

Лабораторна робота №7

Тема: Встановлення HTTP-статус кодів та функції керування буферизацією виводу.

Мета роботи: ознайомитися з HTTP-статусами кодів та функціями керування буферизацією виводу.

Хід роботи:

Завдання 1: Динамічне кешування сторінки з урахуванням статус-коду

Створіть PHP-скрипт `cache_page.php`, який реалізує кешування сторінки залежно від статус-коду відповіді.

Умови:

1. Якщо сторінка віддає статус 200 OK, її вміст зберігається у файлі кешу `cache.html`.
2. Якщо статус 404 Not Found, кешування не відбувається, а файл `cache.html` (якщо він існує) видаляється.
3. Якщо користувач відвідує сторінку повторно, і є кеш-файл, вміст віддається з нього без повторної генерації сторінки.

Лістинг:

```
<?php
```

```
ob_start();
```

```
header("Content-Type: text/html; charset=UTF-8");
```

```
$status = 200;
```

```
$content = "<h1>Це контент сторінки</h1>";
```

```
$cache_file = "cache.html";
```

					Житомирська політехніка.24.121.01.000 – Лр7			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Гнідий Б.С.			Звіт з лабораторної роботи			
Перевір.		Ковтун В.В.						
Керівник								
Н. контр.								
Зав. каф.								
					Літ.	Арк.	Аркушів	
							1	12
					ФІКТ Гр. ВТк-24-1			

```

if (isset($_GET['status']) && $_GET['status'] == 404) {
    $status = 404;
    $content = "<h1>Сторінку не знайдено</h1>";
    if (file_exists($cache_file)) unlink($cache_file);
    http_response_code(404);
} elseif (file_exists($cache_file)) {
    echo file_get_contents($cache_file);
    exit;
} else {
    http_response_code(200);
    echo $content;
    file_put_contents($cache_file, ob_get_contents());
}
ob_end_flush();
?>

```

<h1>Це контент сторінки</h1>

Результат виконання:

Це контент сторінки

Завдання 2. Система захисту від перевантаження сервера (Rate Limiting):

Створіть скрипт `rate_limit.php`, який обмежує кількість запитів від одного користувача та повертає відповідні статус-коди.

Умови:

- Логувати IP-адресу користувача в файл `requests.log`.
- Якщо користувач зробив більше 5 запитів за хвилину, встановлювати статус 429 Too Many Requests і рекомендувати спробувати пізніше.

		Гнідий Б.С.			Житомирська політехніка.24.121.01.000 – Лр7	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		2

- Якщо ліміт не перевищено, видавати 200 ОК та звичайний контент.

Лістинг:

```
<?php
ob_start();

$ip = $_SERVER['REMOTE_ADDR'];
$log_file = "requests.log";
$limit = 5;
$interval = 60;

$time = time();
$entries = file_exists($log_file) ? file($log_file, FILE_IGNORE_NEW_LINES |
FILE_SKIP_EMPTY_LINES) : [];
$entries = array_filter($entries, function($line) use ($time, $interval) {
    $parts = explode("|", $line);
    if (count($parts) < 2) return false;
    return ($time - (int)$parts[1]) < $interval;
});

file_put_contents($log_file, implode("\n", $entries) . "\n", LOCK_EX);
$entry_count = 0;
foreach ($entries as $entry) {
    $parts = explode("|", $entry);
    if (count($parts) >= 2 && $parts[0] === $ip) $entry_count++;
}

if ($entry_count >= $limit) {
    http_response_code(429);
    echo "Забагато запитів. Спробуйте пізніше.";
} else {
```

		Гнідий Б.С.			Житомирська політехніка.24.121.01.000 – Лр7	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		3

```

file_put_contents($log_file, "$ip$time\n", FILE_APPEND);
http_response_code(200);
echo "Ви можете переглянути сторінку.";
}

ob_end_flush();
?>

```

Результат виконання:

Ви можете переглянути сторінку.

Завдання 3. Перехоплення помилок та перенаправлення:

Створіть скрипт `error_handler.php`, який перехоплює фатальні помилки та замість стандартного повідомлення показує кастомну сторінку з відповідним статус-кодом.

Умови:

- Використовуйте `register_shutdown_function()`, щоб обробляти `E_ERROR`.
- Якщо сталася помилка:
 - Очистити буфер `ob_clean()`
 - Встановити статус 500 Internal Server Error
 - Показати сторінку з вибаченням та часом, коли проблема буде вирішена.
- Якщо помилок немає, віддавати 200 ОК.

Лістинг:

`error_handler.php`

		Гнідий Б.С.			Житомирська політехніка.24.121.01.000 – Лр7	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		4

```

<?php
ob_start();

function shutdownHandler() {
    $error = error_get_last();
    if ($error && $error['type'] === E_ERROR) {
        ob_clean();
        http_response_code(500);
        echo "<h1>500 Internal Server Error</h1><p>Ми працюємо над вирішенням
проблеми.</p>";
    }
}

register_shutdown_function('shutdownHandler');

// Фатальна помилка для тесту
undefined_function();

ob_end_flush();
?>

```

Результат виконання:

500 Internal Server Error

Ми працюємо над вирішенням проблеми.

Завдання 4. Гнучка система редиректів:

		Гнідий Б.С.			Житомирська політехніка.24.121.01.000 – Лр7	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		5

Створіть скрипт `redirect_manager.php`, який дозволяє керувати перенаправленнями через конфігураційний файл.

Умови:

1. Є файл `redirects.json` зі списком перенаправлень:

```
{  
  "/old-page": "/new-page",  
  "/deprecated": "/404"  
}
```
2. Якщо користувач заходить на `/old-page`, його потрібно перенаправити на `/new-page` зі статусом 301 Moved Permanently.
3. Якщо сторінка `/deprecated`, видавати 404 Not Found та відповідне повідомлення.
4. Використовувати `ob_start()`, щоб контролювати вивід.

Лістинг:

```
<?php  
ob_start();  
$uri = $_SERVER['REQUEST_URI'];  
$redirects = json_decode(file_get_contents("redirects.json"), true);  
  
if (isset($redirects[$uri])) {  
    $target = $redirects[$uri];  
    if ($target === "/404") {  
        http_response_code(404);  
        echo "Сторінку не знайдено."  
    } else {  
        header("Location: $target", true, 301);  
        exit;  
    }  
} else {
```

		Гнідий Б.С.			Житомирська політехніка.24.121.01.000 – Лр7	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		6

```

        echo "Поточна сторінка: $uri";
    }
    ob_end_flush();
?>

```

```

{
    "/old-page": "/new-page",
    "/deprecated": "/404",
    "/start": "/task1/cache_page.php",
    "/limit": "/task2/rate_limit.php"
}

```

Результат виконання:

Поточна сторінка: /Lab/Lab7/task4/redirect_manager.php

Завдання 5. Динамічне підключення заголовків без видимого виводу:

Створіть PHP-клас Response, який допомагає керувати HTTP-заголовками та буферизацією.

Функціонал класу:

- Метод setStatus(\$code), який встановлює HTTP-статус.
- Метод addHeader(\$header), який додає заголовок (наприклад, Content-Type).
- Метод send(\$content), який очищає буфер, додає заголовки та відправляє відповідь.

Лістинг:

		Гнідий Б.С.			Житомирська політехніка.24.121.01.000 – Лр7	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		7

```

<?php
require_once "Response.php";
ob_start();

$response = new Response();
$response->setStatus(200);
    $response->addHeader("Content-Type: text/html; charset=UTF-8");
$response->send("<h1>Вітаємо!</h1><p>Це відповідь, сформована через
PHP.</p>");
?>

```

```

<?php
class Response {
    private $status;
    private $headers = [];

    public function setStatus($code) {
        $this->status = $code;
        http_response_code($code);
    }

    public function addHeader($header) {
        $this->headers[] = $header;
    }

    public function send($content) {
        ob_clean();
        foreach ($this->headers as $header) {
            header($header);

```

		Гнідий Б.С.			Житомирська політехніка.24.121.01.000 – Лр7	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		8


```

    }
    echo $content;
    ob_end_flush();
}
}
?>

```

Результат виконання:

Вітаємо!

Це відповідь, сформована через PHP.

Завдання 6. Аналіз трафіку та відповідей сервера:

Розробіть traffic_logger.php, який запише інформацію про всі запити в базу даних.

Що зберігати в БД?

- IP-адресу
- Час запиту
- Запитаний URL
- Відповідний HTTP-статус

Додаткові функції:

1. Реалізувати перегляд статистики через stats.php, який рахує кількість 404 помилок за останню добу.
2. Якщо відсоток 404 помилок перевищує 10% від загальної кількості запитів, надсилати адміністратору повідомлення.

Лістинг:

```

<?php
$conn = new mysqli("localhost", "root", "", "lab7");
if ($conn->connect_error) {

```

		Гнідий Б.С.			Житомирська політехніка.24.121.01.000 – Лр7	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		9

```

        die("Помилка з'єднання з базою даних");
    }
    $res = $conn->query("SELECT COUNT(*) as total FROM traffic_logs WHERE created_at >= NOW() - INTERVAL 1 DAY");
    $total = $res->fetch_assoc()['total'] ?? 0;

    $res404 = $conn->query("SELECT COUNT(*) as err FROM traffic_logs WHERE status_code=404 AND created_at >= NOW() - INTERVAL 1 DAY");
    $errors = $res404->fetch_assoc()['err'] ?? 0;

    echo "404 за добу: $errors з $total запитів<br>";
    if ($total > 0 && ($errors / $total) > 0.1) {
        echo " Понад 10% 404 помилок! Повідомлено адміністратора.";
    }
    ?>

<?php
$conn = new mysqli("localhost", "root", "", "lab7");
$ip = $_SERVER['REMOTE_ADDR'];
$url = $_SERVER['REQUEST_URI'];
$status = http_response_code();
$stmt = $conn->prepare("INSERT INTO traffic_logs (ip, url, status_code) VALUES (?, ?, ?)");
$stmt->bind_param("ssi", $ip, $url, $status);
$stmt->execute();
?>

```

Результат виконання:

		Гнідий Б.С.			Житомирська політехніка.24.121.01.000 – Лр7	Арк.
		Ковтун В.В				10
Змн.	Арк.	№ докум.	Підпис	Дата		

404 за добу: 0 з 71 запитів

404 за добу: 0 з 123 запитів

Висновок

У ході лабораторної роботи №7 я ознайомився з HTTP-статус кодами, їх призначенням і застосуванням на практиці. Також було розглянуто механізми буферизації виводу в PHP, що дозволяє ефективно керувати порядком і умовами виведення контенту. Практичне завдання допомогло закріпити теоретичні знання та зрозуміти, як коректно реагувати на помилки, зокрема код 404, і реалізовувати кешування відповідей.

		Гнідий Б.С.			Житомирська політехніка.24.121.01.000 – Лр7	Арк.
		Ковтун В.В				11
Змн.	Арк.	№ докум.	Підпис	Дата		