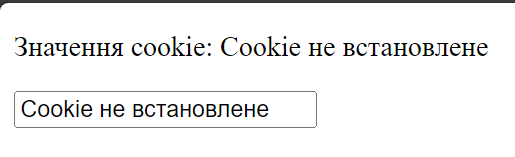
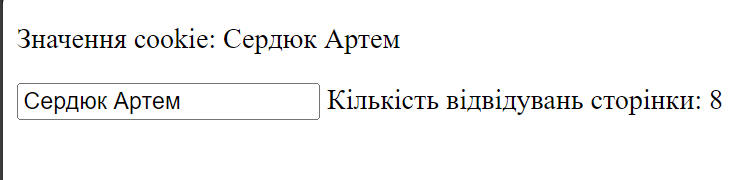
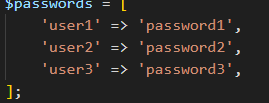
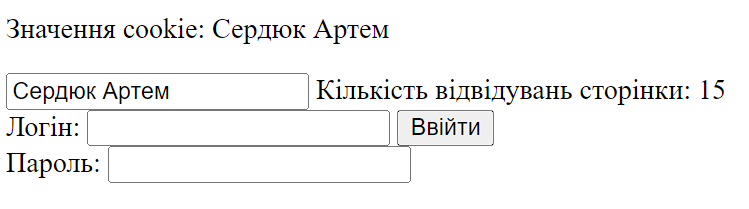
* Створити cookie, встановити значення - ваше ім'я. Написати код з перевіркою передачі і виводом даного cookie на екран. Додати текстове поле для виведення значення cookie в ньому.



* Вивести на екран кількість відвідувань сторінки, використовуючи cookie. Пропонований алгоритм (можливий інший варіант виконання):
  + встановіть змінну для лічильника ($counter), обнуліть її;
  + перевірте, чи встановлений вже cookie, якщо так - то задайте змінній $counter значення cookie;
  + додайте cookie для зберігання кількості відвідувань;
  + нарощуйте лічильник;
  + перевірте, чи встановленый вже cookie, якщо так - то виводите значення cookie.

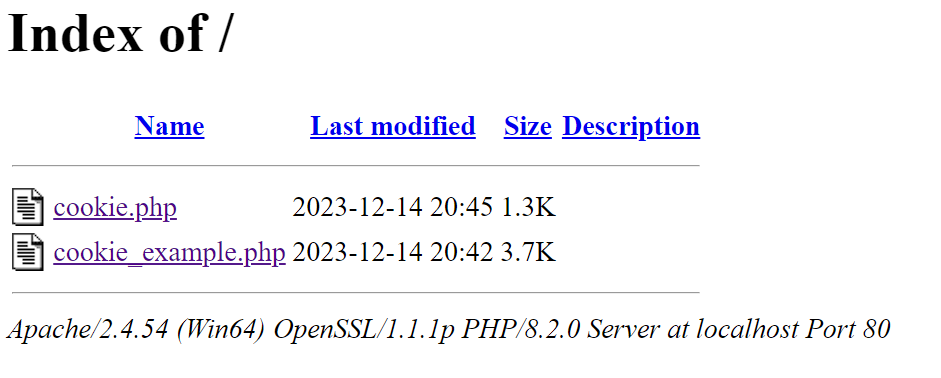


* Створити масив даних для зберігання паролів. Значення паролів зберегти в cookie. В html-код додати текстові поля, вивести в них значення паролів.



Створити файл cookie.php

* При першому запиті сторінки cookie.php користувачем, виводити фразу «Ласкаво просимо!»  
  При повторних запитах:
* Виводити кількість відвідувань. Наприклад: «Ви зайшли на сторінку 5 разів»
* Вказувати дату і час останнього перегляду. Наприклад: «Останнє відвідування: 25-10-2019 16:34:45». Використовуйте функцію date('d/m/Y H:i:s')



1. - \*Що таке cookie:\* Короткий текстовий файл, який зберігається на боці користувача і містить інформацію для взаємодії з веб-сайтом.

- \*Місце розташування:\* Зберігається на боці клієнта (в браузері).

- \*Час життя:\* Визначається при створенні і може бути встановлений для закінчення після певного часу або при закритті браузера.

2. - `setcookie()` в PHP.

3. - Коли потрібно зберігати інформацію на боці користувача, яка може бути використана при подальших відвідуваннях веб-сайту.

4. - `$\_COOKIE` в PHP.

5. - Старт сесії (`session\_start()`).

- Збереження та отримання значень у сесії.

- Закриття сесії або автоматично при закритті браузера.

6. - `session\_start()`.

7.Коли потрібно закрити сесію та видалити всі дані сесії. Наприклад, при виході користувача.