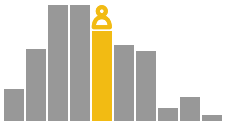
Sami HOUARBI (sami.houarbi@gmail.com)

Campaign: [Scala] PHP #1 **Programming language(s):** PHP **Language:** English **Date:** Sep 14, 2018

SCORE	DURATION	BETTER THAN
42%	0H40	59%
731 / 1,730 pts	/ 0H59	of developers



PHP

42% (731 / 1,730)

Design	25%	(40 / 160)
Language knowledge	89%	(506 / 566)
Problem solving	20%	(139 / 696)
Reliability	15%	(46 / 308)

Question 1: define()

PHP 00:06 / 00:25 40 / 40 pts

Question

Which option is a valid constant declaration in PHP?

Answer

- [x] *define("MAXSIZE", 100);*
- [ ] *#define MAXSIZE 100*
- [ ] *static final MAXSIZE = 100;*
- [ ] *constant MAXSIZE = 100;*

Result

Correct answer
Language knowledge +40pts

Question 2: header()

PHP 00:05 / 00:35 40 / 40 pts

Question

```
<html>
<?php
header('Location: http://www.example.com/');
?>
</html>
```

This code produces an error. What can be done to correct it?

Answer

- [ ] Call function footer() before the end of the script
- [x] **Move the html tag after the call to header()**
- [ ] Replace http://www.example.com with a valid server URL

Result

Correct answer
Language knowledge +40pts

Question 3: Size of a string

PHP 00:18 / 00:20 40 / 40 pts

Question

Which function returns the number of characters in an ASCII string?

Answer

- [ ] length
- [ ] size
- [ ] strsize
- [x] **strlen**
- [ ] len

Result

Correct answer
Language knowledge +40pts

Question 4: Bitwise Operators: &

PHP 00:20 / 00:20 0 / 40 pts

Question

In a base 2 system (binary), what is the value of ***0001 & 0001*** ?

Answer

- [x] *0010*
- [ ] *0000*
- [ ] *0001*

Result

Incorrect answer
Language knowledge ~~+40pts~~

Question 5: Exit

PHP 00:17 / 00:30 40 / 40 pts

Question

Type the name of the language structure equivalent to ***exit()*** which is commonly used in scripts.

Answer

die

Result

Correct answer
Language knowledge +40pts

Correct answer(s)

- die
- die()
- die();

Question 6: "In - Out" variable

PHP 00:55 / 01:00 60 / 60 pts

Question

```
$i = 0;
inc($i);
echo $i;

function inc($arg) {
    $arg++;
}
```

In the signature of ***inc***, what should replace ***$arg*** to display ***1***?

Answer

&$arg

Result

Correct answer
Language knowledge +60pts

Correct answer(s)

- &$arg
- inc(&$arg)
- function inc(&$arg)
- & $arg
- inc(& $arg)
- function inc(& $arg)

Question 7: Evaluate a string

PHP 00:22 / 00:30 40 / 40 pts

Question

Type the name of the function which evaluates a string as PHP code.

Answer

eval

Result

Correct answer
Language knowledge +40pts

Correct answer(s)

- eval
- eval()
- eval ()

Question 8: Unary operator (i++)

PHP 00:09 / 00:25 40 / 40 pts

Question

$i = 0;
echo $i++;

Type the result displayed by this piece of code.

Answer

0

Result

Correct answer
Language knowledge +40pts

Correct answer(s)

- 0

Question 9: Sub string

PHP 00:34 / 00:35 40 / 40 pts

Question

Type the name of the function used to return a portion of a string from a start index and a length.

Answer

substr

Result

Correct answer
Language knowledge +40pts

Correct answer(s)

- substr
- substr()
- substr ()

Question 10: count()

PHP 00:30 / 00:30 0 / 20 pts

Question

Type the name of the function to count all elements in an array.

Answer

array_count_values

Result

Correct answer
Language knowledge ~~±20pts~~

Correct answer(s)

- count
- count()
- sizeof
- sizeof()
- count ()
- sizeof ()

Question 11: Array size

PHP 00:29 / 00:30 40 / 40 pts

Question

```php
<?php
$a = array();
$a[10] = 'ten';
?>
```

What is the size of the array ***$a***?

Answer

- [x] 1
- [ ] 10
- [ ] 11

Result

Correct answer
Language knowledge +40pts

Question 12: String and spaces

PHP 00:29 / 00:30 20 / 20 pts

Question

Enter the name of the function that removes whitespaces from the beginning and end of a string.

Answer

trim

Result

Correct answer
Language knowledge +20pts

Correct answer(s)

- trim
- trim()
- trim();

Question 13: Type of a variable

PHP 00:28 / 00:30 40 / 40 pts

Question

Type the name of the function which displays structured information about a variable (its type and value).

Answer

var_dump

Result

Correct answer
Language knowledge +40pts

Correct answer(s)

- var_dump
- var_dump()
- var_dump ()
- print_r
- print_r()
- print_r ()
- var_export
- var_export()
- var_export ()

Question 14: Object instance

PHP 00:26 / 00:30 20 / 20 pts

Question

class A {}

$a = new A();
$b = new A();
$c = $b;

How many instances of A are created?

Answer

- [ ] 0
- [ ] 1
- [x] 2
- [ ] 3
- [ ] 4

Result

Correct answer
Language knowledge +20pts

Question 15: Move towards zero

PHP 10:25 / 15:00 231 / 300 pts

Question

Implement ***closestToZero*** to return the integer in the array ***$ints*** that is closest to zero. If there are two integers equally close to zero, consider the positive element to be closer to zero (example: if ***$ints*** contains -5 and 5, return 5). If ***$ints*** is empty, return 0 (zero).

Input: integers in ***$ints*** have values ranging from -2147483647 to 2147483647.

Answer

```php
<?php /* PHP 7 code below */?>
<?php
function closestToZero(array $ints) {
    $closest = null;
    if(empty($ints)){
        return 0;
    }
    foreach ($ints as $int){

        if($closest == null || abs(0-$closest) > abs($int-0)){
            $closest = abs($int);
        }
    }

    return $closest;
}
?>
```

Result

- The result is correct with a simple data set [7, 5, 9, 1, 4]
  Problem solving +139pts
- The solution works with 2147483647 or -2147483647
  Reliability ~~+23pts~~
- The solution uses the function abs()
  Language knowledge +46pts
- When two integers are as close to 0, then the positive wins
  Reliability +23pts
- The solution works when the array contains only negative integers
  Reliability ~~+23pts~~
- The solution works with an empty array
  Reliability +23pts
- The solution works when the array contains only two equal negative integers
  Reliability ~~+23pts~~

Question 16: Approximation of π

PHP 15:00 / 15:00 0 / 450 pts

Question

In this exercise we will calculate an approximation of π (Pi).

The technique is as follows:
Take a random point P at coordinated (x, y) such that $0 \leq x \leq 1$ and $0 \leq y \leq 1$. If $x^2 + y^2 \leq 1$, then the point is inside the quarter disk of radius ***1***, otherwise the point is outside.

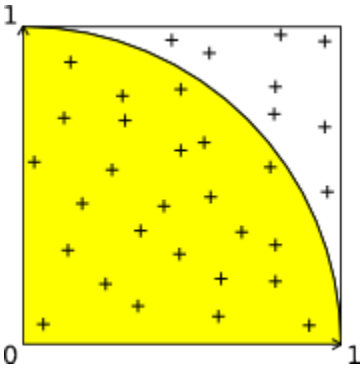


Fig 1. An example using 33 random points.

We know that the probability that a point is inside the quarter disk is equal to ***π/4***.

Write the function ***approx(array $pts)*** who will use the points ***$pts*** to return an approximation of the number π.

Input:
Each item in ***$pts*** is an instance of ***Point***. A point is represented by two numbers ***$x*** and ***$y*** such that $0 \leq \$x \leq 1$ and $0 \leq \$y \leq 1$. ***$pts*** is never NULL and it always contains at least one item.

Answer

```php
<?php /* PHP 7 code below */?>
<?php

class Point {
    public $x, $y;
}

function approx(array $pts) {
    foreach($pts as $pt){
        if(sqrt($pt->x)+sqrt($pt->y)<=1){
            echo 'true';
        }
    }



}

?>
```

Result

Approximation of π is correct (related to pts)
Problem solving ~~+386pts~~

The point P(1, 0) is inside the quarter disk
Reliability ~~±64pts~~

Sami HOUARBI (sami.houarbi@gmail.com)

Question 17: Twins

PHP 04:32 / 15:00 0 / 300 pts

Question

A twin of a word is a word written with the same letters (case insensitive) but not in the same order.

For example ***Silent*** is a twin of ***Listen***.

The ***is_twin($a, $b)*** function should return ***true*** if ***b*** is the twin of ***a*** and ***false*** otherwise. a and b are two non-null strings.

Write the body of the *is_twin($a, $b)* function.

Answer

```php
<?php /* PHP 7 code below */?>
<?php

    function is_twin($a, $b) {
        if($a== strrev($b)){
            return true;
        }
        return false;
    }

?>
```

Result

Simple tests
Problem solving ~~±107pts~~

Case insensitive
Problem solving ~~±64pts~~

Empty and one letter strings
Reliability ~~±43pts~~

Same letters but no twins
Reliability ~~±86pts~~

Question 18: Inheritance

PHP 00:11 / 01:00 20 / 20 pts

Question

Among these two solutions, which one do you prefer?

Solution #1:

```
interface FlyAble {
    public function fly();
}
abstract class AirPlane implements FlyAble {}
abstract class Bird implements FlyAble {}
```

Solution #2:

```
abstract class AirPlane {
    public abstract function fly();
}
abstract class Bird extends AirPlane {}
```

Answer

- [x] **Solution #1**
- [ ] Solution #2

Result

Correct answer
Design +20pts

Question 19: Design pattern 01

PHP 01:18 / 02:00 0 / 60 pts

Question

Type the name of the design pattern illustrated by the code below:

```php
class DatabaseDriver {

    public static function newInstance($driver) {
        $file = dirname(__FILE__).'/drivers/'.$driver.'.php';
        if (include_once($file)) {
            $driverClass = $driver . 'Driver';
            return new $driverClass;
        } else {
            throw new Exception('Database driver not found');
        }
    }
}

// example of use:
$db = DatabaseDriver::newInstance('MySQL');
```

Answer

adapter

Result

Correct answer
Design ~~+60pts~~

Correct answer(s)

- Factory
- Factory method
- Fabrique
- Factory method pattern
- Factory pattern

Question 20: Design pattern 02

PHP 02:12 / 03:00 0 / 60 pts

Question

Type the name of the design pattern illustrated by the code below (1 word only):

```php
interface SequentialAccess {
    public function hasNext();
    public function next();
}

class UsersSequentialAccess implements SequentialAccess {
    private $position;
    private $users;

    public function __construct(array $users) {
        $this->users = $users;
        $this->position = 0;
    }

    public function hasNext() {
        return $this->position < count($this->users);
    }

    public function next() {
        $item = $this->users[$this->position];
        $this->position = $this->position + 1;
        return $item;
    }
}

// example of use:
$seqAccess = new UsersSequentialAccess(
    array('tom', 'bob', 'joe', 'martin')
);

while ($seqAccess->hasNext()) {
    echo $seqAccess->next() . "\n";
}
```

Answer

decorator

Result

Correct answer
Design ~~+60pts~~

Correct answer(s)

- Iterator
- Itérateur
- Iterateur

Question 21: Interface

PHP 00:19 / 00:20 20 / 20 pts

Question

What is the purpose of an interface in object-oriented programming?

Answer

- [ ] To display objects
- [x] **To describe a behavior**
- [ ] To send (or to receive) signals

Result

Correct answer
Design +20pts

Glossary

Language knowledge

This measurement gives an indication of the candidates ability to implement standard solutions to common problems. A developer with a good level of proficiency in this skill will contribute to increase the quality (maintainability, extensibility) of your applications. It does not rely specifically on technology. **This skill is particularly important if, for example, you are looking for a developer who will have to work on the architecture of your applications and to develop long-term solutions.**

Design

Measuring this skill allows us to determine the candidates level of experience in the practice of a specific programming language. **This skill is particularly important if, for example, you are looking for a developer who will have to work on the architecture of your applications and to develop long-term solutions.**

Problem solving

This skill corresponds to the candidates ability to understand and to structure his reasoning in order to find efficient solutions to complex problems. It does not rely specifically on technology. **This skill is particularly important if, for example, you are looking for R&D developers.**

Reliability

Reliability refers to the candidates ability to achieve solutions that address specific cases. Developers with a high reliability score are likely to create more robust applications (less bugs).