PHP 7.1 - Iterable and Void

New Types For Typecasting
August 2016

iterable (it-ah-ah-bul)

This is a multi-type type which includes any objects which implement the Traversable interface, as well as arrays.

It means the function can accept or return a value which will work with the looping constructs.

```
// arrays are iterable.

$Stuff = [ 1, 2, 3 ];
```

```
// objects implementing Iterator are iterable.

class Pile
implements Iterator {
    public function
    Current() { /* ... */ }

    public function
    Key() { /* ... */ }

    public function
    Next() { /* ... */ }

    public function
    Rewind() { /* ... */ }

    public function
    Valid() { /* ... */ }
}
```

```
// generators are iterable.

function
ImAnGenerator() {
    for($A = 1; $A <= 3; $A++) {
    yield $A;
}
```

```php
// process iterables.

function
CanHasIter(Iterable $Stuff):
Void {
    if(is_array($Stuff))
    echo 'by array', PHP_EOL;

    else
    printf('by %s%s', get_class($Stuff), PHP_EOL);

    echo '> ';

    foreach($Stuff as $Thing)
    echo $Thing, ' ';

    echo PHP_EOL;
    return;
}
```

```php
$Stuff = [ 'omg','wtf','bbq' ];
$Pile = (new Pile)->SetData($Stuff);

// give it a direct array.
CanHasIter([1, 2, 3]);

// give it an array variable.
CanHasIter($Stuff);

// give it an iterable object.
CanHasIter($Pile);

// give it an iterable object.
CanHasIter($Pile->StartsWithVowel());

// give it a generator.
CanHasIter(ImAnGenerator());

// give it a string. (intentional error)
CanHasIter('omfg');
```

```
C:\~\Desktop\PHP71> php Code\Iterable.php

by array
> 1 2 3
by array
> omg wtf bbq
by Pile
> omg wtf bbq
by Pile
> omg
by Generator
> 1 2 3

Fatal error: Uncaught TypeError: Argument 1 passed to
CanHasIter() must be iterable, string given
```

void (voyyyyy-duh)

This is a restrictive type which declares "this function returns nothing." aka "i does things, not give things"

It means so much nothing that even `return NULL` is not valid. Function must EOB or end with a flat with `return;`

Can only be used as a return type, not an argument type.

```
// void: hitting the eob without return

function
TotallyVoid():
Void {

    /* ... */

}
```

```
// void: returning absolutely nothing.

function
AlsoTotallyVoid():
Void {

   /* ... */

   return;
}
```

```
// fail: returning anything, even NULL.

function
InvoidableFail():
Void {

    /* ... */

    return NULL;
}
```

```
C:\~\Desktop\PHP71> php Code\Void.php

Fatal error: A void function must not return a value (did
you mean "return;" instead of "return null;"?)
```

@DallasPHP

@bobmagicii

Code:
https://github.com/bobmagicii/dallasphp-201608-php71-iterable-void

Slides:
https://speakerdeck.com/bobmajdakjr/php-7-dot-1-iterable-and-void