

# LiveCode Lessons (copy)

Search

## Topics

[Installing LiveCode Server](#) 6[Using LiveCode Server](#) 4[Interacting with LiveCode Server](#) 6[How do I pass information to LiveCode Server scripts?](#)[How do I handle user input using LiveCode Server?](#)[How Do I Use AJAX with LiveCode Server?](#)[How to upload a file with LiveCode Server \(Current Article\)](#)[How do I use Cookies on LiveCode Server?](#)[Accessing Web Services using LiveCode](#)

## Last Updated

Apr 04, 2016

[Print Article](#)

## Other Resources

### Getting Started with LiveCode

[Get Up and Running with LiveCode](#)[Getting Started with LiveCode Development](#)

### LiveCode Lessons

[How To - Step-By-Step Guides To Tasks In LiveCode](#)[How To - LiveCode Server Tasks](#)[How To - LiveCode Mobile Tasks](#)[How To - LiveCode Sample Scripts](#)[How to - LiveCode Marketplace Products](#)[How to Purchase and License LiveCode](#)

## Tutorials

[Creating a Video Library Application](#)

## Data Grid

[LiveCode Data Grid](#)[Data Grid Tips & Tricks](#)[Converting the Stock Program](#)

## LiveCode Releases

[LiveCode 6.5](#)[LiveCode 6.7](#)[LiveCode 8](#)

## Comments

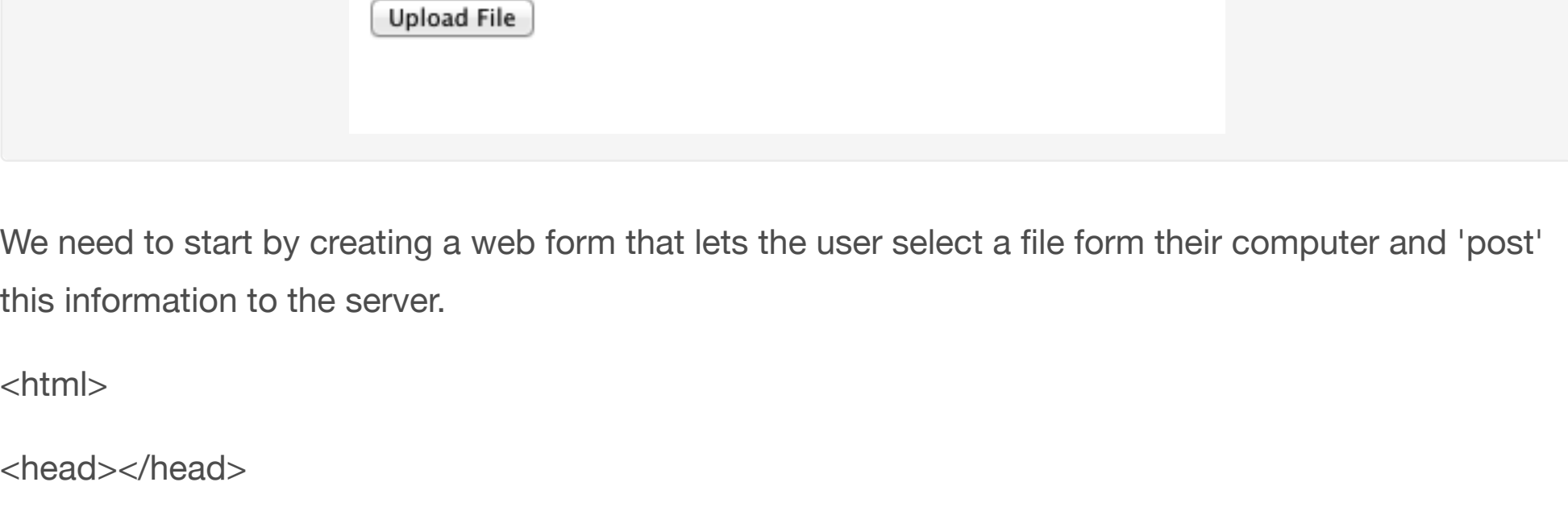
Liquid error: internal for this article

[LiveCode Lessons \(copy\)](#) / [LiveCode Lessons](#) / [How To - LiveCode Server Tasks](#) / [Interacting with LiveCode Server](#) / [How to upload a file with LiveCode Server](#)

## How to upload a file with LiveCode Server

In this example we look at how to use LiveCode server to upload a file. You'll need web server with LiveCode Server in order to follow along with this lesson. If you need hosting, [www.on-rev.com](#) is provided by RunRev Ltd and is pre configured with the latest LiveCode Server engines.

### Create the basic HTML form



We need to start by creating a web form that lets the user select a file from their computer and 'post' this information to the server.

```
<html>

<head></head>

<body>

<H1>Upload Form</H1>

<form enctype="multipart/form-data" action="file_upload_example.lc" method="POST">

Choose a file to upload: <input name="uploadedfile" type="file" /><br />

<input type="submit" value="Upload File" />

</form>

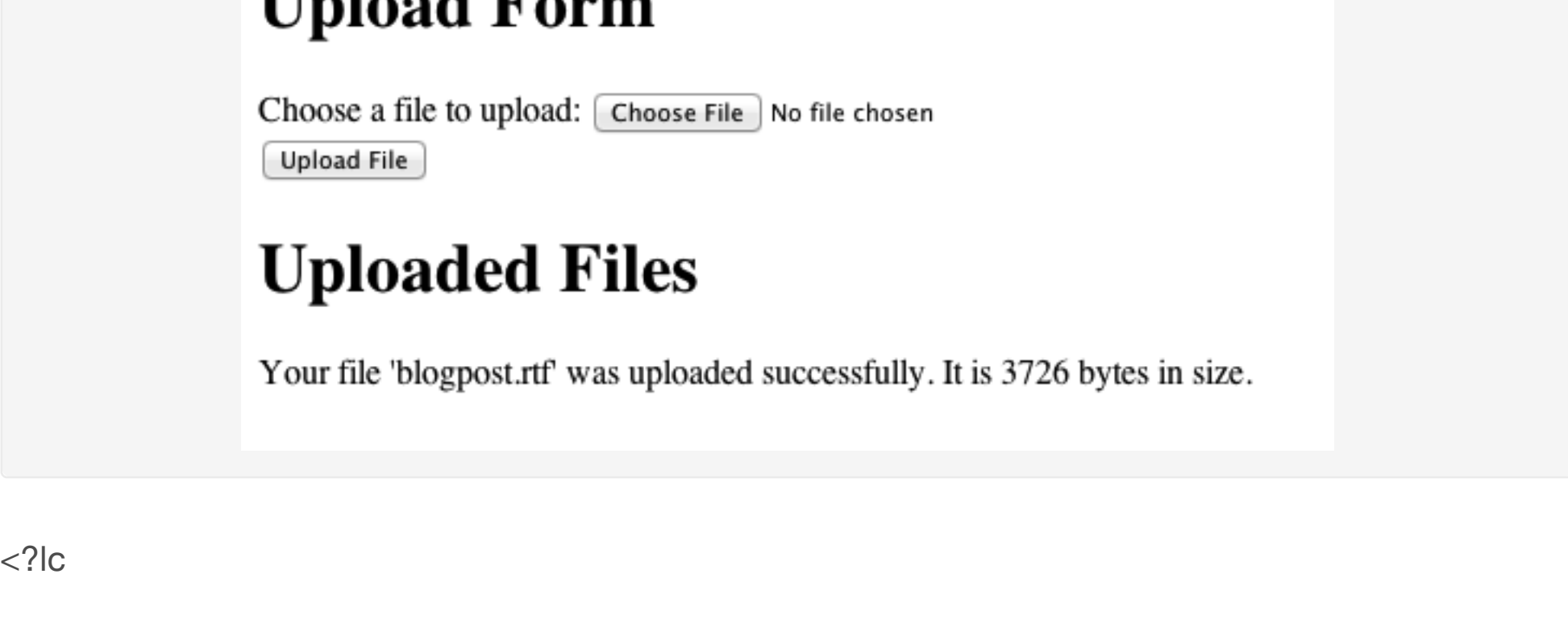
</body>

</html>
```

A few key things to point out:

- 1) **enctype = "multipart/form-data"**: This tells the browser that the form is going to pass multipart data which is what we have to set if we're going to upload files.
- 2) **action = ""**: You'll see in the example that the action is a LiveCode file. This is where the form will post its data too and where our script to handle the upload should go.
- 3) **name="uploadedfile"**: You'll notice that we've created a 'input' form element with the name 'uploadedfile'. The file data will appear in the \$\_FILES array under that key.

### Add the LiveCode Script to check for the uploaded file



```
<?lc

set the errorMode to "inline"

?>

<html>

<head></head>

<body>

<H1>Upload Form</H1>

<form enctype="multipart/form-data" action="file_upload_example.lc" method="POST">

Choose a file to upload: <input name="uploadedfile" type="file" /><br />

<input type="submit" value="Upload File" />

</form>

<H1>Uploaded Files</H1>

<p>

<?lc

if $_FILES[uploadedfile][error] then

put "There was an error uploading your file:" && $_FILES[uploadedfile][error] & "<br />"

else

put $_FILES[uploadedfile][name] into tFileName

put $_FILES[uploadedfile][type] into tFileType

put $_FILES[uploadedfile][size] into tFileSize

put $_FILES[uploadedfile][filename] into tFilePath

put "Your file '" & tFileName & "' was uploaded successfully. It is" && tFileSize && "bytes in size."

end if

?>

</p>

</body>

</html>
```

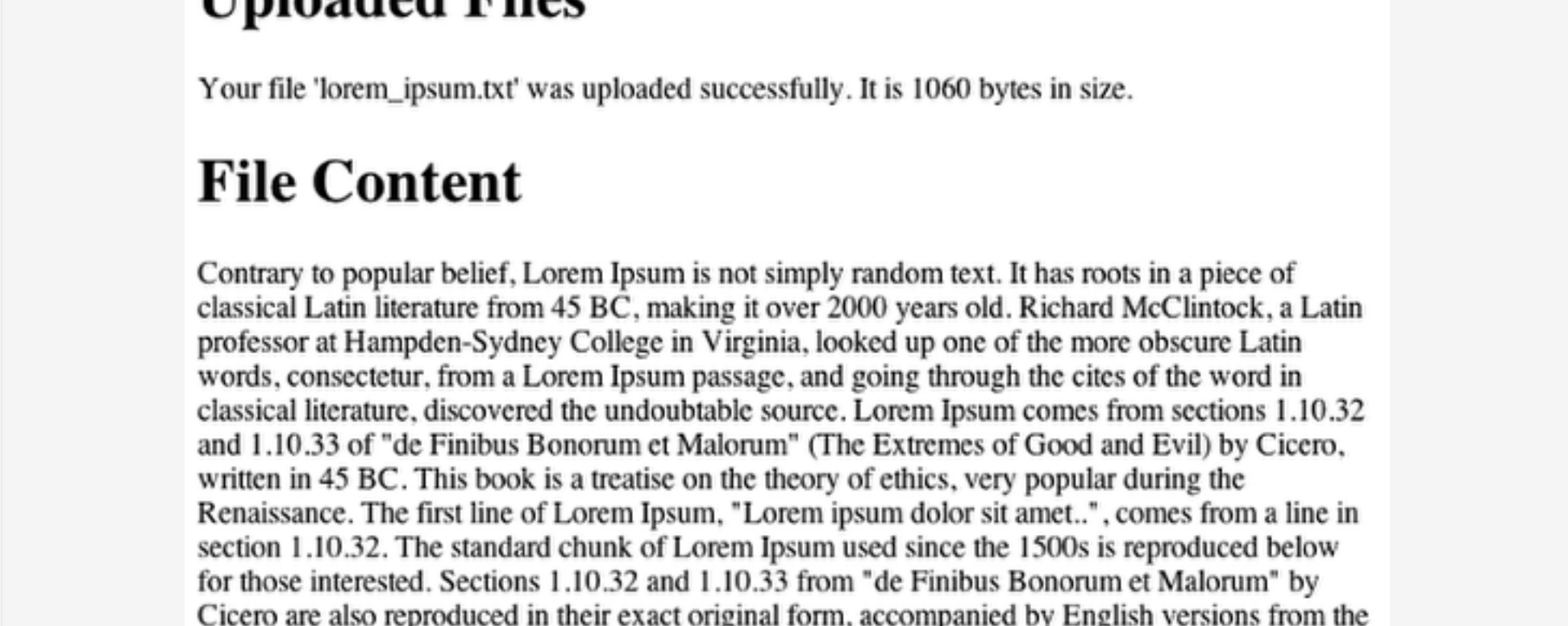
- 1) Copy the above script into a new LC file and change the action to the name of your file.
- 2) Upload it to your server.
- 3) Open a browser and navigate to your file.
- 4) Click 'choose file' and select any file from your local system.
- 5) Wait for it to upload.
- 6) Presto, it should display the name and size of the uploaded file.

You'll notice we've added a little extra HTML header 'Uploaded Files' under which we are displaying the information from our upload (\$\_FILES).

**NOTE:** Notice that all the information, including errors are stored in the \$\_FILES array against a key with the same name as our input box (uploadedfile) in the web form.

**NOTE:** You can access the name, type, size and path of the uploaded file via the \$\_FILES array.

### Do something with the file



In this step I've added code which displays the content of the file if it is found to be a 'text' file. This information is available in the \$\_FILES array.

```
<?lc

set the errorMode to "inline"

?>

<html>

<head></head>

<body>

<H1>Upload Form</H1>

<form enctype="multipart/form-data" action="file_upload_example.lc" method="POST">

Choose a file to upload: <input name="uploadedfile" type="file" /><br />

<input type="submit" value="Upload File" />

</form>

<H1>Uploaded Files</H1>

<p>

<?lc

if $_FILES[uploadedfile][error] then

put "There was an error uploading your file:" && $_FILES[uploadedfile][error] & "<br />"

else

put $_FILES[uploadedfile][name] into tFileName

put $_FILES[uploadedfile][type] into tFileType

put $_FILES[uploadedfile][size] into tFileSize

put $_FILES[uploadedfile][filename] into tFilePath

put "Your file '" & tFileName & "' was uploaded successfully. It is" && tFileSize && "bytes in size."

// If the file is a text file, the display what is in the file

if tFileType is "text/plain" then

put URL ("file:" & tFilePath) into tText

put "<h1>File Content</h1><p>" & tText & "</p>"

end if

end if

?>

</p>

</body>

</html>
```

[Prev: How Do I Use AJAX with LiveCode Server?](#) [Next: How do I use Cookies on LiveCode Server?](#)

## 3 Comments

**Malte** Wednesday Dec 14 2011 at 10:56 AM

Hi,

Thanks for this tutorial!

it seems the file is stored in /tmp  
My questions are, is it ensured the engine has writing permissions ther? How long does the file live there? Is it cleared, after the script finished executing?

All the best,

Malte

**Elanor Buchanan** Thursday Jan 05 2012 at 06:31 AM

Hi Malte

By default the engine should have write permissions to /tmp but this is not ensured, if you want to check you can try writing there and checking the result.

When the engine finishes executing it will clean up any files it thinks it owns. If the engine crashes it will try to clean up the files the next time it is run.

I hope that helps.

Kind regards

Elanor

**Jean Marc** Friday Nov 15 2013 at 09:30 AM

Hi from France!

Thank you for your great help.

I save the successfully uploaded file with

get shell ("rmv" &amp;&amp; tFilePath &amp;&amp; "~/public\_html/" &amp; tFileName)

mv is for move.

All the best too.

Jean marc

## Add your comment

Name\*

Email\*

Comment\*

☐ Subscribe ☐ E-Mail me when someone replies to this comment

Are you human?

☐ I'm not a robot

reCAPTCHA

[Privacy](#) [Terms](#)

Submit Comment