

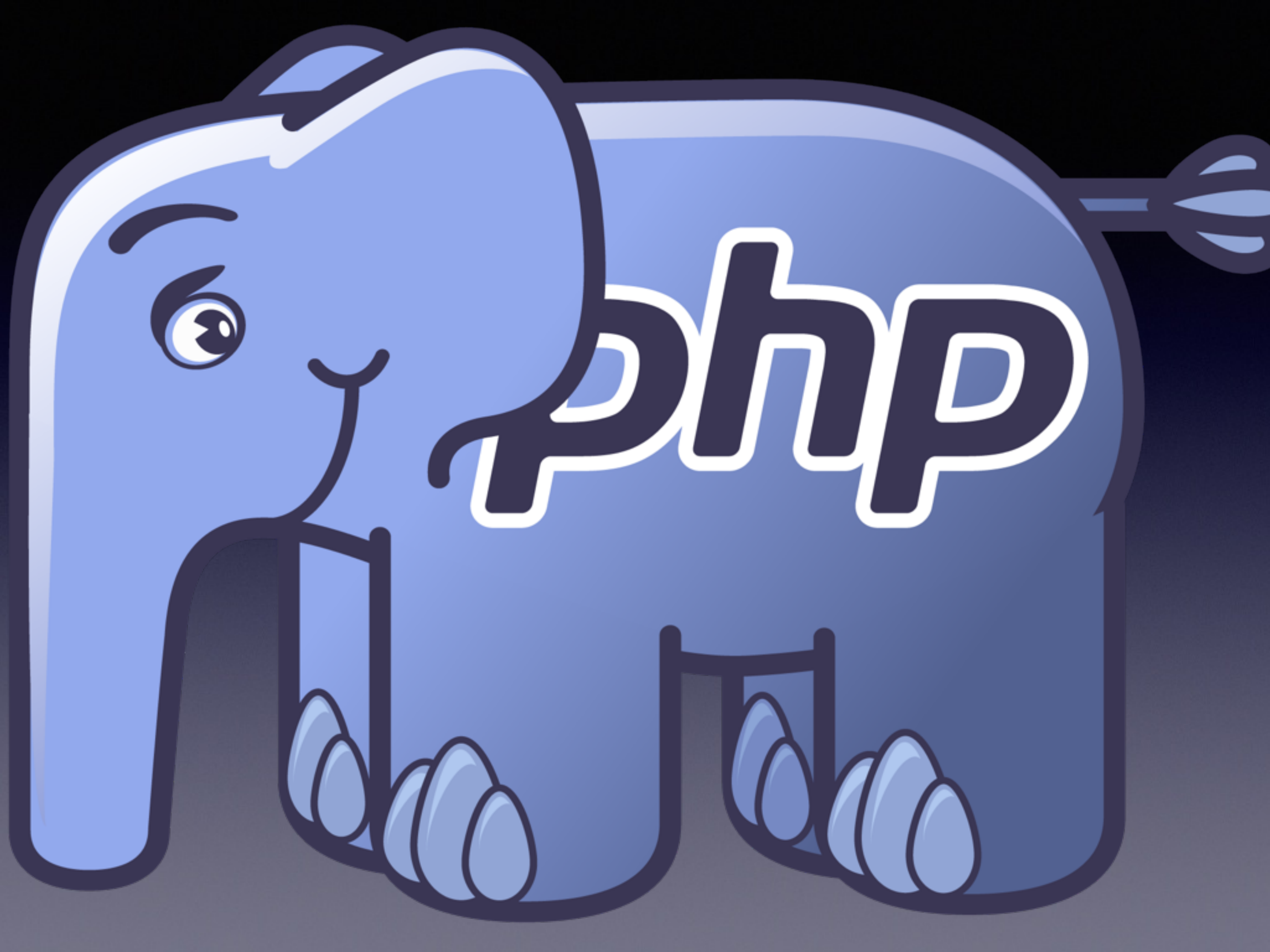
PHP Misses You

Or how I learned to stop hating the language

Your Choice



You're at the Bar



Ternary Statements?

```
1 <?php
2
3 $test = 1;
4 echo ($test === 1) ? "one" :
5      ($test === 2) ? "two" : "three", "\n";
6
```

two

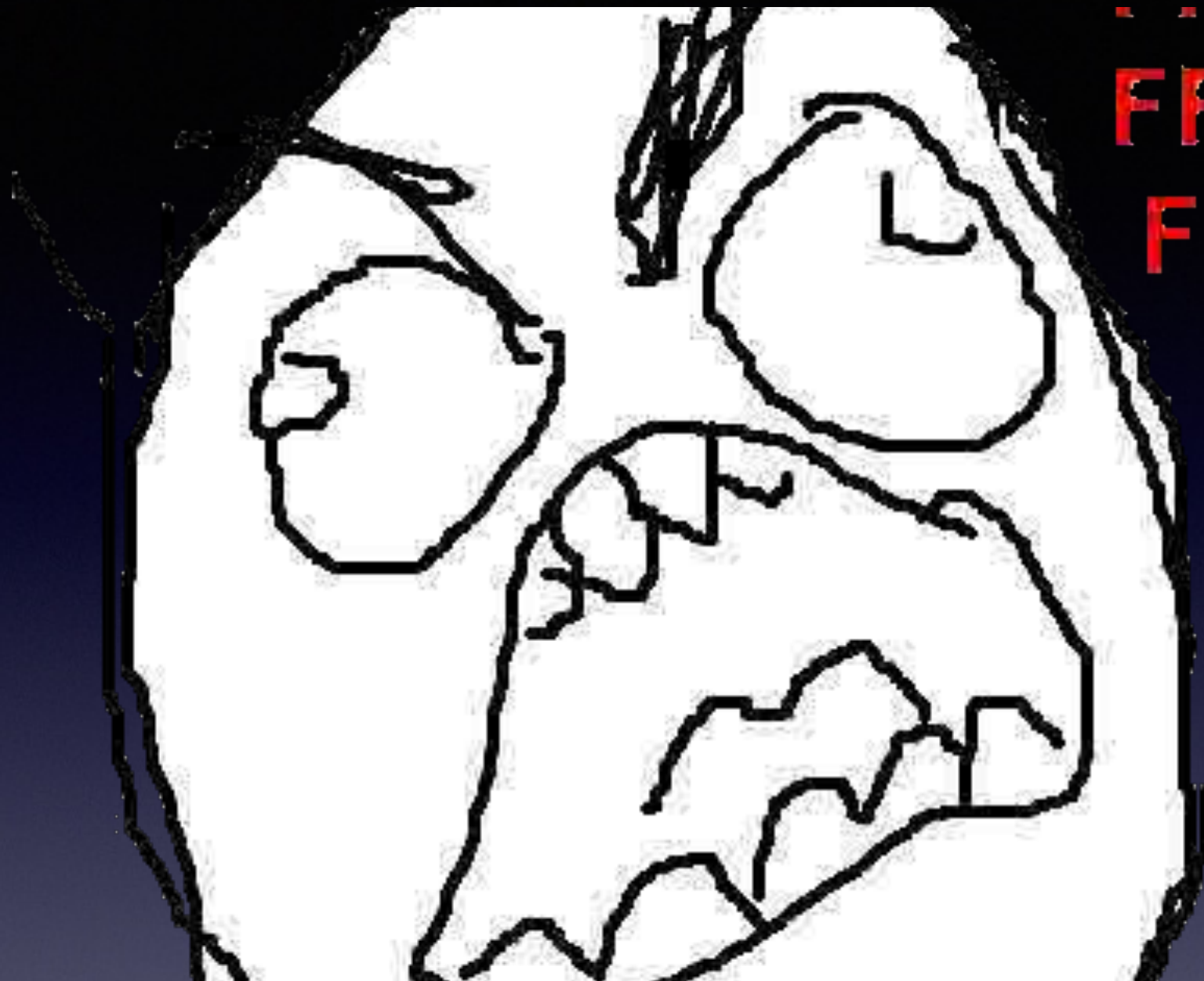
empty()

```
1 <?php
2
3 function test()
4 {
5     return true;
6 }
7
8 var_dump(empty(test()));
9
```

Fatal error: Can't use function return value in write context in php shell code on line #

Naming?

- `strcmp()`
- `strcasecmp()`
- `is_object()`
- `isset()`
- `base64_encode()`
- `urlencode()`



.....
FFFFFFFFF
FFFFFFFFF
FFFUUU
UUUUU
UUUUU
UUUUU
UUUUU
.....

The list goes on...

Time for a change

Who is at fault?

You decide to talk

Before we go any further...

Time to stop writing bad PHP

require_once is dead

```
1 <?php
2
3 require_once "Class1.php";
4 require_once "Class2.php";
5 require_once "Class3.php";
6 require_once "Class4.php";
7 require_once "Class5.php";
8 require_once "Class6.php";
9 require_once "Class7.php";
10 require_once "Class8.php";
11 require_once "Class9.php";
12
```


Long live autoloading

```
1 <?php
2
3 require_once "vendor/autoload.php";
```

- Included with composer.
- Check out php fig PSR-0.

Composer Who?

- PHP has a reasonable package manager now.
- <https://getcomposer.org>
- Packages searchable on <https://packagist.org>
- Installing new packages as simple as:
`$ composer require monolog/monolog:1.9.*`

Object Oriented, procedural, functional

- Decent Object-Oriented design
- Namespaces
- Anonymous functions - closures
- Callbacks
- Procedural still exists.


```
1 <?php
2
3 $someDate = new DateTime( '2014-05-30' );
4 echo $someDate->format( 'm-d-Y' ), "\n";
5
6 echo date_format(date_create( '2014-05-30' ),
                        'm-d-Y' ), "\n";
```

Generators & Iterators

- Iterators are still useful
- Generators are what most people want

Generators

```
1 <?php
2
3 public function getLine($file)
4 {
5     $f = fopen($file, 'r');
6     if (!$f) throw new Exception();
7     while ($line = fgets($f)) {
8         yield $line;
9     }
10    fclose($f);
11 }
```


“What about testing?”

–Ruby

Testing

- PHPUnit isn't the only game in town anymore
- Behat - <http://behat.org>
- Codeception - <http://codeception.com>
- Selenium integration

“Does PHP even event?”

–node.js

REACT

- Event-driven, non-blocking I/O
- Ext-event, LibEv, LibEvent, stream_select()
- Libraries and Examples at <http://reactphp.org>

React

```
1 $loop = React\EventLoop\Factory::create();
2 $socket = new React\Socket\Server($loop);
3
4 $socket->on('connection', function ($conn){
5     $conn->pipe($conn);
6 });
7
8 $socket->listen(4000);
9 $loop->run();
```

So what else is new?

(A lot)

More New

- `empty()` has been enhanced.
- php development server
- OPCache is now in PHP

HHVM

- Great Performance
- Easy to use

HHVM on Ubuntu 14.04

```
1 wget -O - http://dl.hhvm.com/conf/
hhvm.gpg.key | sudo apt-key add -
2 echo deb http://dl.hhvm.com/ubuntu trusty
main | sudo tee /etc/apt/sources.list.d/hhvm.list
3 sudo apt-get update
4 sudo apt-get install hhvm
```


Running HHVM

- Run a file:
1 hhvm hello.php
- Run a server in current directory:
1 sudo hhvm -m server
- Runs in daemon mode as well - see hhvm.com's "Getting Wordpress Running on HHVM"

Password Hashing

```
1 $options = [  
2     'cost' => 11,  
3     'salt' => mcrypt_create_iv(  
4         22,  
5         MCRYPT_DEV_URANDOM  
6     ),  
7 ];  
8 echo password_hash(  
    "self.conf",  
    PASSWORD_BCRYPT,  
    $options  
). "\n";
```

Much more

- Array And String literal dereferencing (“Yo!”[1] === “o”)
- static keyword for late static bindings
- foreach supports non-scalar keys
- etc

So why bother?

- Rekindling an old relationship
- Maintenance
- Servers
- Speed

PHP misses you

But it's doing fine, thanks for asking.

Thank you!