|  |
| --- |
|  |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования  **"МИРЭА - Российский технологический университет"**  **РТУ МИРЭА** |
| Институт информационных технологий (ИТ) |
| Кафедра инструментального и прикладного программного обеспечения (ИППО) |

|  |  |
| --- | --- |
| **ОТЧЕТ**  **ПО ЛАБОРАТОРНОЙ РАБОТЕ №6** | |
| **по дисциплине** |  |
| **«Разработка серверных частей интернет ресурсов»** | |
| Выполнил студент группы ИКБО-30-20 | Бессмертных И.В. |
| Принял ассистент | Благирев М.М. |

|  |  |  |
| --- | --- | --- |
| Практические работы выполнены | « » 2022г. |  |
| «Зачтено» | « » 2022 г. |  |

Москва 2022

СОДЕРЖАНИЕ

[1. Цель работы 3](#_Toc120873403)

[2. Ход работы 4](#_Toc120873404)

[3. Ответы на вопросы к практической работе 5](#_Toc120873405)

[4. Ссылка на удаленный репозиторий проекта 12](#_Toc120873406)

[ЗАКЛЮЧЕНИЕ 13](#_Toc120873407)

[СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 14](#_Toc120873408)

1. Цель работы

Для выполнения каждой части задания предполагается использование различных расширений и сторонних модулей PHP.

Предполагается выполнить модификацию разрабатываемого в процессе первых 5 практических работ интернет-ресурса. Для этого выполнить следующие действия

1. Сгенерировать набор фикстур с помощью выбранного стороннего модуля. Каждая фикстура должна содержать не менее 5 полей. Всего фикстур должно быть не менее 50.
2. На основе этих фикстур построить минимум 3 графика различного типа с помощью сторонних модулей PHP.
3. График превратить в картинку и добавить на него ваш личный полупрозрачный водяной знак с помощью библиотеки GD
4. (строго рекомендуется) или другой библиотеки для работы с изображениями в PHP.
5. Отобразить полученные графики на странице статистики

Предполагается создание стабильной версии интернет-ресурса и сохранение предыдущей функциональности с практических работ 1-5.

В отчете требуется отразить и описать спецификацию реализованной функциональности, провести и отразить тестирование реализованной функциональности. А также описать инструменты для установки и настройки сторонних модулей и процесс их использования для установки инструментов, используемых в данной практической работе.

1. Ход работы

В проекте создана директория graphs, в которой содержатся 3 файла. (Рисунок 2.1). Файл data.php отвечает за данные, draw\_graphics\_diagram.php за круговую диаграмму на странице и draw\_graphics\_line.php за линейный график на странице

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2.1 – Папка проекта

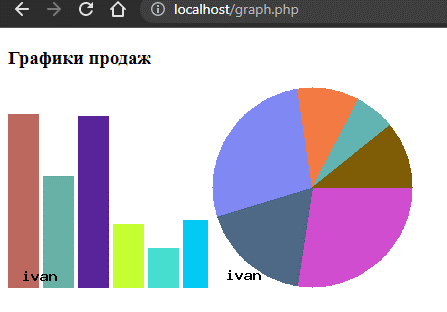


Рисунок 2.2 – Результат работы

1. Ответы на вопросы к практической работе

Что такое Composer и для чего он используется?

Composer — это менеджер для подключения и управления сторонними библиотеками или пакетам в PHP. Его называют пакетный менеджер.

**Опишите работу с календарем в PHP: опишите преобразование дат из одного календаря в другой**

jdtogregorian — Переводит число дней в юлианском летоисчислении в дату по Григорианскому календарю

jdtogregorian(int $julian\_day): string

Переводит число дней в юлианском летоисчислении в строку, содержащую григорианскую дату в формате "месяц/день/год".

cal\_from\_jd — Преобразует дату, заданную в юлианском календаре, в дату указанного календаря

cal\_from\_jd(int $julian\_day, int $calendar): array

cal\_from\_jd() преобразует дату юлианского календаря, заданную в julian\_day, в дату указанного календаря calendar. Поддерживаемые значения calendar: CAL\_GREGORIAN, CAL\_JULIAN, CAL\_JEWISH и CAL\_FRENCH.

**Опишите понятие серверное время и процесс форматирования времени для разных часовых поясов**

Чтобы получить время сервера, используйте функции time() или date() (есть и другие методы, но их должно быть достаточно). Однако time() может вызывать медленно, как и другие функции, которые получают текущее время (поскольку для получения времени требуются системные вызовы). Вы также должны знать, что он будет меняться от вызова к вызову, если выполнение сценария между вызовами занимает более 1 секунды. Это может вызвать проблемы с согласованностью. Обходной путь заключается в использовании $\_SERVER['REQUEST\_TIME'], если это возможно, и устранит ошибки постоянства и проблемы со скоростью, которые могут быть вызваны повторным вызовом time(), date() и т. Д., Которые активно получают текущее системное время.

**Опишите способ получения времени заката и рассвета с использованием языка программирования PHP**

date\_sun\_info ( int $time , float $latitude , float $longitude ).

**Опишите константы, используемые в модуле “Время и дата”**

TZ.

**Приведите принципы арифметики даты и времени?**

+-

**Чем отличается фреймворк от библиотеки? Приведите пример?**

«Фреймворк» отличается от понятия библиотеки тем, что библиотека может быть использована в программном продукте просто как набор подпрограмм близкой функциональности, не влияя на архитектуру программного продукта и не накладывая на неё никаких ограничений.

**Опишите возможные форматы даты и времени и примеры их использования?**

Dd, mm, yy.

**Опишите работу с датой и временем в подходе ООП**

[PHP-расширение Date / Time](https://translate.googleusercontent.com/translate_c?depth=1&rurl=translate.google.ru&sl=en&sp=nmt4&tl=ru&u=http://www.php.net/manual/en/book.datetime.php&xid=17259,1500004,15700021,15700186,15700190,15700259,15700271,15700302&usg=ALkJrhiSYcq-Id_gD1IVUVeHe0wsnyWusw) представляет собой набор классов, которые позволяют вам работать практически со всеми задачами, связанными с датой и временем. Он был доступен с момента выпуска PHP 5.2, и расширение представило несколько новых классов, все из которых сопоставлены с реальными сценариями.

**Опишите использование класса DateTime**

Листинг - Примеры методов DateTime

/\* Методы \*/

public \_\_construct(string $datetime = "now", ?DateTimeZone $timezone = null)

public add(DateInterval $interval): DateTime

public static createFromFormat(string $format, string $datetime, ?DateTimeZone $timezone = null): DateTime|false

public static createFromImmutable(DateTimeImmutable $object): DateTime

public static createFromInterface(DateTimeInterface $object): DateTime

public static getLastErrors(): array|false

public modify(string $modifier): DateTime|false

public static \_\_set\_state(array $array): DateTime

public setDate(int $year, int $month, int $day): DateTime

public setISODate(int $year, int $week, int $dayOfWeek = 1): DateTime

public setTime(

int $hour,

int $minute,

int $second = 0,

int $microsecond = 0

): DateTime

public setTimestamp(int $timestamp): DateTime

public setTimezone(DateTimeZone $timezone): DateTime

public sub(DateInterval $interval): DateTime

public diff(DateTimeInterface $targetObject, bool $absolute = false): DateInterval

public format(string $format): string

public getOffset(): int

public getTimestamp(): int

public getTimezone(): DateTimeZone|false

public \_\_wakeup(): void

**Опишите использование класса DateTimeImmutable**

Листинг - Примеры констант и методов DateTimeImmutable

class DateTimeImmutable implements DateTimeInterface {

/\* Наследуемые константы \*/

const string DateTimeInterface::ATOM = "Y-m-d\TH:i:sP";

const string DateTimeInterface::COOKIE = "l, d-M-Y H:i:s T";

const string DateTimeInterface::ISO8601 = "Y-m-d\TH:i:sO";

const string DateTimeInterface::RFC822 = "D, d M y H:i:s O";

const string DateTimeInterface::RFC850 = "l, d-M-y H:i:s T";

const string DateTimeInterface::RFC1036 = "D, d M y H:i:s O";

const string DateTimeInterface::RFC1123 = "D, d M Y H:i:s O";

const string DateTimeInterface::RFC7231 = "D, d M Y H:i:s \G\M\T";

const string DateTimeInterface::RFC2822 = "D, d M Y H:i:s O";

const string DateTimeInterface::RFC3339 = "Y-m-d\TH:i:sP";

const string DateTimeInterface::RFC3339\_EXTENDED = "Y-m-d\TH:i:s.vP";

const string DateTimeInterface::RSS = "D, d M Y H:i:s O";

const string DateTimeInterface::W3C = "Y-m-d\TH:i:sP";

/\* Методы \*/

public \_\_construct(string $datetime = "now", ?DateTimeZone $timezone = null)

public add(DateInterval $interval): DateTimeImmutable

public static createFromFormat(string $format, string $datetime, ?DateTimeZone $timezone = null): DateTimeImmutable|false

public static createFromInterface(DateTimeInterface $object): DateTimeImmutable

public static createFromMutable(DateTime $object): DateTimeImmutable

public static getLastErrors(): array|false

public modify(string $modifier): DateTimeImmutable|false

public static \_\_set\_state(array $array): DateTimeImmutable

public setDate(int $year, int $month, int $day): DateTimeImmutable

public setISODate(int $year, int $week, int $dayOfWeek = 1): DateTimeImmutable

public setTime(

int $hour,

int $minute,

int $second = 0,

int $microsecond = 0

): DateTimeImmutable

public setTimestamp(int $timestamp): DateTimeImmutable

public setTimezone(DateTimeZone $timezone): DateTimeImmutable

public sub(DateInterval $interval): DateTimeImmutable

public diff(DateTimeInterface $targetObject, bool $absolute = false): DateInterval

public format(string $format): string

public getOffset(): int

public getTimestamp(): int

public getTimezone(): DateTimeZone|false

public \_\_wakeup(): void

}.

**Опишите использование класса DateTimeZone**

Листинг - Примеры констант и методов DateTimeZone

class DateTimeZone {

/\* Константы \*/

const int AFRICA = 1;

const int AMERICA = 2;

const int ANTARCTICA = 4;

const int ARCTIC = 8;

const int ASIA = 16;

const int ATLANTIC = 32;

const int AUSTRALIA = 64;

const int EUROPE = 128;

const int INDIAN = 256;

const int PACIFIC = 512;

const int UTC = 1024;

const int ALL = 2047;

const int ALL\_WITH\_BC = 4095;

const int PER\_COUNTRY = 4096;

/\* Методы \*/

public \_\_construct(string $timezone)

public getLocation(): array|false

public getName(): string

public getOffset(DateTimeInterface $datetime): int

public getTransitions(int $timestampBegin = PHP\_INT\_MIN, int $timestampEnd = PHP\_INT\_MAX): array|false

public static listAbbreviations(): array

public static listIdentifiers(int $timezoneGroup = DateTimeZone::ALL, ?string $countryCode = null): array.

**Какие библиотеки используются для работы с изображениями в PHP**

Goutte, JpGraph

**Опишите основные возможности библиотеки GD**

GD Graphics Library (GD) — программная библиотека, написанная Томасом Баутелом (Thomas Boutell) и другими разработчиками для динамической работы с изображениями.

**Как использовать Composer для подключения библиотек к проекту**

Composer скачивает и устанавливает библиотеки по их имени. Это означает, что сначала нужно найти нужную библиотеку, перейти на её сайт, и найти там в описании её имя. Например, название библиотеки может быть таким: fzaninotto/faker.

**Что такое PEAR? В чём разница работы PEAR и Composer**

PEAR - это аббревиатура от "PHP Extension and Application Repository" (Репозиторий приложений и модулей PHP).

**Как использовать PEAR для установки библиотек?**

pear install <имя библиотеки>.

**Как использовать Composer для обработки зависимостей PEAR?**

all deps, only req deps.

**Что такое PECL?**

PECL — это репозиторий модулей для PHP, написанных на C, доступных через систему пакетов PEAR. PECL был создан, когда возникла проблема удаления некоторых модулей из стандартной поставки PHP. Модули PECL разработаны в соответствии со стандартами кодирования, которые приняты командой разработчиков PHP.

**Как называется репозиторий, содержащий Composer-совместимые библиотеки?**

Pickle.

**С помощью какой библиотеки можно получить детальный отчет о работе приложения?**

PHP BenchMark

**С помощью какой библиотеки можно упростить себе работу с регулярными выражениями в PHP?**

RegExp Builder

**С помощью какой библиотеки возможны быстрые и эффективные запросы на PHP, данная библиотека является аналогом jQuery?**

phpQuery.

**С помощью какой библиотеки возможно простое и эффективное генерирование документов в формате PDF?**

mPDF

**С помощью какой библиотеки возможен эффективный парсинг HTML/XML?**

Goutte

**С помощью какой библиотеки возможно создание диаграмм на движке GOOGLE?**

**GoogChart.**

**С помощью какой библиотеки возможно сканирование конфигурационного файла PHP на предмет безопасности?**

IniScan.

**С помощью какой библиотеки возможен анализ HTML и удаление вредоносного кода для защиты от XSS атак?**

HTML Purifier

**С помощью какой библиотеки возможно построение графиков, диаграмм и другого структурированного контента на PHP.**

JpGraph

**С помощью какой библиотеки облегчается процесс загрузки и валидации файлов на PHP**

Upload

1. Ссылка на удаленный репозиторий проекта

Полный код проекта можно найти по ссылке:

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

ЗАКЛЮЧЕНИЕ

Была создана конфигурация серверного программного обеспечения на наборе LAMP и проверена ее работоспособность. Данная конфигурация будет использоваться для выполнения следующих практических работ по данной дисциплине и для выполнения курсового проектирования.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. «Конспект лекции №1»: [Электронный ресурс]. URL: <https://online-edu.mirea.ru/mod/resource/view.php?id=403993> (дата обращения 01.12.2022)
2. «Руководство по Docker Compose»: [Электронный ресурс]. URL: <https://habr.com/ru/company/ruvds/blog/450312/> (дата обращения 01.12.2022)
3. «Docker Hub»: [Электронный ресурс]. URL: <https://hub.docker.com/> (дата обращения 01.12.2022)