Name : Amitendu Bikash Dhusiya
Stream : BCA
Semester : 6th
University Roll No : 15242723010
Paper Code : BCAC601
Paper Name : Advance Java with Web Application

Main Objective of this Paper: This paper helps to achieve new insights from various perspectives. It sheds light even on various aspect of the course and increases our hunger for knowledge. It is wise to say it contributes a lot in respect to the holistic development of every student.

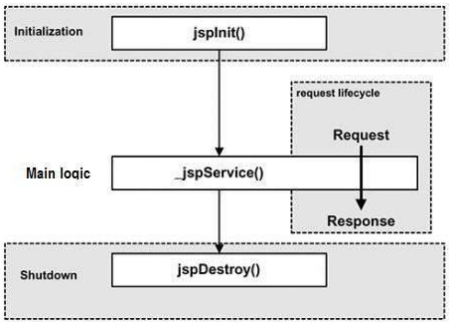
Topics Covered & Assignment Provided: We have managed to gain a comprehensive knowledge on this following topic:

1. Describe properties, bean scopes, and how JavaBeans improve separation of business logic from presentation logic

   JavaBeans use **properties**, which are private data members accessed through public getter and setter methods, ensuring encapsulation and controlled access to data. In JSP, beans can exist in different **scopes**—page (limited to one JSP page), request (valid for a single HTTP request), session (available throughout a user session), and application (shared across the entire application). By moving business logic and data handling into JavaBean classes while keeping presentation code inside JSP, JavaBeans enforce a clear separation of concerns. This reduces scriptlet usage, improves code reusability, enhances maintainability, and supports the MVC architectural pattern in web applications.

2. Explain the complete JSP Life Cycle with diagram.

   The JSP life cycle describes how a JSP page is processed by the web container from creation to destruction. It consists of the following phases:

   - **Translation Phase:** The JSP file is translated into a Servlet source file (`.java`). This happens only the first time the JSP is requested or when it is modified.
   - **Compilation Phase:** The generated Servlet source file is compiled into bytecode (`.class`).
   - **Class Loading & Instantiation:** The compiled Servlet class is loaded into memory and an object is created by the container.
   - **Initialization:** The `jspInit()` method is called once to initialize resources.
   - **Request Processing:** For every client request, the `_jspService()` method is invoked to generate the response.
   - **Destruction:** When the JSP is removed or server shuts down, `jspDestroy()` is called to release resources.

3. Describe the flow of control from client request to database interaction and response generation with proper explanation.

   - **Client Request Initiation:** A user sends an HTTP request (`GET/POST`) from a browser. The request is transmitted over the network to the web server (e.g., Apache Tomcat).

- **Web Server / Servlet Container Processing:** The servlet container:
    i. Identifies the mapped Servlet using web.xml or annotations.
    ii. Creates `HttpServletRequest` and `HttpServletResponse` objects.
    iii. Invokes the `service()` method (which delegates to `doGet()` or `doPost()`).

  At this stage, request parameters are extracted.

- **Controller (*Servlet*) Logic**: The Servlet acts as the **Controller**:
    i. Validates input data.
    ii. Applies business rules.
    iii. Calls the Model layer (Java classes / JavaBeans).
    iv. Prepares data for database interaction.

- **Database Interaction via JDBC**:
  The Model layer:
    i. Loads JDBC Driver.
    ii. Establishes database connection.
    iii. Creates `PreparedStatement` / Statement.
    iv. Executes SQL query (`executeQuery()` / `executeUpdate()`).
    v. Retrieves results in `ResultSet`.
    vi. Closes resources.

  The database processes the SQL and returns results to the application.

- **Data Processing & Response Preparation:**
    i. Retrieved data is stored in JavaBeans or request attributes.
    ii. Servlet attaches data to the request using: `request.setAttribute("data", object);`
    iii. Control is forwarded to a JSP using `RequestDispatcher`.

- **View Generation (JSP):**
    i. The JSP:
        - Accesses request attributes.
        - Uses Expression Language (EL) or JSTL.
        - Generates dynamic HTML content.

- **Response Sent to Client**:
    i. The servlet container converts JSP output into an HTTP response.
    ii. Response is sent back to the client browser.
    iii. Browser renders the HTML page.

4. Explain how JSP, Servlet, and JDBC work together in MVC architecture.

In MVC architecture for Java web applications, JSP, Servlet, and JDBC work together by separating responsibilities into View, Controller, and Model layers. When a client sends an HTTP request, the Servlet acts as the Controller, receiving the request, reading parameters, and performing validation. It then invokes the Model layer, which consists of JavaBeans and JDBC code that handles business logic and interacts with the database by establishing a connection, executing SQL queries, and retrieving results. The processed data is returned to the Servlet, which stores it in request or session scope and forwards the request to a JSP page. The JSP acts as the View, accessing the data using Expression Language (EL) or JSTL and generating dynamic HTML content for the user. Finally, the response is sent back to the client browser. This structured interaction ensures clear separation of concerns, improved maintainability, and scalable application design.

5. Create a JavaBean class for Employee with properties:

- `empId`
- `empName`
- `salary`

Write a **JSP page** to:

- Set bean properties
- Retrieve and display the values
- Explain the scope of the bean

Employee.java

```
package com.demo;
import java.io.Serializable;
public class Employee implements Serializable {
    private int empId;
    private String empName;
    private double salary;
    public Employee() { }
    public int getEmpId() {
        return empId;
    }
    public void setEmpId(int empId) {
        this.empId = empId;
    }
    public String getEmpName() {
        return empName;
    }
    public void setEmpName(String empName) {
        this.empName = empName;
    }
    public double getSalary() {
        return salary;
    }
    public void setSalary(double salary) {
        this.salary = salary;
    }
}
```

App.jsp

```
<%@ page language="java" contentType="text/html;charset=UTF-8" %>
<jsp:useBean id="emp" class="com.demo.Employee" scope="request" />
<!-- Set Bean Properties -->
<jsp:setProperty name="emp" property="empId" value="101" />
<jsp:setProperty name="emp" property="empName" value="John" />
<jsp:setProperty name="emp" property="salary" value="55000" />
<h2>Employee Details</h2>
<!-- Retrieve and Display Properties -->
Employee ID: <jsp:getProperty name="emp" property="empId" /><br>
Employee Name: <jsp:getProperty name="emp" property="empName" /><br>
Salary: <jsp:getProperty name="emp" property="salary" />
```

To gain better insight on each topic of the study, an assignment is provided to us. It is evaluated in a convincing manner by letting us to clarify all our doubts and achieve due momentum on this subject.

Outcome of the Subject: This paper benefits us a lot in every aspect. Moreover, it makes us inclined to understand what our career demands. It is better to acknowledge that this paper abridges the gap between the academic world and professional life. In a nutshell, it reinstates a balance in our life by bringing equilibrium to succeed in the long run professionally.