|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Инструментального и прикладного программного обеспечения

**ПРАКТИЧЕСКАЯ РАБОТА №2**

по дисциплине «Разработка серверных частей интернет-ресурсов»

**Тема практической работы: Основы php.**

**Студент группы** ИКБО-30-20 Бессмертных И.В.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись студента)

**Руководитель практической работы** преподаватель Благирев.М.М.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя)

Работа представлена «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Допущен к работе «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Москва 2022

ОГЛАВЛЕНИЕ

[Цель работы 3](#__RefHeading___Toc487_3036995042)

Ход работы 3

[Вывод 5](#__RefHeading___Toc390_3251243389)

[Ответы на вопросы 6](#__RefHeading___Toc388_3251243389)

[Ссылка на удаленный репозиторий 11](#__RefHeading___Toc386_3251243389)

Список использованной литературы 12

Цель работы

Используя серверную конфигурацию, разработанную в прошлой практической работе выполнить следующие упражнения. Предполагается создать 3 независимых сервиса, устойчивых к минимальному набору самых простых ошибок. Предполагается создание 1 общего проекта с разделенными сервисами, разделением проекта на файлы для разделения функционала и переиспользования файлов. Каждый сервис должен состоять как минимум из 2 файлов.

Ход работы

Для начала сделаем разделение сервисов, рисунок 1.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1 – файлы проекта.

Начнем с первого упражнения. В качестве кода для рисования svg-объектов будем использовать цифры, переданные на страницу с помощью GET-запроса. Далее, получив массив, в котором хранится код, берем его, и в зависимости от кода, с помощью условного оператора, будет рисоваться свой svg-объект, рисунок 2.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2 – пример вывода svg-объекта с кодом 1.

Во втором упражнении мы будем передавать массив как строчку, которую потом спарсим в массив, впоследствии отсортированный сортировкой выбора, в соответствии с 2 вариантом. В результате просто выводим отсортированный в порядке возрастания массив, рисунок 3.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3 – Отсортированный массив.

В последнем упражнении мы должны выполнить команду Unix-а и показать ее работу. Сделаем это с помощью функции exec, рисунок 4.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 4 – Пример выполнения команды.

Вывод

В рамках данной работы в 1 проекте были созданы 3 независимых сервиса, выполнив все предложенные задания я познакомился с основами работы на php.

Ответы на вопросы

1. Конфигурационный файл php.ini

Файл конфигурации (php.ini) считывается при запуске PHP. Для версий серверных модулей PHP это происходит только один раз при запуске веб-сервера. Для CGI и CLI версий это происходит при каждом вызове.

1. Как написать простой скрипт на php.

Для написания простого скрипта на php нужно между <?php и ?> написать некоторый код.

1. Основные правила, связанные с переменными в php

Переменные в PHP представлены знаком доллара с последующим именем переменной. Имя переменной чувствительно к регистру.

Имена переменных соответствуют тем же правилам, что и остальные наименования в PHP. Правильное имя переменной должно начинаться с буквы или символа подчёркивания и состоять из букв, цифр и символов подчёркивания в любом количестве. Это можно отобразить регулярным выражением: ^[a-zA-Z\_\x80-\xff][a-zA-Z0-9\_\x80-\xff]\*$

1. Основные типы данных в php

К скалярным типам данных относятся:

* целочисленный тип (int)
* вещественный (float)
* логический (bool)
* строковый (string)

К нескалярным типам относятся:

* массив (array)
* объект (object)
* внешний ресурс (resource)
* неопределённое значение (null)

1. Какие существуют функции для работы с переменными в php вне зависимости от типа данных

* gettype – возвращает тип переменной;
* empty – проверяет, пуста ли переменная;
* isset – определяет, была ли установлена переменная значением, отличным от null;
* unset – удаляет переменную;
* и другие.

1. Предопределенные переменные в php

PHP предоставляет всем скриптам большое количество предопределённых переменных. Эти переменные содержат всё, от внешних данных до переменных среды окружения, от текста сообщений об ошибках до последних полученных заголовков.

Суперглобальными переменными являются: $GLOBALS, $\_SERVER, $\_GET, $\_POST, $\_FILES, $\_COOKIE, $\_SESSION, $\_REQUEST, $\_ENV.

1. Переменные переменных в php

Переменная переменной берёт значение переменной и рассматривает его как имя переменной.

<?php

$a = 'hello';

$$a = 'world';

?>

Теперь в дереве символов PHP определены и содержатся две переменные:

* $a, содержащая "hello"
* $hello, содержащая "world"

1. Выражения в php

Выражение – это часть кода, которая представляется неким значением (значение – это число, текстовая строка или логическая величина).

Простейшими формами выражения являются переменные и литералы (т.е. числа, строки, константы). С помощью операторов (+,-,\*, / и т.д.) простые выражения можно объединять в более сложные.

1. Арифметические операторы в php

* +$a Идентичность Конвертация $a в int или float, что более подходит.
* -$a Отрицание Смена знака $a.
* $a + $b Сложение Сумма $a и $b.
* $a - $b Вычитание Разность $a и $b.
* $a \* $b Умножение Произведение $a и $b.
* $a / $b Деление Частное от деления $a на $b.
* $a % $b Деление по модулю Целочисленный остаток от деления $a на $b.
* $a \*\* $b Возведение в степень Возведение $a в степень $b.

1. Битовые операции в php

* $a & $b И Устанавливаются только те биты, которые установлены и в $a, и в $b.
* $a | $b Или Устанавливаются те биты, которые установлены в $a или в $b.
* $a ^ $b Исключающее или Устанавливаются только те биты, которые установлены либо только в $a, либо только в $b, но не в обоих одновременно.
* ~ $a Отрицание Устанавливаются те биты, которые не установлены в $a, и наоборот.
* $a << $b Сдвиг влево Все биты переменной $a сдвигаются на $b позиций влево (каждая позиция подразумевает "умножение на 2")
* $a >> $b Сдвиг вправо Все биты переменной $a сдвигаются на $b позиций вправо (каждая позиция подразумевает "деление на 2")

1. Оператор присваивания в php

Базовый оператор присваивания обозначается как "=". На первый взгляд может показаться, что это оператор "равно". На самом деле это не так. В действительности оператор присваивания означает, что левый операнд получает значение правого выражения, (то есть устанавливается значением).

1. Операторы сравнения в php

* Оператор == (равенства)
* Оператор === (идентичности)
* Оператор != или <> (неравенства)
* Оператор !== (неидентичности)
* Оператор > (больше)
* Оператор < (меньше)
* Оператор >= (больше или равно)
* Оператор <= (меньше или равно)

1. Логические операторы в php.

* $a and $b И true, если и $a, и $b true.
* $a or $b Или true, если или $a, или $b true.
* $a xor $b Исключающее или true, если $a, или $b true, но не оба.
* ! $a Отрицание true, если $a не true.
* $a && $b И true, если и $a, и $b true.
* $a || $b Или true, если или $a, или $b true.

1. Условная конструкция в php

Конструкция if является одной из наиболее важных во многих языках программирования, в том числе и PHP. Она предоставляет возможность условного выполнения фрагментов кода. Структура if реализована в PHP по аналогии с языком C:

if (выражение)

инструкция

1. Циклы в php

* while (expr)
* do while (expr)
* for (expr1; expr2; expr3)
* foreach (iterable\_expression as $value)

1. Конструкции switch и match в php

Конструкция switch..case является альтернативой использованию конструкции if..elseif..else. Оператор switch получает некоторое выражение и сравнивает его с набором значений.

Начиная с версии 8.0 в PHP была добавлена поддержка другой, похожей конструкции - match. Она позволяет оптимизировать конструкцию switch. Конструкция match также принимает некоторое выражение и сравнивает его с набором значений.

1. Include и require в php

Разница в том, что функция include() выдает предупреждение, но скрипт продолжит выполнение, а функция require() выдает предупреждение и фатальную ошибку, т. е. скрипт не будет продолжать выполнение. Эти две функции используются для помещения данных файла в другой файл PHP перед его выполнением сервером.

1. Функции в php

PHP поддерживает объявление:

* именованных функций;
* анонимных функций.

// Объявление именованной функции

function gethelloworld(): string

{

return 'Hello World!';

}

// Объявление анонимной функции

$getHelloWorld = function(): string

{

return 'Hello World!';

};

// Короткий синтаксис

$getHelloWorld = fn(): string => 'Hello World!';

Если конструкция return не указана, то функция вернёт значение null.

Ссылка на удаленный репозиторий

https://github.com/Ivanbessmertnykh/Data-Servers

Список использованной литературы

1. Дергачев А. М. Проблемы эффективного использования сетевых сервисов / Научно-технический вестник СПбГУ ИТМО. 2011. No 1 (71). С. 83-87.
2. Розенфельд Л., Морвиль П. Информационная архитектура в Интернете, 2 е издание. – Пер. с англ. – СПб: Символ Плюс, 2005 – 544 с., ил.
3. Спинеллис Д., Гусиос Г. Идеальная архитектура. Ведущие специалисты о красоте программных архитектур. – Пер. с англ. – СПб.: Символ Плюс, 2010 – 528 с., ил.
4. Фаулер, Мартин. Ф28 Архитектура корпоративных программных приложений.: Пер. с англ. — М.: Издательский дом "Вильяме", 2006 — 544 с.: ил. — Парал. тит. англ.
5. Голицына О.Л., Максимов Н.В., Партыка Т.Л., Попов И.И. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ. - 2-е изд. - Москва: ФОРУМ - ИНФРА-М, 2008. - 395 с.
6. Распределенные системы. Принципы и парадигмы / Э. Таненбаум, М. ван Стеен. — СПб.: Питер, 2003. — с. 83-93 — (Серия «Классика computer science»). ISBN 5-272-00053-6-.