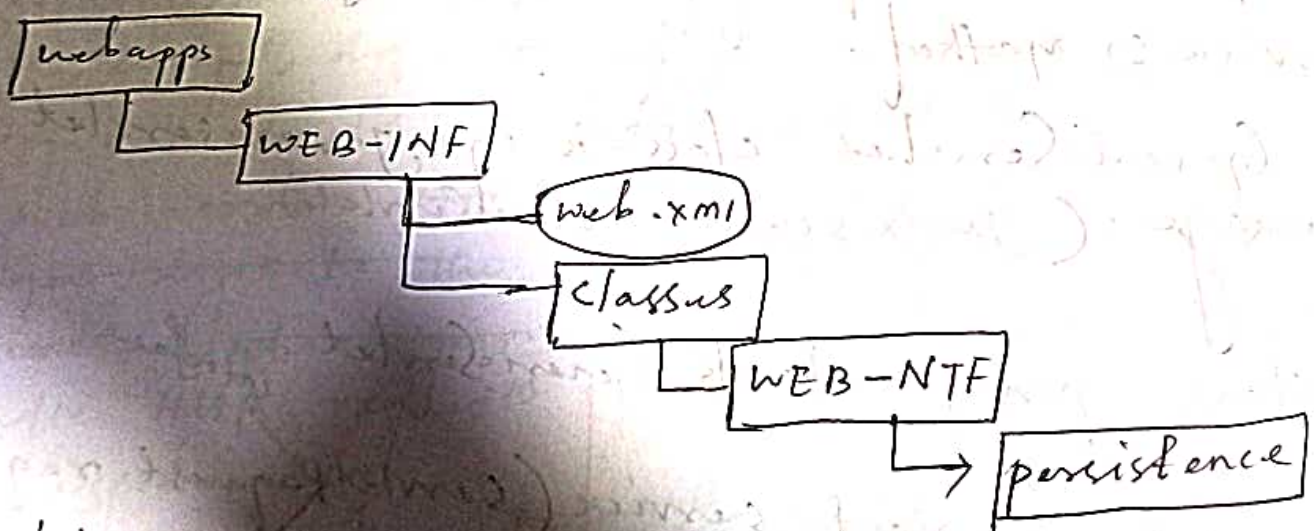
- This method is automatically called by the server whenever a request for a GenericServlet arrives.

- The `service()` method accepts two parameters
- ① A `ServletRequest` Object
 - ② A `ServletResponse` Object.

A `ServletRequest` object allows to read data provided by the client request and the `ServletResponse` object is used to send the response to the client.

web.xml File

- Java web application use a file to determine how URLs map to Servlets.
- This file named `web.xml` and resides in the app's `WEB-INF`



It is a file which give configuration

XML

<web-app>

<servlet>

<servlet-class>com.tg.myservlet</servlet-class>

<servlet-name>myfirstservlet</servlet-name>

</servlet>

<servlet-mapping>

<servlet-name>myfirstservlet</servlet-name>

<url-pattern>/myservlet</url-pattern>

</servlet-mapping>

</web-app>

⑤ what is HttpServlet? abstract class

→ It is works for (http, https).

→ HttpServlet is an abstract class & present in javax.servlet.http.HttpServlet.

→ To create a Servlet, ^{the} class must extend the HttpServlet class and override at least one of its methods like [doGet, doPost, doDelete, doPut, doOption, doTrace, service, service].

→ The HttpServlet class extends the GenericServlet class and implements a Serializable interface.

```

void doPost (HttpServletRequest req, HttpServletResponse resp);
void doGet ( " " );
void service ( " " );

```

```

void service (ServletRequest req, ServletResponse resp);

```

⑥ Difference between doGet() & doPost()
doGet() → This method is used to handle the GET request of the server-side.

→ This method is automatically ~~suppo~~ call by HttpServlet.

→ The GET type request is usually used to process a request.

→ not secured. If data is not encrypted // Fast

Syntax:-

protected void doGet (HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException.

doPost()

→ This method is used to handle the POST request of the server-side.

→ This method allows the client to send data of unlimited length to the webserver at a time.

→ The POST type request is usually used to post-process a request.

→ Secured. | Slow

Syntax:-

protected void doPost (HttpServletRequest req, HttpServletResponse resp) throws SE, IOException

→ whenever we have a `doPost()` in our servlet class, its automatically not supported, we have a `<form action = "url">`. ~~is that form~~ we have whenever we are giving a Submit button by default `[doGet()]` is called, so avoid this conflict we have to use 'post' method inside the form action, like.

`<form action = "url" method = 'post'>`

⑦ What is the difference between `GenericServlet` and `HttpServlet`.

| <u>GenericServlet</u> | <u>HttpServlet</u> |
|--|--|
| ① <code>GenericServlet</code> is protocol independent which can be used with any protocol; | ① <code>HttpServlet</code> is protocol dependent and used only with HTTP protocol. |
| ② In <code>GenericServlet</code> , the <code>service()</code> method is abstract. | ② But <code>service()</code> in <code>HttpServlet</code> is <code>non-abstract</code> . |
| ③ <code>GenericServlet</code> extends <code>Object</code> class and implements <code>Servlet</code> , <code>ServletConfig</code> , & <code>Serializable</code> interfaces. | ③ <code>HttpServlet</code> extends <code>GenericServlet</code> and implements a <code>Serializable</code> interface. |
| ④ <code>GenericServlet</code> supports only <code>service()</code> which does not contain <code>doGet()</code> and <code>doPost()</code> | ④ <code>HttpServlet</code> also supports <code>doGet()</code> , <code>doPost()</code> , <code>doHead()</code> , <code>doPut()</code> , <code>doOptions()</code> , <code>doDelete()</code> , <code>doTrace()</code> . methods |

⑧ what is deployment descriptor?

It is a file used by the Servlet Container to define which Servlets match up with which URLs.

⑨ Explain Servlet lifecycle?

The entire life cycle of a Servlet is managed by the Servlet Container, which uses the `javax.servlet.Servlet`.

How the Servlet Container manages the Servlet object.

The web browser send the request to the server, when it will receive the first request.

① load class in to the memory (loading Servlet class).

② After loading class in to the memory, it instantiate Servlet class (Creating object).

③ Initialise the members [init()].

④ Triggers Service method (to serve the request based on the no. of request, the service method is called).

⑤ Destroy Servlet.

① Loading a Servlet :-

Loading the Servlet class in to the memory

② Instantiate :-

After loading the Servlet class in to the memory it creates an instance of the Servlet. To create a new instance of the Servlet, the Container uses the no-argument constructor.

③ Initializing a Servlet :-

→ After the Servlet is instantiated successfully the Servlet Container initializes the instantiated Servlet object.

→ By using `init()` : `(ServletConfig)` - (one for Servlet.)
If the Servlet fails to initialize, then it informs the Servlet Container by throwing the `ServletException`.

④ Trigger Service method :-

After initializing the Servlet instance is ready to serve the client requests. it performs the following operations like,

→ it creates the `HttpServletRequest` and `HttpServletResponse`

→ After Creating the request and response object is invokes the `ServletService()` (`ServletRequest`, `ServletResponse`) by passing the request and response objects.

⑤ Destroy Servlet -

When a Servlet Container decides to Destroy the Servlet, it performs some tasks.

→ it allows all the threads currently running in the `Service` method of the Servlet instance to complete their jobs.

→ After currently running threads have completed their jobs, the Servlet Container calls the `destroy()` method of the Servlet instance.

After the `destroy()` method is executed, the Servlet Container release all the references of the Servlet instance so that it becomes eligible for garbage collection.

→ `init()`

→ `service()`

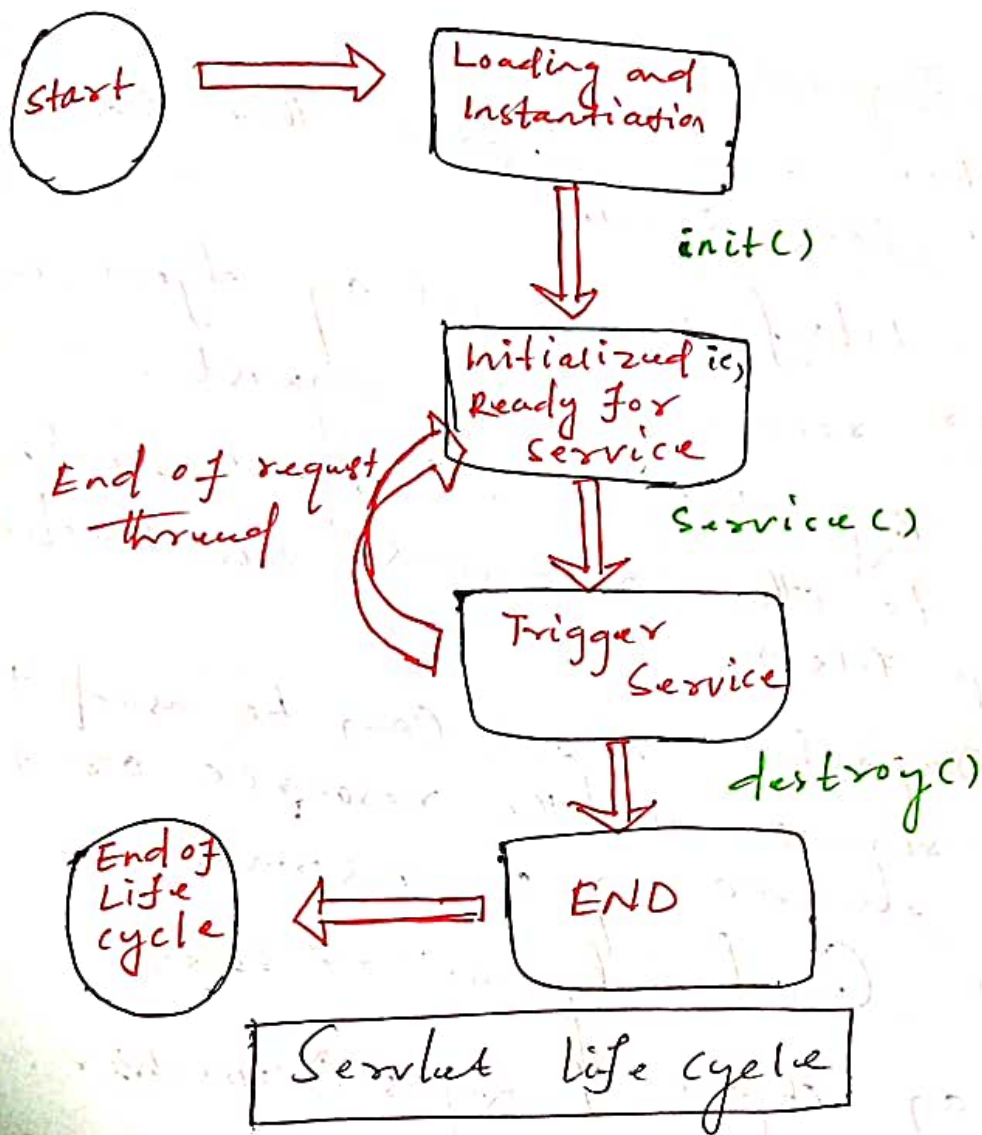
→ `destroy()`



Called only once

based on the no of request the `service` method is called

Called only once



⑩ what is RequestDispatcher?

RequestDispatcher is an interface that comes under package `javax.servlet`.

- Using this interface we get an object in `Servlet` after receiving the request.
- Using the RequestDispatcher object we send a request to other resources which include (HTML, JSP file).
- A RequestDispatcher object can be used to forward a request to the resource or to include the resource in a response. The resource can be dynamic or static.
- To create an object of RequestDispatcher.

```
[RequestDispatcher requestDispatcher = req. getRequestDispatcher  
("stringpath");]
```

- request is the `HttpServletRequest` type object.
- path is a String specifying the path name to the resource. ~~If it is relative,~~

⑪ what is Forward & Include?

```
[requestDispatcher.forward(req, resp);]
```

- This method is used to forward a request from a Servlet to another resource (JSP file, HTML) in the server.
- if it is reloading the same page. (include)

include

`[requestDispatcher.include(req, resp);]`

→ This method is used to include the response of resource (to which the request passed servlet, jsp page, html page) in the current servlet response.

(12) what is welcome file list?

welcome file list of web.xml files. If you don't supply a file name while loading the project of the browser, the tag `<welcome-file-list>` is used to define the files that will be called by the server by default.

Contains

index.html
index.htm
index.jsp
index.html
default.html
default.htm
default.jsp

(13) What is load-on-startup?

The element 'load-on-startup' is used to load the Servlet. The void init() method of Servlet gets executed when the server gets started. The element content of 'load-on-startup' is integer.

→ If the integer is negative. The container loads Servlet at any time.

→ If the integer is 0 or positive. The Servlet marked with lower integer are loaded before Servlets marked with higher integer.

(14) What is PrintWriter?

In Servlets the output can be either character or byte. For character data we can use PrintWriter, for other use ServletOutputStream.

PrintWriter → prints text data to a character stream.

getWriter → Returns a PrintWriter object that can send character text to the client.

[PrintWriter printWriter = resp.getWriter();]

(15) what are different mechanism that can be use for session tracking.

- ① URL Re-writing
- ② Hidden Form field
- ③ Cookies (small)
- ④ Session (HttpSession)

(16) what is query string?

→ A query string is extra info. URL to the right of the path to the Servlet. Using this a programmer can know the data which is sent from the client.

→ it will start with a "?" and the fields are separated by "&"

! Home ? user = madhu & id = 123

(17) what is HttpSession?

→ HttpSession is an interface in terms of which we can store key and value pair [string & object]

→ HttpSession is used to achieve session tracking which means maintaining the state (data) of the user. in order to recognize him.

→

18) How to Create a HttpSession Object?

→ Any kind of object can be stored in the session, be it a text, database.

→ Session is not dependent of the client's browser.

→ Session are secure and transparent.

```
HttpSession session = req.getSession();
```

```
session.setAttribute("id");
```

19) What is cookies and how to create a cookies.

→ It is a textual information about some website, it can not store object.

→ Cookie can store all the information in your browser.

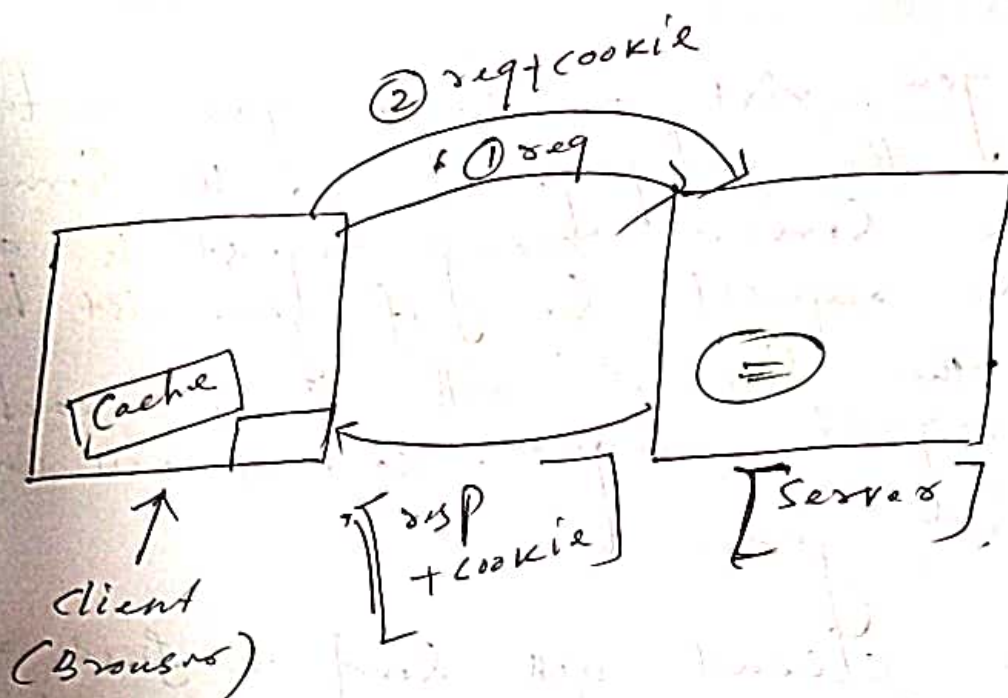
→ When you visit a particular website, some information is saved in your local system so that when you visit the same website again, this website is able to recognize you & show you the results according to your preferences.

→ Cookies can have been long used in the internet history & have developed in a magnificent way.

→ When you visit a website you actually request the web page from the server.
→ For a server, every request is a unique request. So if you visit hundred times, the server will consider each and every request unique. So, it will uniquely remember you.

→ Here client will send the first request to the server and server will create a cookie (small information) & server will send the response along with cookie. and client will receive the response and cookies & store it in a cache form.

→ After that when client will send the second request to the server, it will send request as well as cookie (same cookie whatever server has given so change). & server will identify that & check that cookies, means the client has visited before & if the data is valid then the server will give the valid response according to you as the server will recognize you.



Drawbacks of Cookies

- They can only keep textual information.
- They are browser dependant. Hence if the client disables them, your web application can't make use of them.
- It can not store object.
- It's not secure.

20) What is ServletContext?

ServletContext is the Object created by Servlet Container to share initia parameters or Configuration information to the whole application.

Ex:-

→ ServletContext is a for application

Ex:-

Suppose, one name of the job portal is "abcWebsite.tg" showing the website name at the top of webpages delivered by different Servlets, we need to store the website name in every Servlet, i.e.,
Since the information shared by ServletContext can be accessed by every Servlet, it is better to go with ServletContext and retrieve the website name using
`getServletContext().getInitParameter("Name")`.

②) What is ServletConfig

- ServletConfig is for specific Servlets means one for Servlet.
- ServletConfig is an object containing some initial parameters or configuration information created by Servlet Container and passed to the Servlet during initialization.
- ServletConfig is for a particular Servlet, that means one should store Servlet specific information in web.xml and retrieve them using this object.
- One Servlet can have multiple initparameters and it can store the data in key and value pair.

Ex

Suppose, one is building a job portal and desires to show different email ids (which may get change over time) to recruiter and job applicant, so he decides to write two Servlets one for handling recruiter's and another one for the job applicant.

put email-id as name-value pair for different Servlet inside web.xml which can further be retrieved using getServletConfig()

ServletConfig().getInitParameter("name");

Ex:-

```
public class Recorder extends HttpServlet {  
    protected void doGet(HttpServletRequest request,  
        HttpServletResponse response) throws  
        ServletException, IOException {  
        {
```

```
        String email = getServletConfig().getInitParameter  
            ("Email");
```

```
        String website = getServletContext().getParameter  
            ("website-name");
```

```
        PrintWriter out = response.getWriter();
```

```
        out.println("<center><h1>" + website + "</h1>  
            </center><br><p>Contact us:  
            + email);
```

→ So ServletContext parameter, like

<Context-param> is outside of <servlet>
in web.xml, hence it is independent of
servlet and accessible from whole app.

(22)

Difference between ServletConfig & ServletContext

ServletConfig (object) (Created by Servlet Container)

- ServletConfig is Servlet specific.
- Parameters of ServletConfig are present as key-value pair in `<init-param>` inside `<Servlet>`.
- ServletConfig object is object obtained by `getServletConfig()` method.
- Each Servlet has got its own ServletConfig object.
- Use ServletConfig when only one Servlet reads information shared by it.

ServletContext

- ServletContext is for whole application.
- Parameters of ServletContext are present as key-value pair in `<context-param>` which is outside of `<Servlet>` and inside `<web-app>`.
- ServletContext object is obtained by `getServletContext()` method.
- `getServletContext()` method.
- ServletContext object is only one and used by different Servlets in the application.
- Use ServletContext when whole application reads information shared by it.