

Generar documentación con PHPDocumentor

Requisitos:

- ✓ Sistema Operativo **Linux**.
- ✓ Tener una mínima aplicación PHP (y **comentada correctamente**), como por ejemplo esta, que es la que usaré para este ejemplo de documentación:

```
1  <?php
2
3      class Calcula {
4
5          /**
6           * suma dos numeros
7           * @param float $a
8           * @param float $b
9           * @return float
10          */
11         public function suma($a, $b){
12             return $a+$b;
13         }
14
15         /**
16          * resta dos numeros
17          * @param float $a
18          * @param float $b
19          * @return float
20          */
21         public function resta($a, $b){
22             return $a-$b;
23         }
24     }
25
26
27  ?>
```

- Primero, deberás poner con comentarios, **una breve definición** de lo que hará cada función de tu aplicación.
- Después de la definición, habrá que poner los **parámetros que recibe** la función si es que recibe parámetros. También debes especificar **de que tipo son** los parámetros y **que nombre tienen asignados** estos.
- Por último, deberás poner **que retornará tu función**, es decir, que **tipo de dato**. Por ejemplo; un **float**, un **int**, un **string**, etc.

- ✓ Después de tener en cuenta todo lo anterior, quedan varias cosas más por aclarar. Ahora, vamos a crear una **estructura de directorios** adecuada (dónde quieras), quedando exactamente así:

```
usuario@usuario:~/Escritorio$ tree
.
├── aplicacion
│   └── calcula_com.php
└── documentacion

2 directories, 1 file
usuario@usuario:~/Escritorio$
```

- El directorio **aplicación** contendrá nuestra **aplicación**.
 - Y el directorio **documentación**, en principio no contendrá nada, pero ahí es donde volcaremos la **documentación** que luego generaremos.
- ✓ Por último, instalaremos la herramienta **composer**, la cual nos ayudará a instalar **PHPDocumentor**, lo haremos con el siguiente comando:

```
usuario@usuario:~/Escritorio$ sudo apt install composer
```

- ✓ Ya está todo listo, ahora vamos con los últimos pasos que son bastante sencillos, sólo nos bastarán un par de comandos.

1. Instalaremos la herramienta PHPDocumentor con el siguiente comando:

```
usuario@usuario:~/Escritorio$ composer require --dev phpdocumentor/phpdocumentor dev-master
```

- Después de ejecutar este comando, veremos que se instalarán una serie de paquetes, hay que tener paciencia.
- Después de instalar los paquetes, podremos observar que se han creado nuevos archivos y una nueva carpeta en el directorio en el que hemos ejecutado el comando.

```
usuario@usuario:~/Escritorio$ ls -l
total 124
drwxrwxr-x  2 usuario usuario  4096 mar  6 18:34 aplicacion
-rw-rw-r--  1 usuario usuario    83 mar  6 18:46 composer.json
-rw-rw-r--  1 usuario usuario 109068 mar  6 18:49 composer.lock
drwxrwxr-x  2 usuario usuario  4096 mar  6 18:31 documentacion
drwxrwxr-x 22 usuario usuario  4096 mar  6 18:49 vendor
usuario@usuario:~/Escritorio$
```

2. Y ahora, generaremos la documentación de nuestra aplicación con el siguiente comando:

```
usuario@usuario:~/Escritorio$ php vendor/bin/phpdoc run -d aplicacion/ -t documentacion/

usuario@usuario:~/Escritorio$ php vendor/bin/phpdoc run -d app/ -t documentacionCollecting files .. OK
Initializing parser .. OK
Parsing files
Parsing /home/usuario/Escritorio/app/calcula_com.php
No summary was found for this file
No summary for class \Calcula
Storing cache in "/home/usuario/Escritorio/documentacion" .. OK
Load cache .. 0.002s
Preparing template "clean" .. 0.067s
Preparing 17 transformations .. 0.000s
Build "elements" index .. 0.000s
Replace textual FQCNs with object aliases .. 0.000s
Resolve @link and @see tags in descriptions .. 0.000s
Enriches inline example tags with their sources .. 0.000s
Build "packages" index .. 0.009s
Build "namespaces" index and add namespaces to "elements" .. 0.000s
Collect all markers embedded in tags .. 0.000s
Transform analyzed project into artifacts ..
Applying 17 transformations
Initialize writer "phpDocumentor\Plugin\Core\Transformer\Writer\FileIo"
Initialize writer "phpDocumentor\Plugin\Twig\Writer\Twig"
Initialize writer "phpDocumentor\Plugin\Graphs\Writer\Graph"
Execute transformation using writer "FileIo"
Execute transformation using writer "FileIo"
Execute transformation using writer "FileIo"
Execute transformation using writer "FileIo"
Execute transformation using writer "FileIo"
Execute transformation using writer "twig"
Execute transformation using writer "twig"
Execute transformation using writer "twig"
Execute transformation using writer "twig"
Execute transformation using writer "twig"
Execute transformation using writer "twig"
Execute transformation using writer "twig"
Execute transformation using writer "twig"
Execute transformation using writer "twig"
Execute transformation using writer "twig"
Execute transformation using writer "twig"
```

- Si nos vamos al directorio documentación, veremos un archivo index.html, este, contiene toda la documentación sobre nuestra aplicación, sólo tendremos que abrir dicho archivo en el navegador y listo.

```
usuario@usuario:~/Escritorio$ ls -l documentacion
total 56
drwxr-xr-x 2 usuario usuario 4096 mar  6 20:15 classes
drwxr-xr-x 3 usuario usuario 4096 mar  6 20:15 css
drwxr-xr-x 2 usuario usuario 4096 mar  6 20:15 files
drwxr-xr-x 2 usuario usuario 4096 mar  6 20:15 font
drwxr-xr-x 2 usuario usuario 4096 mar  6 20:15 graphs
drwxr-xr-x 3 usuario usuario 4096 mar  6 20:15 images
-rw-r--r-- 1 usuario usuario 8529 mar  6 20:15 index.html
drwxr-xr-x 3 usuario usuario 4096 mar  6 20:15 js
drwxr-xr-x 2 usuario usuario 4096 mar  6 20:15 namespaces
drwx----- 2 usuario usuario 4096 mar  6 20:15 phpdoc-cache-2e
drwx----- 2 usuario usuario 4096 mar  6 20:15 phpdoc-cache-43
drwxr-xr-x 2 usuario usuario 4096 mar  6 20:15 reports
usuario@usuario:~/Escritorio$
```

API Documentation

Archivo | /home/usuario/Escritorio/documentacion/classes/Calcula.html

API Documentation

Calcula

Summary

Methods

Properties

Constants

suma()

resta()

No protected methods found

No private methods found

No public properties found

No protected properties found

No private properties found

No constants found

N/A

N/A

Methods

suma()

suma(float \$a, float \$b) : float

suma dos numeros

Parameters

FILE

calcula_com.php

PACKAGE

Default

CLASS HIERARCHY

Calcula

Tags

None found

INHERITED FROM

Calcula

Tags

None found