Create test cases for unit testing developers

Getting class comments

<?php

/\*\*

\* @var (0, 0) == 0

\* @test (0, 1) == 1

\* @test (1, 0) == 1

\* @test (1, 1) == 2

\* @test (1, 2) == 4

\*/

class test {

}

function getClassAnnotations($class)

{

$r = new ReflectionClass($class);

$doc = $r->getDocComment();//use for geting class descriptions

preg\_match\_all('#@test(.\*?)\n#s', $doc, $annotations);

return $annotations[1];

}

$test = new test();

Print\_r(getClassAnnotations($test));

Getting function annotations/ comments

class test {

/\*\*

\* @test (0, 0) == 0

\* @test (0, 1) == 5

\* @test (1, 0) == 1

\* @test (1, 1) == 2

\* @test (1, 2) < 4

\*/

function add($a, $b) {

return $a + $b;

}

}

$test = new test();

$r = new ReflectionClass($test);

$fn = $r->getMethod('add')->getDocComment();

$pattern = '/@test(.\*)/';

preg\_match\_all($pattern, $fn, $annotations);

$p="";

foreach($annotations[1] as $a) {

$b = explode(") ",$a);

$COND = str\_replace("("," ",$b[0]);

$operator\_ans = explode(" ",$b[1]);

$operator = $operator\_ans[0];

$ans = $operator\_ans[1];

$t = call\_user\_func\_array(array($test,"add"), explode(",",$COND));

switch($operator) {

case "==";

if( $t == $ans) {

$p .= $a." Passed <br/>";

}

else {

$p .= $a." Faild <br/>";

}

break;

case ">";

if( $t > $ans) {

$p .= $a." Passed <br/>";

}

else {

$p .= $a." Faild <br/>";

}

break;

case "<";

if( $t < $ans) {

$p .= $a." Passed <br/>";

}

else {

$p .= $a." Faild <br/>";

}

break;

case "===";

if( $t === $ans) {

$p .= $a." Passed <br/>";

}

else {

$p .= $a." Faild <br/>";

}

break;

case "!=";

if( $t != $ans) {

$p .= $a." Passed <br/>";

}

else {

$p .= $a." Faild <br/>";

}

break;

case "!==";

if( $t !== $ans) {

$p .= $a." Passed <br/>";

}

else {

$p .= $a." Faild <br/>";

}

break;

}

}

echo $p;