

## ЗАНЯТТЯ 13 ОТРИМАННЯ ДАНИХ З СЕРВЕРУ. ОБРОБКА ПОДІЙ. ОСНОВИ HTML ФОРМ

### ТЕМИ

#### 30. Взаємодія з власним або відкритим API

- Огляд задачі, аргументація необхідності роботи з зовнішніми даними.
- Отримання даних з серверу за допомогою jquery ajax.
- Відправка команд та даних на сервер.
- Приклади взаємодії з серверами, що надають інформацію в публічний доступ (новини google або NY Times, котування валют, тендери prozorro).

#### Практика:

- Практикуємося з кодом для отримання зовнішніх даних та їх обробкою (на вибір ментора)

### РЕКОМЕНДАЦІЇ ДО ПРОВЕДЕННЯ ЗАНЯТТЯ

#### 1. Обговорюємо разом домашнє завдання

- Ментору бажано мати свій варіант реалізованого завдання.
- Дивимось варіанти учасників. Якщо є варіант, який можна швидко доробити - дороблюємо разом.
- Якщо на кількох з працюючих учасників можна виділити типові проблеми - обговорюємо і, при потребі, повторно демонструємо щось з попередніх занять.

#### 2. Отримання даних ззовні - мотивація та постановка задачі для заняття

- Окреслюємо задачу та мотивацію обміну даними між клієнтом та сервером в веб. Оглядом проходимо по поняттях “клієнт” та “сервер”, варто потребу в базі даних (в принципі, той самий інтернет магазин є зрозумілим прикладом, але варто описати процес роботи - закупщики вносять товари, менеджери обробляють замовлення, користувачі шукають товари та роблять замовлення... Також варто навести ще які-небудь видовищні приклади - пошуковий сервер гугла, соц. мережа фейсбук...)
- Нагадуємо про те, що для самої передачі даних в мережі часто використовується JSON. Можна розповісти про те, що для цього дані перетворюються саме в рядкове представлення, тому для кожної технології потрібні “серіалізатори” та “парсери”. Продемонструвати, як виглядає рядок JSON.
- Окреслюємо задачу, яку пропонуємо зробити разом. Задача повинна включати прийняття деякого вводу від користувача, отримання даних з сервера і використання цих даних для виконання дій за командою користувача. Як варіант - можна реалізувати сторінку - конвертер валют, отримуючи дані про курси з сайту НБУ в json-форматі. На сторінці буде поле вводу суми, випадаючий список для вибору валюти, та кнопка “порахувати”. Варіант результату і мінімалістичної реалізації (HTML + JS):

<https://sergdeni33.github.io/WebDevDemoShop/JsStart/js-ajax-pure.html>

<https://github.com/sergdeni33/WebDevDemoShop/blob/master/JsStart/js-ajax-pure.html>

<https://github.com/sergdeni33/WebDevDemoShop/blob/master/JsStart/js/course-converter.js>

**3. Наголошуємо, що для виконання задачі нам необхідно базово освоїти використання обробників подій та елементів керування - та пропонуємо групі створювати сторінку разом з ментором.**

- Створюємо сторінку, в яку додаємо поле вводу (input type='text') (для суми), поле вибору валюти (select + options), кнопку та поле виводу результату.
- По ходу базово розповідаємо про поля вводу тексту та поля вибору в HTML.
  - За поглибленим вивченням рекомендуємо користуватись документацією, а також кажемо, що детальніше про форми та поля вводу поговоримо на наступному занятті.
  - В поле вибору валюти вбиваємо руками два - три варіанти для демонстрації - звертаючи увагу на різницю між заголовком варіанту (текстом, наприклад може бути "Доллар США") та значенням, яке ми будемо використовувати в коді (наприклад, код валюти "USD").
- Базово розповідаємо про обробники подій. Варто згадати, що саме реакція на події (особливо на дії користувача) була однією з основних цілей створення скриптових мов в браузері (javascript). Разом з групою створюємо функцію і призначаємо її обробником події натиснення кнопки "порахувати".
- Приходимо до того, що для подальшої реалізації конвертера валют нам необхідно при показі сторінки користувачу мати в доступі нашого скрипта масив з курсами - щоб при натисненні кнопки у обробнику подій виконати підрахунки. Крім того, в момент показу сторінки ми повинні заповнити список доступних валют, щоб користувач міг реально їх обирати.

**4. Отримання даних з сервера - реалізація**

- Розповідаємо про те, що задача виконати код відразу після завантаження сторінки є важливою та часто зустрічається, і для цього можна також використати обробник події - але вже не такої, яка генерується користувачем, а вбудованої.
- Демонструємо (реалізуємо разом з групою) виклик нашої функції на подію onload в body. Варто згадати, що як правило використовується більш потужний (але і більш складний) варіант призначення цього обробника - і відіслати бажаних до документації.
- Розповідаємо концепцію AJAX. В тому числі про мотивацію виникнення та значення.
- Демонструємо в мінімальному вигляді використання вбудованого об'єкту XMLHttpRequest (вірогідно, варто обійтись без пояснення оператора new - кажемо, що так створюються), без сумісності з 6-м експлорером та повної перевірки проблем. Один з варіантів коду - за посиланням вище.
- Розповідаємо про проблему синхронності та асинхронності (сторінка не повинна підвисати, поки очікує виконання ajax-запитів), базово пояснюємо поняття колбеків (функцій зворотнього виклику), на прикладі призначення обробника onreadystatechange.
- Разом з групою реалізуємо отримання даних про курси валют з сайту НБУ, і в обробнику події отримання результату заповнюємо випадаючий список вибору валюти

для користувача (можна зробити окремою функцією, яку викликати та дати їй масив курсів з сервера). Також в цьому обробнику запам'ятовуємо отримані дані в глобальну змінну.

- Нарешті разом реалізуємо до кінця реакцію на кнопку “Порахувати” - так як тепер при завантаженні сторінки ми отримуємо всі потрібні дані. По ходу показуємо, як отримати обране значення в випадяючому списку валют, та прочитати введену в текстове поле суму.
- Варіант ресурсу з документацією по ajax:  
[https://developer.mozilla.org/ru/docs/Web/Guide/AJAX/Getting\\_Started](https://developer.mozilla.org/ru/docs/Web/Guide/AJAX/Getting_Started)  
(є переклад російською)

### ДОМАШНЄ ЗАВДАННЯ (ТЕОРІЯ)

- [Курс з веб-розробки на W3Schools, тема “jQuery - AJAX Introduction” \(англ\)](#)
- [Курс з веб-розробки на W3Schools, тема “jQuery - AJAX get\(\) and post\(\) Methods” \(англ\)](#)
- [Mozilla Developer Network \(MDN\) «Third party APIs» \(англ\)](#)
- [Mozilla Developer Network \(MDN\) «Fetching data from the server» \(англ\)](#)
- [Офіційний веб-сайт фреймворку jQuery, навчальний розділ “jQuery’s Ajax-Related Methods” \(англ\)](#)

### ДОМАШНЄ ЗАВДАННЯ (ПРАКТИКА)

- Пропонуємо групі додати до свого “каталогу плиткою, якій генерується за масивом даних про товари” (попереднє завдання), наступну функціональність:
  - При завантаженні сторінки, отримати курси валют НБУ, і запропонувати користувачу список вибору валют та кнопку “Застосувати”.
  - По натисненню цієї кнопки, перегенерувати каталог товарів, показавши ціну в обраній валюті (вважати, що в масиві даних для каталога ціна задана в національній валюті).
- Якщо ментору вдасться знайти яке-небудь інше відкрите джерело не складних даних, можна запропонувати групі створити сторінку, яка буде генерувати свій контент на їх основі. Наприклад, це може бути сторінка із списком новин з гугл (апі відкрите, але на даний момент потребує пройти процедуру реєстрації та отримання ключа).
- Можна також запропонувати вивести “курси валют плиткою”, на основі тих самих даних - так само, як в сторінці - прикладі за посиланням.

### ДОДАТКОВІ КОРИСНІ МАТЕРІАЛИ

- [Стаття про OpenProcurement на Вікіпедії \(укр\)](#)
- [Сайт документації по OpenProcurement, розділ «Туторіал. Розглянемо основні правила» \(укр\)](#)
- [Галерея API на офіційному веб-сайт NY Times \(англ\)](#)
- [Google Maps API на офіційному веб-сайті Google \(рос\)](#)

