

Выполнение заданий

Онлайн PHP Песочница:

<https://wtools.io/ru/php-sandbox>

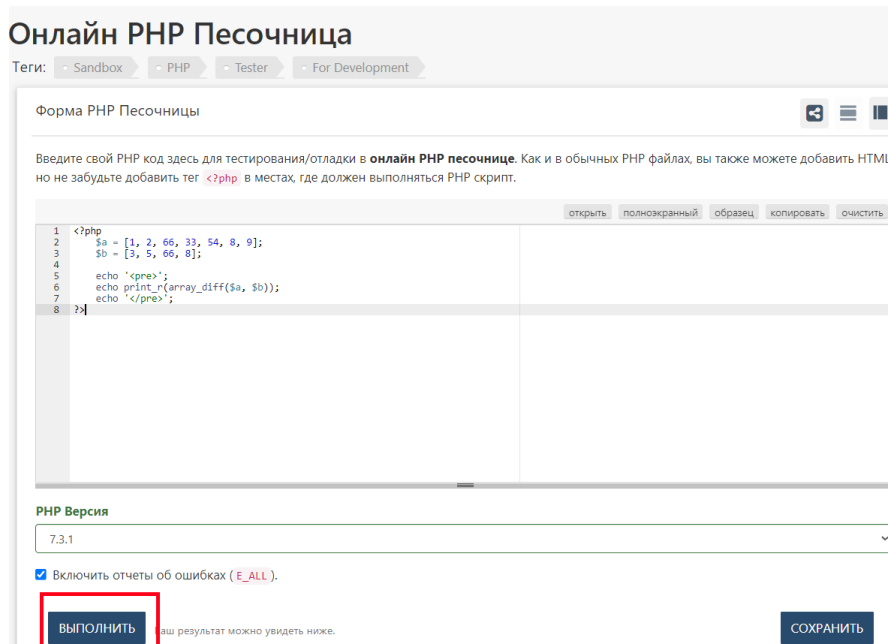


Рисунок 1 – Редактор

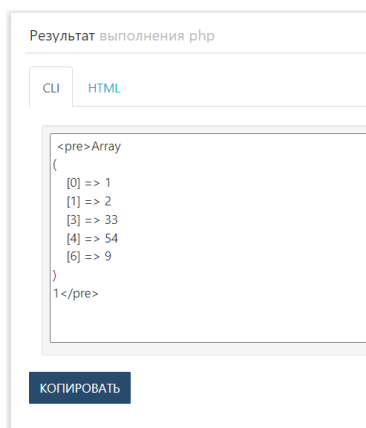


Рисунок 2 – Результат выполнения

При выводе html-верстки можно посмотреть, как это будет выглядеть в браузере на вкладке HTML

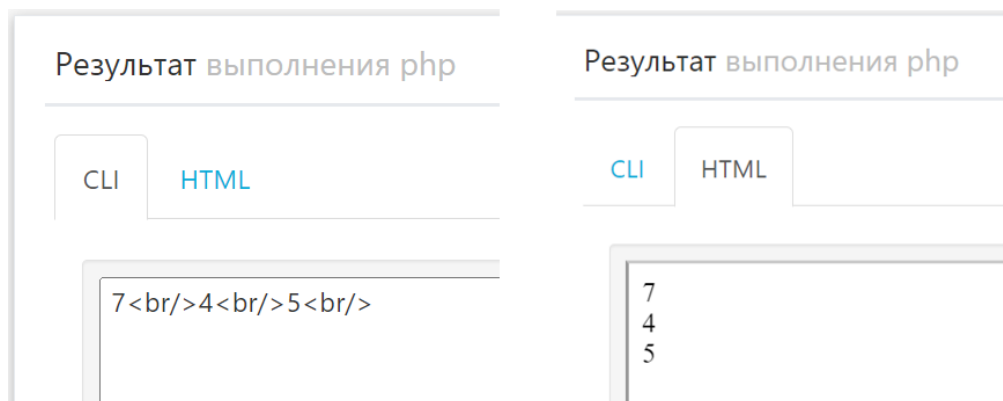


Рисунок 3 – HTML-верстка

. Для вкладки CLI переход на новую строку отображается при помощи символа перехода \n, а на вкладке HTML при помощи тега
.

Вариант 1

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [7, 4, 2, 5, 3];
```

Вывести значения массива, если значение > 3. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
7<br/>4<br/>5<br/>
```

Задание 3

Дан массив:

```
a = ["1a", "j", "some", "4",]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4",] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    5      => "value 1",
    "two"  => "value 1",
    "key"   => 2,
    "key2"  => "2"
];
```

Если значение строго = "2" или ключ = 5 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"  => "value 333",           3: value 1
    "key"   => 2,                   key: 2
    "key2"  => "2"
]
```

Вариант 2

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [6, 3, 2, 4, 1];
```

Вывести значения массива, если значение < 4. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
3<br/>2<br/>1<br/>
```

Задание 3

Дан массив:

```
a = ["abc", "k", "some", "8",]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4",] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    4      => "value 1",
    "two"  => "value 2",
    "key"   => "6",
    "key2"  => 6
];
```

Если значение строго = 6 или ключ = 4 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"  => "value 333",           3: value 1
    "key"   => 2,                    key: 2
    "key2"  => "2"
]
```

Вариант 3

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [7, 3, 4, 8, 5];
```

Вывести значения массива, если значение ≥ 5 . Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
4<br/>7<br/>8<br/>
```

Задание 3

Дан массив:

```
a = ["hq", "v", "some", "9",]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4",] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    4      => "value 1",
    "two"  => "value 2",
    "key"   => "6",
    "key2"  => 6
];
```

Если значение строго = 6 или ключ = 4 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"  => "value 333",           3: value 1
    "key"   => 2,                    key: 2
    "key2"  => "2"
]
```

Вариант 4

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [7, 3, 4, 8, 5];
```

Вывести значения массива, если значение >= 5. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
4<br/>7<br/>8<br/>
```

Задание 3

Дан массив:

```
a = ["hq", "v", "some", "9",]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4",] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    4      => "value 1",
    "two"  => "value 2",
    "key"   => "6",
    "key2"  => 6
];
```

Если значение строго = 6 или ключ = 4 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"  => "value 333",           3: value 1
    "key"   => 2,                    key: 2
    "key2"  => "2"
]
```

Вариант 5

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [1, 6, 4, 3, 7];
```

Вывести значения массива, если значение ≥ 5 . Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
6<br/>7<br/>
```

Задание 3

Дан массив:

```
a = ["qwer", "m", "some", "33",]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4",] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    5      => "value 1",
    "two"  => "value 2",
    "key"   => "6",
    "key2"  => 6
];
```

Если значение строго = "6" или ключ = 5 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"  => "value 333",           3: value 1
    "key"   => 2,                     key: 2
    "key2"  => "2"
]
```

Вариант 6

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [2, 4, 7, 3, 7];
```

Вывести значения массива, если значение <= 4. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
2<br/>4<br/>3<br/>
```

Задание 3

Дан массив:

```
a = ["ttt", "y", "some", "3",]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4",] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    12      => "value 1",
    "two"   => "value 2",
    "key"   => "3",
    "key2"  => 3
];
```

Если значение строго = 3 или ключ = 12 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"   => "value 333",           3: value 1
    "key"   => 2,                     key: 2
    "key2"  => "2"
]
```

Вариант 7

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [4, 5, 7, 1, 8];
```

Вывести значения массива, если значение ≥ 7 . Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
7<br/>8<br/>
```

Задание 3

Дан массив:

```
a = ["ert", "k", "some", "4",]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4",] и варианта 6:

Array

(

[0] => value_1a

[1] => value_j

[2] => value_some

[3] => value_4

)

1

Индекс: 6, значение: value_1a

Индекс: 7, значение: value_j

Индекс: 8, значение: value_some

Индекс: 9, значение: value_4

Задание 4

Дан массив:

```
a = [
    89      => "value 1",
    "two"   => "value 2",
    "key"    => "2",
    "key2"   => 2
];
```

Если значение строго = "2" или ключ = 89 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"   => "value 333",           3: value 1
    "key"    => 2,                     key: 2
    "key2"   => "2"
]
```


Вариант 8

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [5, 3, 4, 8, 2];
```

Вывести значения массива, если значение > 4. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
5<br/>8<br/>
```

Задание 3

Дан массив:

```
a = ["jkl", "p", "some", "8",]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4",] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
```

1

Индекс: 6, значение: value_1a

Индекс: 7, значение: value_j

Индекс: 8, значение: value_some

Индекс: 9, значение: value_4

Задание 4

Дан массив:

```
a = [
    7 => "value 1",
    "two" => "value 2",
    "key" => "5",
    "key2" => 55
];
```

Если значение строго = 55 или ключ = 7 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3 => "value 1",
    "two" => "value 333",
    "key" => 2,
    "key2" => "2"
]
3: value 1
key: 2
```

Вариант 9

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [4, 3, 8, 2, 1];
```

Вывести значения массива, если значение <= 3. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
3<br/>2<br/>1<br/>
```

Задание 3

Дан массив:

```
a = ["wer", "x", "some", "4",]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4",] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    6      => "value 1",
    "two"  => "value 2",
    "key"   => "4",
    "key2"  => 4
];
```

Если значение строго = 4 или ключ = 6 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"  => "value 333",           3: value 1
    "key"   => 2,                    key: 2
    "key2"  => "2"
]
```

Вариант 10

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [6, 7, 3, 5, 2];
```

Вывести значения массива, если значение > 5. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
6<br/>7<br/>
```

Задание 3

Дан массив:

```
a = ["hjk", "b", "some", "8",]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4",] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
```

1

Индекс: 6, значение: value_1a

Индекс: 7, значение: value_j

Индекс: 8, значение: value_some

Индекс: 9, значение: value_4

Задание 4

Дан массив:

```
a = [
    7 => "value 1",
    "two" => "value 2",
    "key" => "4",
    "key2" => 4
];
```

Если значение строго = 4 или ключ = 7 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3 => "value 1",
    "two" => "value 333",
    "key" => 2,
    "key2" => "2"
]
3: value 1
key: 2
```

Вариант 11

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [6, 7, 3, 5, 2];
```

Вывести значения массива, если значение < 5. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
3<br/>2<br/>
```

Задание 3

Дан массив:

```
a = ["qwe", "y", "some", "2",]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4",] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
```

1

Индекс: 6, значение: value_1a

Индекс: 7, значение: value_j

Индекс: 8, значение: value_some

Индекс: 9, значение: value_4

Задание 4

Дан массив:

```
a = [
    5      => "value 1",
    "two"  => "value 2",
    "key"   => "2",
    "key2"  => 2
];
```

Если значение строго = 2 или ключ = 5 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"  => "value 333",           3: value 1
    "key"   => 2,                    key: 2
    "key2"  => "2"
]
```

Вариант 12

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [1, 3, 2, 4, 5];
```

Вывести значения массива, если значение < 4. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
1<br/>3<br/>2<br/>
```

Задание 3

Дан массив:

```
a = ["hjk", "y", "some", "3"]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4"] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    2      => "value 1",
    "two"  => "value 2",
    "key"   => "6",
    "key2"  => 6
];
```

Если значение строго = "6" или ключ = 2 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"  => "value 333",           3: value 1
    "key"   => 2,                    key: 2
    "key2"  => "2"
]
```

Вариант 13

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [5, 3, 2, 4, 5];
```

Вывести значения массива, если значение < 4. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
3<br/>2<br/>
```

Задание 3

Дан массив:

```
a = ["iop", "k", "some", "4"]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4"] и варианта 6:

Array

(

[0] => value_1a

[1] => value_j

[2] => value_some

[3] => value_4

)

1

Индекс: 6, значение: value_1a

Индекс: 7, значение: value_j

Индекс: 8, значение: value_some

Индекс: 9, значение: value_4

Задание 4

Дан массив:

```
a = [
    8      => "value 1",
    "two"  => "value 2",
    "key"   => "4",
    "key2"  => 4
];
```

Если значение строго = "4" или ключ = 8 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"  => "value 333",           3: value 1
    "key"   => 2,                    key: 2
    "key2"  => "2"
]
```

Вариант 13

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [7, 2, 1, 4, 5];
```

Вывести значения массива, если значение ≥ 5 . Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
5<br/>7<br/>
```

Задание 3

Дан массив:

```
a = ["fgh", "a", "some", "3"]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4"] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    6 => "value 1",
    "two" => "value 2",
    "key" => "4",
    "key2" => 4
];
```

Если значение строго = 4 или ключ = 6 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3 => "value 1",
    "two" => "value 333",
    "key" => 2,
    "key2" => "2"
]
3: value 1
key: 2
```

Вариант 15

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [7, 2, 1, 4, 5];
```

Вывести значения массива, если значение < 5. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
2<br/>1<br/>4<br/>
```

Задание 3

Дан массив:

```
a = ["qwe", "a", "some", "5"]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4"] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    9 => "value 1",
    "two" => "value 2",
    "key" => "7",
    "key2" => 7
];
```

Если значение строго = "7" или ключ = 9 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3 => "value 1",
    "two" => "value 333",
    "key" => 2,
    "key2" => "2"
]
3: value 1
key: 2
```


Вариант 16

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [8, 3, 7, 2, 5];
```

Вывести значения массива, если значение < 7. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
3<br/>2<br/>5<br/>
```

Задание 3

Дан массив:

```
a = ["vbb", "a", "some", "5"]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4"] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    7 => "value 1",
    "two" => "value 2",
    "key" => "12",
    "key2" => 12
];
```

Если значение строго = 12 или ключ = 7 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3 => "value 1",
    "two" => "value 333",
    "key" => 2,
    "key2" => "2"
]
3: value 1
key: 2
```

Вариант 17

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [8, 3, 7, 1, 4];
```

Вывести значения массива, если значение < 7. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
3<br/>1<br/>4<br/>
```

Задание 3

Дан массив:

```
a = ["cgu", "a", "some", "8"]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4"] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    2      => "value 1",
    "two"  => "value 2",
    "key"   => "13",
    "key2"  => 13
];
```

Если значение строго = 13 или ключ = 2 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"  => "value 333",           3: value 1
    "key"   => 2,                    key: 2
    "key2"  => "2"
]
```

Вариант 18

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [5, 3, 4, 1, 7];
```

Вывести значения массива, если значение < 5. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
3<br/>4<br/>1<br/>
```

Задание 3

Дан массив:

```
a = ["kqt", "a", "some", "9"]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4"] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    85      => "value 1",
    "two"   => "value 2",
    "key"   => "11",
    "key2"  => 11
];
```

Если значение строго = "11" или ключ = 85 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"   => "value 333",           3: value 1
    "key"   => 2,                     key: 2
    "key2"  => "2"
]
```

Вариант 19

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [5, 3, 4, 1, 7];
```

Вывести значения массива, если значение ≥ 5 . Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
5<br/>7<br/>
```

Задание 3

Дан массив:

```
a = ["vht", "a", "some", "4"]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4"] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    65      => "value 1",
    "two"   => "value 2",
    "key"    => "11",
    "key2"   => 11
];
```

Если значение строго = 11 или ключ = 65 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"   => "value 333",           3: value 1
    "key"    => 2,                     key: 2
    "key2"   => "2"
]
```

Вариант 20

Задание 1

Создать переменную \$task и присвоить ей номер варианта.

Вывести в echo тэг <h1> с содержимым: Мой вариант: \$task, где \$task – значение переменной.

Пример вывода для варианта 14: <h1>Мой вариант: 14</h1>

Задание 2

Дан массив:

```
a = [4, 8, 4, 1, 7];
```

Вывести значения массива, если значение > 4. Каждое значение выводить с новой строки при помощи "
".

Пример вывода:

```
7<br/>8<br/>
```

Задание 3

Дан массив:

```
a = ["giy", "a", "some", "2"]
```

1. Создать новый массив result: все значения из массива a, к каждому значению добавить в начале "value_" (Функция array_push).

2. Вывести массив result при помощи функции print_r.

3. Вывести массив result в формате:

Индекс: <индекс массива + номер варианта>, значение: <значение>

Каждую запись выводить с новой строки при помощи "\n".

Пример вывода для исходного массива ["1a", "j", "some", "4"] и варианта 6:

Array

```
(
    [0] => value_1a
    [1] => value_j
    [2] => value_some
    [3] => value_4
)
1
Индекс: 6, значение: value_1a
Индекс: 7, значение: value_j
Индекс: 8, значение: value_some
Индекс: 9, значение: value_4
```

Задание 4

Дан массив:

```
a = [
    15      => "value 1",
    "two"   => "value 2",
    "key"    => "34",
    "key2"   => 34
];
```

Если значение строго = "34" или ключ = 15 вывести в формате:

<ключ>: <значение>

Каждую запись выводить с новой строки при помощи \n.

Пример вывода для условия значение строго = 2 или ключ = 3 и массива (запись с key2 не попадет!):

```
[
    3      => "value 1",
    "two"   => "value 333",           3: value 1
    "key"    => 2,                     key: 2
    "key2"   => "2"
]
```

