**wat is Cookie?**  
Cookie is small information stored in text file on user’s hard drive by web server. This information is later used by web browser to retrieve information from that machine. Generally cookie contains personalized user data or information that is used to communicate between different web pages.

**Why Cookies are used?**  
Cookies are nothing but the user’s identity and used to track where the user navigated throughout the web site pages. The communication between web browser and web server is stateless.

For example if you are accessing domain http://www.example.com/1.html then web browser will simply query to example.com web server for the page 1.html. Next time if you type page as http://www.example.com/2.html then new request is send to example.com web server for sending 2.html page and web server don’t know anything about to whom the previous page 1.html served.

What if you want the previous history of this user communication with the web server? You need to maintain the user state and interaction between web browser and web server somewhere. This is where cookie comes into picture. Cookies serve the purpose of maintaining the user interactions with web server.

**How cookies work?**  
The HTTP protocol used to exchange information files on the web is used to maintain the cookies. There are two types of HTTP protocol. Stateless HTTP and Stateful HTTP protocol. Stateless HTTP protocol does not keep any record of previously accessed web page history. While Stateful HTTP protocol do keep some history of previous web browser and web server interactions and this protocol is used by cookies to maintain the user interactions.

Whenever user visits the site or page that is using cookie, small code inside that HTML page (Generally a call to some language script to write the cookie like cookies in JAVAScript, PHP, Perl) writes a text file on users machine called cookie.  
Here is one example of the code that is used to write cookie and can be placed inside any HTML page:

Set-Cookie: NAME=VALUE; expires=DATE; path=PATH; domain=DOMAIN\_NAME;

When user visits the same page or domain later time this cookie is read from disk and used to identify the second visit of the same user on that domain. Expiration time is set while writing the cookie. This time is decided by the application that is going to use the cookie.

Generally two types of cookies are written on user machine.

**1) Session cookies:** This cookie is active till the browser that invoked the cookie is open. When we close the browser this session cookie gets deleted. Some time session of say 20 minutes can be set to expire the cookie.  
**2) Persistent cookies:** The cookies that are written permanently on user machine and lasts for months or years.

**Where cookies are stored?**  
When any web page application writes cookie it get saved in a text file on user hard disk drive. The path where the cookies get stored depends on the browser. Different browsers store cookie in different paths. E.g. Internet explorer store cookies on path **“C:\Documents and Settings\Default User\Cookies”**  
Here the “Default User” can be replaced by the current user you logged in as. Like “Administrator”, or user name like “Vijay” etc.  
The cookie path can be easily found by navigating through the browser options. In Mozilla Firefox browser you can even see the cookies in browser options itself. Open the Mozila browser, click on Tools->Options->Privacy and then “Show cookies” button.

**How cookies are stored?**  
Lets take example of cookie written by rediff.com on Mozilla Firefox browser:  
On Mozilla Firefox browser when you open the page rediff.com or login to your rediffmail account, a cookie will get written on your Hard disk. To view this cookie simply click on “Show cookies” button mentioned on above path. Click on Rediff.com site under this cookie list. You can see different cookies written by rediff domain with different names.

**Site:** Rediff.com Cookie name: RMID  
**Name:**RMID (Name of the cookie)  
**Content:** 1d11c8ec44bf49e0… (Encrypted content)  
**Domain:** .rediff.com  
**Path:** / (Any path after the domain name)  
**Send For:** Any type of connection  
**Expires:** Thursday, December 31, 2020 11:59:59 PM

**Applications where cookies can be used:**

**1) To implement shopping cart:**  
Cookies are used for maintaining online ordering system. Cookies remember what user wants to buy. What if user adds some products in their shopping cart and if due to some reason user don’t want to buy those products this time and closes the browser window? When next time same user visits the purchase page he can see all the products he added in shopping cart in his last visit.

**2) Personalized sites:**  
When user visits certain pages they are asked which pages they don’t want to visit or display. User options are get stored in cookie and till the user is online, those pages are not shown to him.

**3) User tracking:**  
To track number of unique visitors online at particular time.

**4) Marketing:**  
Some companies use cookies to display advertisements on user machines. Cookies control these advertisements. When and which advertisement should be shown? What is the interest of the user? Which keywords he searches on the site? All these things can be maintained using cookies.

**5) User sessions:**  
Cookies can track user sessions to particular domain using user ID and password.

**Drawbacks of cookies:**

**1)**Even writing Cookie is a great way to maintain user interaction, if user has set browser options to warn before writing any cookie or disabled the cookies completely then site containing cookie will be completely disabled and can not perform any operation resulting in loss of site traffic.

**2) Too many Cookies:**  
If you are writing too many cookies on every page navigation and if user has turned on option to warn before writing cookie, this could turn away user from your site.

**3) Security issues:**  
Some times users personal information is stored in cookies and if someone hack the cookie then hacker can get access to your personal information. Even corrupted cookies can be read by different domains and lead to security issues.

**4) Sensitive information:**  
Some sites may write and store your sensitive information in cookies, which should not be allowed due to privacy concerns.

This should be enough to know what cookies are. If you want more cookie info see [Cookie Central page.](http://www.cookiecentral.com/c_concept.htm)

**Some Major Test cases for web application cookie testing:**

The first obvious test case is to test if your application is writing cookies properly on disk. You can use the [Cookie Tester application](http://www.html-kit.com/tools/cookietester/)also if you don’t have any web application to test but you want to understand the cookie concept for testing.

**Test cases:**

**1)** As a Cookie **privacy policy** make sure from your design documents that no personal or sensitive data is stored in the cookie.

**2)**If you have no option than saving **sensitive data** in cookie make sure data stored in cookie is stored in encrypted format.

**3)** Make sure that there is **no overuse of cookies** on your site under test. Overuse of cookies will annoy users if browser is prompting for cookies more often and this could result in loss of site traffic and eventually loss of business.

**4) Disable the cookies** from your browser settings: If you are using cookies on your site, your sites major functionality will not work by disabling the cookies. Then try to access the web site under test. Navigate through the site. See if appropriate messages are displayed to user like “For smooth functioning of this site make sure that cookies are enabled on your browser”. There should not be any page crash due to disabling the cookies. (Please make sure that you close all browsers, delete all previously written cookies before performing this test)

**5) Accepts/Reject some cookies:**The best way to check web site functionality is, not to accept all cookies. If you are writing 10 cookies in your web application then randomly accept some cookies say accept 5 and reject 5 cookies. For executing this test case you can set browser options to prompt whenever cookie is being written to disk. On this prompt window you can either accept or reject cookie. Try to access major functionality of web site. See if pages are getting crashed or data is getting corrupted.

**6) Delete cookie:**Allow site to write the cookies and then close all browsers and manually delete all cookies for web site under test. Access the web pages and check the behavior of the pages.

**7) Corrupt the cookies:** Corrupting cookie is easy. You know where cookies are stored. Manually edit the cookie in notepad and change the parameters to some vague values. Like alter the cookie content, Name of the cookie or expiry date of the cookie and see the site functionality. In some cases corrupted cookies allow to read the data inside it for any other domain. This should not happen in case of your web site cookies. Note that the cookies written by one domain say rediff.com can’t be accessed by other domain say yahoo.com unless and until the cookies are corrupted and someone trying to hack the cookie data.

**8 ) Checking the deletion of cookies from your web application page:** Some times cookie written by domain say rediff.com may be deleted by same domain but by different page under that domain. This is the general case if you are testing some ‘action tracking’ web portal. Action tracking or purchase tracking pixel is placed on the action web page and when any action or purchase occurs by user the cookie written on disk get deleted to avoid multiple action logging from same cookie. Check if reaching to your action or purchase page deletes the cookie properly and no more invalid actions or purchase get logged from same user.

**9) Cookie Testing on Multiple browsers:**This is the important case to check if your web application page is writing the cookies properly on different browsers as intended and site works properly using these cookies. You can test your web application on Major used browsers like Internet explorer (Various versions), Mozilla Firefox, Netscape, Opera etc.

**10)** If your web application is using **cookies to maintain the logging state of any user** then log in to your web application using some username and password. In many cases you can see the logged in user ID parameter directly in browser address bar. Change this parameter to different value say if previous user ID is 100 then make it 101 and press enter. The proper access message should be displayed to user and user should not be able to see other users account.

These are some Major test cases to be considered while testing website cookies. You can write multiple test cases from these test cases by performing various combinations. If you have some different application scenario, you can mention your test cases in comments below.