Here's a structured 30-day learning plan to help you learn Java Servlets from beginner to advanced level, covering both theoretical and practical aspects:

Week 1: Basics of Java Servlets

Day 1-3: Introduction to Web Applications and Servlets

- **Theoretical**:
  - Understand web applications and how servers and clients communicate.
  - Learn about the HTTP protocol (GET, POST methods).
  - Introduction to Servlets: what they are, their lifecycle, and how they fit into web development.
- **Practical**:
  - Set up the environment: Install Java Development Kit (JDK), Apache Tomcat, and Eclipse/IntelliJ IDE.
  - Create a simple "Hello World" Servlet.

Day 4-5: Servlet Lifecycle and Deployment

- **Theoretical**:
  - Learn about the Servlet lifecycle (init(), service(), destroy()).
  - Understand deployment descriptors (`web.xml`) and annotations.
- **Practical**:
  - Create a Servlet that handles both GET and POST requests.
  - Deploy your Servlet on Tomcat using `web.xml`.

Day 6-7: Handling Request and Response

- **Theoretical**:
  - Learn about `HttpServletRequest` and `HttpServletResponse` objects.
  - Introduction to request parameters and headers.
- **Practical**:
  - Create a form and a Servlet that handles form data submission using GET and POST.
  - Practice sending headers and cookies.

Week 2: Intermediate Concepts

Day 8-9: Session Management

- **Theoretical**:

  - Learn about sessions, cookies, and URL rewriting.

  - Understand how to maintain state in a web application using `HttpSession`.

- **Practical**:

  - Implement a login page and use sessions to track the user login state.

Day 10-11: ServletConfig and ServletContext

- **Theoretical**:

  - Learn about `ServletConfig` and `ServletContext` for accessing initialization parameters.

  - Understand their difference and use cases.

- **Practical**:

  - Implement an example using `ServletConfig` and `ServletContext` to manage global parameters.

Day 12-14: Filters and Listeners

- **Theoretical**:

  - Understand Servlet Filters and how they are used for logging, security, and request modification.

  - Learn about event listeners and how they work in the Servlet context.

- **Practical**:

  - Implement a simple filter for logging requests and responses.

  - Use listeners to track user session creation and destruction.

Week 3: Working with Databases and JSP

Day 15-17: Connecting Servlets with Databases

- **Theoretical**:
  - Learn JDBC basics and how to connect a Servlet to a database (e.g., MySQL).
  - Understand how to retrieve, insert, update, and delete records from the database.
- **Practical**:
  - Create a CRUD (Create, Read, Update, Delete) application using Servlets and JDBC.

Day 18-20: Introduction to JSP (JavaServer Pages)

- **Theoretical**:
  - Understand the need for JSP and its role in web development.
  - Learn JSP syntax and how to use scriptlets, expressions, and declarations.
- **Practical**:
  - Convert your Servlet-based CRUD app to use JSP for displaying data.
  - Practice forwarding requests from Servlets to JSP using RequestDispatcher.

Week 4: Advanced Concepts

Day 21-23: MVC Architecture and Best Practices

- **Theoretical**:
  - Understand the Model-View-Controller (MVC) design pattern and its importance in web applications.
  - Learn best practices for Servlet and JSP integration.
- **Practical**:
  - Refactor your CRUD application to follow the MVC pattern.

Day 24-26: Exception Handling and Security

- **Theoretical**:
  - Learn how to handle exceptions in Servlets.

  - Understand security in Servlets (authentication, authorization, HTTPS).

- **Practical**:

  - Implement exception handling in your application.

  - Set up basic form-based authentication using Tomcat's security features.

Day 27-28: File Upload and Download in Servlets

- **Theoretical**:

  - Learn about handling file uploads and downloads using Servlets.

  - Understand `Multipart` requests and how to process them.

- **Practical**:

  - Implement a file upload and download functionality in your web application.

Day 29-30: Final Project

- **Theoretical**:

  - Review all key concepts.

- **Practical**:

  - Build a complete project that involves user authentication, database interactions, session management, and a dynamic user interface using JSP.

  - Deploy the project on a remote server like Heroku or AWS for hands-on deployment experience.

Resources:

- **Books**: "Head First Servlets and JSP" by Bryan Basham, Kathy Sierra.

- **Online Courses**:

  - Udemy: [Java Servlets and JSP](https://www.udemy.com/course/jsp-and-servlets/)

  - TutorialsPoint: [Servlets Tutorial](https://www.tutorialspoint.com/servlets/index.htm)

- **Practice Platforms**:

  - Codecademy, LeetCode for basic Java problem-solving.

This plan should give you a solid grasp of Java Servlets both theoretically and practically!