Difference between Cookie and Session

ⓘ ×

What is Cookie?

A cookie is a small file with the maximum size of 4KB that the web server stores on the client computer. Once a cookie has been set, all page requests that follow return the cookie name and value. A cookie can only be read from the domain that it has been issued from. For example, a cookie set using the domain www.guru99.com cannot be read from the domain career.guru99.com. Most of the websites on the internet display elements from other domains such as advertising. The domains serving these elements can also set their own cookies. These are known as third party cookies. A cookie created by a user can only be visible to them. Other users cannot see its value. Most web browsers have options for disabling cookies, third party cookies or both.

What is a Session?

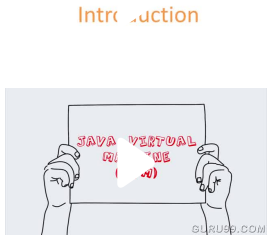
A session is a global variable stored on the server. Each session is assigned a unique id which is used to retrieve stored values. Whenever a session is created, a cookie containing the unique session id is stored on the user's computer and returned with every request to the server. If the client browser does not support cookies, the unique session id is displayed in the URL. Sessions have the capacity to store relatively large data compared to cookies.

The session values are automatically deleted when the browser is closed. If you want to store the values permanently, then you should store them in the database.

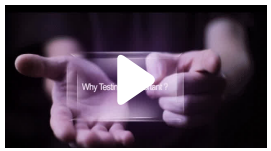
FEATURED VIDEOS

Guru99

Linux Tutorials



NOW
PLAYING



Just like the $_COOKIE array variable, session variables are stored in the $_SESSION array variable. Just like cookies, the session must be started before any HTML tags.

KEY DIFFERENCE

- Cookies are client-side files that contain user information, whereas Sessions are server-side files that contain user information.
- Cookie is not dependent on session, but Session is dependent on Cookie.
- Cookie expires depending on the lifetime you set for it, while a Session ends when a user closes his/her browser.
- The maximum cookie size is 4KB whereas in session, you can store as much data as you like.
- Cookie does not have a function named unsetcookie() while in Session you can use Session_destroy(); which is used to destroy all registered data or to unset some

Why and when to use Cookies?

Http is a stateless protocol; cookies allow us to track the state of the application using small files stored on the user's computer. The path were the cookies are stored depends on the browser. Internet Explorer usually stores them in Temporal Internet Files folder. Personalizing the user experience – this is achieved by allowing users to select their preferences. The page requested that follow are personalized based on the set preferences in the cookies. Tracking the pages visited by a user.

Why and when to use Sessions?

To store important information such as the user id more securely on the server where malicious users cannot temper with them. Sessions are used to pass values from one page to another.

It is also used when you want the alternative to cookies on browsers that do not support cookies, to store global variables in an efficient and more secure way compared to passing them in the URL, developing an application such as a shopping cart that has to temporary store information with a capacity larger than 4KB.

Cookie Vs. Session

Cookie	Session
• Cookies are client-side files that contain user information	• Sessions are server-side files which contain user information
• Cookie ends depending on the lifetime you set for it	• A session ends when a user closes his browser

• You don't need to start cookie as it is stored in your local machine	• In PHP, before using $_SESSION, you have to write session_start(); Likewise for other languages
• The official maximum cookie size is 4KB	• Within-session you can store as much data as you like. The only limits you can reach is the maximum memory a script can consume at one time, which is 128MB by default
• A cookie is not dependent on Session	• A session is dependent on Cookie
• There is no function named unsetcookie()	• Session_destroy(); is used to destroy all registered data or to unset some

Prev (/codeigniter-sending-email.html)

Report a Bug

Next (/laravel-vs-codeigniter.html)

YOU MIGHT LIKE:

PHP

(/object-oriented-programming.html)

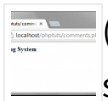
(/object-oriented-programming.html)

PHP

(/php-and-xml.html) (/php-and-xml.html)

PHP XML Tutorial: Create, Parse, Read with Example

PHP



(/php-security.html) (/php-security.html)

PHP Security Function: strip_tags, filter_var, Md5 and sha1

PHP Object Oriented Programming (OOPs) concept Tutorial with Example

(/object-oriented-programming.html)

(/php-and-xml.html)

(/php-security.html)

PHP

(/php-date-functions.html)

(/php-date-functions.html)

PHP Date() & Time Function: How to Get Current Timestamp?

(/php-date-functions.html)

PHP

(/laravel-tutorial.html)

(/laravel-tutorial.html)

Laravel Tutorial for Beginners

(/laravel-tutorial.html)

PHP

(/functions-in-php.html)

(/functions-in-php.html)

PHP Function: How to Define? Built in | String | User Defined

(/functions-in-php.html)

CodeIgniter Tutorial

12) File Upload (/codeigniter-file-upload.html)

13) Send Email using CodeIgniter (/codeigniter-sending-email.html)

14) Cookie Vs Session (/difference-between-cookie-session.html)

15) Laravel vs CodeIgniter (/laravel-vs-codeigniter.html)

16) CodeIgniter Tutorial PDF (/codeigniter-tutorial-pdf.html)

(https://www.facebook.com/guru99com/)
(https://twitter.com/guru99com)
(https://www.linkedin.com/company/guru99/)
(https://www.youtube.com/channel/UC19i1XD6k88KqHlET8atqFQ)

(https://forms.aweber.com/form/46/724807646.htm)

About

About Us (/about-us.html)
Advertise with Us (/advertise-us.html)
Write For Us (/become-an-instructor.html)
Contact Us (/contact-us.html)

Career Suggestion

SAP Career Suggestion Tool (/best-sap-module.html)
Software Testing as a Career (/software-testing-career-complete-guide.html)

Interesting

eBook (/ebook-pdf.html)
Blog (/blog/)
Quiz (/tests.html)
SAP eBook (/sap-ebook-pdf.html)

Execute online

Execute Java Online (/try-java-editor.html)
Execute Javascript (/execute-javascript-online.html)
Execute HTML (/execute-html-online.html)
Execute Python (/execute-python-online.html)

© Copyright - Guru99 2021