

PHP Mini Test

Date:
Name:

1. What is the best way to ensure the safety of the data displayed on HTML page ? (e.g. preventing from XSS attack). (choose 2)

1. Enable built-in security features such as magic_quotes_gpc and safe_mode
2. htmlspecialchars()
3. Use PHP 5's tainted mode
4. strip_tags()
5. urlencode(), urldecode()

2. What would you replace ?????? with, below, to make the string Hello, World! be displayed?

```
<?php
function myfunction() {
    ???????
    print $string;
}

myfunction("Hello, World!");
?>
```

1. There is no way to do this
2. \$string = \$argv[1];
3. \$string = \$_ARGV[0];
4. list(\$string) = func_get_args();
5. \$string = get_function_args()

3. Prepared statements have which of the following advantages over traditional SQL statements? (choose 3)

1. They use less bandwidth
2. They prevent syntax errors
3. They prevent SQL injection
4. They are faster when performing the same query multiple times
5. They allow you to use transactions

Example of prepared statement:

```
$preparedStatement = $db->prepare('SELECT * FROM employees WHERE name = :name');
$preparedStatement->execute(array(':name' => $name));
$rows = $preparedStatement->fetchAll();
```

4. What combination of boolean values for \$a, \$b, \$c, and \$d will result in the variable \$number being equal to 3?

```
<?php
$a = null;
$b = null;
$c = null;
$d = null;

if($a && !$b) {
    if(!$c && !$d) {
        if($d && ($a || $c)) {
            if(!$d && $b) {
                $number = 1;
            } else {
                $number = 2;
            }
        } else {
            $number = 3;
        }
    } else {
        $number = 4;
    }
} else {
    $number = 5;
}
?>
```

1. false, true, true, true
2. true, false, true, false
3. true, true, false, false
4. false, true, true, false
5. false, false, true, false

5. Given the two values below, which of the following possibilities will print "10 foos20 bars"? (choose 2)

```
<?php
$var1 = "10 foos";
$var2 = "20 bars";

print ??????;

?>
```

1. implode("", array(\$var1,\$var2));
2. \$var1 . \$var2
3. \$var1 + \$var2
4. None of the above
5. All of the above

6. What is the output of the following?

```
<?php
function byRef(&$apples)
{
    $apples++;
}
$oranges = 5;
$apples = 5;
byRef($oranges);
echo "I have $apples apples and $oranges oranges";
?>
```

1. I have 6 apples and 6 oranges
 2. I have 6 apples and 5 oranges
 3. I have 5 apples and 6 oranges
 4. I have 5 apples and 5 oranges
-

7. What is the output of?

```
<?php
function apple($apples = 4){
    $apples = $apples / 2;
    return $apples;
}
$apples = 10;
apple($apples);
echo $apples;
?>
```

- a. 2
 - b. 4
 - c. 5
 - d. 10
-

8. In PHP 5's object model, a class can have multiple _____ but only a single direct _____.

1. interfaces, child
 2. children, interface
 3. interfaces, parent
 4. parents, interface
 5. None of the above
-

9. What is the output of the following script?

```
<?php
class A {
    public function bar() {
        print "Hello";
    }
}
class B extends A {
    function bar() {
        print "Goodbye";
    }
}
$c = new B();
$c->bar();
?>
```

Your answer:

10. What is the value of \$d ?

```
$a = "post_processed_string";
$b = array("post_", "_");
$c = array("", " ");
$d = ucwords(str_replace($b,$c,$a));
```

Your answer:

Answers:

1. -> 2, 4

2. -> 4

3. -> 2,3,4

4. -> 2

5. -> 1,2

Optional question: what will be result of 3. option ? (30 = 10+20 – type casting string to interger)

6. -> 3

7. -> d

8. -> 3

9. -> Goodbye

10. -> Processed string