

PHP Includes

Includes are a PHP technique of importing external files into a PHP document. This is similar to the way we can link to an external CSS or JS file client side.

With PHP includes however, we can store HTML formatted content, client-side or PHP code, and any other text in external documents.

Advantages:

- Rapid development. A single file can be used in many places.
- Simply to make changes. Just like a linked CSS file, we can edit the whole site from one file.

Disadvantages:

- Sometimes difficult to affect changes to a simple page only.

The external files that are imported can be any format: .html, .txt, .php.

Hint: A very common mistake is to create external includes using a text editor and leaving in template HTML formatting (doctype, etc.) When this file is imported, the resulting document can end up having multiple sets of doctype, <head>, <body>, etc.

Syntax:

```
<?php
```

```
include ("pathto/file.php");
```

```
?>
```

Activity

Create a simple HTML formatted document called *includes.php*. Somewhere in the <body>, create a PHP block, then an include statement to an external file called *includedfile.php*. Create this external document with some simply HTML formatting and content (such as "<h1>Here is external content</h1>").

Note how the first document imports the included file content.

Try changing the path or filename to deliberately create an error: Note the error message.

Includes Site Activity

Give the folder *includes_site_giveto_students* to students.

Inside, create an */includes/* folder.

From the design.htm file, create the *header.php* and *footer.php*.

- Take all code/content above the PAGE CONTENT HERE comment and paste into header.php
- Save as header.php in the */includes/* folder.
- Take all code below END PAGE CONTENT HERE comment and paste into footer.php

Create a file called index.php.

```
<?php
```

```
include ("includes/header.php");
```

```
?>
```

```
<h1>Welcome to the E.A. Poe Website</h1>
```

Page specific content goes here

```
<?php
```

```
include ("includes/footer.php");
```

```
?>
```

Load the site, and note how the index page loads with all the design from the includes.

Create your site nav: Change links in header.php to expected file names.

Create the second page file (using index as a template), *berenice.php*. Use content provided.

Switch statement to create individual page changes.

Since we are using includes to get a consistent site design AND to simplify development, we may lose some of the individuality of each page.

Here's one way we can still do specific things for each page: Create variable that change depending on the page loaded.

Thus, we can change necessary features like the title tag (for usability and SEO), additional content like what is loaded into a sidebar, etc.

`$_SERVER` is an array (many items stored in one variable) related to the current file, the IP of the user, the document that referred to the current file, etc. Many useful features.

`$_SERVER['PHP_SELF']` is the current file name and its path

`basename($_SERVER['PHP_SELF'])` is the current filename only; no path.

This code goes in the very top of header.php, and is thus included in all pages.

```
*****/  
$thisFile = basename($_SERVER['PHP_SELF']);  
  
switch ($thisFile){  
    case "index.php":  
        $thisPageTitle = "Edgar Allen Poe";  
        $thisSideBarFile = "includes/summaries/home.txt";  
        break;  
    case "berenice.php":  
        $thisPageTitle = "Edgar Allen Poe - Berenice";  
        $thisSideBarFile = "includes/summaries/berenice.txt";  
        break;  
}
```

Note: This code is NOT finished: Student will have to finish all pages on their own.

This can then be used as a:

1. Dynamic <title>
 <title><?php echo \$thisPageTitle; ?></title>
2. Dynamic sidebar include:

```
<h2>Summary</h2>
```

```
<?php  
include ($thisSideBarFile);  
?>
```

Students can finish all pages on their own. This activity will be assessed as part of Lab 1.