

Как правильно вызвать функцию func с одним параметром?

☐ call func(1);

☐ func(param=4);

☐ invoke func(3);

☐ func(2);

Какая функция в PHP возвращает имя сессии?

☐ session()

☐ session\_register()

☐ session\_name()

☐ session\_id()

Что выведет следующий код

```
<?php
$note = 3;
switch($note){
default:
    echo "c";
case 1:
    echo "a";
    break;
case 2:
    echo "b";
}
?>
```

Что напечатает следующий код

```
<?php
$mas = array('a', 1, array(1,4,6));
echo $mas[2];
?>
```

Результат выполнения кода

```
$x = 1;
for ($i=0; $i<3; $i++){
    $x += $i;
    $s = $$x;
    unset($s);
    print($s);
}
```

Чем отличаются одинарные и двойные кавычки при оформлении строковых переменных в PHP?

☐ Внутри двойных кавычек происходит подстановка значений переменных, а внутри одинарных – нет.

☐ Строки в двойных кавычках имеют нумерацию символов с 1

☐ Строки в одинарных кавычках имеют терминальный символ /0 в стиле C

☐ Внутри одинарных кавычек происходит подстановка значений переменных, а внутри двойных – нет.

☐ Они ничем не отличаются

Что будет храниться в ячейке с индексом 1 массива после выполнения данного кода

```
<?
$a[] = "a";
$a[] = "b";
$a[] = "c";
?>
```

Каким будет результат выполнения следующего фрагмента кода

```
<?php
echo floor((0.1+0.7)*10);
?>
```

Какие из перечисленных ключевых слов относятся к оператору ветвления?

☐ else

☐ if

☐ elseif

☐ then

Что выведет на экран данный скрипт?

```
<?php
function test($a) {
    return $a;
}
$sone = test(5) && false && test(6);
$two = $sone ? test(5) : test(7);
echo $two;
?>
```

Что означает константа \_\_FILE\_\_?

☐ Имя файла, куда нужно сериализовать объекты

☐ Ссылка на объект текущего файла

☐ Имя файла, в котором хранятся настройки

☐ Имя файла, в котором расположен скрипт

Результат кода

```
<?php
define('myvalue', "10");
$array[10] = "Dog";
$array[] = "Human";
$array['myvalue'] = "Cat";
$array["Dog"] = "Cat";

print "The value is: ";
print $array[myvalue]."\n";
```

Что выдает такой код?

```
$var = 5;
$i = 0;
while(--$var) {
    $i += $var--;
}
echo $i;
```

Может ли переменная менять свой тип в процессе работы программы?

☐ Да

☐ Нет

Что напечатает следующий код

```
echo(int)'1 февраля 2009 года';
```

Каким будет результат выполнения следующего кода

```
<?php
$t="next";
echo "a href=" . $t . ">";
?>
```

Для чего предназначена функция isset()

☐ Она проверяет, была ли инициализирована переменная

☐ Она проверяет, существует ли массив

☐ Она проверяет, существует ли объект

☐ Ничего из вышеперечисленного

Каков будет результат выполнения следующего кода?

```
$a = array(1=>"aaa", 2=>"aaa", 3=>"aaa");
foreach ($a as $b=>$c) {
    echo $c;
}
```

- ☐ Произойдет синтаксическая ошибка
- ☐ 0
- ☐ 1
- ☐ раздвиги
- ☐ Будет выведена пустая строка
- ☐ 123

Укажите верные способы объявления массивов

- ☐ \$b = array("a", "b", "c");
- ☐ \$b = array("a">1, "b">2, "c">3);
- ☐ \$b = array[] ["a", "b", "c"];
- ☐ \$b = new array[];
- ☐ \$b = array("a">1, "b">2, "c">3);

Какой результат выполнения следующего кода?

```
<?php
for ($i = 0; $i < 5; ++$i) {
    if ($i == 2) continue;
    print '$i';
}
?>
```

- ☐ 0124
- ☐ 2
- ☐ Ничего из вышеперечисленного
- ☐ 0134
- ☐ 012345
- ☐ 134

Какой выведется результат?

```
$str = '1234567890';
echo $str[$str[1]] - $str[$str[3]] + $str[$str[5]];
```

- ☐ синтаксическая ошибка
- ☐ 4
- ☐ 3
- ☐ 5
- ☐ 1

В каком порядке будут располагаться ключи в массиве \$a после выполнения следующего кода?

```
$a = array();
$a[1] = $a[] = $a[2] = true;
```

- ☐ 1, 2 (перезапись [2])
- ☐ 1, 0, 2
- ☐ 2, 3, 1
- ☐ 2, 0, 1
- ☐ Ошибка синтаксиса: нельзя считать из \$a[]

```
<?php
$str1 = "Hello World!";
$str2 = "str1";
echo $$str2;
```

- Какой будет результат?
- ☐ String
  - ☐ Код не скомпилируется
  - ☐ Hello World!
  - ☐ str1

Что будет напечатано этим скриптом?

```
$t = true, false;
echo $t;
```

- ☐ Возникнет ошибка
- ☐ 1
- ☐ 0
- ☐ true
- ☐ false

Какое из перечисленных выражений может быть корректным?

- ☐ \$s = fwrite("a string here");
- ☐ \$s = fwrite(\$fp, "a string here");
- ☐ \$s = fwrite("a string here", \$fp);
- ☐ Ничего из указанного

Какая из функций обработки текста не поддерживается в PHP?

- ☐ ucfirst()
- ☐ strsplit()
- ☐ strtolower()
- ☐ strtoupper()

Какая функция PHP добавляет тег <br> возле каждого символа перевода строки в некоторой переменной строкового типа?

- ☐ htmlentities()
- ☐ nl2br()
- ☐ strip\_tags()
- ☐ convert()

Каков будет массив \$b после выполнения данного кода?

```
$a = array(1, 3, 5);
$b = array(2, 4, 6);
$b += $a;
```

- ☐ Возникнет ошибка
- ☐ \$b = array(2, 4, 6)
- ☐ \$b = array(3, 7, 11)
- ☐ \$b = array(2, 4, 6, 1, 3, 5)
- ☐ \$b = array(1, 2, 3, 4, 5, 6)

Результат выполнения скрипта?

```
<?php
$a = 1;
echo "${a}";
```

- ☐ 1
- ☐ \$a
- ☐ Возникнет сообщение об ошибке
- ☐ "\${a}"

Какой из операторов PHP позволит определить остаток от целочисленного деления?

- ☐ /
- ☐ %
- ☐ \
- ☐ mod
- ☐ //

Какие из следующих синтаксических конструкций допустимы в PHP?

- ☐ @\$var
- ☐ \$0x0
- ☐ \$(0x0)
- ☐ \$var
- ☐ &\$var

Что выведет следующий код?

```
$arr = array(1,3,5);
$count = count($arr);
if ($count == 0) {
    echo "Массив пуст.";
} else {
    echo "Массив состоит из $count элементов.";
}
```

- ☐ Массив пуст.
- ☐ Массив состоит из 3 элементов.
- ☐ Массив состоит из 0 элементов.

Какой результат следующего кода?

```
$Rent = 250;
function Expenses($Other)
{
    $Rent = 250 + $Other;
    return $Rent;
}
Expenses(50);
echo $Rent;
```

- ☐ Возникнет ошибка
- ☐ 300
- ☐ 250
- ☒ 200

Что из этого необходимо соблюдать при работе с базой данных для предотвращения SQL-инъекций?

- ☐ Экранировать все специальные символы прежде чем вставлять данные в SQL-запрос
- ☐ Для хранения строчных данных в базе использовать только колонки типа VARCHAR
- ☐ Пропускать все данные из формы через функцию urlencode()
- ☐ Применять хэш SHA1 на все данные прежде чем вставлять их в базу
- ☐ Обращаться функцией addslashes() все данные, полученные из базы

Каким тегом может начинаться PHP код? укажите верные варианты

- ☐ <script language="php">
- ☐ <%
- ☐ <?php
- ☐ <\$
- ☐ <?
- ☐ все из перечисленных

С помощью, какой функции можно преобразовать текстовое представление даты на английском языке в метку времени Unix?

- ☐ strftime()
- ☐ strtotime()
- ☐ strptime()
- ☐ mktime()

Что получим в результате?

```
$a = 3;  
$b = '3';  
if ($a == $b)  
    echo 'Sa равно $b';  
else echo 'Sa не равно $b';
```

- ☐ \$a равно \$b
- ☐ 3 не равно 3
- ☐ \$a не равно \$b
- ☐ 3 равно 3
- ☐ Ошибка

Что выведет данный фрагмент кода.

```
<?php  
define('FOO', 10);  
$array = array(10 => FOO, "FOO" => 20);  
print $array[$array[FOO]] * $array["FOO"];  
>
```

- ☐ Сообщение об ошибке
- ☐ 0
- ☐ 200
- ☐ 100

Дан следующий код. Какие из этих переменных будут хранить значение "Rules"?

```
$var = "PHP";  
$$var = "Rules";
```

- ☐ \$PHP
- ☐ \$Rules
- ☐ \$"PHP"
- ☐ \$["PHP"]
- ☐ \$var

Какое значение будет выведено на экран при выполнении следующего кода:

```
<?php  
class foo {  
    private $value = 42;  
  
    public function &getValue() {  
        return $this->value;  
    }  
    public function echoValue(){  
        echo $this->value;  
    }  
}  
  
$obj = new foo;  
$obj->value = $obj->getValue();  
$obj->value = 33;  
$obj->echoValue();  
>
```

- ☐ Сообщение об ошибке
- ☐ 33
- ☐ 42
- ☐ 0

данный код выведет ошибку. В какой строке.

```
<?php  
abstract class B {  
    public $var = "B";  
  
    function __construct(){  
        $this->var = "B";  
    }  
}  
  
class E extends B {  
    $var2 = new E();  
    function __construct() {  
        $this->var = "E";  
    }  
  
    function __clone() {  
        $this->var = "C";  
    }  
}  
  
$var1 = new B();  
$var2 = new E();  
$var3 = $var1;  
$var4 = clone($var2);  
$var5 => $var4->var;  
$var2->var = "A";  
  
echo $var1 = new B();  
echo $var5 => $var4->var;  
echo $var4 = clone($var2);  
function __clone() {  
    echo $var3 = $var1;
```

Вам необходимо установить следующие области видимости для атрибутов класса:  
1) Базе  
2) Потомки, предки, и текущий класс;  
3) Только текущий класс

Какие ключевые слова нужно использовать в каждом из этих трёх случаев?

- ☐ 1) private  
2) public  
3) protected
- ☐ 1) private  
2) protected  
3) public
- ☐ 1) public  
2) private  
3) protected
- ☐ 1) public  
2) protected  
3) private
- ☐ 1) protected  
2) private  
3) public

Чем отличается абстрактный класс от интерфейса?

- ☐ Интерфейс не может содержать реализации
- ☐ Абстрактный класс не может содержать реализации
- ☐ Класс, реализующий интерфейс, обязан предоставить имплементацию всех методов этого интерфейса
- ☐ Ничем не отличаются

Что напечатает следующий код:

```
<?php  
class Class {  
    public $value;  
}  
  
$o = new Class;  
$o->newValue = 1;  
  
$a = $o;  
$a->newValue = 2;  
  
echo $o->newValue;
```

- ☐ 1
- ☐ 2
- ☐ Ошибка, так как newValue не определена
- ☐ Ничего не напечатает

```
interface Singleton {  
    public static function getInstance();  
}  
  
class singleton implements Singleton {  
    private static $instance = false;  
    final public static function getInstance() {  
        return (self::$instance) ? self::$instance : self::$instance = new self;  
    }  
  
    private function __construct(){}  
    private function __clone(){}  
    function f1(){  
        echo 'Singleton::f1()';  
    }  
}  
  
class c1 extends singleton{  
    function f1(){  
        echo 'c1::f1()';  
    }  
}  
  
class c2 extends singleton{  
    function f1(){  
        echo 'c2::f1()';  
    }  
}  
  
c1::getInstance()->f1();  
c2::getInstance()->f1();  
  
echo c2::f1(x2::f1());  
echo c1::f1(x2::f1());  
echo Singleton::f1(Singleton::f1());  
echo c1::f1(x1::f1());  
echo Fatal Error  
echo Notice
```

**Примечание [A1]:** Пояснение:  
Позднее статическое связывание в паттерне "Одиночка" через перенаправленный вызов self. self использует класс, к которому она принадлежит, в данном случае это class singleton. getInstance() всегда возвращает единственный и тот же самый экземпляр объекта. Первым была вызвана инициализация объекта c1, все последующие инициализации будут возвращать первый инициализированный объект

Какие из утверждений являются верными относительно интерфейсов (PHP 5.4+)?

- ☐ Абстрактный класс может реализовывать несколько интерфейсов
- ☐ Интерфейс может расширять несколько интерфейсов
- ☐ Методы с одинаковыми именами и сигнатурами могут существовать в нескольких интерфейсах, которые реализует данный класс
- ☐ Класс может реализовывать несколько интерфейсов

Что выведет данный скрипт в результате своей работы?

```
<?
class MyClass {
    public $myVar;
    function __constructor() {
        $this->myVar = 5;
    }
}

$a = new MyClass;
echo $a->myVar;
?>
```

- ☐ 5
- ☐ 0
- ☐ Ничего не выведет
- ☐ Выведет ошибку

Что выведет данный код?

```
<?php
1. class Fruit
2. {
3.     protected $type = ' ';
4.     protected $color = 'green';
5.     public function getType()
6.     {
7.         return $this->type;
8.     }
9. }
10.
11. final public function getColor()
12. {
13.     return $this->color;
14. }
15. }
16.
17. final class Apple extends Fruit {
18.     protected $type = ' ';
19.     protected $color = 'red';
20. }
21. final public function getColor()
22. {
23.     return "Apple:".$this->color;
24. }
25. }
26.
27. $generalFruit = new Fruit();
28. $apple = new Apple();
29. echo $generalFruit->getType()." ".$generalFruit->getColor();
30. -. $apple->getType()." ".$apple->getColor();
```

- ☐ Fruit Green Apple Solid Red
- ☐ Ошибка в строке 28
- ☐ Fruit Green Apple Green
- ☐ Ошибка в строке 23
- ☐ Ошибка в строке 21

В какой строке кода допущена ошибка.

```
<?php
2: class dog {
3:     public function bark() {
4:         echo "Woof!";
5:     }
6: };
7:
8: $foo = new dog;
9: $foo->bark();
?>
```

- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 6
- ☐ 8
- ☐ 9

В этом коде нет ошибок.

Что из следующих утверждений верно относительно ключевых слов final и abstract?

- ☐ Абстрактный класс не может иметь final методов
- ☐ final класс не может иметь final методов
- ☐ final класс не может иметь абстрактных методов
- ☐ Абстрактный класс не может иметь не абстрактные методы

Что выведет следующий код?

```
<?php
class People {
    public function greeting() {
        echo "A nice smile for you. ";
    }
}

trait HyParents {
    public function greeting() {
        parent::greeting();
        $this->greeting();
        echo "A big hug for you. ";
    }
}

class Hyson extends People {
    use HyParents;
    public function greeting() {
        echo "A big kiss for you. ";
    }
}

class Hydad extends People {
    use HyParents;
    public function greeting() {
        echo "A strong handshake for you. ";
    }
}

$person = new Hyson();
$person->greeting();
?>
```

- ☐ A strong handshake for you. A big hug for you.
- ☐ A nice smile for you. A big hug for you. A strong handshake for you.
- ☐ A nice smile for you. A big hug for you. A strong handshake for you.
- ☐ A nice smile for you. A big kiss for you. A big hug for you.
- ☐ A big kiss for you. A big hug for you. A strong handshake for you.

В чем отличие записей Class::method() и \$Class->method() ?

- ☐ Первая запись неверна для php
- ☐ Вторая запись неверна для php
- ☐ Вызов статического и динамического метода соответственно
- ☐ Ни в чем

Какой фрагмент вы используете, чтобы создать экземпляр объекта?

- ☐ Object = new Object();
- ☐ Object = Object(new);
- ☐ \$Object: new Object();
- ☐ Object&=Object();
- ☐ Object->new Object();

Что будет находиться в свойстве \$css объекта \$c?

```
class c {
    public $css;
    function make_bold(){
        $this->css .= 'bold';
        return $this;
    }
    function make_italic(){
        $this->css .= 'italic';
        return $this;
    }
}

$c = new c();
$c->make_bold()->make_italic();
echo $c->css;
```

- ☐ ничего, код вызовет ошибку
- ☐ bold
- ☐ italic
- ☐ bolditalic

**Примечание [A2]:** Пояснение: getColor() уже объявлен финальной функцией в классе Fruit, поэтому в Apple её нельзя переопределить

Что будет содержать переменная \$c после выполнения следующего кода?

```
$a = ['a', 'b', 3 => 'c', 'd'];  
$b = ['e', 'd', 'c', 'b'];  
  
$c = array_values($a + $b);
```

Ответ:

- ☐ ['a', 'b', 'c', 'd', 'c']
- ☐ ['a', 'b', 'c', 'd']
- ☐ ['a', 'b', 'd']
- ☐ Код завершится с ошибкой из-за неподдерживаемой операции

Какие из функций позволяют получить целиком содержимое файла?

Ответ:

- ☐ fgets
- ☐ file\_get\_contents
- ☐ fread
- ☐ file

Какой стандартной функцией PHP нужно заполнить пропуск, чтобы получить указанный результат?

```
$s = 'abc def ghijkl!';  
$s2 = ____($s);  
  
// $s2 === 'Abc def ghijkl!'
```

Ответ:

- ☐ ucfirst
- ☐ strtoupper
- ☐ ucwords
- ☐ wordwrap

Сколько экземпляров объекта создастся в памяти при интерпретации в PHP следующей строки кода?

```
$object = new ClassName();  
  
$newObject = $object;
```

Ответ:

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ Ни одного

Какие из перечисленных возможностей не поддерживаются PHP?

Ответ:

- ☐ Абстрактные классы
- ☐ Интерфейсы
- ☐ Публичные и приватные методы
- ☐ Множественное наследование

Какой стандартной функцией PHP можно получить массив букв ['b', 'c', 'd', 'e']?

Ответ:

- ☐ range
- ☐ array\_range
- ☐ array\_combine
- ☐ fill

Какой из режимов функции fopen позволяет открыть файл для записи и помещает указатель на конец файла?

Ответ:

- ☐ w
- ☐ a
- ☐ w+
- ☐ r+

Зачем используют Round Robin DNS?

Ответ:

- ☐ Чтобы снизить суммарную нагрузку на backend-сервера
- ☐ Чтобы клиенты быстрее получали нужный IP адрес backend-сервера
- ☐ Чтобы равномерно распределять нагрузку на backend-сервера
- ☐ Чтобы защищаться от возможных атак хакеров

Каким будет результат выполнения кода?

```
$num = [11, 12, 13];  
  
foreach ($num as $val) {  
    $val++;  
}  
  
print_r($num);
```

Ответ:

- ☐ Array(12, 13, 14)
- ☐ Array(11, 12, 13)
- ☐ Зависит от версии PHP, в версиях 5.3 и 5.5 результат будет разным.
- ☐ Сообщение об ошибке

Какой ассоциативный массив содержит данные, переданные следующей формой?

```
<form action="script.php">  
    <input type="text" name="field" />  
</form>
```

Ответ:

- ☐ \$\_GET
- ☐ \$\_POST
- ☐ \$\_COOKIE
- ☐ \$\_REQUEST

Каким будет результат выполнения кода?

```
trait Foo {  
    private $name = 'Foo';  
}  
  
class Bar {  
    use Foo;  
    private $name = 'Bar';  
  
    public function getName() {  
        echo $this->name;  
    }  
}
```

Ответ:

- ☐ Foo
- ☐ Bar
- ☐ Сообщение об ошибке

Что удаляет стандартная функция delete в PHP?

Ответ:

- ☐ Файлы
- ☐ Директории
- ☐ Файлы и директории
- ☐ Не существует такой стандартной функции

Что такое машина Тьюринга?

Ответ:

- ☐ Название первой ЭВМ
- ☐ Простейшая вычислительная машина
- ☐ Механический калькулятор
- ☐ Паровой двигатель

Необходимо спроектировать полноценную базу данных для хранения и обработки трех сущностей — книги, авторы книг, магазины, в которых продаются книги. Сколько таблиц должно быть в такой базе данных, чтобы она соответствовала 3-ей нормальной форме, если в качестве СУБД используется MySQL?

- Ответ:
- ☐ 3
  - ☐ 4
  - ☐ 5
  - ☐ 6

Каким будет результат выполнения кода?

```
namespace Foo;

class Bar {
    private function baz() {}
}

$class = new ReflectionClass('Foo\Bar');
$method = $class->getMethod('baz');
echo $method->getName();
```

Ответ:

- ☐ baz
- ☐ Пустая строка.
- ☐ Сообщение об ошибке.

Файл `some_file.php` существует. Каким будет результат выполнения кода, если этот файл пустой?

```
echo include 'some_file.php';
```

- Ответ:
- ☐ 0
  - ☐ 1
  - ☐ Сообщение об ошибке
  - ☐ true

В `memcache` по ключу "k" хранится "42". После вызова какой функции значение в "k" станет равно "13"?

- Ответ:
- ☐ `memcache_add("k", "13")`
  - ☐ `memcache_increment("k", "13")`
  - ☐ `memcache_set("k", "13")`
  - ☐ `memcache_get("k", "13")`

В чем отличие методов POST и GET?

- Ответ:
- ☐ В GET данные передаются в заголовке запроса, а в POST — в теле
  - ☐ В GET данные передаются в адресной строке, а в POST — через шифрованный тоннель
  - ☐ GET-данные передаются вместе с запросом, а POST — после него
  - ☐ Данные, передаваемые по GET, всегда в кодировке UTF8, а по POST — в любой

В чем отличие записей `Class::method()` и `$Class->method()` ?

- Ответ:
- ☐ Первая запись не соответствует синтаксису PHP
  - ☐ Первую запись можно использовать только в PHP5
  - ☐ Вызов статического и динамического метода, соответственно
  - ☐ Вторую запись можно использовать, только если `$Class` — объект

С помощью какой стандартной команды Linux можно посмотреть занятый и доступный объемы HDD?

- Ответ:
- ☐ `du`
  - ☐ `df`
  - ☐ `da`
  - ☐ `dd`

[PHP 5.6, 7.0] Как правильно передать переменную `$a` в функцию, чтобы получить указанный результат?

```
$a = [[1, 2], [3, 4], [5, 6, 7]];
$r = array_merge(____);

// $r == [1, 2, 3, 4, 5, 6, 7]
```

- Ответ:
- ☐ `...$a`
  - ☐ `$a...`
  - ☐ `$a[...]`
  - ☐ `$a[]`

Какой будет результат выполнения кода?

```
abstract class Checker {
    abstract public function check(array $params = null);
}

class Counter extends Checker {
    public function check(array $params) {
        echo count($params);
    }
}

$c = new Counter();
$c->check([5]);
```

- Ответ:
- ☐ 0
  - ☐ Fatal error
  - ☐ 5
  - ☐ 1

С помощью какой стандартной команды Linux можно создать ссылку на файл?

- Ответ:
- ☐ `ls`
  - ☐ `ln`
  - ☐ `mklink`
  - ☐ `anch`

Какие утверждения корректны для таблиц InnoDB в MySQL?

- Ответ:
- ☐ Поддерживают транзакции
  - ☐ Позволяют хранить таблицы, размером превышающим размер файла в операционной системе
  - ☐ Каждая таблица хранится в отдельном файле
  - ☐ Не поддерживает внешние ключи

Какой тип данных возвращает функция `array_unique()` ?

- Ответ:
- ☐ integer
  - ☐ array
  - ☐ boolean
  - ☐ mixed

Какой оператор необходимо использовать в запросе, чтобы получить результат?

Table1		Table2	
Key	Value	Key	Value
1	One	2	Red
2	Two	3	White
3	Three	4	Black

Результат:

2	Two	Red
3	Three	White

- Ответ:
- ☐ `join`
  - ☐ `right join`
  - ☐ `left outer join`
  - ☐ `union`

Какую функцию следует использовать для определения последовательности вызова функций?

- Ответ:
- ☐ `debug_backtrace`
  - ☐ `debug_print_backtrace`
  - ☐ `var_dump`
  - ☐ `trigger_error`

Какой из перечисленных стандартных функций PHP должна быть функция \$func, чтобы дать показанный ниже результат \$r?

```
$a = ['a', 'b', 3 => 'c', 'd'];  
$b = ['e', 'd', 'b', 'c'];  
  
$r = $func($a, $b);  
  
// array_values($r) == ['c']
```

Ответ:

- ☐ array\_intersect\_assoc
- ☐ array\_intersect
- ☐ array\_intersect\_key
- ☐ Ни одной из перечисленных

Какой из следующих сетевых протоколов является устаревшим?

Ответ:

- ☐ IMAP
- ☐ UDP
- ☐ IPX
- ☐ FTP

В memcache по ключу "k" нет данных. После вызова какой функции значение в "k" станет равно "13"?

Ответ:

- ☐ memcache\_add("k", "13")
- ☐ memcache\_increment("k", "13")
- ☐ memcache\_set("k", "13")
- ☐ memcache\_get("k", "13")

С помощью какой стандартной команды Linux можно изменить права на доступ к файлу?

Ответ:

- ☐ chown
- ☐ chroot
- ☐ chattr
- ☐ chmod

С помощью какой команды можно посмотреть план выполнения запроса в MySQL?

Ответ:

- ☐ explain
- ☐ explain analyze
- ☐ analyze
- ☐ С помощью всех вышеперечисленных

Что будет содержать переменная \$r после выполнения следующего кода?

```
$a = ['a' => 2, 'b' => 3];  
  
$r = array_combine(  
    array_keys($a),  
    array_fill(0, count($a), [])  
);
```

Ответ:

- ☐ ['a' => [], 'b' => []]
- ☐ ['a' => 'b', 'b' => 'a']
- ☐ ['a', 'b', [], [], []]
- ☐ ['a', 2, 'b', 3]

Как следует заполнить пробелы (\_\_\_), чтобы получить в результате "53"?

```
namespace Foo;  
function strlen($string) {  
    return 5;  
}
```

```
namespace Bar;  
use Foo;  
  
echo ___('foo'), ___('bar');
```

Ответ:

- ☐ strlen, strlen
- ☐ strlen, strlen
- ☐ Foostrlen, strlen
- ☐ Foostrlen, strlen

Чем отличаются одинарные и двойные кавычки при оформлении строковых переменных в PHP?

Ответ:

- ☐ Внутри одинарных кавычек происходит подстановка значений переменных, а внутри двойных – нет.
- ☐ Внутри двойных кавычек происходит подстановка значений переменных, а внутри одинарных – нет.
- ☐ Строки в одинарных кавычках имеют терминальный символ /0 в стиле C.
- ☐ Строки в двойных кавычках имеют нумерацию символов с 1

Что будет выведено после выполнения следующего кода?

```
$r = array_reverse([10 => 1, 20 => 2, 30 => 3]);  
  
switch (true) {  
    case $r === [3, 2, 1]: echo "a"; break;  
    case $r === [10, 20, 30]: echo "b"; break;  
    case $r === [1 => 10, 2 => 20, 3 => 30]: echo "c"; break;  
    default: echo "d";  
}
```

Ответ:

- ☐ a
- ☐ b
- ☐ c
- ☐ d

[PHP 5.6, 7.0] Что будет содержать переменная \$r после выполнения следующего кода?

```
$a = ['a' => true, 'f' => 0, 's' => null];  
  
$r = $a['s'] ?? $a['f'] ?? 13;
```

Ответ:

- ☐ 0
- ☐ 13
- ☐ null
- ☐ false

Что необходимо делать каждый раз перед началом использования постоянного соединения к MySQL?

Ответ:

- ☐ ROLLBACK
- ☐ COMMIT
- ☐ Снимать логи с таблиц
- ☐ fork, затем продолжать выполнение в форке