Session Tracking

Servlet

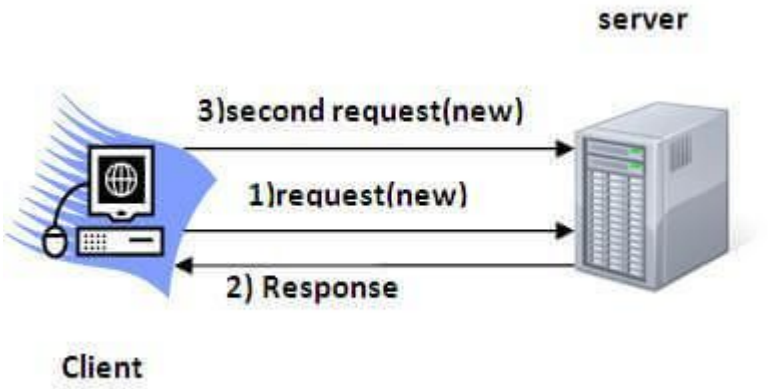
What is session and session tracking?

Session simply means a particular interval of time.

Session tracking in Servlets is a mechanism that allows a web application to maintain state and associate data with a specific user across multiple HTTP requests. Since HTTP is a stateless protocol, the server doesn't inherently maintain any information about previous interactions with a user. Session tracking helps overcome this limitation by providing a way to recognize and remember individual users as they navigate through a website or web application.

Types of session tracking

1. Cookies
2. Hidden Form Field
3. URL Rewriting
4. HttpSession



What are cookies?

Cookies are small pieces of data that a web server sends to a user's web browser, and the browser stores them locally. They are commonly used in web development for various purposes.

Storage of Information:

Cookies are used to store information that can be retrieved later when the user visits the same website.

Real-life example: When you log in to a website and select the "Remember Me" option, a cookie is created to store your login credentials. The next time you visit the site, you are automatically logged in without entering your username and password again.

HTTP Headers:

Cookies are sent as HTTP headers between the web server and the browser.

Real-life example: When you visit an online store and add items to your cart, the website's server sends a response with a Set-Cookie header to instruct your browser to store a cookie containing your cart information.

Tracking User Behavior:

Cookies are frequently used by advertisers and analytics tools to track user behavior on websites.

Real-life example: When you browse online stores, you may notice that you see ads for products you recently viewed. This is possible because the website placed a cookie in your browser to track your browsing history and display relevant ads.

Privacy Concerns:

Cookies can raise privacy concerns because they can potentially be used to track users across different websites.

Real-life example: Some users may be uncomfortable with the idea that their online behavior is being tracked by advertising companies using cookies. This has led to the development of browser features like "Do Not Track" and increased awareness of online privacy.

Adding cookies in servlet

```
Cookie cookie = new Cookie("username", "john_doe");

cookie.setMaxAge(3600); // Cookie will expire in 1 hour (in seconds)

response.addCookie(cookie);

response.setContentType("text/html");
response.getWriter().println("Cookie added successfully");
```

Deleting cookies from servlet

```
Cookie cookieToDelete = new Cookie("username", "");

cookieToDelete.setMaxAge(0);

// Add the cookie to the response to instruct the client's browser to delete it

response.addCookie(cookieToDelete);

response.setContentType("text/html");

response.getWriter().println("Cookie deleted successfully");
```

Getting cookies values

```
Cookie[] cookies = request.getCookies();
      if (cookies != null) {
         for (Cookie cookie : cookies) {
            String cookieName = cookie.getName();
            String cookieValue = cookie.getValue();
               response.setContentType("text/html");
               response.getWriter().println("Hello, " + cookieValue);
            }
         }
      } else {
         response.setContentType("text/html");
          response.getWriter().println("No cookies found");
       }
```

Task : Create a login logout system using cookies.