**Лабораторна робота**

*Тема роботи:* Консоль розробника Chrome

*Мета роботи:* вивчення основних можливостей консолі розробника браузера Chrome

*Завдання*

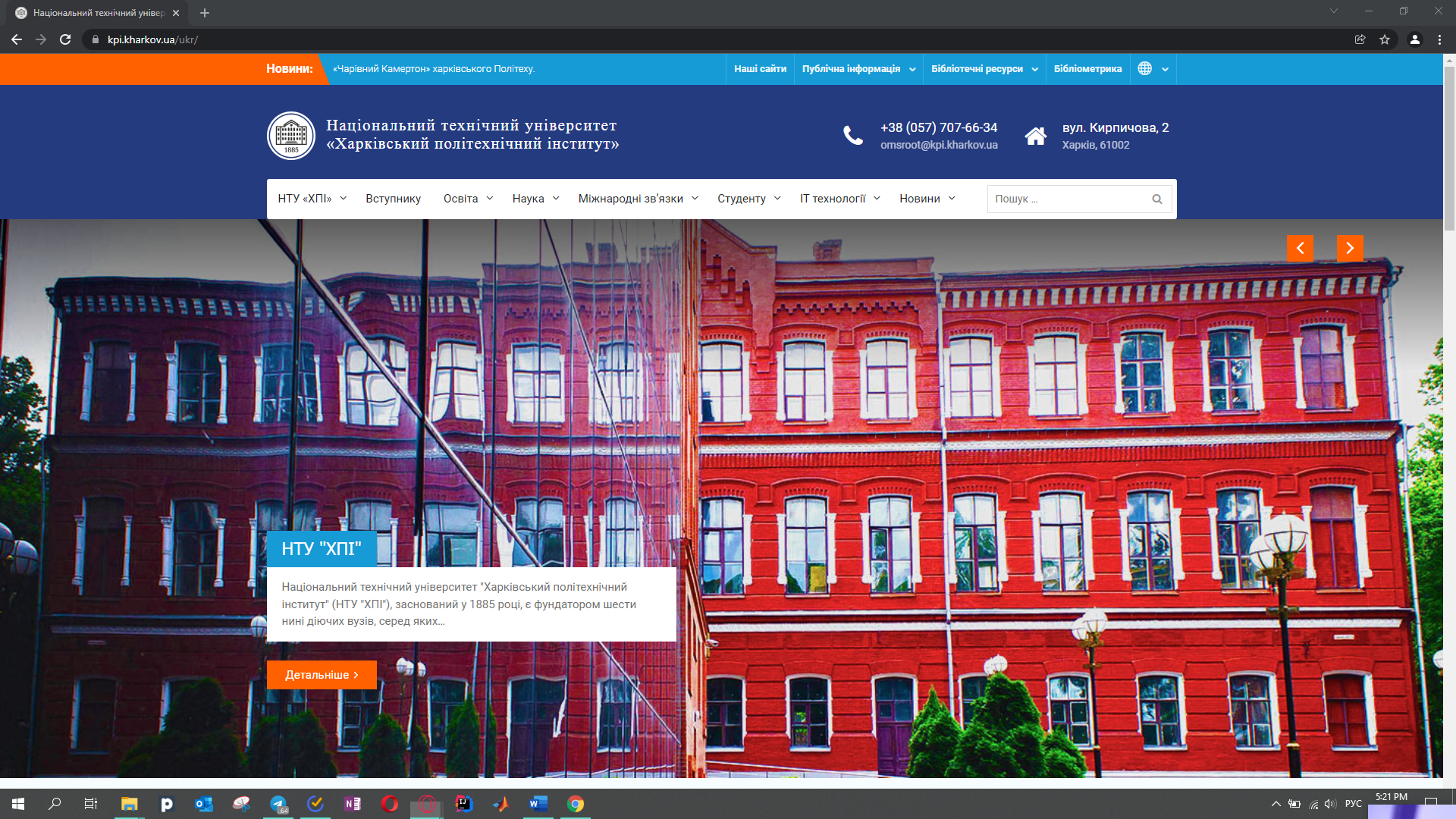
* Відпрацювати матеріал лекції, вказану літературу.

*При виконанні лабораторної роботи:*

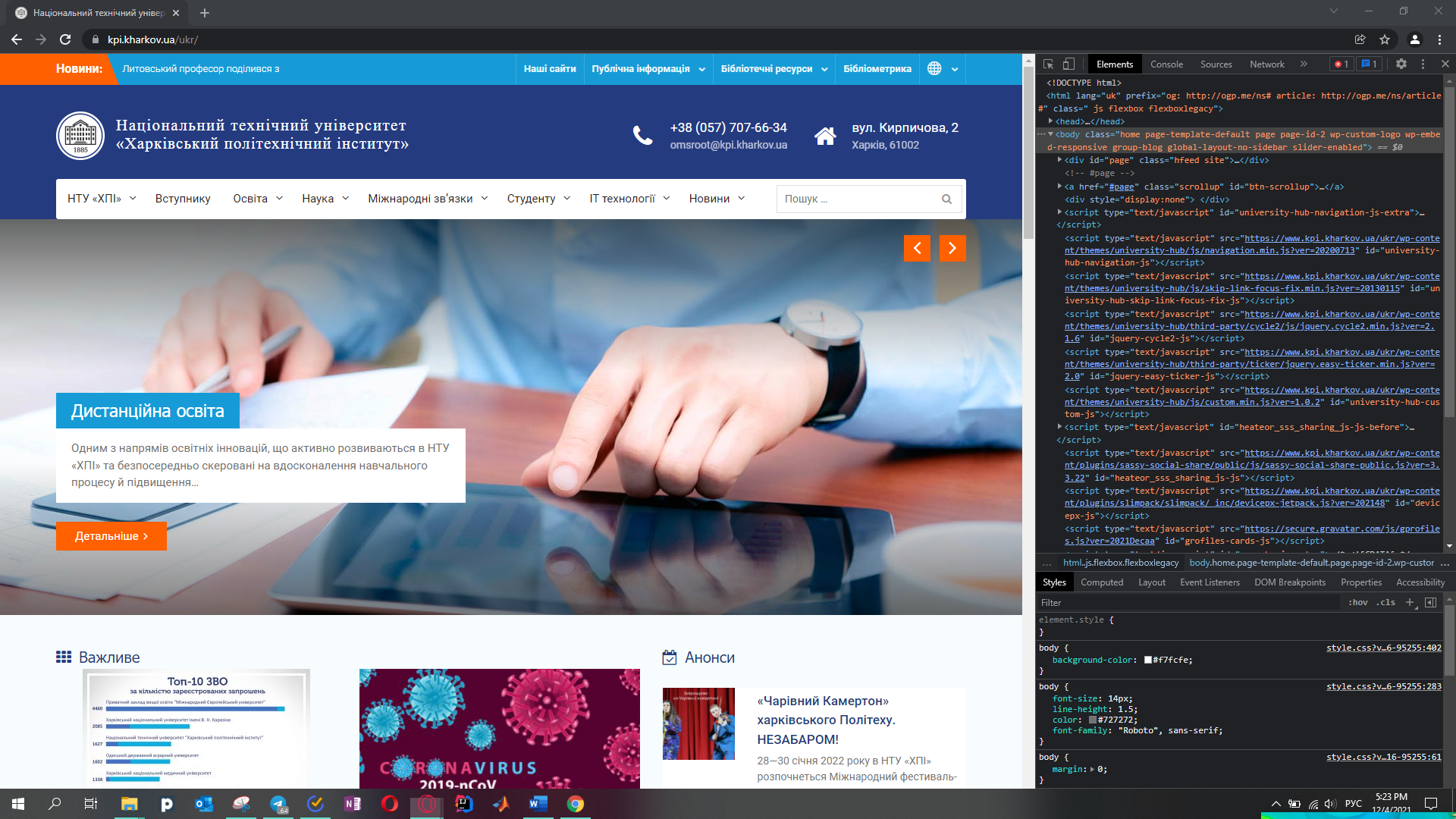
1. Відкройте сторінку НТУ ХПІ у браузері Chrome
2. Відкрийте консоль в браузері Chrome
3. Подивіться рендеринг сторінки в браузері
4. Передівиться знайдені при виконанні скрипта помилки в коді
5. Відкрийте DOM сторінку.
6. Переглянутиьте відкриті, відредагуйте вихідні файли сайту
7. Відстежите, які чинники ділянки коду роблять її повільною.
8. Повторіть для обраного вами сайту
9. Звіт по лабораторній роботі представляється у вигляді покрокових скрінів при виконанні пунктів 1- 7.

*Звiт*

1. Переходим на сайт университета по ссылке: <https://www.kpi.kharkov.ua/ukr/> в браузері Chrome:

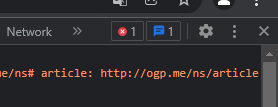


2. За допомогою гарячої клавіші F12 відкриваємо консоль. Справа знаходиться сама консоль:

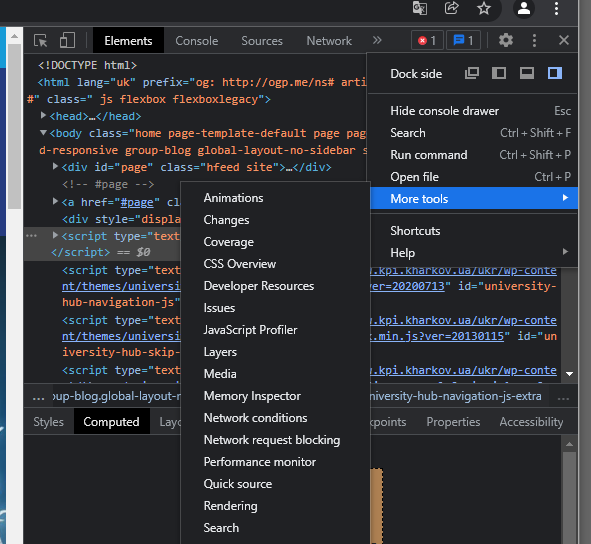


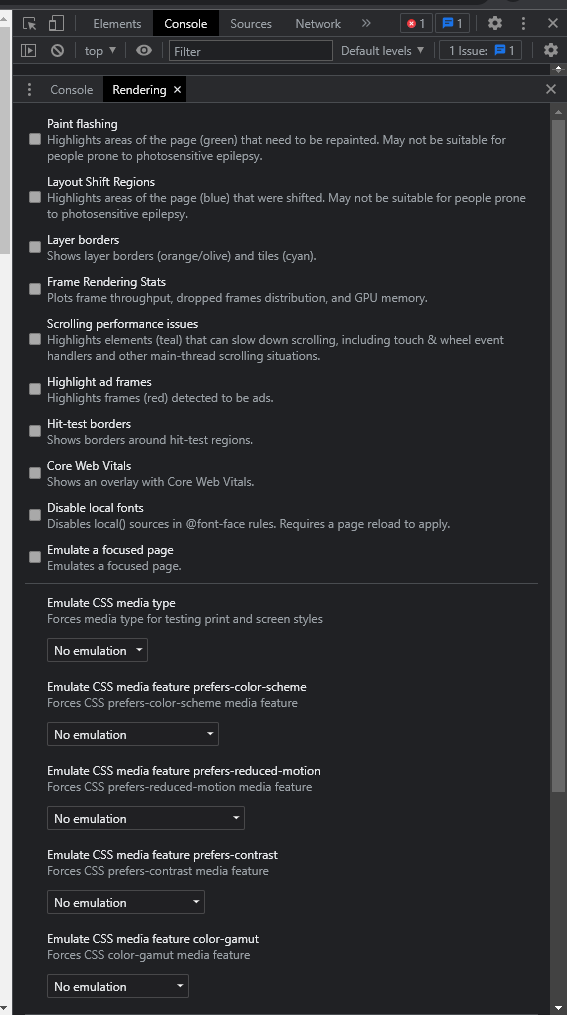
3. В консолі браузера є вкладка Elements (містить весь html/css код сторінки та дозволяє вибрати елементи для дослідження, а також редагувати їх). Панель Elements показує розмітку сторінки так само, як вона рендерується у браузері.

Натискаємо на трикрапку:

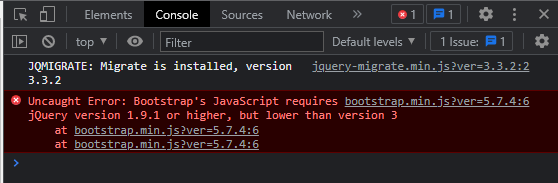


Вибираємо More tools -> Rendering:

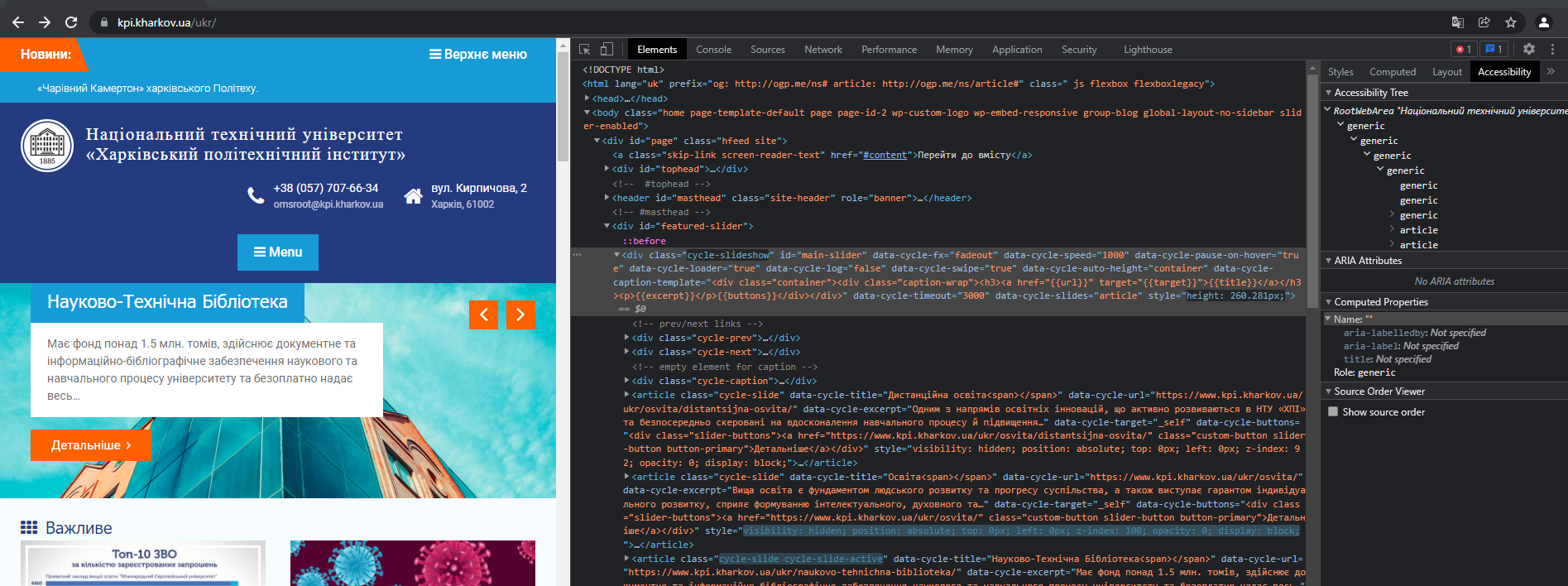




4. Переходимо на вкладку Console. Тут можна побачити, що на сторінці сталася помилка з некоректною версією jQuery та Bootstrap.



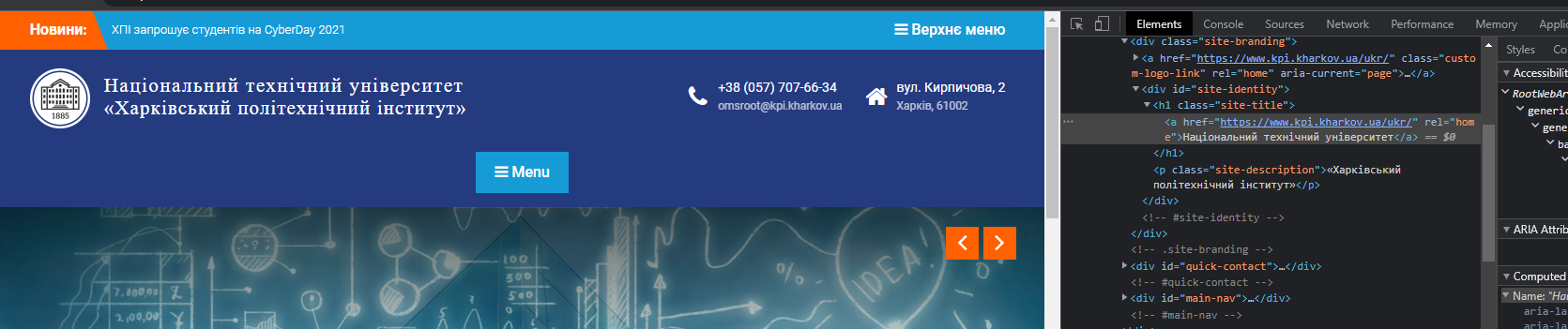
5 Щоб відкрити DOM, потрібно натиснути правою кнопкою миші(далi ПКМ) на сторінку та вибрати пункт Inspect.



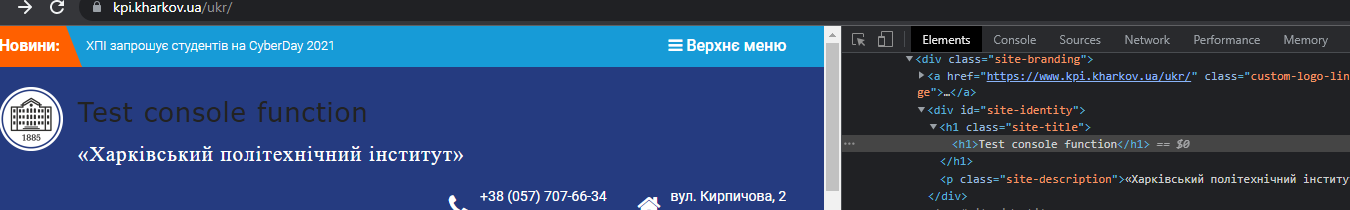
Праворуч можна побачити консоль із DOM-деревом.

6. Щоб відредагувати код на сторінці, потрібно вибрати частину коду, який цікавить, тег, натиснути ПКМ і вибрати пункт «Edit as HTML». Після цього можна додати шматок свого коду.

До:



Пiсля:



*Контрольні питання*

*1. Посилання на поточний вибраний елемент в консолі*

Виберіть тег у панелі елементів та введіть $0 у консолі, щоб створити посилання на поточний вибраний елемент.

*2. Додавання CSS і зміна стану тега*

CSS вказується у head тегі. Таблиці стилів можуть бути додані на веб-сторінку трьома способами:

- Таблиці пов'язаних стилів:

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

<title> Стиль </title>

<link rel="stylesheet" type="text/css" href="mysite.css">

<link rel="stylesheet" type="text/css" href="http://www.mysite.ru/main.css">

</head>

- Таблиці глобальних стилів:

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

<title> Стиль </title>

<style type="text/css">

H1 {

font-size: 120%;

font-family: Verdana, Arial, Helvetica, sans-serif;

color: #336;

}

</style>

</head>

Внутрішні стилі

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

<title> Стиль </title>

</head>

*3. Інструменти для перегляду, редагування й додавання ключів/значень у відповідні сховища даних*

Сховище даних – необхідно для зберігання процесів. Браузер зберігає на жорсткий диск

дані різних типів, наприклад, файли cookie. У новій специфікації HTML (HTML5) є визначення

терміна «веб-база даних»: це повноцінна (хоча і полегшена) браузерная база даних.

* Elements → описує всі елементи сторінки.
* Console → виявляє можливі і критичні помилки коду.
* Sources → показує вміст файлів на сторінці.
* Network → вказує код відповіді сервера, час завантаження сторінки і її розмір.
* Security → відображає інформацію про сертифікат SSL.
* Audits → дозволяє провести технічний аудит мобільного або десктопной версії сторінки.

*4. Аналіз завантажених сторінкою ресурсів*

Код сторінки включає:

* назви title, description;
* дані мікророзмітки Schema.org, Open Graph або інших словників;
* дані JavaScript;
* мова відображення контенту на сторінці;
* підключені лічильників аналітики, генераторів заявок та інших сервісів;
* витікаючі посилання на інші сторінки і сайти;
* розташування картинок, заголовків і текстових блоків;
* розміри і тип шрифтів, кольору елементів.

*5. Модуль відображення браузера*

Модуль відображення – відповідає за виведення запитаного вмісту на екран. Наприклад, якщо запитується HTML-документ, модуль відображення виконує синтаксичний аналіз коду HTML і CSS і виводить результат на екран. *10. Основні компоненти браузера.*

Web Components (веб-компоненти) - це набір функцій що додаються консорціумом W3C до специфікацій HTML та DOM які дозволять створення повторно використовних віджетів чи компонентів у застосунках та документах веб. Модель компонентів дозволяє інкапсуляцію та інтероперабельність окремих елементів HTML.

Web компоненти складаються з 4 основних деталей, які можуть використовуватись окремо або поодинці:

* Custom Elements (спеціальні елементи) - API для визначення нових HTML елементів
* Shadow DOM - інкапсульований DOM та стилі, з композицією
* HTML Imports - декларативні методи імпортування HTML документів в інші документи
* HTML Templates - шаблон, який дозволяє документам містити інертні частини DOM

*6. Основні компоненти браузера.*

1. Інтерфейс – включає адресний рядок, кнопки «Назад» і «Вперед», меню закладок і т. д. До нього відносяться всі елементи, крім вікна, в якому відображається запитувана сторінка.
2. Механізм браузера – управляє взаємодією інтерфейсу і модуля відображення.
3. Модуль відображення – відповідає за виведення запитаного вмісту на екран. Наприклад, якщо запитується HTML-документ, модуль відображення виконує синтаксичний аналіз коду HTML і CSS і виводить результат на екран.
4. Мережеві компоненти – призначені для виконання Інтернет-дзвінків, таких як HTTP-запити. Їх інтерфейс не залежить від типу платформи, для кожного з яких є власні реалізації.
5. Виконавча частина користувацького інтерфейсу – використовується для відтворення основних віджетів, таких як вікна і поля зі списками. Її універсальний інтерфейс також не залежить від типу платформи. Виконавча частина завжди застосовує методи призначеного для користувача інтерфейсу конкретної операційної системи.
6. Інтерпретатор JavaScript – використовується для синтаксичного аналізу і виконання коду JavaScript.
7. Сховище даних – необхідно для зберігання процесів. Браузер зберігає на жорсткий диск дані різних типів, наприклад, файли cookie. У новій специфікації HTML (HTML5) є визначення терміна «веб-база даних»: це повноцінна (хоча і полегшена) браузерная база даних.