**Лабораторна робота №2**

**Використання функцій, рядків, масивів та форм в РНР**

**Мета:** Ознайомитись з роботою масивів, рядків, функцій та форм

Виконання лабораторної роботи:

Завдання 1:

1. Створіть 4 текстових поля з мітками *Текст*, *Знайти,* *Замінити і Результат*. Виконати  можливість заміни символів і вивести результат.
2. Користувач вводить назви міст через пробіл. Переставте назви так, щоб назви були впорядковані за алфавітом.
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

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

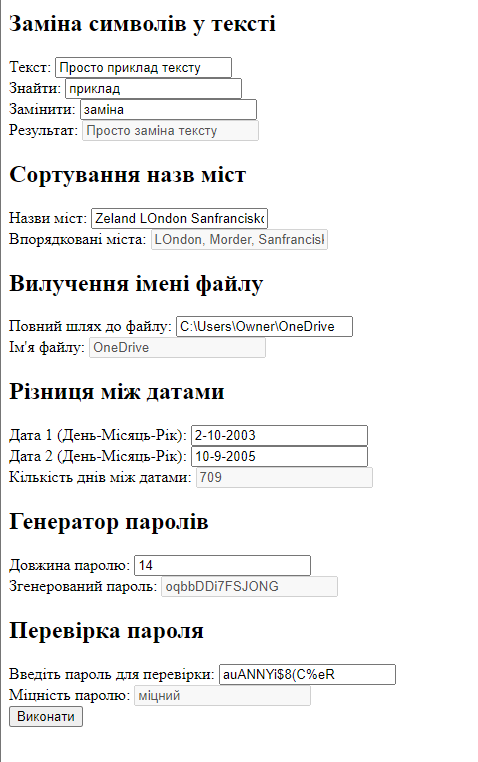
1. Генератор паролів: Напишіть програму, яка генерує випадковий пароль заданої довжини. Пароль може містити великі та малі літери, цифри та спеціальні символи.

*Перевірка пароля:* Реалізуйте функцію, яка перевіряє, чи є введений користувачем пароль достатньо міцним. Вважайте пароль міцним, якщо він містить принаймні одну велику літеру, одну малу літеру, одну цифру та один спеціальний символ, і його довжина є не менше 8 символів.

*Виконання завдання:*

<?php  
function generatePassword($length = 8) {  
 $chars = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789!@#$%^&\*()";  
 $password = "";  
 for ($i = 0; $i < $length; $i++) {  
 $password .= $chars[rand(0, strlen($chars) - 1)];  
 }  
 return $password;  
}  
  
function checkPasswordStrength($password) {  
 return preg\_match("/^(?=.\*[a-z])(?=.\*[A-Z])(?=.\*\d)(?=.\*[@#$%^&\*()])[A-Za-z\d@#$%^&\*()]{8,}$/", $password);  
}  
  
// Обробка введення форми  
if ($\_SERVER["REQUEST\_METHOD"] == "POST") {  
 // Заміна символів у тексті  
 $text = $\_POST["text"] ?? "";  
 $search = $\_POST["search"] ?? "";  
 $replace = $\_POST["replace"] ?? "";  
 $result = str\_replace($search, $replace, $text);  
  
 // Сортування назв міст  
 $cities = $\_POST['cities'] ?? '';  
 $citiesArray = preg\_split('/[\s,]+/', $cities);  
 $citiesArray = array\_map('trim', $citiesArray);  
 $citiesArray = array\_filter($citiesArray);  
 sort($citiesArray);  
 $sortedCities = implode(', ', $citiesArray);  
  
 // Вилучення імені файлу  
 $filePath = $\_POST["filePath"] ?? "";  
 $fileName = pathinfo($filePath, *PATHINFO\_FILENAME*);  
  
 // Різниця між датами  
 $date1 = isset($\_POST["date1"]) ? DateTime::*createFromFormat*("d-m-Y", $\_POST["date1"]) : null;  
 $date2 = isset($\_POST["date2"]) ? DateTime::*createFromFormat*("d-m-Y", $\_POST["date2"]) : null;  
 $diff = $date1 && $date2 ? $date1->diff($date2)->days : 0;  
  
 // Генерація пароля  
 $passwordLength = $\_POST["passwordLength"] ?? 8;  
 $generatedPassword = generatePassword($passwordLength);  
  
 // Перевірка пароля  
 $inputPassword = $\_POST["inputPassword"] ?? "";  
 $passwordStrength = checkPasswordStrength($inputPassword) ? "міцний" : "слабкий";  
}  
?>  
  
<!DOCTYPE html>  
<html lang="uk">  
<head>  
 <meta charset="UTF-8">  
 <title>Лабораторна робота</title>  
</head>  
<body>  
  
<form method="post">  
 <h2>Заміна символів у тексті</h2>  
 Текст: <input type="text" name="text" value="<?php echo htmlspecialchars($text); ?>"><br>  
 Знайти: <input type="text" name="search" value="<?php echo htmlspecialchars($search); ?>"><br>  
 Замінити: <input type="text" name="replace" value="<?php echo htmlspecialchars($replace); ?>"><br>  
 Результат: <input type="text" value="<?php echo htmlspecialchars($result); ?>" disabled><br>  
  
 <h2>Сортування назв міст</h2>  
 Назви міст: <input type="text" name="cities" value="<?php echo htmlspecialchars($cities); ?>"><br>  
 Впорядковані міста: <input type="text" value="<?php echo htmlspecialchars($sortedCities); ?>" disabled><br>  
  
 <h2>Вилучення імені файлу</h2>  
 Повний шлях до файлу: <input type="text" name="filePath" value="<?php echo htmlspecialchars($filePath); ?>"><br>  
 Ім'я файлу: <input type="text" value="<?php echo htmlspecialchars($fileName); ?>" disabled><br>  
  
 <h2>Різниця між датами</h2>  
 Дата 1 (День-Місяць-Рік): <input type="text" name="date1" value="<?php echo htmlspecialchars($\_POST["date1"]); ?>"><br>  
 Дата 2 (День-Місяць-Рік): <input type="text" name="date2" value="<?php echo htmlspecialchars($\_POST["date2"]); ?>"><br>  
 Кількість днів між датами: <input type="text" value="<?php echo htmlspecialchars($diff); ?>" disabled><br>  
  
 <h2>Генератор паролів</h2>  
 Довжина паролю: <input type="text" name="passwordLength" value="<?php echo htmlspecialchars($passwordLength); ?>"><br>  
 Згенерований пароль: <input type="text" value="<?php echo htmlspecialchars($generatedPassword); ?>" disabled><br>  
  
 <h2>Перевірка пароля</h2>  
 Введіть пароль для перевірки: <input type="text" name="inputPassword" value="<?php echo htmlspecialchars($inputPassword); ?>"><br>  
 Міцність паролю: <input type="text" value="<?php echo htmlspecialchars($passwordStrength); ?>" disabled><br>  
 <input type="submit" value="Виконати">  
</form>  
  
</body>  
</html>

Результат виконання завдання:



Зобр 2.1 Результат виконання першого завдання

Завдання 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

1. 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

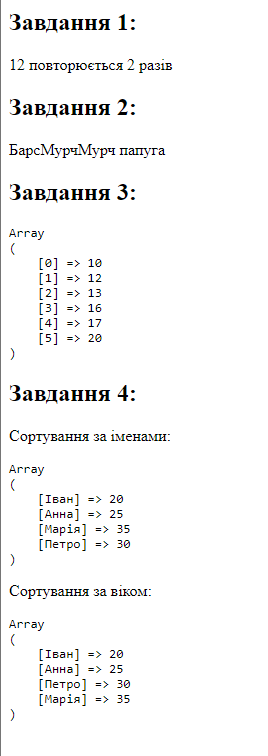
4) Створіть асоціативний масив, де ключами будуть імена користувачів, а

значеннями - їхній вік. Реалізуйте функцію для сортування цього масиву в залежності від параметру, переданого у функцію: (віком користувачів або їх імен в алфавітному порядку).

Виконання завдання:

<?php  
  
// Завдання 1: Функція для виведення елементів масиву, що повторюються  
function findDuplicates($array) {  
 $duplicates = array\_count\_values($array);  
 foreach ($duplicates as $value => $count) {  
 if ($count > 1) {  
 echo "$value повторюється $count разів<br>";  
 }  
 }  
}  
  
// Завдання 2: Функція-генератор імен тварин  
function generateAnimalName($parts) {  
 $animals = ['кішка', 'собака', 'хом\'як', 'папуга'];  
 $name = '';  
 for ($i = 0; $i < count($parts); $i++) {  
 $index = rand(0, count($parts) - 1);  
 $name .= $parts[$index];  
 }  
 return $name . ' ' . $animals[rand(0, count($animals) - 1)];  
}  
  
// Завдання 3: Функція для об'єднання, вилучення повторів та сортування масиву  
function mergeArraysAndSort($array1, $array2) {  
 $mergedArray = array\_merge($array1, $array2);  
 $uniqueArray = array\_unique($mergedArray);  
 sort($uniqueArray);  
 return $uniqueArray;  
}  
  
// Завдання 4: Сортування асоціативного масиву за іменами або віком  
function sortUsers($users, $sortBy = 'ім\'я') {  
 if ($sortBy === 'вік') {  
 asort($users);  
 } else {  
 ksort($users);  
 }  
 return $users;  
}  
  
// Функція для створення масиву з випадковою довжиною та випадковими значеннями  
function createArray() {  
 $length = rand(3, 7);  
 $array = [];  
 for ($i = 0; $i < $length; $i++) {  
 $array[] = rand(10, 20);  
 }  
 return $array;  
}  
  
// Приклади масивів та асоціативного масиву  
$array1 = createArray();  
$array2 = createArray();  
$parts = ['Барс', 'Мурч', 'Шерст'];  
$users = [  
 'Анна' => 25,  
 'Петро' => 30,  
 'Іван' => 20,  
 'Марія' => 35  
];  
  
?>  
  
<!DOCTYPE html>  
<html lang="uk">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0">  
 <title>Робота з масивами</title>  
</head>  
<body>  
  
<h2>Завдання 1:</h2>  
<p><?php findDuplicates($array1); ?></p>  
  
<h2>Завдання 2:</h2>  
<p><?php echo generateAnimalName($parts); ?></p>  
  
<h2>Завдання 3:</h2>  
<p><pre><?php print\_r(mergeArraysAndSort($array1, $array2)); ?></pre></p>  
  
<h2>Завдання 4:</h2>  
<p>Сортування за іменами:</p>  
<p><pre><?php print\_r(sortUsers($users)); ?></pre></p>  
  
<p>Сортування за віком:</p>  
<p><pre><?php print\_r(sortUsers($users, 'вік')); ?></pre></p>  
  
</body>  
</html>

Результат виконання завдання:



Зобр 2.2 Результат виконання другого завдання

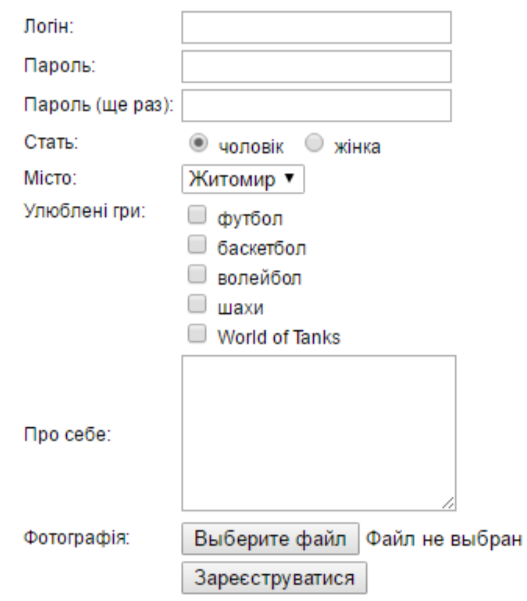
Завдання 3:

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

**AddDefaultCharset utf-8**

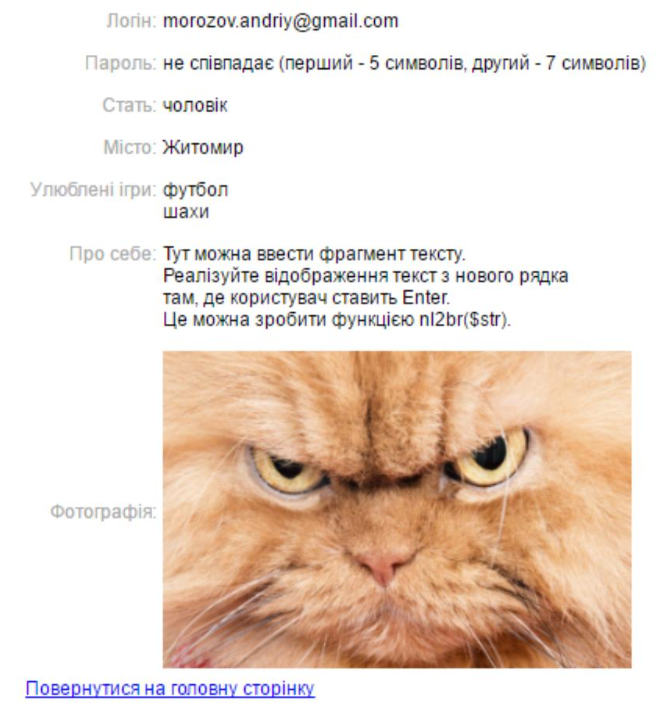
2) Створити файл **index.php**

3) На головній сторінці створіть форму з такими елементами:

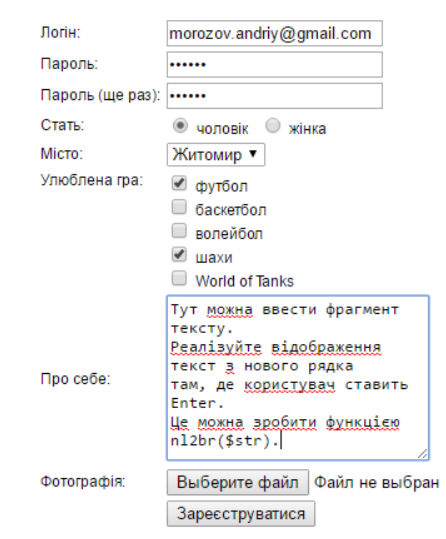


*Не забувайте, щоб форма могла передавати файли, у тезі форм потрібно вказувати атрибути* **enctype="multipart/form-data" method="post"**

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



5) Запишіть усі передані дані у сесію. Реалізуйте заповнення значень форми на головній сторінці значеннями, які зберігаються у формі. Тобто, щоб при переході на головну сторінку елементи форми заповнилися тими значеннями, які збережені у сесії (окрім фотографії):



6) На головній сторінці додайте іконки для вибору мови:



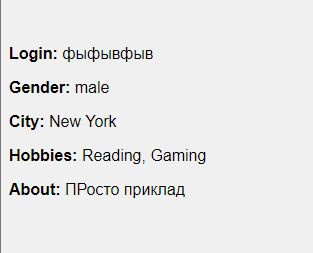
7) Кожну іконку оберніть у посилання вигляду: **index.php?lang=ukr** і реалізуйте у файлі **index.php** обробник GET-запиту з вибраною мовою. Вибрану мову запишіть у кукі-змінну з часом життя півроку і при відображенні файла **index.php** реалізуйте читання кукі-змінної і виведення фрази вигляду:

**Вибрана мова: Українська**

Виконання завдання:

<?php  
session\_start();  
  
// Масив міст  
$cities = array("New York", "London", "Paris");  
  
// Отримання значення мови з GET або з кукі  
$lang = isset($\_GET['lang']) ? $\_GET['lang'] : (isset($\_COOKIE['lang']) ? $\_COOKIE['lang'] : 'ukr');  
  
// Запис мови у куку на півроку  
setcookie('lang', $lang, time() + (6 \* 30 \* 24 \* 60 \* 60));  
  
// Встановлення мови  
if ($lang == 'ukr') {  
 $language = "Українська";  
} else {  
 $language = "English";  
}  
  
// Заповнення значень форми з сесії  
$login = isset($\_SESSION['formData']['login']) ? $\_SESSION['formData']['login'] : '';  
$gender = isset($\_SESSION['formData']['gender']) ? $\_SESSION['formData']['gender'] : '';  
$city = isset($\_SESSION['formData']['city']) ? $\_SESSION['formData']['city'] : '';  
$hobbies = isset($\_SESSION['formData']['hobbies']) ? $\_SESSION['formData']['hobbies'] : array();  
$about = isset($\_SESSION['formData']['about']) ? $\_SESSION['formData']['about'] : '';  
  
?>  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Registration Form</title>  
 <link rel="stylesheet" href="assets/css/styles.css">  
</head>  
<body>  
<h1>Registration Form</h1>  
<form action="process\_form.php" method="post" enctype="multipart/form-data">  
 <label for="login">Login:</label>  
 <input type="text" name="login" id="login" value="<?php echo $login; ?>"><br>  
  
 <label for="password">Password:</label>  
 <input type="password" name="password" id="password"><br>  
  
 <label for="confirm\_password">Confirm Password:</label>  
 <input type="password" name="confirm\_password" id="confirm\_password"><br>  
  
 <label for="gender">Gender:</label>  
 <select name="gender" id="gender">  
 <option value="male" <?php if ($gender == 'male') echo 'selected'; ?>>Male</option>  
 <option value="female" <?php if ($gender == 'female') echo 'selected'; ?>>Female</option>  
 </select><br>  
  
 <label for="city">City:</label>  
 <select name="city" id="city">  
 <?php foreach ($cities as $ct): ?>  
 <option value="<?php echo $ct; ?>" <?php if ($city == $ct) echo 'selected'; ?>><?php echo $ct; ?></option>  
 <?php endforeach; ?>  
 </select><br>  
  
 <label for="hobbies">Hobbies:</label><br>  
 <input type="checkbox" name="hobbies[]" value="Reading" <?php if (in\_array('Reading', $hobbies)) echo 'checked'; ?>>Reading<br>  
 <input type="checkbox" name="hobbies[]" value="Gaming" <?php if (in\_array('Gaming', $hobbies)) echo 'checked'; ?>>Gaming<br>  
 <input type="checkbox" name="hobbies[]" value="Cooking" <?php if (in\_array('Cooking', $hobbies)) echo 'checked'; ?>>Cooking<br>  
  
 <label for="about">About:</label><br>  
 <textarea name="about" id="about" cols="30" rows="5"><?php echo $about; ?></textarea><br>  
  
 <label for="photo">Photo:</label>  
 <input type="file" name="photo" id="photo"><br>  
  
 <input type="submit" value="Register">  
</form>  
  
<!-- Додавання іконок для вибору мови -->  
<a href="index.php?lang=ukr"><img src="assets/images/ukr\_icon.png" alt="Ukrainian Flag"></a>  
<a href="index.php?lang=eng"><img src="assets/images/eng\_icon.png" alt="English Flag"></a>  
  
<!-- Виведення обраної мови -->  
<p>Вибрана мова: <?php echo $language; ?></p>  
</body>  
</html>

Результат виконання завдання:



Зобр 2.3 Результат виконання третього завдання

Завдання 4:

1. Створіть папку з назвою "**Function**" і в ній створіть файл **func.php** з власними функціями (sin, cos, tg, my\_tg(x), xy, 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:



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

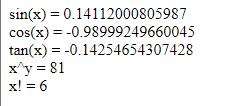
Виконання завдання:

<!DOCTYPE html>  
<html>  
<head>  
 <title>Calculator</title>  
</head>  
<body>  
<h2>Calculator</h2>  
<form action="calculate.php" method="post">  
 <label for="x">Enter x:</label>  
 <input type="text" name="x" id="x">  
 <br><br>  
 <label for="y">Enter y:</label>  
 <input type="text" name="y" id="y">  
 <br><br>  
 <input type="submit" value="Calculate">  
</form>  
</body>  
</html>

}

<?php  
  
include 'Function/func.php';  
  
if ($\_SERVER["REQUEST\_METHOD"] == "POST") {  
 $x = $\_POST["x"];  
 $y = $\_POST["y"];  
  
 // Виклик функцій  
 $sin\_x = my\_sin($x);  
 $cos\_x = my\_cos($x);  
 $tg\_x = my\_tg($x);  
 $pow\_xy = my\_pow($x, $y);  
 $factorial\_x = factorial($x);  
  
 // Виведення результатів  
 echo "sin(x) = $sin\_x<br>";  
 echo "cos(x) = $cos\_x<br>";  
 echo "tan(x) = $tg\_x<br>";  
 echo "x^y = $pow\_xy<br>";  
 echo "x! = $factorial\_x<br>";  
}  
  
?>

Результат виконання завдання:



Зобр 2.4 Результат виконання четвертого завдання

Висновок: В ході виконання цієї лабораторної роботи я вдосконалив свої навички роботи з середовищем програмування php за допомогою практики над масивами

функціями та формами.