**Лабораторная работа №17**

**Многомерные массивы в PHP (часть 2)**

**Заполнение многомерных массивов**

Пусть теперь мы хотим в цикле создать какой-нибудь многомерный массив с числами.

Например, вот такой двухмерный массив:

<?php

[[1, 2, 3], [1, 2, 3], [1, 2, 3]]

?>

Решим поставленную задачу, применив два вложенных цикла. Внешний цикл будет создавать подмассивы, а внутренний - заполнять эти подмассивы числами:

<?php

$arr = [];

for ($i = 0; $i < 3; $i++) {

for ($j = 0; $j < 3; $j++) {

$arr[$i][$j] = $j + 1; // заполняем подмассив числами

}

}

var\_dump($arr);

?>

**Задача №1**

Сформируйте с помощью двух вложенных циклов следующий массив:

[[1, 2, 3, 4, 5], [1, 2, 3, 4, 5], [1, 2, 3, 4, 5]]

**Задача №2**

Сформируйте с помощью двух вложенных циклов следующий массив:

[['x', 'x', 'x', 'x'], ['x', 'x', 'x', 'x'], ['x', 'x', 'x', 'x']]

**Задача №3**

Сформируйте с помощью трех вложенных циклов следующий массив:

[

[

[1, 2, 3, 4, 5],

[1, 2, 3, 4, 5],

],

[

[1, 2, 3, 4, 5],

[1, 2, 3, 4, 5],

],

[

[1, 2, 3, 4, 5],

[1, 2, 3, 4, 5],

],

]

**Заполнение числами по порядку**

В предыдущих примерах все числа в подмассивах были одинаковыми. Давайте теперь сделаем так, чтобы числа возрастали, вот так:

<?php

[[1, 2, 3], [4, 5, 6], [7, 8, 9]]

?>

Для этого нужно сделать специальную переменную-счетчик, которая будет увеличивать свое значение на 1 при каждой итерации внутреннего цикла. Значение этого счетчика мы и будем записывать в массив, вот так:

<?php

$arr = [];

$k = 1; // счетчик

for ($i = 0; $i < 3; $i++) {

for ($j = 0; $j < 3; $j++) {

$arr[$i][$j] = $k; // записываем счетчик

$k++; // увеличиваем счетчик

}

}

var\_dump($arr);

?>

Можно уменьшить код, занеся определение счетчика в первый цикл, а k++ - во второй:

<?php

$arr = [];

for ($i = 0, $k = 1; $i < 3; $i++) {

for ($j = 0; $j < 3; $j++, $k++) {

$arr[$i][$j] = $k;

}

}

var\_dump($arr);

?>

Можно также сделать увеличение счетчика после присваивания:

<?php

$arr = [];

for ($i = 0, $k = 1; $i < 3; $i++) {

for ($j = 0; $j < 3; $j++) {

$arr[$i][$j] = $k++;

}

}

var\_dump($arr);

?>

Обратите внимание на то, что в данном случае должно быть обязательно k++, а не ++k, так как второй вариант сначала выполнит увеличение счетчика, а уже потом запись в массив (то есть в результате массив начнется с двойки, а не с единицы).

**Задача №4**

Сформируйте с помощью двух вложенных циклов следующий массив:

<?php

[[1, 2], [3, 4], [5, 6], [7, 8]]

?>

**Задача №5**

Сформируйте с помощью двух вложенных циклов следующий массив:

<?php

[[2, 4, 6], [8, 10, 12], [14, 16, 18], [20, 22, 24]]

?>

**Задача №6**

Сформируйте с помощью трех вложенных циклов следующий трехмерный массив:

<?php

[[[1, 2], [3, 4]], [[5, 6], [7, 8]]]

?>

**Массив ассоциативных массивов в PHP**

Очень часто в PHP появляются массивы, содержащие набор неких сущностей, например, список пользователей или список продуктов.

Вот пример такого массива:

<?php

$users = [

[

'name' => 'user1',

'age' => 31,

'salary' => 1000,

],

[

'name' => 'user2',

'age' => 32,

'salary' => 2000,

],

[

'name' => 'user3',

'age' => 33,

'salary' => 3000,

],

];

?>

Как вы видите, здесь мы имеем дело с массивом ассоциативных массивов. Не смотря на то, что этот массив двухмерный, как правило для его перебора используется один foreach, перебирающий подмассивы. А части самих подмассивов просто получают по ключу и записывают в нужном порядке и формате.

Давайте для примера выведем столбец наших работников в некотором формате:

<?php

foreach ($users as $user) {

echo $user['name'] . ': ' . $user['salary'] . '$, ' . $user['age'] . '<br>';

}

?>

**Задача №7**

Дан следующий массив:

<?php

$products = [

[

'name' => 'мясо',

'price' => 100,

'amount' => 5,

],

[

'name' => 'овощи',

'price' => 200,

'amount' => 6,

],

[

'name' => 'фрукты',

'price' => 300,

'amount' => 7,

],

];

?>

Выведите с помощью этого массива столбец продуктов в каком-нибудь придуманном вами формате.

Добавление элементов

Давайте добавим в наш массив еще одного юзера:

<?php

$users[] = [

'name' => 'name4',

'age' => 34,

'salary' => 4000,

];

?>

**Задача №8**

Добавьте в массив с продуктами из предыдущей задачи еще один продукт.

**Конвертация многомерных массивов в PHP**

Пусть у нас есть список юзеров, хранящийся в следующей структуре:

<?php

$users = [

[

'id' => 11,

'name' => 'name11',

'surname' => 'surname11',

],

[

'id' => 14,

'name' => 'name14',

'surname' => 'surname14',

],

[

'id' => 17,

'name' => 'name17',

'surname' => 'surname17',

],

];

?>

Давайте напишем код, который переформатирует наш массив вот в такой:

[

11 => [

'id' => 11,

'name' => 'name11',

'surname' => 'surname11',

],

14 => [

'id' => 14,

'name' => 'name14',

'surname' => 'surname14',

],

17 => [

'id' => 17,

'name' => 'name17',

'surname' => 'surname17',

],

]

Решим поставленную задачу с помощью цикла:

<?php

$result = [];

foreach ($users as $user) {

$result[$user['id']] = $user;

}

var\_dump($result);

?>

**Задача №9**

Дан список городов и их стран, хранящийся в следующей структуре:

[

[

'country' => 'Россия',

'city' => 'Москва',

],

[

'country' => 'Беларусь',

'city' => 'Минск',

],

[

'country' => 'Россия',

'city' => 'Питер',

],

[

'country' => 'Россия',

'city' => 'Владивосток',

],

[

'country' => 'Украина',

'city' => 'Львов',

],

[

'country' => 'Беларусь',

'city' => 'Могилев',

],

[

'country' => 'Украина',

'city' => 'Киев',

],

]

Напишите код, которой переделает структуру данных вот в такую:

[

'Россия'=> [

'Москва', 'Питер', 'Владивосток',

],

'Беларусь'=> [

'Минск', 'Могилев'

],

'Украина'=> [

'Киев', 'Львов'

],

]

**Задача №10**

Дан список событий за определенные даты, хранящийся в следующей структуре:

[

[

'date' => '2019-12-29',

'event' => 'name1'

],

[

'date' => '2019-12-31',

'event' => 'name2'

],

[

'date' => '2019-12-29',

'event' => 'name3'

],

[

'date' => '2019-12-30',

'event' => 'name4'

],

[

'date' => '2019-12-29',

'event' => 'name5'

],

[

'date' => '2019-12-31',

'event' => 'name6'

],

[

'date' => '2019-12-29',

'event' => 'name7'

],

[

'date' => '2019-12-30',

'event' => 'name8'

],

[

'date' => '2019-12-30',

'event' => 'name9'

],

]

Напишите код, которой переделает структуру данных вот в такую:

[

'2019-12-29'=> ['name1', 'name3', 'name5', 'name7'],

'2019-12-30'=> ['name4', 'name8', 'name9'],

'2019-12-31'=> ['name2', 'name6'],

]

**Задача №11**

Дан список событий за определенные даты, хранящийся в следующей структуре:

[

'2019-12-29'=> ['name1', 'name2', 'name3', 'name4'],

'2019-12-30'=> ['name5', 'name6', 'name7'],

'2019-12-31'=> ['name8', 'name9'],

]

Напишите код, которой переделает структуру данных вот в такую:

[

[

'date' => '2019-12-29',

'event' => 'name1'

],

[

'date' => '2019-12-29',

'event' => 'name2'

],

[

'date' => '2019-12-29',

'event' => 'name3'

],

[

'date' => '2019-12-29',

'event' => 'name4'

],

[

'date' => '2019-12-30',

'event' => 'name5'

],

[

'date' => '2019-12-30',

'event' => 'name6'

],

[

'date' => '2019-12-30',

'event' => 'name7'

],

[

'date' => '2019-12-31',

'event' => 'name8'

],

[

'date' => '2019-12-31',

'event' => 'name9'

],

]