

Forum:[Hack Forums](#)

Thread: [Moving from procedural to OOP?](#)

.eXe - July 5, 2018, 12:57 am

How difficult is it to move from procedural to OOP? I've always coded in proc, and never even touched OOP. Are there any preparation steps I should take? Any pitfalls to look out for?

Quad - July 5, 2018, 1:09 am

shouldn't be too difficult if you're good at procedural.

in the simplest cases you will be changing from

PHP Code

```
<?php
```

```
do_something($resource, $paramater)
```

to

PHP Code

```
<?php
```

```
$resource->do_something($paramater);
```

d4rkc4st - July 6, 2018, 7:02 am

(Jul 5, 2018 12:57 AM)Evil.eXe Wrote: [\[CITE\]](#) How difficult is it to move from procedural to OOP? I've always coded in proc, and never even touched OOP. Are there any preparation steps I should take? Any pitfalls to look out for?

There must be tons of tutorials and books out there since PHP became OO. e.g. <https://code.tutsplus.com/tutorials/object-oriented-php-for-beginners--net-12762>

.eXe - July 6, 2018, 7:11 am

(Jul 6, 2018 07:02 AM)d4rkc4st Wrote: [\[CITE\]](#) There must be tons of tutorials and books out there since PHP became OO. e.g. <https://code.tutsplus.com/tutorials/object-oriented-php-for-beginners--net-12762>

Using PHP in an OOP setting isn't what I asked, it was the best way to move from only using procedural to OOP.

m0noxide - July 6, 2018, 1:25 pm

Look up the S.O.L.I.D design principles for OOP and that should steer you in the right direction.

vercenta - July 8, 2018, 8:22 am

(Jul 6, 2018 07:11 AM)Evil.eXe Wrote: [\[CITE\]](#) Using PHP in an OOP setting isn't what I asked, it was the best way to move from only using procedural to OOP.

Just make sole purposes functions for everything and then call it a class, pretty much what OOP is lol.

Link2 - July 9, 2018, 2:03 am

Hi,

i wrote a little example for you:

gist.github.com/mariusgerum/a32e803e1022956759c4764dad74dec9 (sorry i can't post real links yet, so just copy&paste it into your browser).

As you can see the logic resides in one single place where you actually need it instead of spreading that shit all over your entire codebase.