**Практическая работа №5 «Создание серверных сценариев с использованием языка PHP». Массивы**

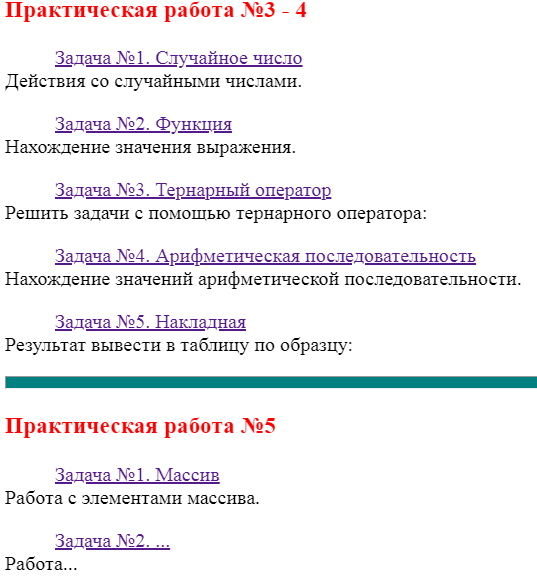
**Цель:**

* Изучить основы языка PHP: массивы.
* Решать задачи на массивы.

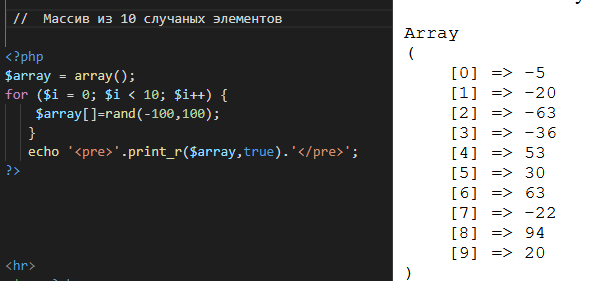
**Порядок выполнения работы**

**Задача 1 Действия со случайными числами**

* В стартовой странице практической работы в файле **index.php**  в конце документа установите черту и допишите информацию стартовой страницы: **Практическая работа №5.** На этой же стартовой странице проекта продолжайте размещать ссылки на задачи и их краткое описание. Файлы с задачами определяются именем «**workN\_M.php**», где N – номер работы, M - номер задачи.

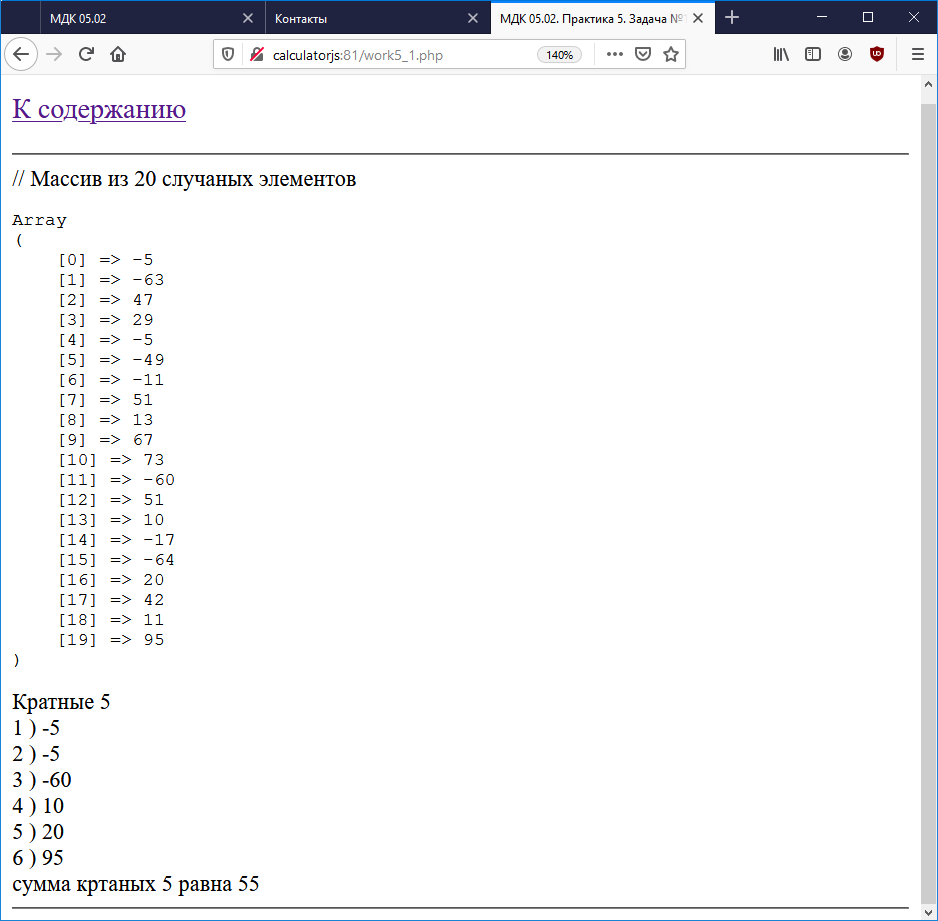


* Изучите теорию о массивах:



Напишите программный код, позволяющий решить задачи, приведенные ниже.

|  |  |
| --- | --- |
|  | Заполните массив из 20 элементов случайными числами в промежутке  [х1, х2] (с отрицательными числами). Выведите полученные элементы на страницу.  Определите:   1. сумму элементов, кратных «5»; выведите полученные элементы, определите их количество, найдите среди них максимум; 2. минимум среди четных; выведите четные элементы; 3. сравните максимум среди четных и максимум среди нечетных элементов; 4. замените все отрицательные элементы их абсолютным значением; 5. среднее арифметическое элементов массива с p1-го по p2-й элементы; 6. произведение первых пяти неотрицательных элементов; 7. количество элементов, больших среднего арифметического всех его элементов; выведите найденные элементы; 8. количество элементов, равных максимальному значению этого массива |



|  |  |
| --- | --- |
|  | Выведите на страницу дни недели по образцу: понедельник – день 1-й. Используйте массив. |
|  | Приготовьте 6 картинок. Рассмотрите код, введите его. Протестируйте.      Установите стиль:   * на тело документа - выравнивание контента по центру, темный цвет заливки; * на заголовок – шрифтовое оформление;           Протестируйте сайт.  Выполните добавление еще одной картинки в массив, просмотрите результат (затем добавьте еще 2). |
|  | Добавьте на страницу навигационное меню (например: Главная, О нас, Контакты), используя массив для хранения пунктов меню.  Примечание: значением элемента массива является имя файла, на который будет указывать ссылка, например: index.php, about.php, contact.html |
|  | Выполните реализацию задачи с помощью ассоциативного массива. Определите суммарную и среднюю стоимость всех товаров. Выведите самые дорогие товары (самые дешевые).  //Вывод суммы с форматом -  number\_format(13 707.33 - [Число]  // 2 - [знаков после .], '.' - [разделитель],' ' - [пробел между тысячными]);  echo '<h2 style="text-align: center;">Всего: '.number\_format(array\_sum($M\_Vsego), 2, '.', ' ') .' руб.</h2>'; |
|  | Создайте ассоциативный массив, содержащий список студентов по группам (ГРУППА => СТУДЕНТЫ)  Просмотрите созданный массив на странице, выводя название групп и связанных с ними студентов. Определите количество студентов в каждой группе |
|  | Создайте ассоциативный массив, содержащий имена и информацию личного характера (телефон, город проживания и аватар) 7 семи человек. Выведите информацию в таблицу. |
|  | Создайте ассоциативный массив, содержащий список стран, а также название столицы, язык, площадь занимаемой территории, количество населения и флаг каждой из них.  Просмотрите созданный массив на странице. Определите самые густонаселенные страны. |
|  | Создайте ассоциативный массив, содержащий список выполненных и невыполненных задач. Выведите информацию в виде списка, в котором выполненные дела будут как-либо выделены. |
| **10** | Напишите функцию quotient, которая принимает два параметра (делимое и делитель) и возвращает частное (результат деления). Если делитель равен нулю, то функция должна вернуть сообщение: “Делитель не может быть нулем”.  quotient(50, 2) -> 25  quotient(0, 2) -> 0  quotient(2, 0) -> Делитель не может быть нулем |
| **11** | Вам необходимо запрограммировать работу кнопки переключения некоторого устройства.  Известно, что в вашей системе могут быть устройства со следующими типами:   * Окно (close и open) * Дверь (close и open) * Светильник (off и on) * Холодильник (off и on) * Кофеварка (off и on)   Ваша задача написать функцию toggleDevice, которая принимает тип устройства и его текущее значение и возвращает новое состояние.  toggleDevice(‘door’, ‘close’) -> ‘open’  toggleDevice(‘light’, ‘off’) -> ‘on’ |
| **12** | Напишите функцию search, которая принимает массив и искомый элемент, верните true если исходный элемент найден в массиве, и false если такого элемента в массиве нет.  search([1, 7, 8], 8) -> true  search([1, 7, 8], 9) -> false  search([‘alex’, ‘andrey’, ‘carla’], ‘victor’) -> false  search([‘alex’, ‘andrey’, ‘carla’], ‘carla’) -> true |
| **13** | Написать функцию getAdults, которая принимает массив пользователей, по каждому из которых известен возраст и имя.  Результатом работы функции должен стать массив, содержащий сведения о пользователях, старше 18 лет. |
| **14** | Сделайте таблицу основных функций для работы с массивами:   |  |  |  | | --- | --- | --- | | Функция | Назначение | Пример | |  |  |  |  * проверка наличия элемента в массиве * добавление элемента в массив (в начало, в конец) * удаление элемента (из начала, с конца) * получение уникальных значений * получение ключей ассоциативного массива * перемешивание элементов и переворачивание массива * сортировка (sort, rsort, asort, ksort, arsort, krsort, usort) * array\_splice * array\_slice * разделение строки на элементы массива * вывод элементов массива в строку * объединение элементов массивов, их пересечение |
| **15** | Вернитесь к решенным задачам и напишите новые функции с опорой на готовые функции для работы с массивами |
| **16** | Добавьте сортировку по различным критериям в уже решенные вами задачи. |
| **17** | Придумайте свою задачу и реализуйте ее. |

Просмотрите результат в браузере.

**Контрольные вопросы**

1. Что такое массив, виды массивов
2. Параметры массива