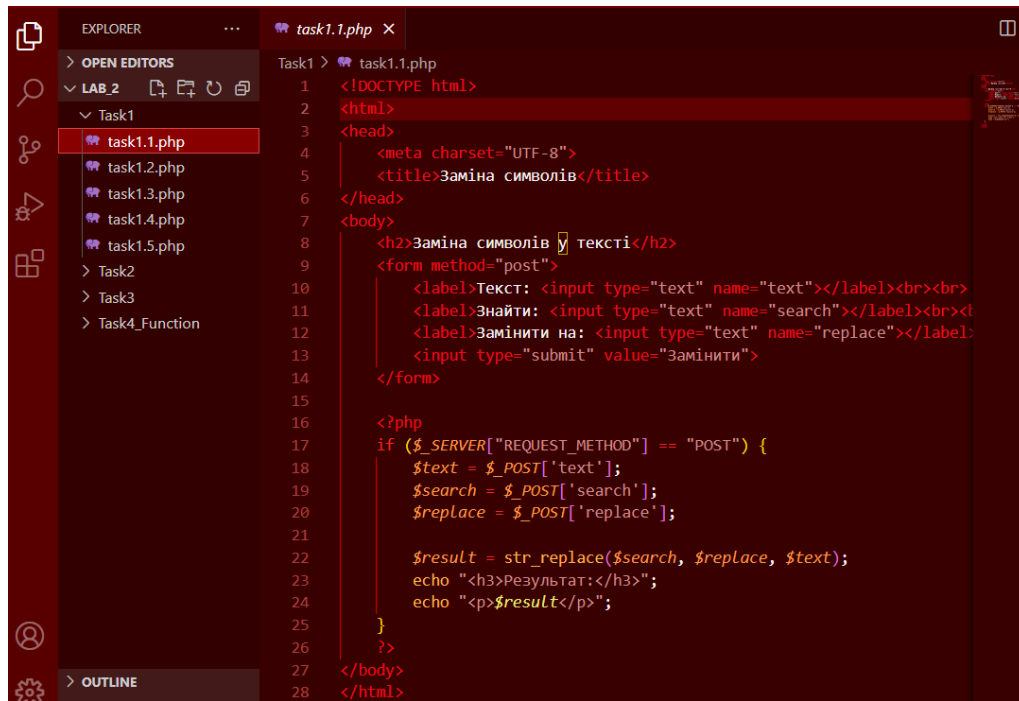


## Лабораторна робота №2

### Тема: Використання функцій, рядків, масивів та форм в PHP

#### Завдання 1. Робота з рядками (5 завдань)

1. Створіть 4 текстових поля з мітками *Текст*, *Знайти*, *Замінити* і *Результат*. Виконати можливість заміни символів і вивести результат.



```
task1.1.php
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="UTF-8">
5 <title>Заміна символів</title>
6 </head>
7 <body>
8 <h2>Заміна символів у тексті</h2>
9 <form method="post">
10 <label>Текст: <input type="text" name="text"></label><br><br>
11 <label>Знайти: <input type="text" name="search"></label><br><br>
12 <label>Замінити на: <input type="text" name="replace"></label>
13 <input type="submit" value="Замінити">
14 </form>
15
16 <?php
17 if ($_SERVER["REQUEST_METHOD"] == "POST") {
18     $text = $_POST['text'];
19     $search = $_POST['search'];
20     $replace = $_POST['replace'];
21
22     $result = str_replace($search, $replace, $text);
23     echo "<h3>Результат:</h3>";
24     echo "<p>$result</p>";
25 }
26 ?>
27 </body>
28 </html>
```

### Заміна символів у тексті

Текст:

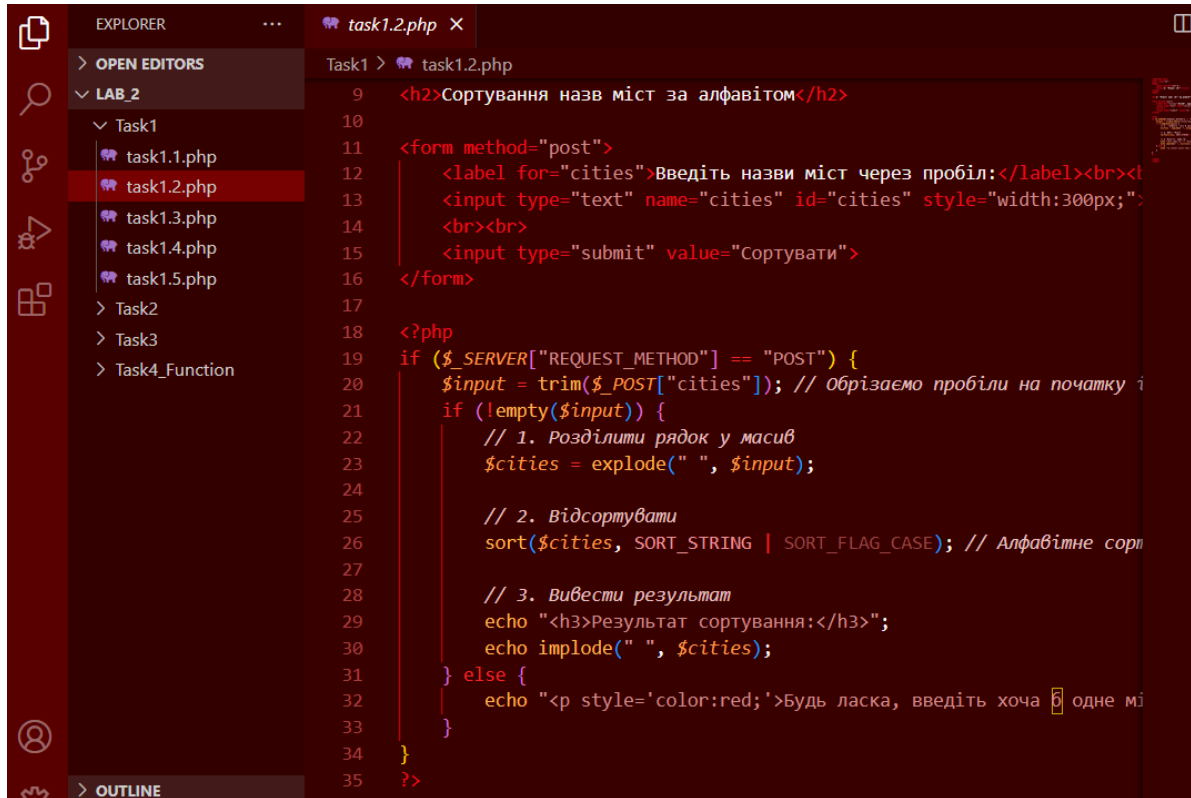
Знайти:

Замінити на:

**Результат:**

зло

2. Користувач вводить назви міст через пробіл. Переставте назви так, щоб назви були впорядковані за алфавітом.



```
9 <h2>Сортування назв міст за алфавітом</h2>
10
11 <form method="post">
12 <label for="cities">Введіть назви міст через пробіл:</label><br><input
13 <input type="text" name="cities" id="cities" style="width:300px;">
14 <br><br>
15 <input type="submit" value="Сортувати">
16 </form>
17
18 <?php
19 if ($_SERVER["REQUEST_METHOD"] == "POST") {
20     $input = trim($_POST["cities"]); // Обрізаємо пробіли на початку і
21     if (!empty($input)) {
22         // 1. Розділити рядок у масив
23         $cities = explode(" ", $input);
24
25         // 2. Відсортувати
26         sort($cities, SORT_STRING | SORT_FLAG_CASE); // Алфавітне сортування
27
28         // 3. Вивести результат
29         echo "<h3>Результат сортування:</h3>";
30         echo implode(" ", $cities);
31     } else {
32         echo "<p style='color:red;'>Будь ласка, введіть хоча б одне місто";
33     }
34 }
35 ?>
```

## Сортування назв міст за алфавітом

Введіть назви міст через пробіл:

Сортувати

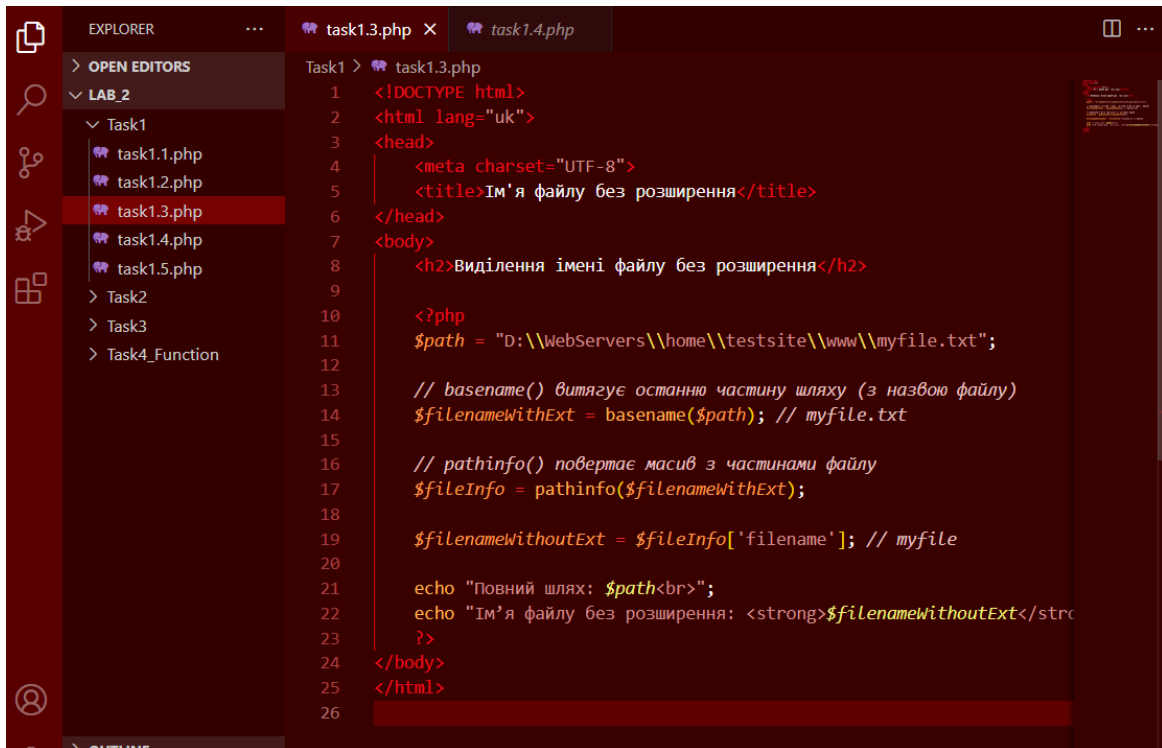
Результат сортування:

житомир кіїв львів одеса

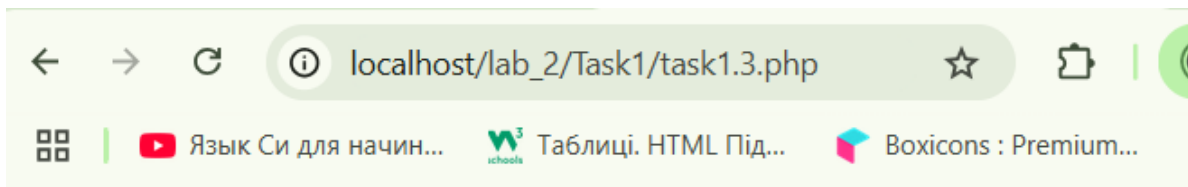
3. Дан рядок, що містить повне ім'я файлу, наприклад:

**D: \ WebServers \ home \ testsite \ www \ myfile.txt**

Виділіть з цього рядка ім'я файлу без розширення



```
1 <!DOCTYPE html>
2 <html lang="uk">
3 <head>
4     <meta charset="UTF-8">
5     <title>Ім'я файлу без розширення</title>
6 </head>
7 <body>
8     <h2>Виділення імені файлу без розширення</h2>
9
10    <?php
11        $path = "D:\\WebServers\\home\\testsite\\www\\myfile.txt";
12
13        // basename() витягує останню частину шляху (з назвою файлу)
14        $filenameWithExt = basename($path); // myfile.txt
15
16        // pathinfo() повертає масив з частинами файлу
17        $fileInfo = pathinfo($filenameWithExt);
18
19        $filenameWithoutExt = $fileInfo['filename']; // myfile
20
21        echo "Повний шлях: $path<br>";
22        echo "Ім'я файлу без розширення: <strong>$filenameWithoutExt</strong>";
23    ?>
24 </body>
25 </html>
26
```

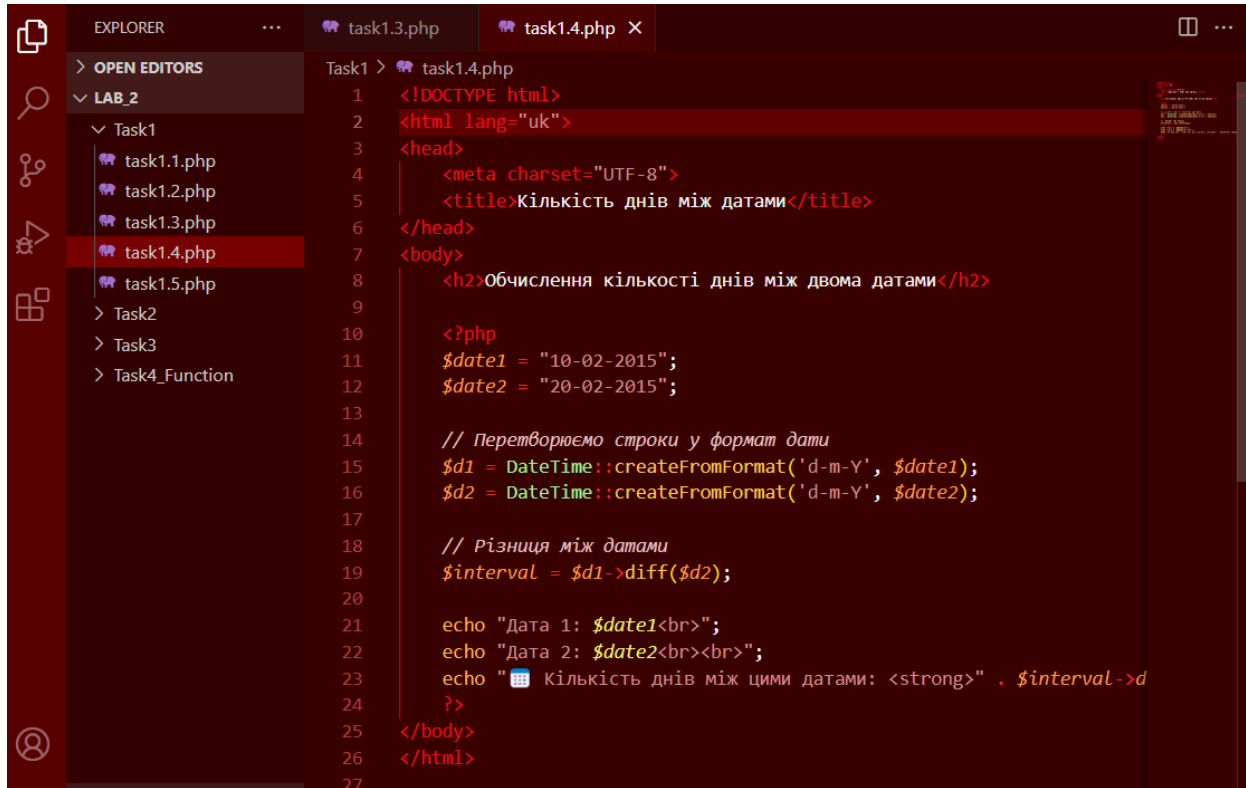


## Виділення імені файлу без розширення

Повний шлях: D:\WebServers\home\testsite\www\myfile.txt

Ім'я файлу без розширення: **myfile**

4. У двох рядках містяться дати виду День-Місяць-Рік (наприклад, 10-02-2015). Визначте кількість днів між датами.



The screenshot shows a code editor with a sidebar on the left displaying a file explorer. The main editor area shows a PHP file named 'task1.4.php'. The code is as follows:

```
1 <!DOCTYPE html>
2 <html lang="uk">
3 <head>
4     <meta charset="UTF-8">
5     <title>Кількість днів між датами</title>
6 </head>
7 <body>
8     <h2>Обчислення кількості днів між двома датами</h2>
9
10    <?php
11        $date1 = "10-02-2015";
12        $date2 = "20-02-2015";
13
14        // Перетворюємо строки у формат дати
15        $d1 = DateTime::createFromFormat('d-m-Y', $date1);
16        $d2 = DateTime::createFromFormat('d-m-Y', $date2);
17
18        // Різниця між датами
19        $interval = $d1->diff($d2);
20
21        echo "Дата 1: $date1<br>";
22        echo "Дата 2: $date2<br><br>";
23        echo "📅 Кількість днів між цими датами: <strong>" . $interval->d
24        ?>
25    </body>
26 </html>
27
```

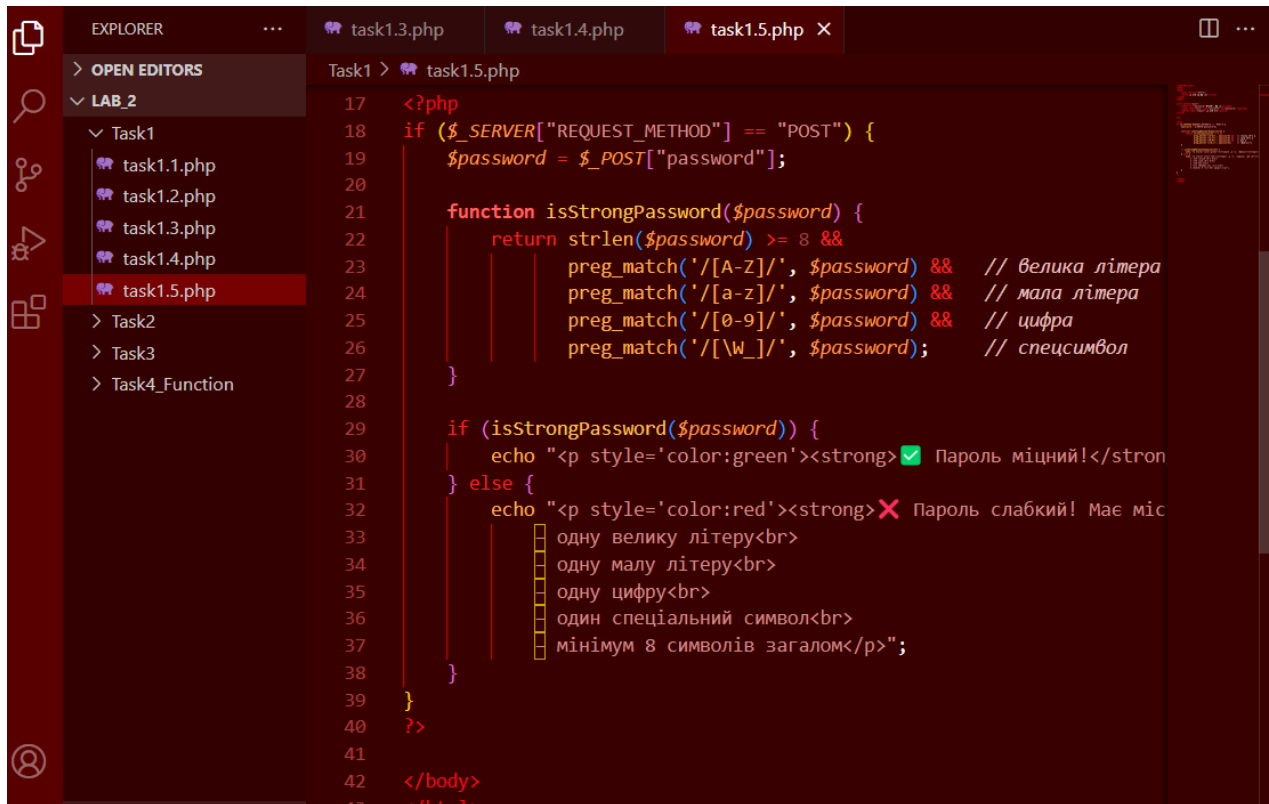
## Обчислення кількості днів між двома датами

Дата 1: 10-02-2015

Дата 2: 20-02-2015

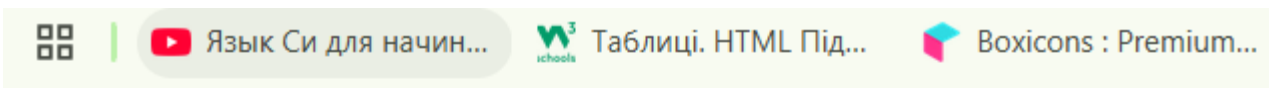
📅 Кількість днів між цими датами: **10**

5. Генератор паролів: Напишіть програму, яка генерує випадковий пароль заданої довжини. Пароль може містити великі та малі літери, цифри та спеціальні символи.



The screenshot shows a code editor with a sidebar on the left displaying a file explorer. The main editor area shows a PHP file named 'task1.5.php'. The code is as follows:

```
17 <?php
18 if ($_SERVER["REQUEST_METHOD"] == "POST") {
19     $password = $_POST["password"];
20
21     function isStrongPassword($password) {
22         return strlen($password) >= 8 &&
23             preg_match('/[A-Z]/', $password) && // велика літера
24             preg_match('/[a-z]/', $password) && // мала літера
25             preg_match('/[0-9]/', $password) && // цифра
26             preg_match('/[\W_]/', $password); // спецсимвол
27     }
28
29     if (isStrongPassword($password)) {
30         echo "<p style='color:green'><strong>✔ Пароль міцний!</strong>";
31     } else {
32         echo "<p style='color:red'><strong>✘ Пароль слабкий! Має містити:
33         <ul style='list-style-type: none; padding-left: 20px;'>
34             <li>одну велику літеру<br>
35             <li>одну малу літеру<br>
36             <li>одну цифру<br>
37             <li>один спеціальний символ<br>
38             <li>мінімум 8 символів загалом</li>
39         </ul>
40     }
41 }
42 </body>
43 </html>
```



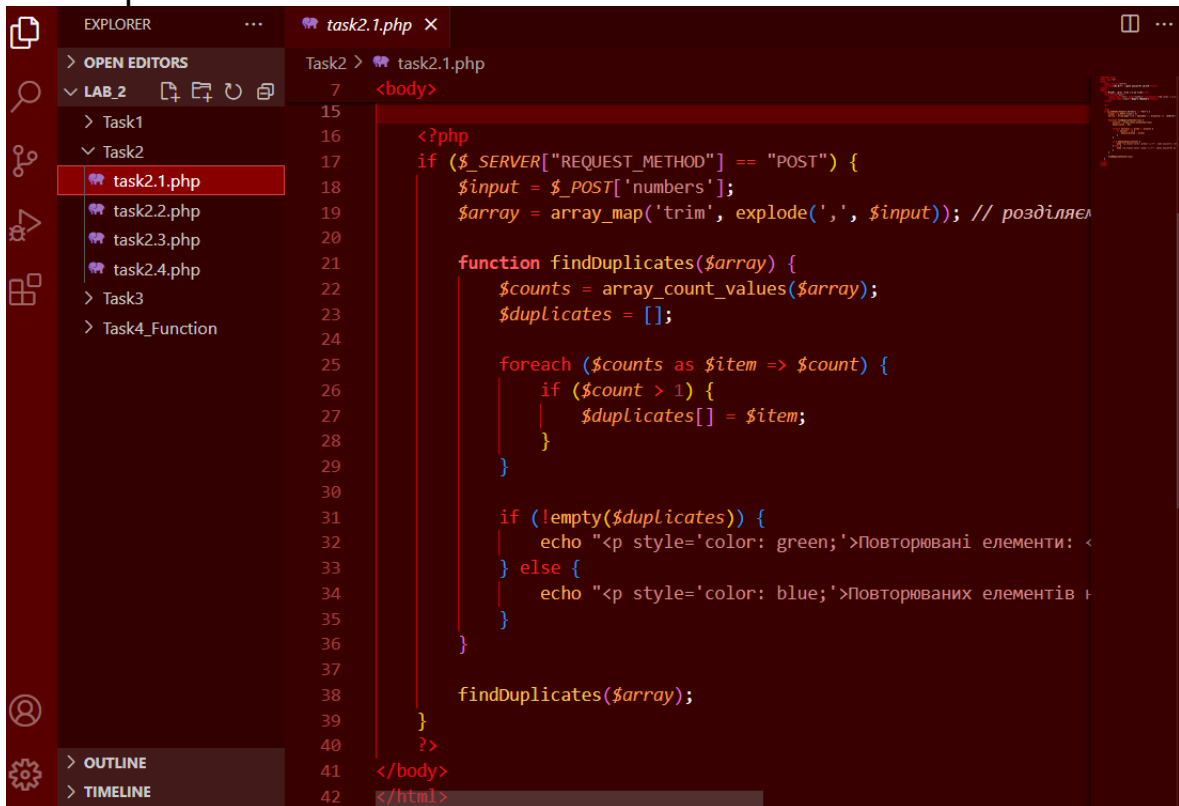
The taskbar at the bottom of the screen shows several application icons: a window icon, a YouTube icon with the text 'Язык Си для начин...', a 'w3schools' logo, a 'Таблиці. HTML Під...' icon, and a 'Boxicons : Premium...' icon.

Введіть пароль:

✔ Пароль міцний!

## Завдання 2. Робота з масивами (4 завдання)

1. Написати функцію, яка приймає масив. Вивести елементи масиву, що повторюються.



```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $input = $_POST['numbers'];
    $array = array_map('trim', explode(',', $input)); // розділяєм

    function findDuplicates($array) {
        $counts = array_count_values($array);
        $duplicates = [];

        foreach ($counts as $item => $count) {
            if ($count > 1) {
                $duplicates[] = $item;
            }
        }

        if (!empty($duplicates)) {
            echo "<p style='color: green;'>Повторювані елементи: <
        } else {
            echo "<p style='color: blue;'>Повторюваних елементів н
        }
    }

    findDuplicates($array);
}
?>
</body>
</html>
```

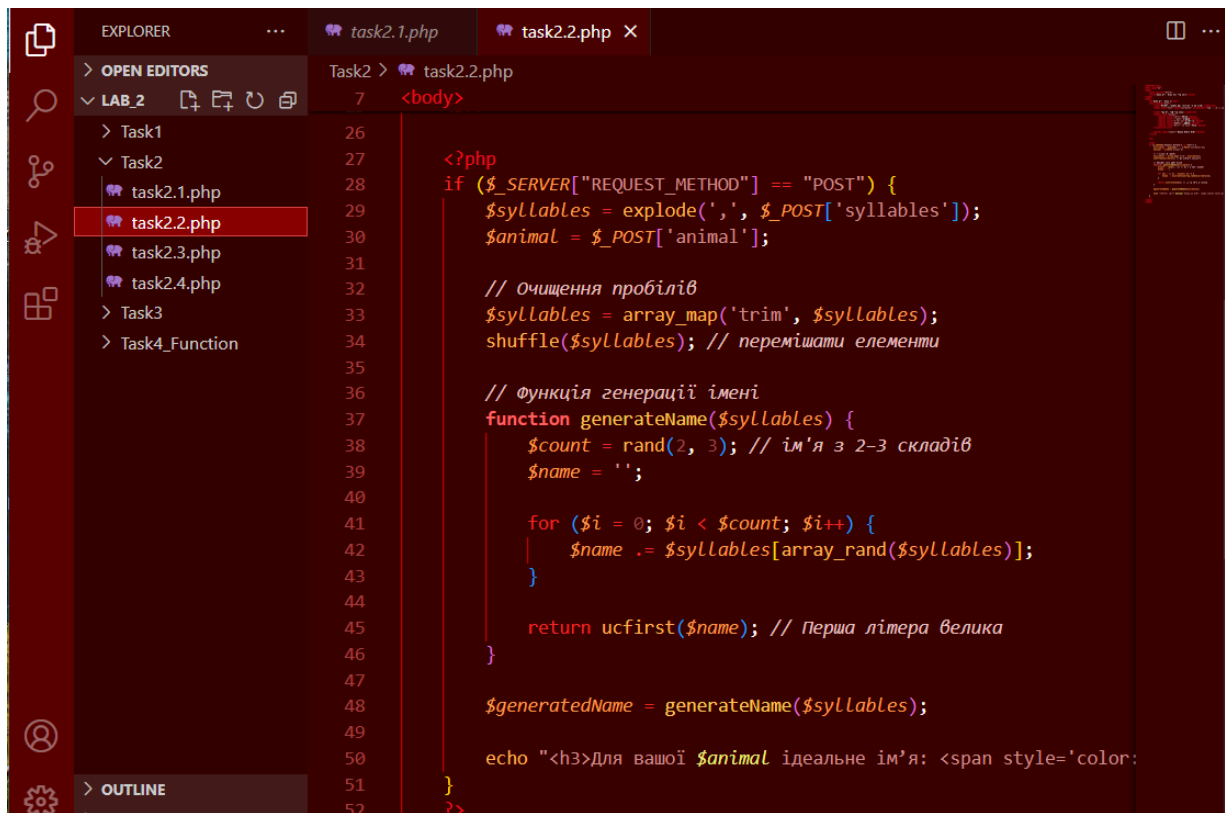
**Введіть масив чисел (через кому):**

Наприклад: 1,2,3,2,4,5,1

Знайти дублікати

Повторювані елементи: 1, 6

2. Написати функцію-генератор імен. Функція приймає масив з складів або символів. Функція має повертати згенероване ім'я кішки, собаки, хом'ячка або іншої тваринки



```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $syllables = explode(',', $_POST['syllables']);
    $animal = $_POST['animal'];

    // Очищення пробілів
    $syllables = array_map('trim', $syllables);
    shuffle($syllables); // перемішати елементи

    // Функція генерації імені
    function generateName($syllables) {
        $count = rand(2, 3); // ім'я з 2-3 складів
        $name = '';

        for ($i = 0; $i < $count; $i++) {
            $name .= $syllables[array_rand($syllables)];
        }

        return ucfirst($name); // Перша літера велика
    }

    $generatedName = generateName($syllables);

    echo "<h3>Для вашої $animal ідеальне ім'я: <span style='color:"
```

## Генератор імен 🐾

Введіть склади або символи через кому:

Оберіть тип тваринки:

Кішка ▼

Згенерувати ім'я

Для вашої хом'ячка ідеальне ім'я: **бетка**

### 3. Створіть 2 функції:

- функцію **createArray()**, яка створює масив з випадковою довжиною (від 3 до 7) і випадковими значеннями (від 10 до 20)
- функцію , яка приймає 2 масиви і виконує наступні дії:
  - З'єднує обидва масиви в один
  - Вилучає повторювані елементи з отриманого масиву
  - Відсортовує отриманий масив за зростанням і повертає його
- Створіть 2 масиви за допомогою функції **createArray()** та продемонструйте виконання функції з 2-го пункту

```
> Task1 16
> Task2 17
  task2.1.php 18
  task2.2.php 19
  task2.3.php 20
  task2.4.php 21
> Task3 22
> Task4_Function 23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49

<?php
// Функція для створення випадкового масиву
function createArray() {
    $length = rand(3, 7);
    $array = [];

    for ($i = 0; $i < $length; $i++) {
        $array[] = rand(10, 20);
    }

    return $array;
}

// Функція для об'єднання, очищення від повторів і сортування
function processArrays($arr1, $arr2) {
    $combined = array_merge($arr1, $arr2); // з'єднання
    $unique = array_unique($combined); // видалення дубів
    sort($unique); // сортування за
    return $unique;
}

// Якщо кнопка натиснута
if (isset($_POST['generate'])) {
    $array1 = createArray();
    $array2 = createArray();
    $result = processArrays($array1, $array2);

    echo "<h3>Масив 1: [" . implode(" ", $array1) . "]</h3>";
    echo "<h3>Масив 2: [" . implode(" ", $array2) . "]</h3>";
    echo "<h3>Результат: [" . implode(" ", $result) . "]</h3>";
}
?>
</body>
</html>
```

#### 12 34 Робота з випадковими масивами

Згенерувати та обробити масиви

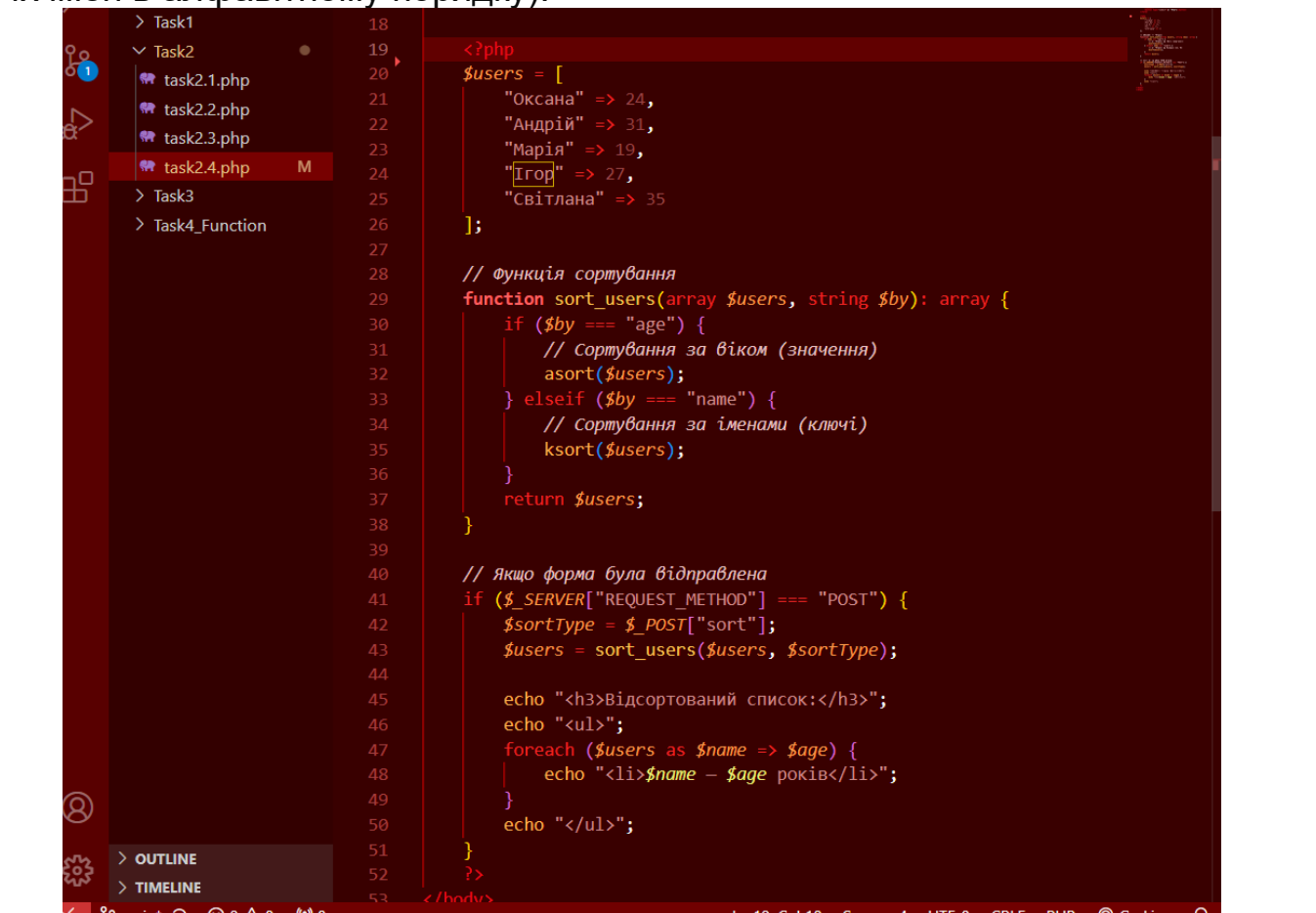
**Масив 1:** [10, 15, 12, 18]

**Масив 2:** [12, 14, 11, 17, 19, 18]

**Результат:** [10, 11, 12, 14, 15, 17, 18, 19]



4) Створіть асоціативний масив, де ключами будуть імена користувачів, а значеннями - їхній вік. Реалізуйте функцію для сортування цього масиву в залежності від параметру, переданого у функцію: (віком користувачів або їх імен в алфавітному порядку).



```
18
19
20 <?php
21 $users = [
22     "Оксана" => 24,
23     "Андрій" => 31,
24     "Марія" => 19,
25     "Ігор" => 27,
26     "Світлана" => 35
27 ];
28
29 // Функція сортування
30 function sort_users(array $users, string $by): array {
31     if ($by === "age") {
32         // Сортування за віком (значення)
33         asort($users);
34     } elseif ($by === "name") {
35         // Сортування за іменами (ключі)
36         ksort($users);
37     }
38     return $users;
39 }
40
41 // Якщо форма була відправлена
42 if ($_SERVER["REQUEST_METHOD"] === "POST") {
43     $sortBy = $_POST["sort"];
44     $users = sort_users($users, $sortBy);
45
46     echo "<h3>Відсортований список:</h3>";
47     echo "<ul>";
48     foreach ($users as $name => $age) {
49         echo "<li>$name — $age років</li>";
50     }
51     echo "</ul>";
52 }
53 </body>
```

## Сортування користувачів

Сортувати за:

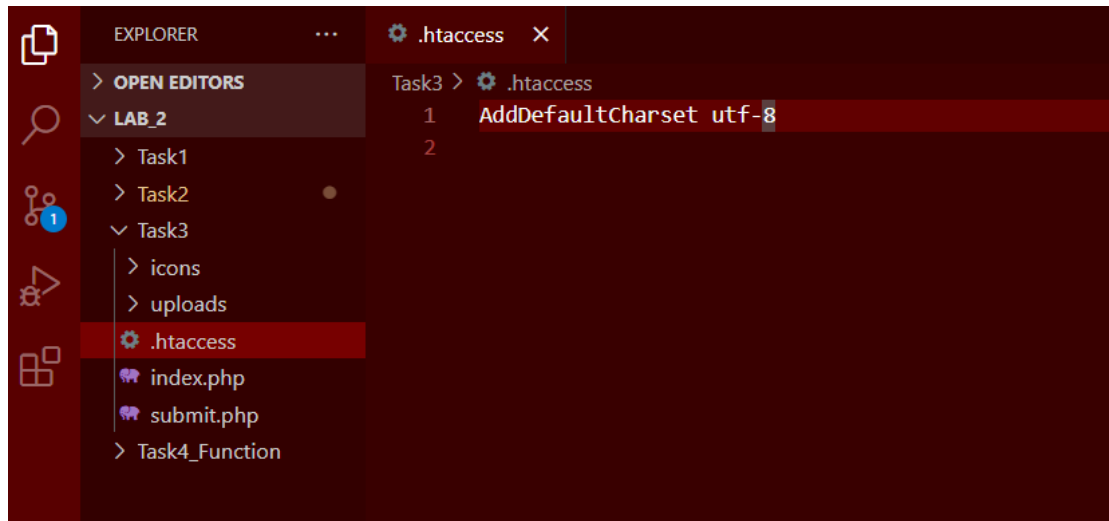
**Відсортований список:**

- Ігор — 27 років
- Андрій — 31 років
- Марія — 19 років
- Оксана — 24 років
- Світлана — 35 років

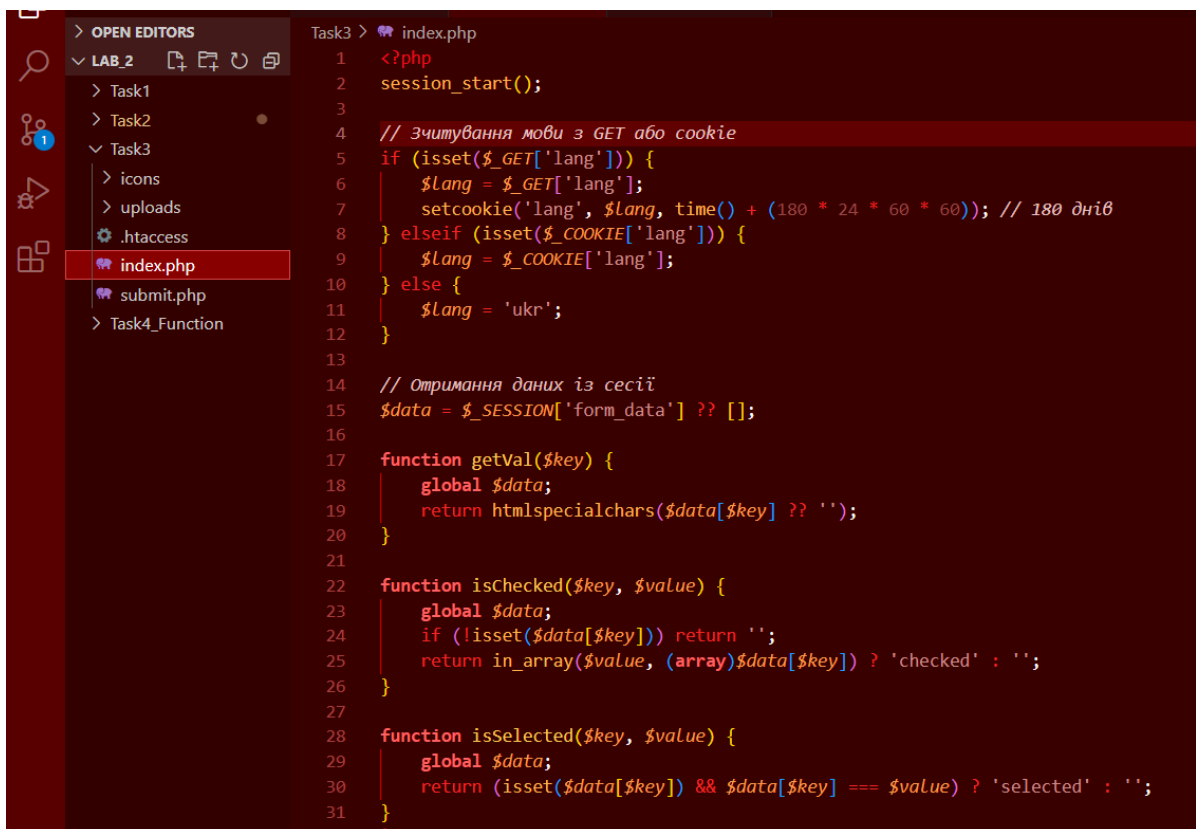
### Завдання 3. Робота з формою (1 завдання)

1) Створити файл .htaccess, написати у ньому рядок:

**AddDefaultCharset utf-8**



2) Створити файл index.php



3) На головній сторінці створіть форму з такими елементами:

Логін:

Пароль:

Пароль (ще раз):

Стать: ☒ чоловік ☐ жінка

Місто:

Улюблені гри: ☐ футбол  
☐ баскетбол  
☐ волейбол  
☐ шахи  
☐ World of Tanks

Про себе:

Фотографія:  Файл не выбран

```
Task3 > index.php
40 <body>
41 </div>
42 <p><strong>Вибрана мова:</strong> <?= ($lang == 'ukr') ? 'Українська' : 'English' ?></p>
43 </div>
44 <form action="submit.php" method="post" enctype="multipart/form-data">
45     Логін: <input type="email" name="login" value="<?= getVal('login') ?>"><br>
46     Пароль: <input type="password" name="password1"><br>
47     Пароль (ще раз): <input type="password" name="password2"><br>
48
49     Стать:
50     <label><input type="radio" name="gender" value="чоловік" <?= getVal('gender') == 'чоловік' ? 'checked' : '' ?>> чоловік</label>
51     <label><input type="radio" name="gender" value="жінка" <?= getVal('gender') == 'жінка' ? 'checked' : '' ?>> жінка</label><br>
52
53     Місто:
54     <select name="city">
55         <?php
56         $cities = ['Житомир', 'Київ', 'Львів', 'Харків'];
57         foreach ($cities as $city) {
58             echo "<option value='$city' ". isselected('city', $city) . ">$city</option>";
59         }
60     </select><br>
61
62     Улюблені ігри:<br>
63     <label><input type="checkbox" name="games[]" value="футбол" <?= isChecked('games', 'футбол') ?>> футбол</label><br>
64     <label><input type="checkbox" name="games[]" value="баскетбол" <?= isChecked('games', 'баскетбол') ?>> баскетбол</label><br>
65     <label><input type="checkbox" name="games[]" value="волейбол" <?= isChecked('games', 'волейбол') ?>> волейбол</label><br>
66     <label><input type="checkbox" name="games[]" value="шахи" <?= isChecked('games', 'шахи') ?>> шахи</label><br>
67     <label><input type="checkbox" name="games[]" value="World of Tanks" <?= isChecked('games', 'World of Tanks') ?>> World of Tanks</label><br>
68
69     Про себе:<br>
70     <textarea name="about"><?= getVal('about') ?></textarea><br>
71
72     Фотографія: <input type="file" name="photo"><br>
73
74     <button type="submit">Зареєструватися</button>
75 </form>
76 </div>
77 </body>
78 </html>
```



**Вибрана мова:** Українська

Логін:

Пароль:

Пароль (ще раз):

Стать: ☐ чоловік ☒ жінка

Місто:

Улюблені ігри:

- ☒ футбол
- ☐ баскетбол
- ☐ волейбол
- ☐ шахи
- ☐ World of Tanks

Про себе:

Фотографія:  04 (1).jpg

\

4) На іншій сторінці реалізуйте прийом даних цієї форми та відображення переданої інформації, у т.ч. фотографії. Наприклад так:

Логін: morozov.andriy@gmail.com

Пароль: не співпадає (перший - 5 символів, другий - 7 символів)

Стать: чоловік

Місто: Житомир

Улюблені ігри: футбол  
шахи

Про себе: Тут можна ввести фрагмент тексту.  
Реалізуйте відображення текст з нового рядка  
там, де користувач ставить Enter.  
Це можна зробити функцією nl2br(\$str).

Фотографія:



[Повернутися на головну сторінку](#)

Логін: anastasiaIgorivna@gmail.com

Пароль: не співпадає (перший - 11 символів, другий - 0 символів)

Стать: жінка

Місто: Житомир

Улюблені ігри:  
футбол

Про себе:

Фото:



#### Завдання 4. Робота з функціями (1 завдання)

1. Створіть папку з назвою **"Function"** і в ній створіть файл **func.php** з власними функціями (sin, cos, tg, my\_tg(x), x<sup>y</sup>, x!). Потім цей файл підключити за допомогою конструкцій include, require)
- 2) Створіть файл **index.php**, який буде головною сторінкою і міститиме форму для вводу даних та виклик функцій:

<b>x</b>	<b>y</b>	
<input type="text" value="4"/>	<input type="text" value="3"/>	<input type="button" value="="/>

- 3) Створіть файл **calculate.php**, який буде обробляти введені дані, викликати функції та виводити результати:

<b>x<sup>y</sup></b>	<b>x!</b>	<b>my_tg(x)</b>	<b>sin(x)</b>	<b>cos(x)</b>	<b>tg(x)</b>
64	24	1.15782128235	-0.756802495308	-0.653643620864	1.15782128235

```
1 <?php
2 include_once('Function/func.php');
3
4 $x = $_POST['x'];
5 $y = $_POST['y'] ?? null;
6 $op = $_POST['operation'];
7 $result = '';
8
9 switch ($op) {
10     case 'x_pow_y':
11         $result = x_pow_y($x, $y);
12         break;
13     case 'fact':
14         $result = fact($x);
15         break;
16     case 'my_tg':
17         $result = my_tg($x);
18         break;
19     case 'sin':
20         $result = sin_x($x);
21         break;
22     case 'cos':
23         $result = cos_x($x);
24         break;
25     case 'tg':
26         $result = tg_x($x);
27         break;
28     default:
29         $result = 'Невідома операція';
30 }
31
32 echo "<h2>Результат: $result</h2>";
33 echo "<a href='index.php'>Назад</a>";
34 ?>
```

# Калькулятор математичних функцій

Введіть x:

Введіть y (для  $x^y$ ):

Оберіть операцію:

▼

---

Результат: 1024

[Назад](#)

Висновок: ознайомилась з роботою масивів, рядків, функцій та форм.