

## Worksheet No.: 2

**Student Name:** Harsh

**UID:** 24MCA20045

**Branch:** MCA (General)

**Section/Group:** 1-A

**Semester:** 2nd

**Date of Performance:**

25/01/2025

**Subject Name:** Advanced internet programming lab **Subject Code:** 24CAP-652

### 1. Aim of the practical:

Create a servlet that describes how to use the HttpSession object to find out the creation time and the last-accessed time for a session. We would associate a new session with the request if one does not already exist.

### 2. Apparatus

Software requirements (Jdk, NetBeans, Tomcat)

### 3. Coding

- **HTML part**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Session Tracker</title>
</head>
```

<body>

<h1>Check Session</h1>

<form action="checksession" method="post">  
 <label for="username">Username:</label>  
 <input type="text" name="username" required>

<label for="id">User ID:</label>  
 <input type="text" name="id" required>

<button type="submit">Submit</button>  
</form>

<hr>

<h3>If you want to destroy the session:</h3>  
<form action="SessionTracker" method="post">  
 <input type="hidden" name="destroySession" value="true"/>  
 <button type="submit">Destroy Session</button>  
</form>

<div class="session-info">  
 <!-- The session info will be displayed here based on the servlet response -->  
</div>

</body>

</html>

- **Servlet part:**

```
import java.io.*;  
import java.text.SimpleDateFormat;  
import java.util.Date;  
import jakarta.servlet.ServletException;  
import jakarta.servlet.annotation.WebServlet;  
import jakarta.servlet.http.HttpServlet;  
import jakarta.servlet.http.HttpServletRequest;  
import jakarta.servlet.http.HttpServletResponse;
```

```
import jakarta.servlet.http.HttpSession;

@WebServlet("/checksession")
public class checksession extends HttpServlet {

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
    ServletException, IOException {
        // Retrieve form data
        String username = request.getParameter("username");
        String id = request.getParameter("id");
        String destroySession = request.getParameter("destroySession");

        // Get or create a session
        HttpSession session = request.getSession(false); // Do not create a session if it doesn't exist
        SimpleDateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
        long expiryTimeInMillis = 30 * 1000; // 30 seconds
        String message;

        if ("true".equalsIgnoreCase(destroySession) && session != null) {
            // Destroy the current session if requested
            session.invalidate();
            message = "Session destroyed! Please log in again.";
            session = null;
        } else if (session == null) {
            // Create a new session
            session = request.getSession();
            String loginTime = dateFormat.format(new Date());
            session.setAttribute("loginTime", loginTime);
            session.setAttribute("expiryTime", System.currentTimeMillis() + expiryTimeInMillis);
            session.setAttribute("username", username);
            session.setAttribute("id", id);
            message = "Welcome! You are logged in.";
        } else {
            // Existing session: check expiry
            long currentTime = System.currentTimeMillis();
            long expiryTime = (long) session.getAttribute("expiryTime");
            if (currentTime > expiryTime) {
                // Session expired: invalidate and create a new session
            }
        }
    }
}
```

```
session.invalidate();
session = request.getSession();
String loginTime = dateFormat.format(new Date());
session.setAttribute("loginTime", loginTime);
session.setAttribute("expiryTime", currentTime + expiryTimeInMillis);
session.setAttribute("username", username);
session.setAttribute("id", id);
message = "Session expired! Starting a new session. Welcome!";
} else {
    // Session is still active
    message = "Welcome Back!";
}
}

// Retrieve session details
String sessionId = (session != null) ? session.getId() : "No Session";
String sessionCreationTime = (session != null)
    ? dateFormat.format(new Date(session.getCreationTime()))
    : "N/A";
String lastAccessTime = (session != null)
    ? dateFormat.format(new Date(session.getLastAccessedTime()))
    : "N/A";
String loginTime = (session != null) ? (String) session.getAttribute("loginTime") : "N/A";
long remainingTime = (session != null)
    ? ((long) session.getAttribute("expiryTime") - System.currentTimeMillis()) / 1000
    : 0;

// Set session timeout to 30 seconds for inactivity
if (session != null) {
    session.setMaxInactiveInterval(30);
}

// Set response content type
response.setContentType("text/html");
PrintWriter out = response.getWriter();

// Display session and user data
out.println("<html><body>");
out.println("<h1>" + message + "</h1>");
```

```

if (session != null) {
    out.println("<p><b>Username:</b> " + session.getAttribute("username") + "</p>");
    out.println("<p><b>ID:</b> " + session.getAttribute("id") + "</p>");
    out.println("<hr>");
    out.println("<h2>Session Information</h2>");
    out.println("<p><b>Session ID:</b> " + sessionId + "</p>");
    out.println("<p><b>Session Creation Time:</b> " + sessionCreationTime + "</p>");
    out.println("<p><b>Last Accessed Time:</b> " + lastAccessTime + "</p>");
    out.println("<p><b>Login Time:</b> " + loginTime + "</p>");
    out.println("<p><b>Remaining Time (seconds):</b> " + remainingTime + "</p>");
    out.println("<p><b>Session Timeout (seconds):</b> 30</p>");
    out.println("<hr>");
    out.println("<form action='SessionTracker' method='post'>");
    out.println("<input type='hidden' name='destroySession' value='true'/>");
    out.println("<button type='submit'>Destroy Session</button>");
    out.println("</form>");
} else {
    out.println("<p><b>Note:</b> No active session. Please log in again.</p>");
}
out.println("</body></html>");
}
}

```

#### 4. Result/Output/Writing Summary:

### Check Session

Username:

User ID:

If you want to destroy the session:

## Welcome! You are logged in.

Username: Harsh

ID: 24MCA20045

### Session Information

Session ID: F17062FC9D23E41093DBAA973E640ADC

Session Creation Time: 2025-01-26 23:17:07

Last Accessed Time: 2025-01-26 23:17:07

Login Time: 2025-01-26 23:17:07

Remaining Time (seconds): 30

Session Timeout (seconds): 30

## Welcome Back!

Username: Harsh

ID: 24MCA20045

### Session Information

Session ID: F17062FC9D23E41093DBAA973E640ADC

Session Creation Time: 2025-01-26 23:17:07

Last Accessed Time: 2025-01-26 23:17:07

Login Time: 2025-01-26 23:17:07

Remaining Time (seconds): 17

Session Timeout (seconds): 30

## Session expired! Starting a new session. Welcome!

Username: Harsh

ID: 24MCA20045

### Session Information

Session ID: BCB555394605922D2088B9ADB930B12F

Session Creation Time: 2025-01-26 23:17:38

Last Accessed Time: 2025-01-26 23:17:38

Login Time: 2025-01-26 23:17:38

Remaining Time (seconds): 30

Session Timeout (seconds): 30

**5. Learning outcomes (What I have learnt):**

- Understand how to manage user sessions using HttpSession.
- Learn to handle session expiry and create new sessions.
- Gain skills to process user requests like session destruction.
- Understand dynamic HTML generation with Print Writer.
- Learn to configure session timeout for better control.

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.	Worksheet		8 Marks
2.	Viva		10 Marks
3.	Simulation		12 Marks
	Total		30 Marks

**Teacher Signature** |