

# 1. Браузерні DevTools.

## Контрольні запитання

- 11.01. Які є відмінності між URN (Uniform Resource Name), URL (Uniform Resource Locator) та URI (Uniform Resource Identifier)?
- 11.02. З яких частин складається URL?
- 11.03. Що таке доменне ім'я?
- 11.04. Що таке IP-адреса?
- 11.05. Що таке код статусу відповіді веб-сервера? Які є класи цих кодів? Що кожен із них означає?
- 11.06. Що таке заголовки HTTP?
- 11.07. Яка відмінність між методами GET та POST запиту веб-клієнта?
- 11.08. Що таке веб-сторінка? Який вона має формат?

## Завдання для аудиторної роботи

[A]01.01. Відкрити будь-який сайт і виконати наступні завдання:

- Обрати довільний елемент: правою кнопкою миші на елементів, далі натиснути кнопку «Перевірити». Змінити його текст.
- Знайти елемент зі стилем. Додати або замінити колір цього CSS-стилю.
- У вкладці Console вивести document.title і змінити його на власний.
- Перевірити чи сторінка адаптована до екранів різної ширини.
- Відкрити вкладку Network і перезавантажити сторінку. Знайти, скільки запитів зроблено та який із них найбільший за розміром.
- У вкладці Performance зробити запис (Record) під час скролу сторінки й знайти, які процеси займають найбільше часу.
- У вкладці Application подивитися, що зберігається в LocalStorage. Змінити значення й оновити сторінку.
- Запустити Lighthouse-аудит і проаналізувати, які SEO чи Performance-помилки він знайшов.
- У вкладці Security перевірити сертифікат сайту (чи дійсний він та хто видав).

[A]01.02. Створіть простий файл **index.html** з кількома блоками та стилями.

- У DevTools через **Elements** та **Styles** змінити кольори, шрифти, padding у заданого стилю.

- Напишіть код з помилкою, наприклад

```
<script>
function brokenFunc() {
    console.Log(user.name);
}
brokenFunc();
</script>
```

У **Console** знайти текст помилки, а в **Sources** зупинитися на breakpoint-i, щоб подивитись, які змінні доступні.

- Додайте у свій HTML велику картинку (наприклад більшу за 3МБ) і кілька скриптів із CDN

```
<script src="https://code.jquery.com/jquery-3.7.1.min.js"></script>
<script
src="https://cdn.jsdelivr.net/npm/lodash@4.17.21/lodash.min.js"></s
cript>
<script
src="https://cdn.jsdelivr.net/npm/axios/dist/axios.min.js"></script
>
<script
src="https://cdn.jsdelivr.net/npm/vue@3.5.11/dist/vue.global.prod.js"></script>
<script
src="https://unpkg.com/react@18/umd/react.development.js"></script>
<script src="https://unpkg.com/react-dom@18/umd/react-
dom.development.js"></script>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap
.bundle.min.js"></script>
```

- У **Network** заміряйте час завантаження сторінки та визначте, що найбільше її «гальмує».
- Додати скрипт

```
localStorage.setItem("theme", "dark");
```

У **LocalStorage (Application)** знайти цей ключ, змінити його на "light" і перевірити в консолі, що значення справді оновилось.

- Додати код

```
async function loadData() {
    let res =
        await fetch("https://jsonplaceholder.typicode.com/posts/1");
    let data = await res.json();
    console.Log(data);
}
loadData();
```

У **Sources** поставити breakpoint після await та подивитись, як DevTools «чекає» відповідь.

- Додати HTML

```
<button id="btn">Click me</button>
<script>
  document.getElementById("btn")
    .addEventListener("click", () => {
      alert("Button clicked!");
    });
</script>
```

У **Event Listeners** (панель праворуч у вкладці **Elements**) знайти, яка подія прив'язана до кнопки.

## Завдання для самостійної роботи

[B]01.01. Нема :(