



.(/).

(/)

(/LassoDocs/)

[View Cart \(/user-cart/\)](#)

Login

(/LassoTalk/)

LassoDocs >

All the technical info
to get you coding quickly

LassoSoft >

LassoTalk >

Search

[Home \(/LassoDocs/\)](#)

[Reference \(/LassoDocs/LanguageReferenceCategories\)](#)

[Lasso Guide \(http://lassoguide.com/index.html\)](http://lassoguide.com/index.html)

[TagSwap \(/TagSwap\)](#)

[Migration \(/Migration-Guide\)](#)

[Articles \(/Available-Articles-List\)](#)

[Tutorials \(/Lasso-Tutorials\)](#)

[Contact \(/Contact-LassoSoft\)](#)

Tutorials

Understanding Cookies and Sessions

Cookies and Sessions are inextricably intertwined, and fundamental to creating rich, interactive internet applications.

What we will be covering in this tutorial:

- What is a "cookie"
- What is a "session"
- Setting and reading cookies
- Using sessions
- Troubleshooting
- Further reading

What is a "Cookie"?

A cookie is a small piece of text stored on a user's computer by their browser. Common uses for cookies are authentication, storing of site preferences, shopping cart items, and server session identification.

Each time the users' web browser interacts with a web server it will pass the cookie information to the web server. Only the cookies stored by the browser that relate to the domain in the requested URL will be sent to the server. This means that cookies that relate to www.example.com will not be sent to www.exampledomain.com.

In essence, a cookie is a great way of linking one page to the next for a user's interaction with a web site or web application.

What is a "Session"?

Table of Contents

Understanding Cookies and Sessions

[What is a "Cookie"?](#)

[What is a "Session"?](#)

[Setting and reading cookies](#)

[Using Sessions](#)

[Troubleshooting](#)

[Further Reading](#)

[Cookies](#)

[Sessions](#)

[Types and methods used in this tutorial](#)

Categories

[Lasso 9](#)

[Security](#)

[Sessions](#)

Random User Tips

Email_Immediate

To send an email immediately and
bypass the queue, use

[email_immediate](#)

[\(/LassoDocs/languageReference/obj/Email_Immediate\)](/LassoDocs/languageReference/obj/Email_Immediate)

A session can be defined as a server-side storage of information that is desired to persist throughout the user's interaction with the web site or web application.

Instead of storing large and constantly changing information via cookies in the user's browser, only a unique identifier is stored on the client side (called a "session id"). This session id is passed to the web server every time the browser makes an HTTP request (ie a page link or AJAX request). The web application pairs this session id with it's internal database and retrieves the stored variables for use by the requested page.

Setting and reading cookies

Using the [cookie_set] method we can set cookies to store information for use in later pages. The following code shows how easy it is to store a user's details such as their name and email address which they may have entered on a "Contact Us" form. This would then allow later pages to pre-populate forms with this information.

```
Cookie_Set(  
  'UserDetails'='John Doe|john.doe@example.com',  
  -Domain='example.com',  
  -Expires='1440',  
  -Path='/'  
)
```

In this example the cookie named "UserDetails" contains the user name and email address delimited by a "pipe" character. This can be read and interpreted, then output in the following code.

```
local( userDetails = decode_url(cookie('UserDetails'))->split('|'))  
  
if(#userDetails->size)={  
  'User Name = '+#userDetails->get(1)  
  '<br />'  
  'Email Address = '+#userDetails->get(2)  
^}
```

Using Sessions

To store information that is not appropriate to store client-side, we use sessions. Lasso has built in session handling, and deals with the setting and retrieval of the cookie itself. It will automatically set and retrieve the session id, which is the only thing stored client-side.

To set up a new session, we first start the session, then add to it the variables we would like to store in it. Those variables are stored within Lasso's session database.

```
// Start the session.
session_start(
    'mySessionName',
    -expires = 1440,
    -usecookie=true
)

// Add variables to the session
if(session_result('mySessionName') != 'load') => {
    session_addVar('mySessionName', 'sv_userId')
    session_addVar('mySessionName', 'sv_userName')
    session_addVar('mySessionName', 'sv_userEmail')
    session_addVar('mySessionName', 'sv_favouriteColour')
}

!var_defined('sv_userId') ?    var('sv_userId' = integer)
!var_defined('sv_userName') ?  var('sv_userName' = string)
!var_defined('sv_userEmail') ? var('sv_userEmail' = string)
!var_defined('sv_favouriteColour') ? var('sv_favouriteColour' = 'red')
```

How this works:

Lines 2-6: Initializing the session. This needs to happen on every page you wish the session information to be readable and writable. Here we start a session named 'mySessionName' with an idle expiry of 1440 minutes (1 day) and set to track via cookie.

Lines 9-14: The "if" conditional here checks to see if the session_result shows a new session and adds new variables to the session if it is new.

Lines 16-19: Each line in this section is checking if a variable has been defined, and if not then the variable is defined and declared with a default type and value.

The variables added to the session can now be used just like any other variable in the page. These variables are both readable and writable. This means that when the value is changed, it will be stored in the session with the new value.

On subsequent pages if the session is initialized, you will be able to continue reading and modifying these variables.

Troubleshooting

If you're having problems with sessions, there are a few things to check.

Check the time on the server and the client PC

If the clock on the server and/or the client PC is incorrect, the session may expire prematurely. For example if your session idle expiry time is short (ie 15 minutes) and the clocks differ by greater than that amount, the session will expire before it can be used on the next page.

Ensure the client browser is accepting cookies

Browsers allow users to decide to not allow cookies to be set. If a user has blocked cookies you will have to use links to persist sessions. See the [session documentation \(/lassoDocs/Language-Guide-Sessions\)](#) for more information.

Further Reading

Cookies

- [Lasso 9 Language Reference: Cookie \(/lassoDocs/languageReference/obj/cookie\)](/lassoDocs/languageReference/obj/cookie).
- [W3C Specification \(http://www.w3.org/Protocols/rfc2109/rfc2109\)](http://www.w3.org/Protocols/rfc2109/rfc2109).
- [Wikipedia: HTTP Cookies \(http://en.wikipedia.org/wiki/HTTP_cookie\)](http://en.wikipedia.org/wiki/HTTP_cookie).

Sessions

- [Lasso 9 Language Guide: Sessions \(/lassoDocs/Language-Guide-Sessions\)](/lassoDocs/Language-Guide-Sessions).
- [Session Tracking \(http://javapapers.com/servlet/explain-the-methods-used-for-session-tracking/\)](http://javapapers.com/servlet/explain-the-methods-used-for-session-tracking/).
- [W \(http://en.wikipedia.org/wiki/Session_\(computer_science\)\)](http://en.wikipedia.org/wiki/Session_(computer_science))[ikipedia \(http://en.wikipedia.org/wiki/Session_\(computer_science\)\)](http://en.wikipedia.org/wiki/Session_(computer_science)): [Sessions \(http://en.wikipedia.org/wiki/Session_\(computer_science\)\)](http://en.wikipedia.org/wiki/Session_(computer_science)).

Types and methods used in this tutorial

More detail on the types and methods used in this tutorial can be found by visiting the following links:

- [Cookie Set \(/lassoDocs/languageReference/obj/cookie_set\)](/lassoDocs/languageReference/obj/cookie_set).
- [Cookie \(/lassoDocs/languageReference/obj/cookie\)](/lassoDocs/languageReference/obj/cookie).
- [Session Start \(/lassoDocs/languageReference/obj/session_start\)](/lassoDocs/languageReference/obj/session_start).
- [Session Result \(/lassoDocs/languageReference/obj/session_result\)](/lassoDocs/languageReference/obj/session_result).
- [Session AddVar \(/lassoDocs/languageReference/obj/session_addvar\)](/lassoDocs/languageReference/obj/session_addvar).



Please note that periodically LassoSoft will go through the notes and may incorporate information from them into the documentation. Any submission here gives LassoSoft a non-exclusive license and will be made available in various formats to the Lasso community.