# Лабораторна робота №5. Система авторизації з використанням сесії.

**Мета**:навчитися використовувати механізм сесій PHP для системи авторизації. **опанувати навички ділення HTML-сторінок на окремі блоки та генерація динамічних HTML сторінок з використанням мови програмування PHP. Express.js.**

**Теоретичні відомості**

Протокол HTTP є протоколом «без збереження стану». Це означає, що даний протокол не має вбудованого способу збереження стану між двома транзакціями.

**Сесії** та **куки** (**cookies**) призначені для зберігання відомостей про користувачів при переходах між сторінками сайту. При використанні сесій дані зберігаються у тимчасових файлах на сервері. Файли з куками зберігаються на комп’ютері користувача, і за запитом відсилаються броузером сервера.

При роботі з сесіями розрізняють наступні етапи:

відкриття сесії

реєстрація змінних сесії і їх використання закриття сесії

Найпростіший спосіб відкриття сесії полягає у використанні функції session\_start, яка викликається на початку PHP-сценарію:

bool session\_start ( void )

Ця функція перевіряє, чи існує ідентифікатор сесії, і, якщо неіснує, то створює його. Якщо ідентифікатор поточної сесії вже існує, то завантажуються зареєстровані змінні сесії.

Після ініціалізації сесії з’являється можливість зберігати інформацію в глобальному масиві $\_SESSION. Наприклад, є файл index.php у якому в масив

$\_SESSION зберігається змінна і масив.

<?php

// Ініціюємо сесію session\_start();

// Записуємо значення в сесію

$\_SESSION['name'] = "value" ;

//Записуємо масив у сесію

$arr = array("first", "second", "third");

$\_SESSION ['arr'] = $arr;

//Виводимо посилання на іншу сторінку

echo "<a href="other.php">інша сторінка</a>" ;

?>

На сторінках, де відбувається виклик функції session\_start (), значення даних змінних можна витягти з масиву $\_SESSION. У наступному прикладі

коду приводиться вміст сторінки other.php, де беруться дані, раніше записані у сесію на сторінці index.php.

<?php

//Ініціюємо сесію session\_start();

//Виводимо вміст суперглобального масиву $\_SESSION print\_r($\_SESSION);

?>

Після завершення роботи з сесією спочатку потрібно видалити всі змінні сесії (session\_unset), а потім викликати функцію session\_destroy.

Cookies – це текстові дані, що зберігаються на стороні клієнта, і містять пари «ім’я – значення», з якими пов’язаний URL, за яким броузер визначає чи потрібно надсилати cookies на сервер.

Установка cookies проводиться за допомогою функції setcookie.

bool setcookie ( string name [, string value [, INT Expire [, String Path [, String Domain [, INT Secure ]]]]] )

де name – ім’я встановлюваного cookie;

value – значення, що зберігається в cookie з ім’ям name ;

expire – час в секундах з початку епохи-юнікс, по закінченні якого поточні cookie стають недійсними;

path – шлях, по якому доступний cookie; domain – домен, з якого доступний cookie;

secure – директива, яка визначає, чи доступний cookie не по протоколу HTPPS. За замовчуванням ця директива має значення 0, що означає можливість доступу до cookie по звичайному протоколу HTTP.

Сookies треба обов’язково встановлювати перед відправкою в броузер будь-яких заголовків (будь-якої інформації що повинна відображатися на сторінці), оскільки самі cookies встановлюються у вигляді заголовків.

Значення, що зберігається в cookie можна отримати через суперглобальний масив $\_COOKIE.

За замовчуванням cookies встановлюються на один сеанс роботи з броузером, однак можна задати для них більш тривалий термін існування. Це дуже зручна і корисна властивість, оскільки в цьому випадку користувачеві не потрібно надавати свої дані знову при кожному відвідуванні сайту.

Видалити cookie просто. Для цього треба викликати функцію setcookie і передати їй ім’я того cookie, який підлягає видаленню.

**Завдання**

1. Додати до сайту, створеного на попередніх лабораторних роботах, модуль авторизації3 (auth), який буде викликатися за посиланням

«Авторизація» з верхньої або нижньої частини будь якої сторінки сайту (тобто для авторизації не відводитися окремий пункт головного меню).

1. Створити у модулі скрипт index.php який приймає, окрім параметру mod – ім’я модулю, параметр opt – вказую на одну з можливих дій процесу авторизації (login або logout), та виводить форму авторизації4. Форма повинна мати наступну елементи вводу та атрибути:
	* «Логін користувача» (ім’я параметру – username, type=’text’);
	* «Пароль користувача» (ім’я параметру – username, type = ‘password’);
	* кнопка «Авторизація»;
	* приховані параметри mod=”auth” та opt = “login”;
	* action = “index.php”;
	* method = “POST”.
2. Срипт index.php повинен реалізувати наступні дій в залежності від значення параметру opt:
	* якщо такого параметру не існує і користувач вже зареєстрований вивести ім’я користувача, інакше вивести форму авторизації;
	* login – перевіряти реєстраційні дані5, що вводить користувач; якщо данні коректні вивести відповідне повідомлення та зареєструвати його ім’я у змінній сесії ($\_SESSION[“user”] = “admin”), а якщо данні не коректні вивести відповідне повідомлення та форму авторизації;
	* logout – завершити сеанс роботи користувача; видалити всі дані сесії, завершити сесію та виконати перехід на головну сторінку сайту.
3. Якщо користувач зареєстрований (isset($\_SESSION[“user”]) == true) змінити посилання «Авторизація» на посилання «Імя користувача – Завершити сеанс» що виконує завершення сеансу роботи користувача
4. Отримані сторінки та пов'язані з ними дані (папка з картинками, файл стилів) скопіювати в кореневу папку веб-сервера.
5. Протестувати роботу сторінок на віддаленому сервері.
6. опанувати навички ділення HTML-сторінок на окремі блоки та генерація динамічних HTML сторінок з використанням мови програмування PHP. Express.js.
7. Оформити звіт про виконання лабораторної роботи.

**Контрольні питання**

1. Яка різниця між сесією ту cookie?
2. Як виконати перехід на іншу сторінку сайту у коді PHP?
3. Як встановити змінну сесії?
4. Як правильно завершити роботу сесії?

3 Під «додаванням модулю» слід розуміти створення нової папки у папці modules з кореневої директорій сайту.

4 Може бути реалізована у окремому скрипті

5 Для простоти вважати що існує один користувач «admin» з паролем «111»

1. Де правильно викликати функції session\_start()?