

CLOSE ENCOUNTERS

with
PHP

Level 4

Composer & Autoloading

Refactoring to a Standard

PHP Standards Recommendations (PSR)

PSRs are recommended by the PHP Framework Interop Group, or the PHP-FIG.

PSR-1. Basic Coding Standard

PSR-2. Coding Style Guide

PSR-3. Logger Interface

PSR-4. Autoloading Standard

PSR-6. Caching Interface

PSR-7. HTTP Message Interface

CLOSE
ENCOUNTERS
with
PHP

PHP Standards Recommendations (PSR)

PSRs are recommended by the PHP Framework Interop Group, or the PHP-FIG.

PSR-1. Basic Coding Standard

*Sets standard coding elements to ensure
a good fit between shared PHP projects*

PSR-2. Coding Style Guide

Sets a standard for readability within our code

Both of these PSRs and more can be found at go.codeschool.com/psr

CLOSE
ENCOUNTERS
with
PHP

Coding to a Standard With PSR-1

PHP Standards Recommendations #1 is the PHP Basic Coding Standard.

app/src/app.php

```
<?php
function validate_date($date_string)
{
    if ($time = strtotime($date_string)) {
        return date('F jS Y', $date_string);
    } else {
        return $date_string . ' does not look valid.';
    }
}
```

*Files **SHOULD** either declare symbols (functions)*

or

... cause side effects (generate output)...

```
...
if (!empty($date) && !empty($email) && !empty($description)) {
    echo validate_date($date);
}
```

*...but **SHOULD NOT** do both*

Creating a Validation Function File

Create a validation.php file nested in the app folder.



1. Create a new file in the **app** folder named **validation.php**

app/src/validation.php

Creating a Validation Function File

Create a validation.php file nested in the app folder.



1. Create a new file in the **app** folder named **validation.php**
2. Move **validate_date** function code to the new file

app/src/validation.php

```
<?php
function validate_date($date_string)
{
    if ($time = strtotime($date_string)) {
        return date('F jS Y', $date_string);
    } else {
        return $date_string . ' does not look valid.';
    }
}
```


Creating a Validation Function File

Create a validation.php file nested in the app folder.



1. Create a new file in the **app** folder named **validation.php**
2. Move **validate_date** function code to the new file
3. In **app.php** require the new **validation.php** file

app/src/app.php

```
<?php
require __DIR__ . '/validation.php';
...
if (!empty($date) && !empty($email) && !empty($descri

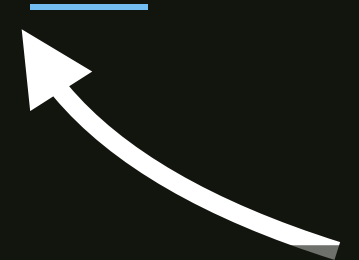
echo validate_date($date);

if (filter_var($email, FILTER_VALIDATE_EMAIL)) {
    echo "<p>Email: $email</p>";
}
echo '<p>' . htmlspecialchars($description) . '</p>';
```


Requiring With an Absolute Path

app/src/app.php

```
<?php  
require __DIR__ . '/validation.php';
```



... *__DIR__ is a “magic constant” in PHP that gives us an absolute path of the current file*

```
if (!empty($date) && !empty($email) && !empty($description)) {  
  
    echo validate_date($date);  
  
    if (filter_var($email, FILTER_VALIDATE_EMAIL)) {  
        echo "<p>Email: $email</p>";  
    }  
    echo '<p>' . htmlspecialchars($description) . '</p>';  
}  
}
```

More magic constants can be found at go.codeschool.com/php-magic-constants

Refactoring Recap

Let's walk through what we learned in this section.

- PSRs are recommendations made by the PHP Framework Interop Group, or PHP-FIG
- Cleaning up our code using the PSR-1 Basic Coding Standard
- PHP magic constant of **__DIR__**

CLOSE ENCOUNTERS

with
PHP