Q.1) Write a Program to make use of following JSP implicit objects:
i. out: To display current Date and
Time. ii. request: To get header
information.
iii. response: To Add Cookie iv. config: get the
parameters value defined in <init-param>
v. application: get the parameter value defined in <context-
param> vi. session: Display Current Session ID vii.
pageContext: To set and get the attributes.
viii. page: get the name of Generated Servlet

→<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<%@ page import="java.util.Date" %>
<%@ page import="javax.servlet.http.Cookie" %>

```
<html>
<head>
  <title>JSP Implicit Objects Demo</title>
</head>
<body>
  <h1>JSP Implicit Objects Example</h1>

  <!-- i. out: Display current Date and Time -->
  <h2>1. Current Date and Time (out):</h2>
  <%
    Date date = new Date();
    out.println("Current Date and Time: " + date.toString());
  %>

  <!-- ii. request: Get header information -->
  <h2>2. Request Header (request):</h2>
  <%
    String browser = request.getHeader("User-Agent");
    out.println("User-Agent: " + browser);
  %>

  <!-- iii. response: Add Cookie -->
  <h2>3. Add Cookie (response):</h2>
  <%
    Cookie ck = new Cookie("user", "jsp_user");
```

```
        response.addCookie(ck);
        out.println("Cookie 'user=jsp_user' has been added.");
    %>

    <!-- iv. config: Get <init-param> value -->
    <h2>4. Init Parameter Value (config):</h2>
    <%
        String initVal = config.getInitParameter("myInitParam");
        out.println("Init Parameter 'myInitParam': " + initVal);
    %>

    <!-- v. application: Get <context-param> value -->
    <h2>5. Context Parameter Value (application):</h2>
    <%
        String contextVal = application.getInitParameter("myContextParam");
        out.println("Context Parameter 'myContextParam': " + contextVal);
    %>

    <!-- vi. session: Display Current Session ID -->
    <h2>6. Session ID (session):</h2>
    <%
        out.println("Current Session ID: " + session.getId());
    %>

    <!-- vii. pageContext: Set and Get Attributes -->
    <h2>7. PageContext Attributes (pageContext):</h2>
    <%
        pageContext.setAttribute("message", "Hello from PageContext!");
        String message = (String) pageContext.getAttribute("message");
        out.println("Attribute 'message': " + message);
    %>

    <!-- viii. page: Get class name of generated servlet -->
    <h2>8. Page Object Info (page):</h2>
    <%
        out.println("Class of 'page' object: " + page.getClass().getName());
    %>

</body>
</html>
```

```
<web-app xmlns="http://java.sun.com/xml/ns/javaee" version="3.0">

    <!-- Context Parameter -->
    <context-param>
        <param-name>myContextParam</param-name>
        <param-value>ThisIsContextParam</param-value>
    </context-param>

    <!-- Servlet for JSP with init-param -->
    <servlet>
        <servlet-name>ImplicitObjectsDemo</servlet-name>
        <jsp-file>/ImplicitObjectsDemo.jsp</jsp-file>
        <init-param>
            <param-name>myInitParam</param-name>
            <param-value>ThisIsInitParam</param-value>
        </init-param>
    </servlet>

    <servlet-mapping>
        <servlet-name>ImplicitObjectsDemo</servlet-name>
        <url-pattern>/ImplicitObjectsDemo.jsp</url-pattern>
    </servlet-mapping>

</web-app>
```

Output

1. Current Date and Time (out):

Current Date and Time: Sat Apr 20 12:00:00 IST 2025

2. Request Header (request):

User-Agent: Mozilla/5.0 ...

3. Add Cookie (response):

Cookie 'user=jsp_user' has been added.

4. Init Parameter Value (config):

Init Parameter 'myInitParam': ThisIsInitParam

5. Context Parameter Value (application):

Context Parameter 'myContextParam': ThisIsContextParam

6. Session ID (session):
Current Session ID: ABC123XYZ

7. PageContext Attributes (pageContext):
Attribute 'message': Hello from PageContext!

8. Page Object Info (page):
Class of 'page' object: org.apache.jsp.ImplicitObjectsDemo_jsp

Q2 Write a JDBC program to update number_of_students of “BCA Science” to 1000.Create a table Course (Code,name, department,number_of_students). Insert values in the table.

→Course (
  code VARCHAR(10),
  name VARCHAR(50),
  department VARCHAR(50),
  number_of_students INT
)

```
import java.sql.*;

public class UpdateCourseJDBC {
   public static void main(String[] args) {
      // JDBC URL, username and password of MySQL server
      String jdbcURL = "jdbc:mysql://localhost:3306/your_database_name"; // Replace with your DB
      String dbUser = "root"; // Replace with your DB username
      String dbPassword = "password"; // Replace with your DB password

      try {
         // Load MySQL JDBC Driver
         Class.forName("com.mysql.cj.jdbc.Driver");

         // Connect to the database
         Connection conn = DriverManager.getConnection(jdbcURL, dbUser, dbPassword);
         Statement stmt = conn.createStatement();

         // Create table if not exists
         String createTable = "CREATE TABLE IF NOT EXISTS Course (" +
               "code VARCHAR(10), " +
               "name VARCHAR(50), " +
               "department VARCHAR(50), " +
```

```
            "number_of_students INT)";
        stmt.executeUpdate(createTable);

        // Insert sample data
        String insertData = "INSERT INTO Course (code, name, department,
number_of_students) VALUES " +
            "('C101', 'BCA Science', 'Computer Science', 500), " +
            "('C102', 'MCA', 'Computer Applications', 300), " +
            "('C103', 'BBA', 'Business Administration', 200)";
        stmt.executeUpdate("DELETE FROM Course"); // clear previous entries
        stmt.executeUpdate(insertData);

        // Update number_of_students for 'BCA Science'
        String updateQuery = "UPDATE Course SET number_of_students = 1000 WHERE
name = 'BCA Science'";
        int rowsAffected = stmt.executeUpdate(updateQuery);
        System.out.println("Rows Updated: " + rowsAffected);

        // Show all courses after update
        ResultSet rs = stmt.executeQuery("SELECT * FROM Course");
        System.out.println("\nCourse Table After Update:");
        System.out.printf("%-10s %-20s %-25s %-10s%n", "Code", "Name", "Department",
"Students");
        System.out.println("-----------------------------------------------------------");
        while (rs.next()) {
           System.out.printf("%-10s %-20s %-25s %-10d%n",
                rs.getString("code"),
                rs.getString("name"),
                rs.getString("department"),
                rs.getInt("number_of_students"));
        }

        // Close resources
        rs.close();
        stmt.close();
        conn.close();

    } catch (Exception e) {
        e.printStackTrace();
    }
```

```
    }
}
```