|  |  |
| --- | --- |
| 10down voteaccepted | Cookie  A cookie is just a key-value pair that is stored in the user's browser. A cookie is sent to your browser as part of the HTTP response that contains the web page you requested.  When your browser receives a cookie, it stores it, and sends it back to the server with every subsequent request it makes on the same website.  Because cookies are part of the HTTP request and response headers, they are somewhat limited in size.  Typical information stored in cookies:   * Session IDs (see below) * Tracking IDs (Google Analytics, etc.) * User preferences (preferred language or currency, etc.)   For larger, or sensitive data, you typically store values in the session. The cookie is only there to identify the proper session.  A cookie can be configured to only live until the browser window is closed, or have a configurable lifetime (1 week, 1 month, 1 year, whatever). If you visit the website again during this period, your browser will send the cookie with every request.  Session  A session is a set of data that is stored on the server, usually as key-value pairs. A session is assigned a pseudo-random, secret ID that is usually stored in the user's browser using a cookie, for example SESSID=abcdef123456789. The session ID typically matches the name of a file containing the session data on the server.  Sessions are usually short-lived, and automatically deleted if unused for some time (20 minutes or so).  Typical information stored in a session:   * ID of the user currently logged in * Shopping cart * ... anything you can think of, that can be safely deleted when the session expires   Example  Let's say I visit a website for the first time. The website detects that I didn't send a session cookie, so it creates a session for me. It creates a session file on the server, such as /tmp/sess\_abcdef123456789.  Then it sends a cookie header with the HTTP response that contains the web page:  HTTP/1.1 200 OK  Set-Cookie: SESSID=abcdef123456789  My browser stores this cookie. If I visit another page on the same server, my browser will send this cookie with the request:  GET /cart HTTP/1.1  Cookie: SESSID=abcdef123456789  When receiving the second request, the server can check if there's a session file with this ID, and use it to retrieve the session data.  Your web programming language will offer support for sessions, and should handle most of this complexity for you. You can usually directly use the session array/object, which will be already populated with the session data *specific to the user visiting your website*, and will be automatically saved if you update the session data; this should be totally transparent to you.  Security  When logging in a user to your website, always store the user ID in the session. **Never trust a user ID stored in a cookie to load user data.**  It's very easy to forge a cookie. If you were to load user information based on a user ID stored in a cookie, it would be easy to change the user ID in this cookie to gain access to any user's account on your website.  On the other hand, if you store the user ID in the session, which is assigned a pseudo-random session ID, it will be hard for an attacker to guess the session ID that is currently assigned to the user. |