

Useful PHP Things & REST

CS518

Dr. Justin F. Brunelle

Includes & Requires

- Includes:
 - Copy the code in the file if you can find the file
 - `<?php include 'commonFile.php'; ?>`
- Requires:
 - Copy the code in the file; halt and catch fire if you can't find it
 - `<?php require 'commonFile.php'; ?>`
 - `<?php require 'fileThatDoesntExist.php'; ?>`
- Examples:
 - <http://www.cs.odu.edu/~jbrunelle/cs518/examples/phpThings/>

File I/O

- `$text = file_get_contents($filename);`
- `file_put_contents($filename, $text);`

REST

- Representational State Transfer
- All nouns
 - HTTP POST `http://localhost/users/jbrunelle`
 - HTTP GET `http://localhost/users/jbrunelle`
- NO VERBS!
 - HTTP GET `http://localhost/createUsers.php?username=jbrunelle&position=professor`

HTTP Actions

- GET – give me a resource/list of resources
- POST – create a resource
- PUT – update/create/replace a resource
- PATCH – update this feature of the resource
- DELETE – delete the resource

- HEAD – give me metadata for this resource
- OPTIONS – give me available actions for this resource

Examples

POST /oduScores/opponents HTTP/1.1

Host: api.example.org

Accept: application/json

Content-Type: application/json

Content-Length: 28

```
{  
  "name": "Hampton",  
  "score": 30  
}
```

Examples

GET /oduScores/opponents/Hampton

Examples

PATCH /oduScores/opponents/Hampton HTTP/1.1

Host: api.example.org

Accept: application/json

Content-Type: application/json

Content-Length: 10

```
{  
  "score": 17  
}
```


Examples

GET /oduScores/opponents

PHP Sessions and Cookies

PHP Sessions

- Like a cookie, but variables/data not stored on the client
- Ends when browser/window is closed
- Persists across multiple pages

```
<?php
session_start();

print_r($_SESSION);
?>
```

Ending Sessions

```
<?php  
// remove all session variables  
session_unset();  
  
// destroy the session  
session_destroy();  
?>
```

PHP Cookies

- File delivered to the browser from the server
- PHP can access the contents of the cookie

```
setcookie(name, value, expire, path, domain, secure, httponly);
```

```
<?php
$cookie_name = "user";
$cookie_value = "John Doe";
setcookie($cookie_name, $cookie_value, time() +
(86400 * 30), "/"); // 86400 = 1 day

if(!isset($_COOKIE[$cookie_name])) {
    echo "Cookie named '" . $cookie_name . "' is not set!";
} else {
    echo "Cookie '" . $cookie_name . "' is set!<br>";
    echo "Value is: " . $_COOKIE[$cookie_name];
}
?>
```

PHP Removing Cookie (eating cookie?)

```
<?php
// set the expiration date to one hour
ago
setcookie("user", "", time() - 3600);
?>
```

```
<?php
if(count($_COOKIE) > 0) {
    echo "Cookies are enabled.";
} else {
    echo "Cookies are disabled.";
}
?>
```