INSY-413

WEB TECHNOLOGY & INTERNET

Theoretical and practical course, 4 Credits

Blessed assurance, Jesus is mine
O what a foretaste of glory divine
Heir of salvation, purchase of God
Born of His Spirit, washed in His blood

This is my story, this is my song
Praising my Savior all the day long
This is my story, this is my song
Praising my Savior all the day long

COURSE CONTENT

⇒ **Prerequisite: Java Programming, Web Design**

- Understanding the Internet, WWW and Web Application
- Servlets Technology
- JSP Application Development
- Java Server Faces Technology
- Web Services
- Mini Project

Module 2 – Understanding Internet ,WWW, and Web Application

UPON COMPLETION OF THE MODULE THE STUDENT SHOULD BE ABLE TO:

- Describe the Internet and its history
- Explain the World Wide Web, HTTP, DNS, URL, Port, WEB Servers, Application Servers, and WEB Browsers
- Examine Internet Protocol Suite
- Understand Web Application Building Block
- Implementing a simple Web Application

Internet

- Internet is a short form of the technical term ***internetwork***, the result of interconnecting computer networks with special gateways or routers.
- **The Internet is a global network of interconnected computers and servers that communicate with each other through standardized protocols**.
- It enables the exchange of information, data, and communication between devices worldwide, providing a platform for various services such as email, online shopping, social media, and more.
- **It operates without a central governing body.**
- However, to maintain interoperability, all technical and policy aspects of the underlying core infrastructure and the principal name spaces are administered by the **Internet Corporation for Assigned Names and Numbers (ICANN).**

History of Internet

- The Internet has its roots in the **1960s** when the U.S. Department of Defense created ARPANET, a precursor to the modern Internet.
- Over the decades, the Internet evolved through the development of new technologies, such as the **World Wide Web, email, and the rise of mobile devices.**
- **In the 1990s, the commercialization of the Internet began**, leading to an explosion of growth and the development of e-commerce, social media, and other online services.
- Today, the Internet is a vital tool for communication, commerce, and information exchange, with billions of people accessing it globally.

World Wide Web

- WWW is a system of interlinked hypertext documents accessed via the Internet.
- The World Wide Web, or simply **Web**, **is a way of accessing information over the medium of the Internet**.
- **It is an information-sharing model that is built on top of the Internet.**
- **The Web uses the HTTP protocol,** only one of the languages spoken over the Internet, **to transmit data.**
- It was created in 1989 by British computer scientist Tim Berners-Lee, who envisioned a way for people to share information and access it easily from anywhere in the world.

World Wide Web Consortium (W3C)

- The World Wide Web Consortium (W3C) is an **international standards organization** founded in 1994 by Tim Berners-Lee, the inventor of the World Wide Web.
- Its mission is to lead the Web to its full potential by developing and promoting open standards that ensure its long-term growth and interoperability.
- The W3C develops technical standards, such as HTML, CSS, and JavaScript, and provides guidelines and tools to improve accessibility and the user experience on the Web.
- It also fosters collaboration between members, which include governments, corporations, and individuals, to advance the Web as a platform for innovation and creativity.

Web Page

- A web page is a **document** or information resource that is **suitable for the World Wide Web** and **can be accessed through a web browser and displayed on a monitor or mobile device**.
- This information is usually in HTML or XHTML format and may provide navigation to other web pages via hypertext links.
- Web pages frequently include other resources such as style sheets, scripts, and images in their final presentation.
- Web pages are requested and served from web servers using Hypertext Transfer Protocol (HTTP).
- Web pages may consist of files of static text and other content stored **within the web server's** file system (**static web pages**), or may be **constructed by server-side software when they are requested** (**dynamic web pages**).

Website

- A website is a collection of related web pages containing images, videos, and other digital assets that are hosted on a single server and can be accessed through a single domain name.
- All publicly accessible websites collectively constitute the World Wide Web.
- Web sites can be static or dynamic.
- A dynamic website is a type of website that generates and displays different content each time it is accessed, based on user input or changes to the database. This is in contrast to a static website, which displays the same content every time it is visited.
- Dynamic websites are often used for e-commerce, social media, and other applications that require user interaction and the ability to update content in real-time. They are built using server-side technologies, such as PHP, Ruby on Rails, or ASP.NET, that enable the server to process user requests and generate customized content on the fly.

Assignment 1. Understanding Web Applications

- Create a Simple Web Application with
- a student admission page (the form should have at least 10 fields)
- a login page

Use HTML, style with CSS, and validate the forms with JavaScript.

Servlets Technology

Servlets

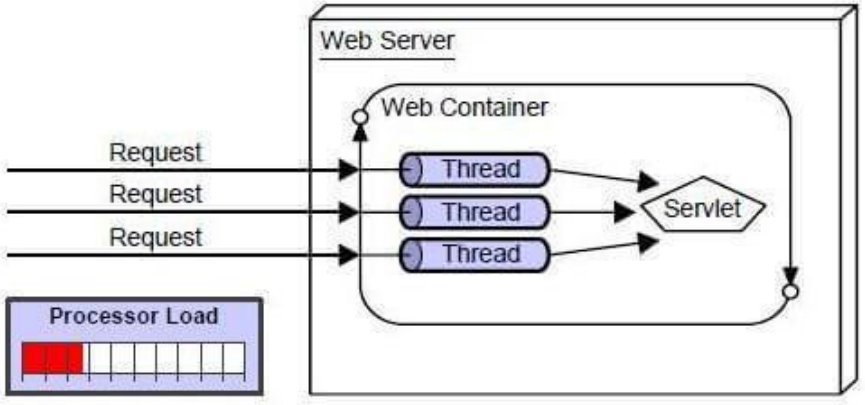
- **Servlet** technology is used to create a web application (resides at server side and generates a dynamic web page).
- **A Servlet is a Java class** that is **used to extend** the capabilities of servers that host applications accessed by means of a request-response programming model.
- The **server-side extensions** are nothing but the technologies that are used to create dynamic Web pages.
- Servlets can be used to **process HTTP requests**, **handle form submissions**, **manage sessions**, and **perform other server-side tasks**.
- Servlets are part of Java Enterprise Edition (Java EE) and are **standard technology for building web-based applications**.

Servlets (Cnt'd)

- **Servlet** technology is robust and scalable because of Java language.
- Before Servlet, CGI (Common Gateway Interface) scripting language was common as a server-side programming language.
- However, there were many disadvantages to CGI technology:
  - If the number of clients increases, it **takes more time for sending the response**.
  - **For each request, it starts a process**, and the web server is limited to start processes.
  - It uses **platform-dependent language** e.g. C, C++, perl.

Advantages of Servlets

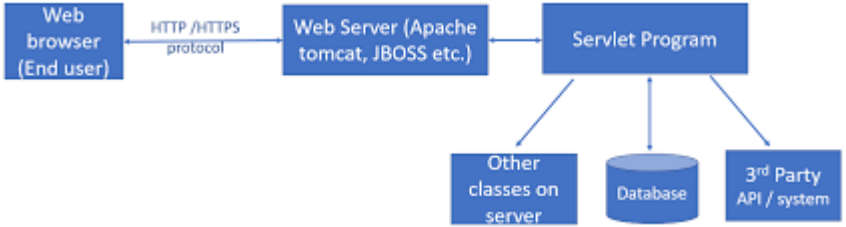
- **Better performance:** because it creates a thread for each request, not process.
- **Portability:** because it uses Java language.
- **Robust:** JVM manages Servlets, so we don't need to worry about the memory leak, garbage collection, etc.
- **Secure:** because it uses Java language.



Servlet Architecture

Execution of Servlets basically involves six basic steps:

- The clients send the request to the webserver.
- The web server receives the request.
- The web server passes the request to the corresponding servlet.
- The servlet processes the request and generates the response in the form of output.
- The servlet sends the response back to the webserver.
- The web server sends the response back to the client and the client browser displays it on the screen.



The Servlet Container

- **Servlet container**, also known as **Servlet engine** is an integrated set of objects that provide a run time environment for Java Servlet components.
- In simple words, it is a system that manages Java Servlet components on top of the Web server to handle the Web client requests.

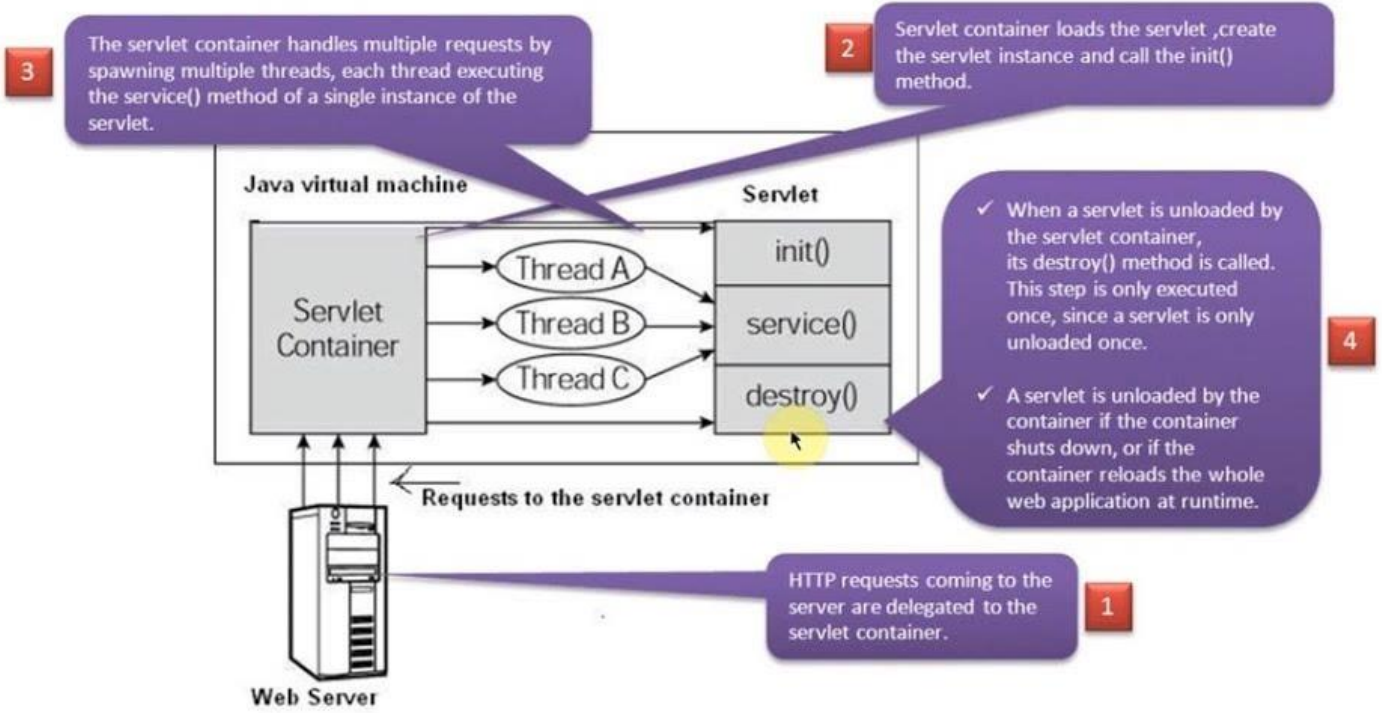
The Servlet Container

Services provided by the Servlet container :

- **Network Services:** Loads a Servlet class. The loading may be from a local file system, a remote file system or other network services. The Servlet container provides the network services over which the request and response are sent.
- **Decode and Encode MIME-based messages:** Provides the service of decoding and encoding MIME-based messages.
- **Manage Servlet container:** Manages the lifecycle of a Servlet.
- **Resource management** Manages the static and dynamic resources, such as HTML files, Servlets, and JSP pages.
- **Security Service:** Handles authorization and authentication of resource access.
- **Session Management:** Maintains a session by appending a **session ID** to the URL path.

Servlets & Jsp

How Servlets works



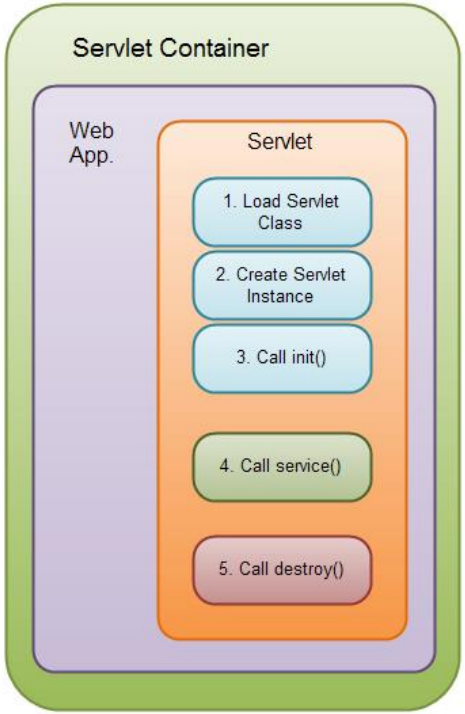
Servlet API Interfaces and Classes

- The **javax.servlet** package contains many interfaces and classes that are used by the servlet or web container. These are not specific to any protocol.
- The **javax.servlet.http** package contains interfaces and classes that are responsible for http requests only.

javax.servlet Package	javax.servlet.http Package
Interfaces: Servlet ServletConfig ServletContext ServletRequest ServletResponse RequestDispatcher Filter FilterChain FilterConfig ServletRequestListener ServletRequestAttributeListener ServletContextListener ServletContextAttributeListener	**Interfaces:** HttpServletRequest HttpServletResponse HttpSession HttpSessionListener HttpSessionAttributeListener HttpSessionBindingListener HttpSessionActivationListener
Classes: GenericServlet ServletRequestWrapper ServletResponseWrapper ServletInputStream ServletOutputStream ServletContextEvent ServletContextAttributeEvent ServletRequestEvent ServletRequestAttributeEvent ServletException UnavailableException	**Classes:** Cookie HttpServlet HttpServletRequestWrapper HttpServletResponseWrapper HttpSessionEvent HttpSessionBindingEvent

Methods of Servlet interface

- Servlet interface needs to be implemented for creating any servlet (either directly or indirectly).
- It provides 3 life cycle methods that are used to **initialize** the servlet, to **service** the requests, and to **destroy** the servlet and 2 non-life cycle methods:
  - **getServletConfig()** returns the object of ServletConfig
  - **getServletInfo()** returns information about servlet such as writer, copyright, version etc.



Steps to create a servlet

There are given 6 steps to create a **servlet example**. These steps are required for all the servers. The servlet example can be created by three ways:

- By implementing Servlet interface,
- By inheriting GenericServlet class, (or)
- By inheriting HttpServlet class

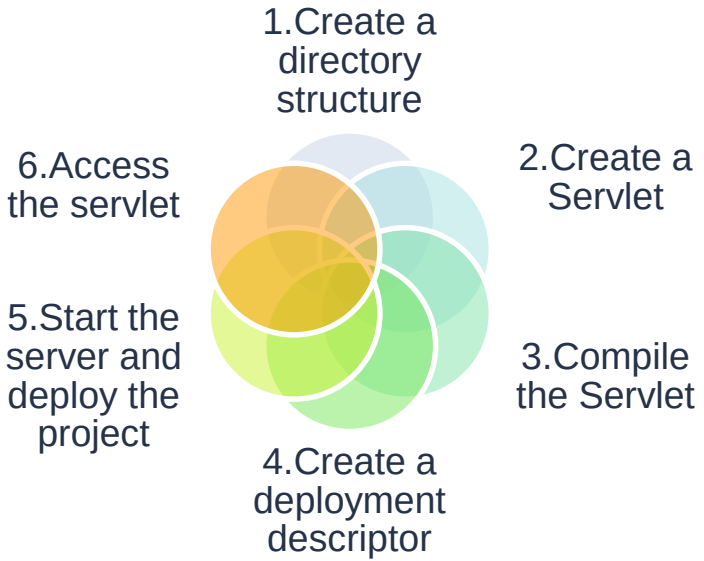
The mostly used approach is by extending HttpServlet because it provides http request specific method such as doGet(), doPost(), doHead() etc.

Steps to create a servlet

There are given 6 steps to create a **servlet example**. These steps are required for all the servers. The servlet example can be created by three ways:

- By implementing Servlet interface,
- By inheriting GenericServlet class, (or)
- By inheriting HttpServlet class

The mostly used approach is by extending HttpServlet because it provides http request specific method such as doGet(), doPost(), doHead() etc.



Notes on GET and POST Methods

GET Method

Appends form-data into the URL in name/value pairs	The length of a URL is limited (about 3000 characters)	Never use GET to send sensitive data! (will be visible in the URL)	Useful for form submissions where a user wants to bookmark the result	GET is better for non-secure data, like query strings in Google

POST Method

Appends form-data inside the body of the HTTP request (data is not shown in URL)	Has no size limitations	Form submissions with POST cannot be bookmarked

Assignment 2. Understanding Web Applications

- Using Servlets create a Simple Number Converter as shown by the sample page (right)
- Perform the CRUD Operations for the Student Admission module. Use Hibernate for persistence.
- Implement the Authentication using servlets.