**Servlet-question-answer:**

**1.What is web-application?**

\*A web-application is an application program that is usually stored on a remote server, and users can access it through the use of **Software** known as **web-browser.**

**\*Web browser**

A web browser (also referred to as an Internet browser or simply a browser) is application software for accessing the World Wide Web or a local website.

Example: google-chrome, opera, firefox etc......

**2.List different internet protocols?**

**1.TCP/IP (Transmission Control Protocol/ Internet Protocol)**

**These are a set of standard rules that allows different types of computers to communicate with each other**

**The functionality of TCP/IP is divided into 4 layers**

**1.Application Layer:**

**2.Transport Layer:**

**3.Internet Layer:**

**4.Network Layer:**

**2. SMTP(Simple Mail Transfer Protocol):**

**These protocols are important for sending and distributing outgoing emails.**

**3. FTP (File Transfer Protocol):**

This protocol is used for transferring files from one system to the other. This works on a client-server model.

**4. HTTP(HyperText Transfer Protocol):**

This protocol is used to transfer hypertexts over the internet and it is defined by the www(world wide web) for information transfer. This protocol defines how the information needs to be formatted and transmitted.

5. HTTPS(HyperText Transfer Protocol Secure):

HTTPS is an extension of the Hypertext Transfer Protocol (HTTP). It is used for secure communication over a computer network with the SSL/TLS protocol for encryption and authentication.

**4.What is the difference between static and dynamic web application?**

**Static Website:**

A static website is one with stable content, where every user sees the exact same thing on each individual page.

A static website is made up of webpages created using [HTML](https://www.wix.com/encyclopedia/definition/html-hypertext-markup-language), [CSS](https://www.wix.com/encyclopedia/definition/cascading-style-sheets-css) and Javascript (all examples of web development languages). Each page on a static website is stored as a single HTML file,

**Dynamic Website:**

a dynamic website is one where content is pulled on-the-fly, allowing its content to change with the user.

Built using server side language and technology, dynamic websites allow for the content of each page to be delivered and displayed dynamically, or on-the-fly, according to user behavior or from user-generated content.

**5.What is web server?**

A web server is software and hardware that uses [HTTP](https://www.techtarget.com/whatis/definition/HTTP-Hypertext-Transfer-Protocol) (Hypertext Transfer Protocol) and other protocols to respond to [client](https://www.techtarget.com/searchenterprisedesktop/definition/client) requests made over the World Wide Web.

The main job of a web server is to display website content through storing, processing and delivering webpages to users.

Web server software is accessed through the domain names of websites and ensures the delivery of the site's content to the requesting user.

**6.What is application server?**

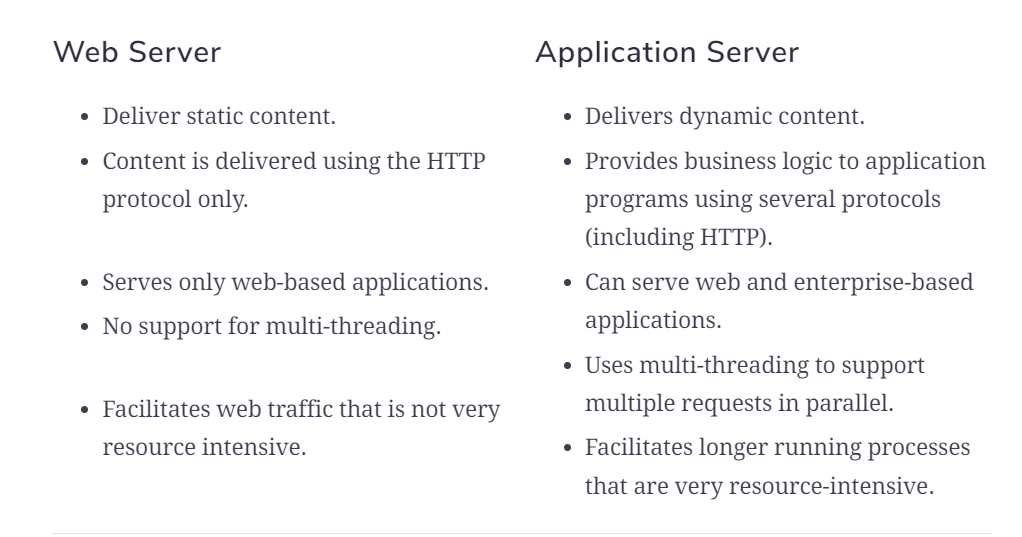
Application Server is a type of server designed to install, operate, and host applications.

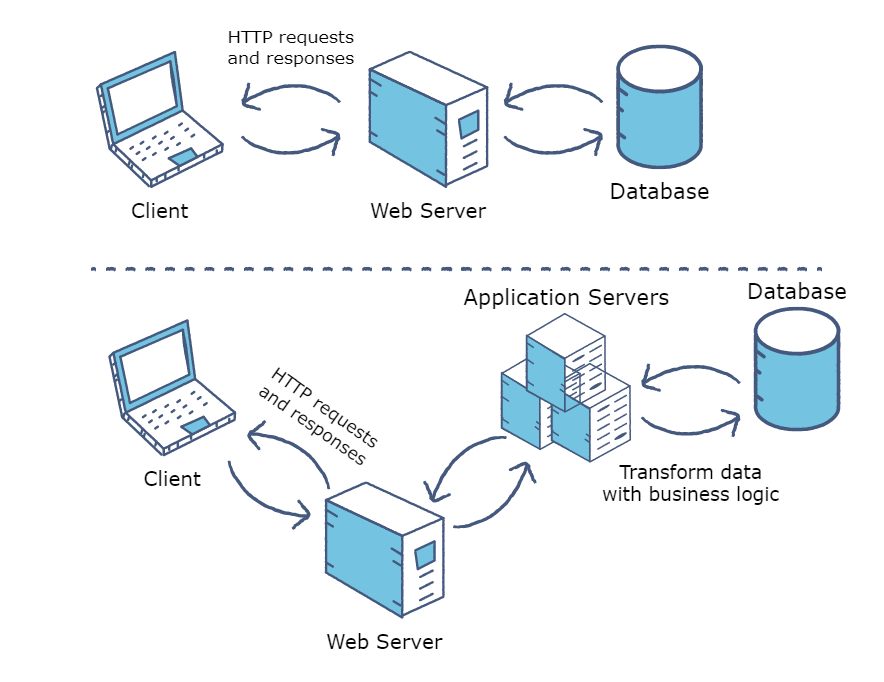
An application server is a program that resides on the server-side, and it’s a server programmer providing business logic behind any application. This server can be a part of the network or the distributed network.

They are basically used in a web-based application that has 3 tier architecture. The position at which the application server fits in is described below:

Tier-1=web server, tier-2=application server, tier-3=database server

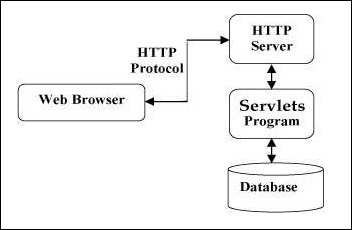
**7.What is the difference between web server and application server?**



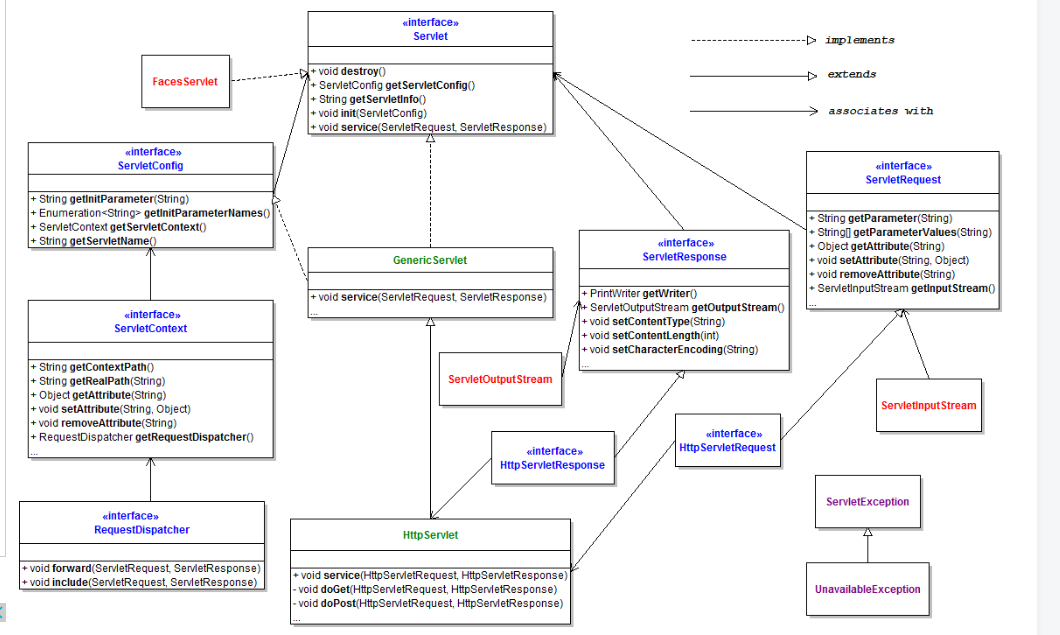


**8.What is servlet?**

* 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 a web component that is deployed on the server to create a dynamic web page
* Servlet is an interface.



**9.What is the difference between Generic servlet and http servlet?**



**Servlet:**

1. The Servlets runs as a thread in a web-container instead of in a separate OS process.
2. Only one object is created first time when first request comes, another request share the same object.
3. Servlet is platform independent.
4. Servlet is fast.

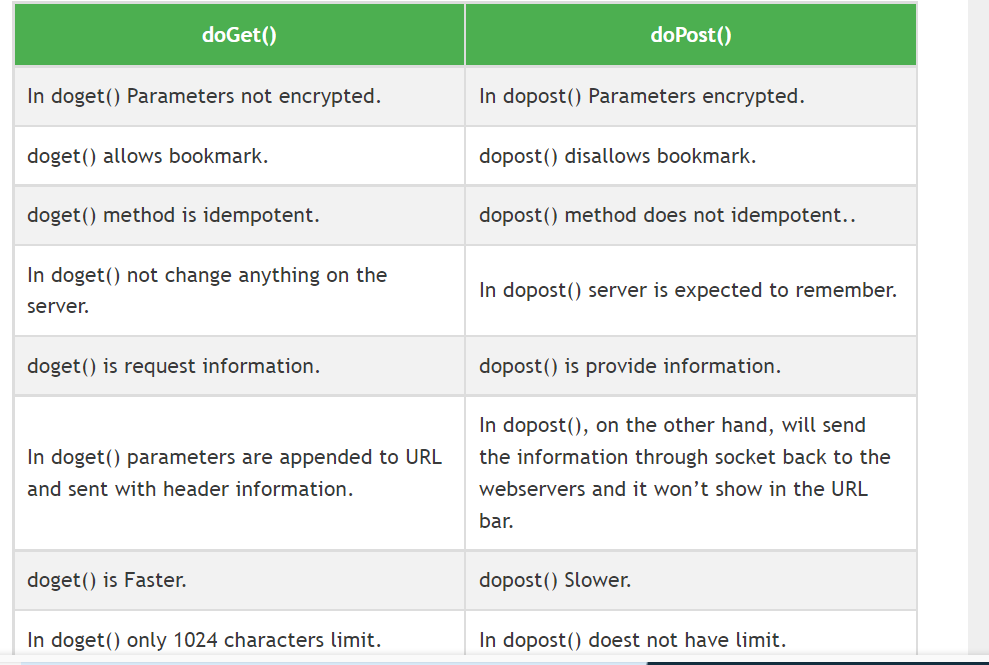
**Generic Servlet:**

1. General for all protocol.
2. Implements Servlet Interface.
3. Use Service method.

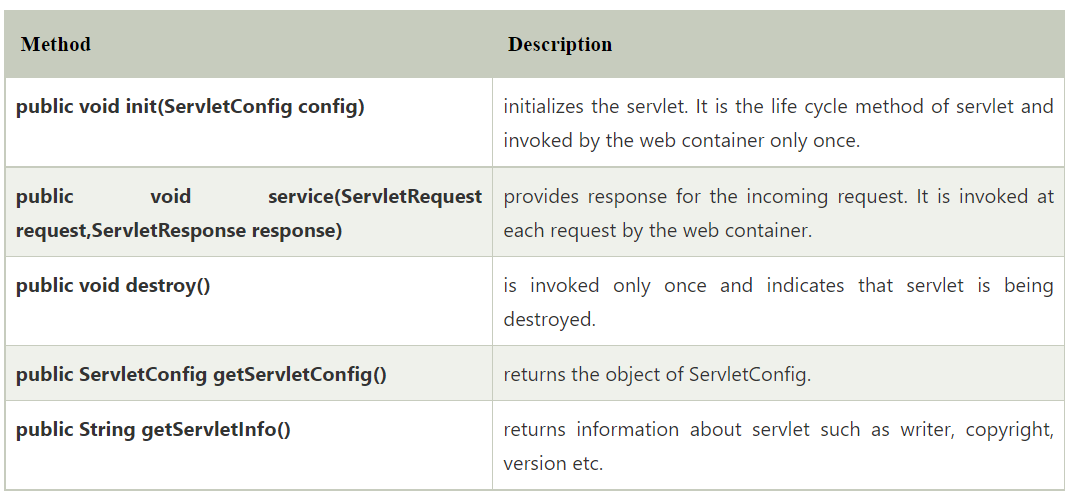
10.List and explain Http Servlet Methods?

1. **public void service(ServletRequest req,ServletResponse res)** dispatches the request to the protected service method by converting the request and response object into http type.
2. **protected void service(HttpServletRequest req, HttpServletResponse res)** receives the request from the service method, and dispatches the request to the doXXX() method depending on the incoming http request type.
3. **protected void doGet(HttpServletRequest req, HttpServletResponse res)** handles the GET request. It is invoked by the web container.
4. **protected void doPost(HttpServletRequest req, HttpServletResponse res)** handles the POST request. It is invoked by the web container.

**11.What is the difference between doGet and doPost?**



**12.What are the different methods of servlet interface (List and explain)?**



**13.** **What is deployment descriptor?**

In a java web application a file named web.xml is known as deployment descriptor.

It is a xml file and <web-app> is the root element for it. When a request comes web server uses web.xml file to map the URL of the request to the specific code that handle the request.

**14. what is the difference between servlet context and servlet config?**

|  |  |
| --- | --- |
| **servlet context** | **servlet config** |
| ServletConfig object is one per servlet class. | ServletContext object is global to the entire web application. |
| Object of ServletConfig will be created during the initialization process of the servlet. | Object of ServletContext will be created at the time of web application deployment |
| We have to give the request explicitly in order to create the ServletConfig object for the first time | ServletContext object can be available even before giving the first request |
| getServletConfig() method is used to obtain Servletconfig object | getServletContext() method is used to obtain ServletContext object |
| In web.xml — <init-param> tag will be appear under <servlet-class> tag. | In web.xml — <context-param> tag will be appear under <web-app> tag. |

**15. How to configure the config parameters in web.xml?**

**Note:**

**<init-param>**

**<param-name>loan</param-name>**

**<param-value>farmer</param-value>**

**</init-param>**

**Step-1: we will create one servlet class.**

**Class:**

**public class StateA extends HttpServlet{**

**@Override**

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

**ServletConfig config = getServletConfig();**

**System.out.println(config.getInitParameter("loan"));**

**}**

**}**

**Web.xml:**

**<web-app>**

**<display-name>Archetype Created Web Application</display-name>**

**<!-- state-a-mapping -->**

**<servlet>**

**<servlet-name>StateA</servlet-name>**

**<servlet-class>com.ty.StateA</servlet-class>**

**<init-param>**

**<param-name>loan</param-name>**

**<param-value>farmer</param-value>**

**</init-param>**

**</servlet>**

**<servlet-mapping>**

**<servlet-name>StateA</servlet-name>**

**<url-pattern>/statea</url-pattern>**

**</servlet-mapping>**

**</web-app>**

**16.How to configure the context parameter?**

**Class:**

public class StateA extends HttpServlet{

@Override

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

ServletContext context = getServletContext();

System.out.println(context.getInitParameter("url"));

}

}

**Web.xml**

<web-app>

<display-name>Archetype Created Web Application</display-name>

<context-param>

<param-name>url</param-name>

<param-value>Mysql:jdbc</param-value>

</context-param>

<servlet>

<servlet-name>StateA</servlet-name>

<servlet-class>com.ty.StateA</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>StateA</servlet-name>

<url-pattern>/statea</url-pattern>

</servlet-mapping>

</web-app>

**17.What is session tracking what are the different ways to implement it?**

**Session** simply means a particular interval of time.

**Session Tracking** is a way to maintain state (data) of a user. It is also known as **session management** in servlet.

### **Why use Session Tracking?**

**To recognize the user,** it is used to recognize the particular user.

### **Session Tracking Techniques:**

There are four techniques used in Session tracking:

1. Cookies
2. Hidden Form Field
3. URL Rewriting
4. HttpSession

**Note:** Here we will learn only Cookies and HttpSession.

**18.What is cookie?**

A **cookie** is a small piece of information in the form of key and value pair which is stored in the client side. That is persisted between the multiple client requests.

Cookies are small files that websites send to your device that the sites then use to monitor you and remember certain information about you like what’s in your shopping cart on an e-commerce site, or your login information.

**Note:** Cookies can store only String Values.

**19.How to create a cookie?**

**Cookie cookie = new Cookie("myName", name);**

**resp.addCookie(cookie);**

**Class:**

@WebServlet(value = "/statec")

public class StateC extends HttpServlet {

@Override

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

String name = req.getParameter("myName");

String age = req.getParameter("myAge");

int myAge = Integer.parseInt(age);

Cookie cookie = new Cookie("myName", name);

resp.addCookie(cookie);

PrintWriter printWriter = resp.getWriter();

printWriter.write("<html><body>");

printWriter.write("<h1>recieved the data</h1>");

printWriter.write("<h3><a href='stated'>proceed</a></h3>");

printWriter.write("</body></html>")

}

}

**Class:**

@WebServlet(value = "/stated")

public class StateD extends HttpServlet{

@Override

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

Cookie[] cookies = req.getCookies();

for (Cookie cookie : cookies) {

System.out.println(cookie.getName()+"\t"+cookie.getValue());

}

PrintWriter printWriter = resp.getWriter();

printWriter.write("<html><body><h1>Welcome to stateD</h1></body></html>");

}

}

**Html:**

<!DOCTYPE html>

<html>

<head>

<meta charset="ISO-8859-1">

<title>Insert title here</title>

</head>

<body>

<form action="statec">

Name:<input type="text" name="myName"><br>

Age:<input type="number" name="myAge"> <br>

<input type="submit" value="Submit">

</form>

</body>

</html>

**20.How to destroy the cookie?**

**21.What is HttpSession?**

A session contains information specific to a particular user across the whole application.

In web terminology, a session is simply the limited interval of time in which two systems communicate with each other.

HttpSession provides information of user login and it will for whole application if user logout then he can't back to his profile and other parts of application. if he wants to access then he must login again.

**Note:** In session we can store the data in key and value pair.

Where key=String and Value=Object

**22.What is the use of getSession()?**

getSession() returns the valid session object associated with the request, identified in the session cookie that is encapsulated in the request object.

\* **HttpSession httpSession = req.getSession();**

Here getSession() method will get the HttpSession object to set the attributes for httpsession.

**httpSession.setAttribute("userName", email);**

\*Here HttpSession will create only one object for entire object.

**23. What is request dispatcher? Explain**

\*The Request Dispatcher interface provides the facility of dispatching the request to another resource it may be html, servlet or jsp.

\* This interface can also be used to include the content of another resource also. It is one of the ways of servlet collaboration.

We have two important method

1.include(req,resp);

2.forward(req,resp);

**Note:** If we want to include one html page to another html page

RequestDispatcher request = req.getRequestDispatcher("HtmlPageName");

Request.include(req,res);

**Note:** If we want to forward one html page to another html page

RequestDispatcher request = req.getRequestDispatcher("HtmlPageName");

Request.forward(req,res);

**24.Difference between Forward and include?**

**25.Difference between forward and send redirect?**

**26.What is send Redirect?**

sendRedirect() method redirects the response to another resource, inside or outside the server. It makes the client/browser to create a new request to get to the resource. It sends a temporary redirect response to the client using the specified redirect location URL.

resp.sendRedirect("https://www.geeksforgeeks.org/url-rewriting-using-java-servlet/");

**27.Explain Servlet Lifecycle?**

\*Load class in a memory.

\*Class Object will be created.

\*Create request and response object created

\*Invoke service method

**Note:** Servlet is a singleton class.

**What is Singleton class?** -> it will create only one object throughout all the application.

1.Servlet Load into memory & Get Instantiated.

2.Initialize the member {init()}

3.Invoke service method {service (req,res)}

4.Servlet destroyed {destroy()}

**28.What is JSP?**

\*Java Server Pages (JSP) is **a Java standard technology that enables you to write dynamic**, data-driven pages for your Java web applications.

\*Inside Jsp we write java code also what is not possible in html

**Note:** In jsp we will use JSP tag.

**29.Explain Each JSP Tag?**

We have multiple tags in jsp:

**Declaration Tag:**

<%! %> when you want to declare something <%! Int sum=0 %>

**Expression Tag:**

**<%= %>** To display any value of a variable. <% = sum %>

**Scriplet Tag:**

We use scriplet tag to write business logic:

<% %>

<%

For(int I=0;I<=5;I++){

%>

<h1><%=I%></h1>

<%>}<%>

Note: Whatever will write in declaration tag loaded in class.

Scriplet and expression will load in service method ().

**31.Explain JSP Life Cycle?**