

1. Ի՞նչի համար է օգտագործվում PHP լեզուն:

1. Մերվերի և տվյալների հենքերի հետ աշխատելու համար:
2. Մերվերի վրա աշխատելու համար:
3. Տվյալների հենքերի հետ աշխատելու համար:

2. Ի՞նչ է MySQL- ն:

1. Ծրագրավորման լեզու է տվյալների հենքերի հետ աշխատելու համար:
2. Տվյալների հենքերի կառավարման համակարգ է:
3. PHP- ի և տվյալների հենքերի տեխնոլոգիա է:

3. Ո՞ր նիշով է ներկայացվում փոփոխականը:

1. \$
2. &
3. #

4. Ինչո՞վ են հաստատունները տարբերվում փոփոխականներից:

1. Տարբերություններ չկան:
2. Փոփոխականները օգտագործվում են միայն ֆունկցիաներում:
3. Հաստատունները չեն կարող վերասահմանվել:

5. Ի՞նչ տրամաբանական գործողություններ կան:

1. >, <, ==, !=, <=, >=, ===
2. >, <, ==, !=, <=, >=
3. >, <, ==, !=, <=, >=, ===

6. Որտե՞ղ է նշված ճիշտ արտածման օպերատորը:

1. show " ";
2. echo " ";
3. message " ";

7. Ի՞նչ է էկվիվալենտության օպերատորը:

1. Այն համեմատում է փոփոխականների համապատասխանությունը նաև ըստ տիպերի:
2. Նման օպերատոր գոյություն չունի:
3. Այն համեմատում է փոփոխականների համապատասխանությունը, բայց տիպերը հաշվի չի առնում:

8. Որտե՞ղ է ճիշտ տրված PHP- ի մեկնաբանությունը:

1. #
2. /* /*
3. <!-- -->

9. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
$x = 1;  
$y = 5;  
$z = 2.1;  
echo $z != $y && $x < $y;
```

1. "false"
2. 0
3. 1

10. Ի՞նչ թվաբանական գործողություններ կան PHP- ում:

1. +, -, *, /, %
2. +, -, *, /
3. +, -, *, /, ^

11. Որտե՞ղ է PHP կոդը ճիշտ բացված և փակված:

1. <?php ?>
2. <php ?>
3. <? php>

12. Ի՞նչ տիպի փոփոխականներ կան PHP-ում:

1. int, float, string, char, bool
2. int, float, string, char
3. int, float, string, bool

13. Ինչպե՞ս արտածել \$arr զանգվածի բոլոր տարրերը:

1. Դա հնարավոր է անել միայն ցիկլի միջոցով:

2. `echo $arr;`

3. `print_r($arr);`

14. Ո՞ր օպերատորը PHP-ում գոյություն չունի:

1. `+=`

2. `!==`

3. `!==`

15. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
echo "1" + "2";
?>
```

1. 12

2. Շարահյուսական սխալ:

3. 3

16. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
function myfunc(&$a) {
    $a++;
}
$b = 5;
myfunc($b);
echo $b;
?>
```

1. 6

2. Սխալ, ֆունկցիայի մեջ փոփոխականի անվան դիմաց՝ լրացուցիչ «&» նիշի պատճառով:

3. 5

17. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
function myfunc($x = 0) {
    echo $x;
}
```

```
myfunc();  
myfunc(5);  
?>
```

1. 0
2. 5
3. 05

18. Ի՞նչ կարտաձվի տրված կոդի կատարման արդյունքում:

```
<?php  
$x = 5;  
echo "Փոփոխական $x = 5";  
?>
```

1. Փոփոխական x = 5
2. Փոփոխական \$x = 5
3. Փոփոխական 5 = 5

19. Տրված է հետևյալ տողը. \$str = "123456": Ինչպես այս տողից ստանալ "5" ենթատողը:

1. \$str[5];
2. \$str[4];
3. \$str[6];

20. PHP-ի որ ֆունկցիան է թույլ տալիս գտնել ենթատողի դիրքը տրված տողում:

1. substr()
2. strpos()
3. str_repeat()

21. PHP-ի որ ֆունկցիան է թույլ տալիս գտնել ենթատողի դիրքը տրված տողում, եթե տողի պարունակությունը հայերեն տեքստ է:

1. mb_strpos()
2. strpos()
3. str_repeat()

22. Ի՞նչ կարտաձվի տրված կոդի կատարման արդյունքում:

```
<?php  
$mystring = 'PHP: Hypertext Preprocessor ';
```

```

$findme = 'Pr';
$pos = strpos($mystring, $findme);

if ($pos === false) {
    echo " '$findme' տողը չի պարունակվում '$mystring' տողում";
} else {
    echo " '$findme' տողը պարունակվում է '$mystring' տողի";
    echo " $pos դիրքում ";?>

```

1. Ոչինչ չի արտածվի:
2. 'Pr' տողը պարունակվում է 'PHP: Hypertext Preprocessor ' տողի 18 դիրքում
3. 'Pr' տողը պարունակվում է 'PHP: Hypertext Preprocessor ' տողի 15 դիրքում

23. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```

<?php
    $mystring = 'PHP: Hypertext Preprocessor ';
    $findme = 'PH';
    $pos = strpos($mystring, $findme);

    if ($pos == false) {
        echo " '$findme' տողը չի պարունակվում '$mystring' տողում";
    } else {
        echo " '$findme' տողը պարունակվում է '$mystring' տողի";
        echo " $pos դիրքում ";
    }
?>

```

1. Ոչինչ չի արտածվի:
2. 'PH' տողը չի պարունակվում 'PHP: Hypertext Preprocessor ' տողում
3. 'PH' տողը պարունակվում է 'PHP: Hypertext Preprocessor ' տողի 0 դիրքում

24. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```

<?php
    $mystring = 'PHP: Հիպերտեքստային Պրեպրոցեսսոր';
    $findme = 'Պր';
    $pos = strpos($mystring, $findme);

    if ($pos === false) {
        echo " '$findme' տողը չի պարունակվում '$mystring' տողում";
    } else {
        echo " '$findme' տողը պարունակվում է '$mystring' տողի";
        echo " $pos դիրքում ";
    }
?>

```

1. 'Պր' տողը չի պարունակվում 'PHP: Հիպերտեքստային Պրեպրոցեսսոր ' տողում:

2. 'Պր' տողը պարունակվում է 'PHP: Հիպերտեքստային Պրեպրոցեսոր' տողի 34 դիրքում
3. 'Պր' տողը պարունակվում է 'PHP: Հիպերտեքստային Պրեպրոցեսոր' տողի 20 դիրքում

25. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
$mystring = 'PHP: Հիպերտեքստային Պրեպրոցեսոր';
$findme = 'Պր';
$pos = mb_strpos($mystring, $findme);

if ($pos === false) {
    echo " '$findme' տողը չի պարունակվում '$mystring' տողում";
} else {
    echo " '$findme' տողը պարունակվում է '$mystring' տողի";
    echo " $pos դիրքում ";
}
?>
```

1. ' Պր ' տողը չի պարունակվում 'PHP: Հիպերտեքստային Պրեպրոցեսոր' տողում
2. 'Պր' տողը պարունակվում է 'PHP: Հիպերտեքստային Պրեպրոցեսոր' տողի 34 դիրքում
3. 'Պր' տողը պարունակվում է 'PHP: Հիպերտեքստային Պրեպրոցեսոր' տողի 20 դիրքում

26. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
echo strpos('abcde abcde', 'd', 5);
?>
```

1. bool(false)
2. 9
3. 3

27. PHP-ի որ ֆունկցիան է թույլ տալիս գտնել տրված տողում ենթատողի վերջին մուտքի դիրքը:

4. substr()
5. strpos()
6. strrpos()

28. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
```

```
$str = "0123456789a123456789b";  
var_dump(strrpos($str, '6', -6));  
?>
```

1. **int(6)**
2. 16
3. 4

29. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php  
$str = "0123456789a123456789b";  
var_dump(strrpos($str, '6'));  
?>
```

1. **int(16)**
2. int(6)
3. 6

30. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php  
$str = "0123456789a123456789b";  
var_dump(strrpos($str, '6', -16));  
?>
```

1. int(16)
2. int(6)
3. **bool(false)**

31. PHP-ի որ ֆունկցիան է թույլ տալիս գտնել տրված տողում ենթատողի վերջին մուտքի դիրքը և զգայուն չի ռեգիստրի նկատմամբ:

1. substr()
2. strpos()
3. **stripos()**

32. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php  
$haystack = 'abcdabcdabcd';  
$needle = 'cDa';  
$pos = strrpos($haystack, $needle);  
  
if ($pos === false) {
```

```

        echo "($needle) չի հայտնաբերվել ($haystack)-ում";
    } else {
        echo " ($needle)-ի վերջին մուտքը ($haystack) -ի ($pos) դիրքում է";
    }
?>

```

1. (cDa)-ի վերջին մուտքը (abcdabcdabcd) -ի (6) դիրքում է
- 2.(cDa)-ի վերջին մուտքը (abcdabcdabcd) -ի (2) դիրքում է
- 3.(cDa) չի հայտնաբերվել (abcdabcdabcd)-ում

33. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```

<?php
    $haystack = 'աբգդեզաբգդեզ';
    $needle   = 'Եզ';
    $pos      = strrpos($haystack, $needle);

    if ($pos === false) {
        echo "($needle) չի հայտնաբերվել ($haystack)-ում";
    } else {
        echo " ($needle)-ի վերջին մուտքը ($haystack) -ի ($pos) դիրքում է";
    }
?>

```

1. (Եզ)-ի վերջին մուտքը (աբգդեզաբգդեզ) -ի (10) դիրքում է
- 2.(Եզ) չի հայտնաբերվել (աբգդեզաբգդեզ)-ում
- 3.(Եզ)-ի վերջին մուտքը (աբգդեզաբգդեզ) -ի (5) դիրքում է

34. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```

<?php
    $haystack = 'աբգդեզաբգդեզ';
    $needle   = 'Եզ';
    $pos      = mb_strrpos($haystack, $needle);

    if ($pos === false) {
        echo "($needle) չի հայտնաբերվել ($haystack)-ում";
    } else {
        echo " ($needle)-ի վերջին մուտքը ($haystack) -ի ($pos) դիրքում է";
    }
?>

```


1. (Եզ) չի հայտնաբերվել (արգդեզարգդեզ)-ում
- 2.(Եզ)-ի վերջին մուտքը (արգդեզարգդեզ) -ի (5) դիրքում է
- 3.(Եզ)-ի վերջին մուտքը (արգդեզարգդեզ) -ի (10) դիրքում է

**35. PHP-ի որ ֆունկցիան է տողի սկզբից և վերջից հեռացնում բացատները, ինչպես նաև
 "\n", "\r", "\t", "\0", "\x0B" ղեկավարման պայմանանշանները:**

1. trim()
- 2.strpos()
- 3.stripos()

36. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
    $text = "\tՀայաստան ";
    $text1 = trim($text);
    var_dump($text);

?>
```

1. string(22) "\tՀայաստան"
- 2.string(22) "Հայաստան"
- 3.bool(false)

37. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
    $text = "Hello World";
    $text1 = trim($text, "Hdle");
    var_dump($text1);

?>
```

1. string(5) "Hello W"
- 2.string(5) "World"
- 3.string(5) "o Wor"

38. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
    $text = "\x09Հայաստան\x0AԵրևան\x1B";
    $text1 = trim($text, "\x00..\x1F");
    var_dump($text1);

?>
```

1. `string(27) "Հայաստան\x0Երևան"`
2. `string(27) "\x00Հայաստան Երևան\x1F"`
3. `string(27) "Հայաստան Երևան"`

39. PHP-ի որ ֆունկցիան է տողը փոխում ներքին ռեգիստրի:

1. `strtolower();`
2. `strtoupper()`
3. `strhome()`

40. PHP-ի որ ֆունկցիան է տողը փոխում վերին ռեգիստրի:

1. `strtolower();`
2. `strtoupper()`
3. `strhome()`

41. PHP-ի որ ֆունկցիան տողը չի փոխում վերին ռեգիստրի:

1. `strtoupper()`
2. `mb_strtoupper()`
3. `mb_strtolower()`

42. PHP-ի որ ֆունկցիան տողի յուրաքանչյուր բառի առաջին սիմվոլը դարձնում է մեծատառ:

1. `strtoupper()`
2. `ucwords()`
3. `trim()`

43. Ի՞նչ կարտաձվի տրված կոդի կատարման արդյունքում:

```
<?php
    echo mb_strtolower('ԱԲԳԴԵ');
?>
```

1. `աբգդե`

2.ԱԲԳԴԵ

3.Աբգդե

44. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
    $bar = 'ARAM KARAPETYAN';
    echo ucwords(strtolower($bar));
?>
```

1. ARAM KARAPETYAN

2.Aram karapetyan

3.Aram Karapetyan

45. PHP-ի որ ֆունկցիան կարող է տողից ստանա դրա որոշակի մասը:

1. strtoupper()

2.substr()

3 strtolower()

46. PHP-ի որ ֆունկցիան է տողում պահանջվող ենթատողերը փոխարինում տրված ենթատողով, հաշվի առնելով ռեգիստրը:

1. strtoupper()

2.str_ireplace()

3.str_replace()

47. PHP-ի որ ֆունկցիան է տողում պահանջվող ենթատողերը փոխարինում տրված ենթատողով, առանց հաշվի առնելու ռեգիստրը:

1. strtoupper()

2.str_ireplace()

3.str_replace()

48. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
    $bar = 'ARAM KARAPETYAN';
    echo str_replace('A', '!', $bar);
?>
```

1. !RAM K!RAPETYAN
- 2.RM KRPETYN
3. !R!M K!R!PETY!N

49. PHP-ի որ ֆունկցիան է որպես փաստարկ վերցնում ASCII կոդի նիշը և վերադարձնում է այս կոդին համապատասխանող իրական նիշը:

1. chr()
2. ord()
3. trim()

50. PHP-ի որ ֆունկցիան է նիշը փոխարինում համապատասխանող ASCII կոդով:

1. chr()
2. ord()
3. trim()

51. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
    $armenian = "ԱՐՄԵՆ";
    echo $armenian[0]. '<br />';
    echo $armenian[1]. '<br />';
    echo $armenian[2]. '<br />';
    echo $armenian[3];
?>
```

1. ԱՐՄԵ
2. ԱՐՄԵՆ
- 3.



52. Որտե՞ղ է սխալ մեկնաբանվում UTF-8 կոդավորումը:

1. Այսօր բոլոր նոր ծրագրերը կենտրոնանում են UTF-8 կոդավորման վրա, որը դե ֆակտո ստանդարտ է աշխարհի բոլոր լեզուների կոդավորման համար:
2. UTF-8- ին անցումը հնարավորություն է տալիս մեկ փաստաթղթում տարբեր լեզուներով տեքստերի օգտագործել, բայց թույլ չի տալիս ընդլայնել կոդավորումը, երբ հայտնվում են նոր նիշեր:

3.PHP 7-ում UTF-8 կոդավորման նիշերը կարող են նշվել հատուկ շարահյուսության միջոցով \ u հաջորդականությանը հաջորդում է ձևավոր փակագծերում նիշի տասնվեցական կոդը:

53. Որտե՞ղ է սխալ մեկնաբանվում UTF-8 կոդավորումը:

1. UTF-8 -ում անգլերեն նիշերը զբաղեցնում են մեկ բայթ:
- 2.UTF-8 -ում հայերեն նիշերը զբաղեցնում են չորս բայթ:
- 3.mbstring ընդլայնումը ապահովում է բազմաբայթային կոդավորումը:

54. Ի՞նչ կարտաձվի տրված կոդի կատարման արդյունքում:

```
<?php
    $english = "Program";
    $armenian = "Ծրագիր";
    print "strlen() = " . strlen($english) . " $english -ի համար և " .
    strlen($armenian) . " $armenian -ի համար <br/>";
?>
```

1. strlen() = 7 Program -ի համար և 12 Ծրագիր -ի համար
- 2.strlen() = 7 Program -ի համար և 6 Ծրագիր -ի համար
- 3.Error

55. Ի՞նչ կարտաձվի տրված կոդի կատարման արդյունքում:

```
<?php
    $english = "Program";
    $armenian = "Ծրագիր";
    print "mb_strlen() = " . mb_strlen($english) . " $english -ի համար և " .
    mb_strlen($armenian) . " $armenian -ի համար <br/>";
?>
```

1. mb_strlen() = 7 Program -ի համար և 12 Ծրագիր -ի համար
- 2.mb_strlen() = 7 Program -ի համար և 6 Ծրագիր -ի համար
- 3.Error

56. PHP-ի որ ֆունկցիան է ստեղծում պանակ:

1. readdir()
- 2.rmdir()
- 3.mkdir()

57. PHP-ի որ ֆունկցիան է ջնջում պանակը:

1. readdir()
- 2.rmdir()
- 3.mkdir()

58. PHP-ի որ ֆունկցիան է կարդում պանակի պարունակությունը:

1. `scandir()`
2. `rmdir()`
3. `mkdir()`

59. PHP-ի որ ֆունկցիան է վերադարձնում պանակում ըստ կարգի հաջորդ տարրի անունը:

1. `readdir()`
2. `rmdir()`
3. `mkdir()`

60. PHP-ի որ ֆունկցիան է փոխում ընթացիկ պանակը:

1. `chdir()`
2. `rmdir()`
3. `mkdir()`

61. PHP-ի որ ֆունկցիան չի օգտագործվում ֆայլից կարդալու համար:

1. `file_get_contents()`
2. `file()`
3. `file_put_contents()`

62. PHP-ի որ ֆունկցիան չի օգտագործվում ֆայլում գրելու համար:

1. `file_get_contents()`
2. `fputs()`
3. `file_put_contents()`

63. PHP-ի որ ֆունկցիան չի օգտագործվում զանգվածի տարրերի քանակը որոշելու համար:

1. count()
2. sizeof()
3. compact()

64. PHP-ի որ ֆունկցիան է խառնում զանգվածի տարրերը պատահականորեն:

1. count()
2. Shuffle()
3. compact()

65. PHP-ի որ ֆունկցիան է թույլ տալիս փոփոխականների հավաքածուից ստեղծել ասոցիացված զանգված:

1. count()
2. Shuffle()
3. compact()

66. PHP-ի որ ֆունկցիան է թույլ տալիս փոփոխականների հավաքածուից ստեղծել ասոցիացված զանգված:

1. count()
2. Shuffle()
3. compact()

67. PHP-ի որ ֆունկցիան է թույլ տալիս զանգվածի տարրերը դասավորել հակառակ հերթականությամբ:

1. arsort()
2. asort()
3. compact()

68. PHP-ի որ ֆունկցիան է թույլ տալիս զանգվածի տարրերը դասավորել աճման կարգով:

1. arsort()
2. asort()
3. compact()

69. Որտե՞ղ է սխալ մեկնաբանվում զանգվածի տարրերի դասավորման լրացուցիչ պարամետրը:

1. SORT_REGULAR - տեսակավորման տիպը բնորոշում է ծրագրավորողը:

2.SORT_NUMERIC - թվերի դասավորում:

3.SORT_STRING -տողերի դասավորում:

70. PHP-ի որ ֆունկցիան է թույլ տալիս զանգվածի տարրերը դասավորել աճման կարգով ըստ բանալիի:

1. arsort()

2.ksort()

3.krsort()

71. PHP-ի որ ֆունկցիան է թույլ տալիս զանգվածի տարրերը դասավորել հակառակ հերթականությամբ ըստ բանալիի:

1. arsort()

2.ksort()

3.krsort()

72. PHP-ի որ ֆունկցիայի միջոցով է հնարավոր իրականացնել զանգվածի տարրերի բնական դասավորում:

1. natsort()

2.ksort()

3.krsort()

73. PHP-ի որ ֆունկցիան է թույլ տալիս երկու գոյություն ունեցող զանգվածներից ստեղծել նոր զանգված:

1. array_combine()

2.natsort()

3.krsort()

74. Որտե՞ղ է ճիշտ մեկնաբանվում ասոցիատիվ զանգվածի հետ իրականացվող գործողությունը:

1. array_values() ֆունկցիան ասոցիատիվ զանգվածից դուրս է բերում բանալիները:

2.array_keys()-ը վերադարձնում է միայն զանգվածի արժեքները:

3.array_flip()-ը փոխում է տեղերով բանալիները և արժեքները:

75. Որտե՞ղ է սխալ մեկնաբանվում զանգվածի հետ իրականացվող գործողությունը:

1. Որպեսզի ստանանք զանգված միայն եզակի արժեքներով, պետք է կիրառել array_unique() ֆունկցիան:

2.Array_column () ֆունկցիան օգտակար կլինի, եթե անհրաժեշտ է բազմաչափ զանգվածի որոշակի սյունակ արտածել:

- 3.Նախորդ երկու մեկնաբանություններն էլ սխալ են:

76. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
```

```
$array = [  
    ['id' => 1, 'title' => 'mek'],  
    ['id' => 2, 'title' => 'erku'],  
    ['id' => 3, 'title' => 'ereq'],  
];
```

```
$sids = array_column($array, 'id');  
print_r($sids);
```

```
?>
```

1. Array ([0] => 'mek' [1] => 'erku' [2] => 'ereq')

2.Array ([0] => 1 [1] => 2 [2] => 3)

- 3.Array ([1] => 0 [2] => 1 [3] => 2)

77. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
```

```
$array = [  
    ['id' => 1, 'title' => 'mek'],  
    ['id' => 2, 'title' => 'erku'],  
    ['id' => 3, 'title' => 'ereq'],  
];
```

```
$sids = array_column($array, 'title');  
print_r($sids);
```

```
?>
```

1. Array ([0] => 'mek' [1] => 'erku' [2] => 'ereq')

2.Array ([0] => 1 [1] => 2 [2] => 3)

3.Array ([0] => mek [1] => erku [2] => ereq)

78. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```
<?php
```

```
$array = ['a', 'a', 'b', 'b', 'b', 'c', 'b', 'a', 'd', 'd'];
```

```

$uniques = array_unique($array);
print_r($uniques);

?>

```

1. Array ([0] => a [2] => b [5] => c [8] => d)
2. Array ([0] => a [1] => b [2] => c [3] => d)
3. Array ([3] => a [4] => b [1] => c [2] => d)

79. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```

<?php
    $keys = ['կապույտ', 'կանաչ', 'նարնջագույն'];
    $values = ['blue', 'green', 'orange'];
    $array = array_combine($keys, $values);
    print_r($array);

?>

```

1. Array ([1] => blue [2] => green [3] => orange)
2. Array ([blue] => կապույտ [green] => կանաչ [orange] => նարնջագույն)
3. Array ([կապույտ] => blue [կանաչ] => green [նարնջագույն] => orange)

80. Որտե՞ղ է սխալ մեկնաբանվում զանգվածի հետ իրականացվող գործողությունը:

1. array_shift () - առաջին տարրի հեռացում:
2. array_pop () - վերջին տարրի հեռացում:
3. array_push () - ավելացնել տարրեր զանգվածի սկզբում:

81. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```

<?php
    $queue = array("orange", "banana");
    array_unshift($queue, "apple", "raspberry");
    print_r($queue);

?>

```

1. Array ([0] => apple [1] => raspberry [2] => orange [3] => banana)
2. Array ([0] => orange [1] => banana [2] => apple [3] => raspberry)
3. Array ([0] => banana)

82. Ի՞նչ կարտածվի տրված կոդի կատարման արդյունքում:

```

<?php
    $queue = array("orange", "banana");
    array_push ($queue, "apple", "raspberry");
    print_r($queue);

?>

```

1. Array ([0] => apple [1] => raspberry [2] => orange [3] => banana)
2. Array ([0] => orange [1] => banana [2] => apple [3] => raspberry)
3. Array ([0] => banana)

83. Որտե՞ղ է սխալ մեկնաբանվում սուպերգլոբալ փոփոխականները:

1. Սուպերգլոբալ փոփոխականներ են անվանում PHP-ում նախորոշված փոփոխականները, որոնք ունեն տեսանելիության գլոբալ մակարդակ:
2. Սուպերգլոբալ փոփոխականները նախապես հայտարարված փոփոխականներ են, որոնք տեսանելի են PHP կոդի ցանկացած հատվածում:
3. Սուպերգլոբալ փոփոխականները չեն կարող լինել ասոցիատիվ զանգվածներ:

84. Որտե՞ղ է սխալ մեկնաբանվում \$ _SERVER զանգվածում առկա տվյալները:

1. PHP_SELF — ցույց է տալիս այն ֆայլի անունը, որը տվյալ պահին գործարկված է:
2. SERVER_ADDR — ցույց է տալիս այն սերվերի IP հասցեն, որի վրա գործարկված է տվյալ ֆայլը:
3. REQUEST_METHOD — ցույց է տալիս, թե ինչ մեթոդով է ավարտվել ծրագրի աշխատանքը:

85. Նշվածներից ո՞րը սուպերգլոբալ փոփոխական չէ:

1. \$ _SERVER
2. \$ _POST
3. \$ _directory

86. Նշվածներից ո՞րը սուպերգլոբալ փոփոխական չէ:

4. \$ _COOKIE
5. \$ _SCRIPT
6. \$ _SESSION

87. Որտե՞ղ է սխալ մեկնաբանվում սեսիաները:

1. Մեկ սցենարի մեջ սեսիան պետք է սկզբնաբաժնավորել միայն մեկ անգամ:
2. Սեսիայի մեջ ինչ-որ բան գրելու համար նախ պետք է սկզբնաբաժնավորել այն session() ֆունկցիայով:
3. Կարելի է գրել որևէ տեղեկատվություն սեսիայում և այնտեղից կարդալ այն կայքի այս կամ այլ սցենարի հաջորդ գործարկման ժամանակ:

88. Որտե՞ղ է սխալ մեկնաբանվում սեսիայի փոփոխականների ոչնչացումը:

1. Սեսիայի փոփոխականը կարող է ոչնչացվել unset ֆունկցիայի միջոցով:
- 2.Եթե անհրաժեշտ է տվյալ օգտագործողի համար ոչնչացնել սեսիայի բոլոր փոփոխականները ապա unset-ի փոխարեն պետք է օգտագործել session_destroy() ֆունկցիան:
- 3.session_destroy() ֆունկցիան կարելի է կանչվել միայն այն ժամանակ, երբ սեսիան չի սկսվել session_start-ի միջոցով:

89. Որտե՞ղ է սխալ մեկնաբանվում կուկիների հետ աշխատանքը:

1. Cookie-ն օգտագործողի զննարկիչում տվյալների երկարաժամկետ պահպանման միջոց է:
- 2.Տվյալ դոմենի համար Cookie-ների քանակի սահմանափակում չկա:
- 3.Ինչպես սեսիաների դեպքում կուկիներում կարելի է գրել մինչ էկրանին որևէ բան արտածելը:

90. Որտե՞ղ է սխալ մեկնաբանվում կուկիների հետ աշխատանքը:

1. Կարելի է ինչ-որ բան գրել cookie- ներում` օգտագործելով setcookie ֆունկցիան, որը առաջին պարամետրը վերցնում է որպես այս cookie- ի անուն, իսկ երկրորդը` որպես արժեքը:
- 2.Կարելի է երկարացնել cookie- ի կյանքի ժամկետը օգտագործելով setcookie ֆունկցիայի երրորդ պարամետրը, որով նշվում է cookie- ի կյանքի տևողությունը:
- 3.setcookie ֆունկցիայում երրորդ պարամետրը սովորաբար գրվում է հետևյալ կերպ` ժամանակի ներկա պահը + N րոպե:

91. Որտե՞ղ է սխալ տրված cookie-ի կյանքի ժամկետը:

1. // 1 ժամ setcookie("test"," տեստ", time() + 60);
- 2.// 1 օր setcookie("test"," տեստ", time() + 3600*24); :
- 3.// 1 ամիս setcookie("test"," տեստ", time() + 3600*24*30);

92. Որտե՞ղ է սխալ մեկնաբանվում կուկիների հետ աշխատանքը:

1. Կուկիները կարելի է կարդալ \$_COOKIE գլոբալ զանգվածի միջոցով:
- 2.cookie- ի ոչնչացման համար անհրաժեշտ է որպես setcookie ֆունկցիայի երրորդ պարամետր նշանակել ժամանակի ներկա պահը:
- 3.Ժամանակի ներկա պահը կարելի է ստանալ second ֆունկցիայի միջոցով:

93. Ի՞նչ պետք է տեղադրվի «...» - ի փոխարեն տվյալների հենքում հարցում կատարելու համար:

```
$host = 'localhost';  
$database = 'db1';  
$user = 'root';  
$password = '';  
$link = mysqli_connect($host, $user, $password, $database);  
$res = mysqli_query(..., 'SET NAMES utf8');
```

1. \$host
2. \$link
3. \$database

94. Ո՞րն է mysqli_connect ֆունկցիայի արգումենտների ներկայացման ճիշտ կարգը:

1. \$database, \$user, \$password, \$host
2. \$host, \$user, \$password, \$database
3. \$host, \$password, \$user, \$database

95. Որտե՞ղ է սխալ մեկնաբանվում ներկայացված ֆունկցիաները:

1. Մաքրել եզրային բացատները` trim:
2. Տողից հեռացնել HTML թեգերը` strip_tags:
3. Օգտագործողի ներածված տվյալներին ավելացնել սլեշներ` stripslashes: