

Quality Assurance 2.0 -> Lesson 16. Web технології.

Клієнт-серверна архітектура

Протокол HTTP(s) — HyperText Transfer Protocol

HTTP — це клієнт-серверний протокол, призначений для передачі гіпертекстових документів (тобто документів, які можуть містити посилання, що дозволяють здійснити перехід до інших документів).

HTTP — це протокол передачі даних в Інтернеті. Він використовується для взаємодії між клієнтом і сервером, коли є намір отримати доступ до веб-сторінки або інших даних з Інтернету.

HTTP працює на основі запитів і відповідей. Коли адреса веб-сторінки вводиться у веб-браузер, він створює HTTP запит до сервера, на якому розміщена ця веб-сторінка. Запит містить інформацію про те, яку веб-сторінку треба отримати, а також інші дані, такі як тип вашого веб-браузера.

Типи запитів HTTP:

- **GET-запит:** Використовується для отримання даних із сервера. Коли відкривається веб-сторінка, браузер надсилає запит GET, щоб отримати вміст цієї сторінки. У тестуванні GET-запитів важливо перевірити, чи дані завантажуються правильно і відображаються коректно.
- **POST-запит:** Використовується для надсилання даних на сервер. Це як надсилання форми на веб-сайті. QA Engineer повинен перевірити, що дані, що надсилаються через POST-запит, обробляються правильно, і сервер зберігає їх так, як очікується.
- **PUT-запит:** Використовується для оновлення наявних даних на сервері. Наприклад, запит PUT може змінювати інформацію про користувача. У тестуванні PUT-запитів важливо переконатися, що оновлення відбуваються коректно та без втрати даних.
- **DELETE-запит:** Використовується для видалення даних на сервері. Якщо у програмі передбачена функція видалення, QA Engineer повинен перевірити, що вона працює правильно і безпечно.
- **PATCH-запит:** Використовується для часткового оновлення даних на сервері, змінюючи лише певні атрибути об'єкта. Це корисно, коли потрібно оновити частину інформації, не торкаючись інших даних.

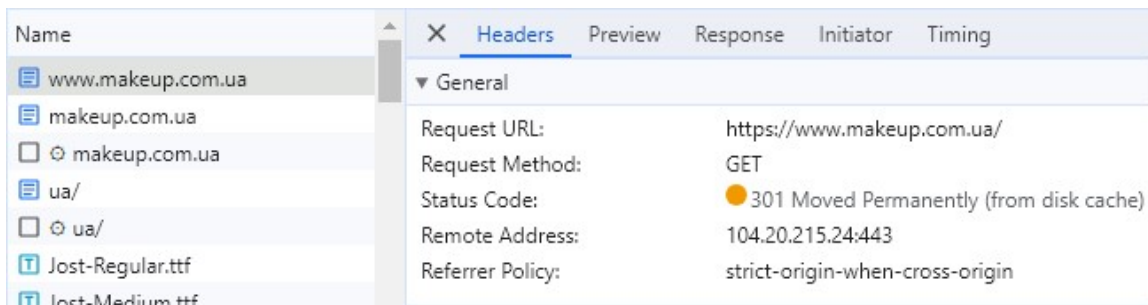
Статус кодів відповіді від сервера:

- **1xx (Інформаційні):** сервер отримав запит клієнта, але відповідь ще не готова.
- **2xx (Успіх):** запит клієнта був успішно виконаний, відповідь є переданою до клієнта, все в порядку.
- **3xx (Перенаправлення):** серверу потрібно зробити ще щось, щоб отримати відповідь, наприклад, перейти на іншу сторінку.
- **4xx (Помилка клієнта):** клієнт зробив щось невірно, наприклад, спробував відкрити неіснуючу сторінку.
- **5xx (Помилка сервера):** сервер з якоїсь причини не може виконати запит клієнта.

Завдання

Використовуємо інструменти розробника (DevTools) у браузері Google Chrome (Натиснути F12).

1. Проаналізувати запит <https://www.makeup.com.ua/> , його метод і статус

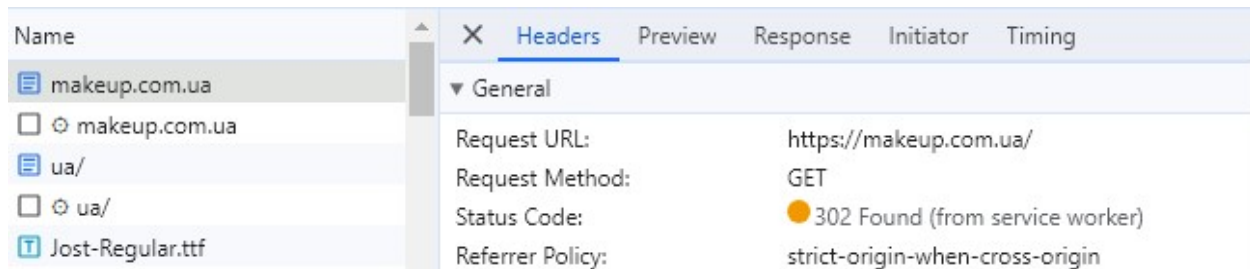


Request Method: GET - отримати дані з ресурсу по URL

Request URL: https://www.makeup.com.ua/

Status Code: 301 Moved Permanently (from disk cache) - переміщено назавжди, використовується для постійного переспрямування, тобто посилання або записи, які повертають цю відповідь, повинні бути оновлені.

2. Проаналізувати запит <https://makeup.com.ua/> , його метод і статус

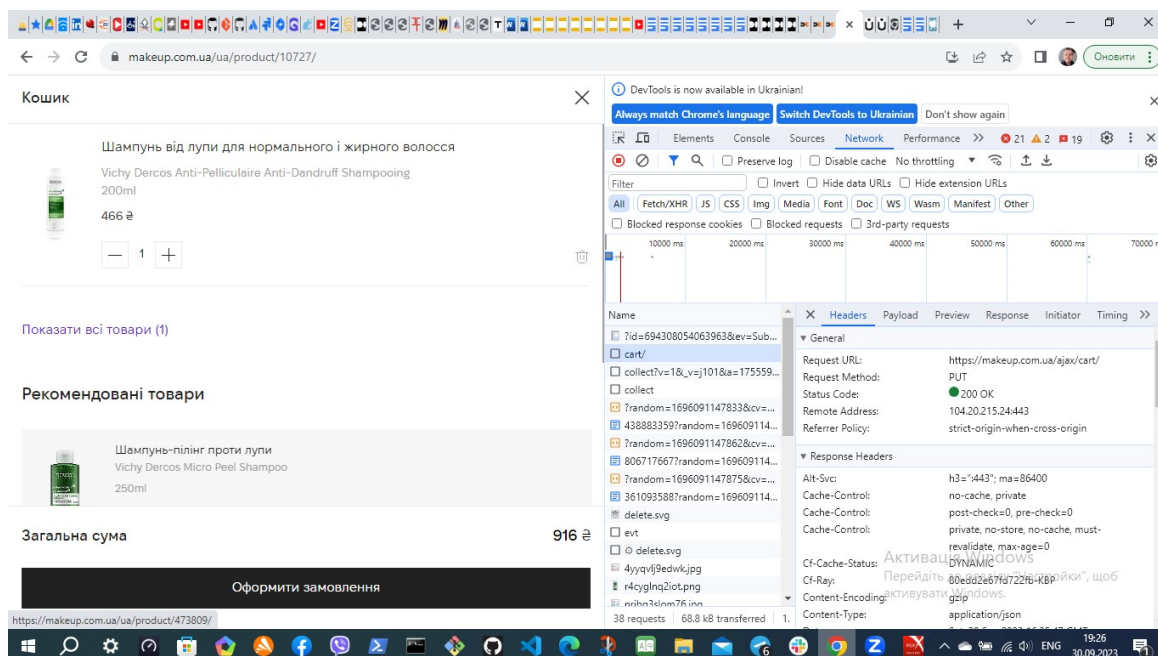


Request Method: **GET** - отримати дані з ресурсу по URL

Request URL: <https://makeup.com.ua/>

Status Code: **302 Found (from service worker)** - вказує на те, що певну URL-адресу було тимчасово переміщено в нове місце. Щоразу, коли відвідувачі, роботи Google або інші пошукові системи отримують доступ до вихідної URL-адреси, переспрямування 302 надає автоматичну відповідь із зазначенням нової адреси.

3. Додати будь-який товар в кошик і проаналізувати запит cart, його метод і статус

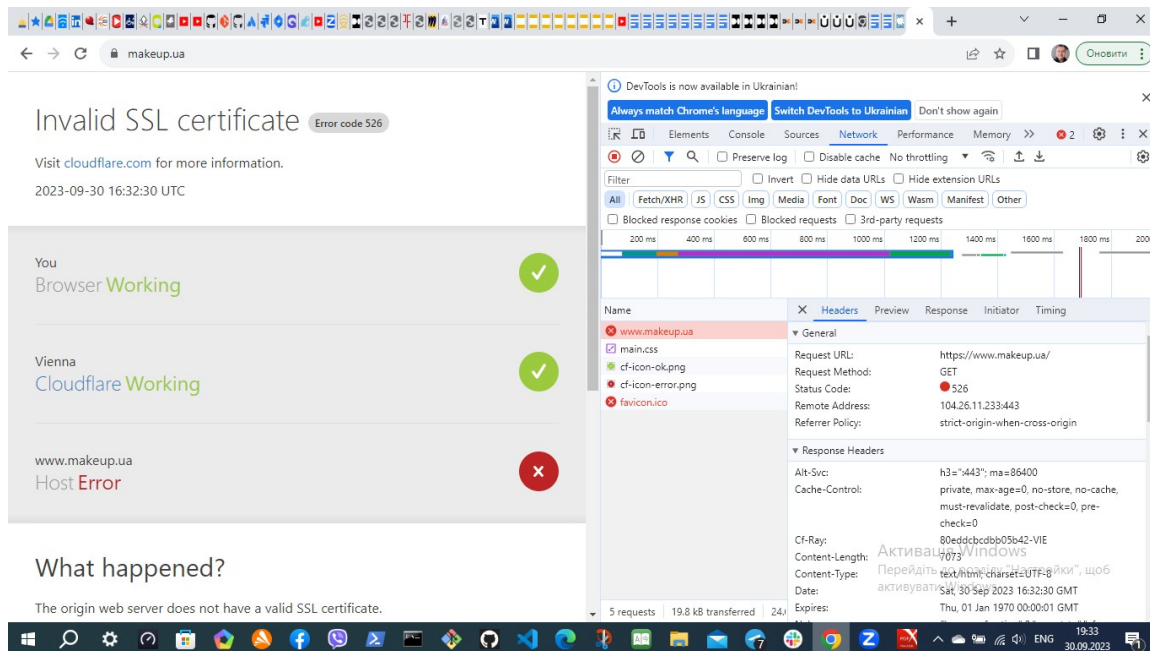


Request Method: **PUT** - додати дані до ресурсу по URL

Request URL: <https://makeup.com.ua/ajax/cart/>

Status Code: 200 OK - код відповіді статусу успіху, вказує на успішне виконання запиту, ресурс було отримано та передано в тілі повідомлення.

4. Вбити в адресному рядку <https://www.makeup.ua/> проаналізувати запит і статус.



Request Method: GET - отримати дані з ресурсу по URL

Request URL: <https://www.makeup.ua/>

Status Code: 526 - запит є не виконаний, це помилка https:// протоколу, яка виникає, коли веб-сайт не може встановити безпечне з'єднання, що вказує на невдалий або неповний SSL handshake.