

JavaScript, JSON – XHR, Fetch, axios

Początek laboratorium:

- w XAMPP uruchomić Apache oraz MySQL, następnie przejść do phpMyAdmin,
- pobrać na pulpit archiwum Lab014_AI1_start.zip, w którym umieszczony jest projekt startowy do wykonania zadań oraz rozpakować to archiwum,
- przejść do rozpakowanego folderu oraz w przypadku posiadania innych ustawień niż domyślne (np. połączenia z bazą), wykonać ich zmianę w .env.example oraz start...,
- uruchomić skrypt start.bat (Windows, 2x kliknięciem) lub start.sh (inne systemy, przez polecenie *bash start.sh*),
- zainstalować program Postman oraz pominąć zakładanie konta: *lightweight API client*.

Zadania (JavaScript, JSON):

Zadanie 14.1:

Zapoznać się z następującymi zagadnieniami:

- format JSON, składnia,
- typy danych wartości JSON'a,
- poprawność składniowa dokumentu JSON (Syntax Rules),
- JSON vs JavaScript Objects,
- funkcja JSON.stringify,
- funkcja toJSON (wewnątrz obiektu JS),
- funkcja JSON.parse.

<https://www.json.org/json-en.html>

<https://www.techiedelight.com/json-introduction/>

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Working_with_objects

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/JSON/stringify

<https://thecodebarbarian.com/what-is-the-tojson-function-in-javascript.html>

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/JSON/parse

Zadanie 14.2:

Otworzyć terminal *cmd* (Command Prompt) w VSCode.

Uruchomić serwer deweloperski *php* dla przy użyciu komendy *serve artisan'a*.

W przeglądarce przejść do *Telescope* oraz pozostawić go otwartym.

```
php artisan serve
```

<http://localhost:8000/telescope/requests>

Zadanie 14.3:

Przejsć do folderu *public/lab14*.

W przeglądarce internetowej przejść pod adres dotyczący pierwszych zadań.

Otworzyć *Narzędzia dla deweloperów* (F12).

Wykonując dane zadanie należy odkomentować obszar tego zadania.

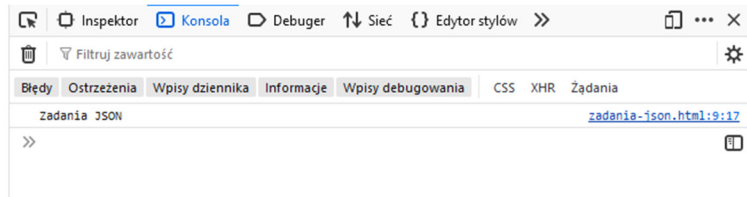
Po zakończonym zadaniu należy zakomentować obszar tego zadania.

Odkomentowanie/zakomentowanie obszaru w *VSCode*: zaznaczyć obszar oraz *Ctrl + /*.

Po zmianach w kodzie należy odświeżać stronę.

<http://localhost:8000/lab14/zadania-json.html>

Otworzyć *Narzędzia dla deweloperów* (F12).



Zadanie 14.4:

W nawiązaniu do obiektów *user* i *user2*:

- wyjaśnić różnicę między *const* oraz *let*,
- wymienić typy danych wartości *JSON'a*,
- wyjaśnić zastosowanie funkcji *JSON.stringify*,
- rozpatrzyć wywołania tej funkcji z różnymi parametrami *,
- wyjaśnić wpływ *toJSON* (wewnątrz obiektu) na działanie *JSON.stringify*.

–

Zadanie 14.5:

Zwalidować poniższy *JSON* w narzędziu *JSONLint*. Zmodyfikować go, dodając w nim celowo błędy oraz wykonywać ponownie walidację *poprawności składniowej*.

W nawiązaniu do ciągów znaków *json3* i *json4*:

- wyjaśnić zastosowanie funkcji *JSON.parse*,
- modyfikować *json4* i sprawdzać działanie obsługi błędów.

<https://jsonlint.com/>

```
{
  "name": "Jan",
  "age": 30,
  "technologies": ["PHP", "Java"]
}
```

Zadanie 14.6: *

Sprawdzić działanie drugiego, opcjonalnego parametru funkcji *JSON.parse*, którym jest funkcja pozwalająca zmodyfikować tworzony obiekt.

–

Zadania (JavaScript, XHR):

Zadanie 14.7:

Zapoznać się z następującymi zagadnieniami:

- Ajax,
- XMLHttpRequest (XHR),
- wykonywanie żądań HTTP za pomocą obiektu XHR,
- synchroniczność vs asynchroniczność,
- 5 stanów obiektu XHR.

<http://localhost:8000/telescope/requests>

<https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX>

<https://developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest>

<https://www.loginradius.com/blog/engineering/ajax-and-xhr-using-plain-javascript/>

https://developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest/Synchronous_and_Asynchronous_Requests

https://developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest/readystatechange_event

- 0 – XMLHttpRequest.UNSET
- 1 – XMLHttpRequest.OPEN
- 2 – XMLHttpRequest.HEADERS_RECEIVED
- 3 – XMLHttpRequest.LOADING
- 4 – XMLHttpRequest.DONE

<http://localhost:8000/lab14/zadania-xhr.html>

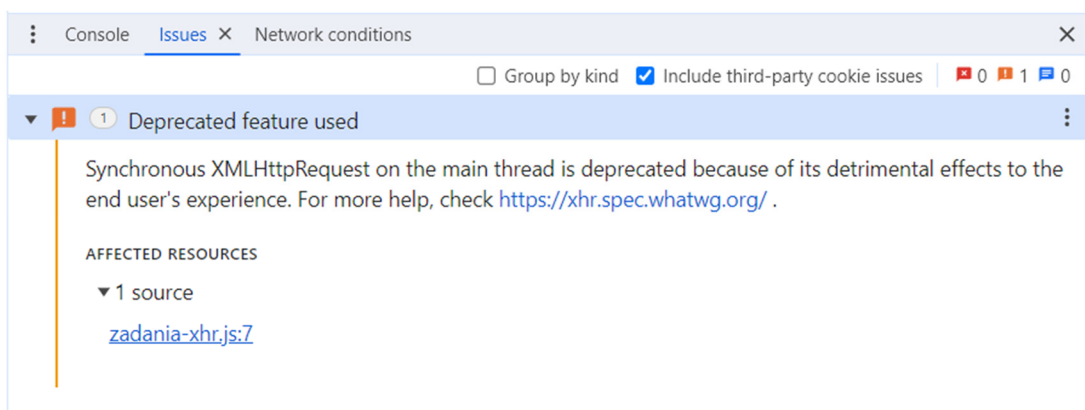
Zadanie 14.8:

Sprawdzić działanie wykonania synchronicznego żądania pobierającego wszystkie kraje. Zwrócić uwagę na komunikaty „Przed” i „Po”.

Wyjaśnić jakie problemy mogą spowodować takie żądania.

```
xhr.open("GET", "...", false); // `false` makes the request synchronous
```

⚠ Użycie synchronicznej wersji metody XMLHttpRequest w głównym wątku jest przestarzałe z powodu szkodliwych efektów ubocznych. Więcej informacji można znaleźć na stronie <https://xhr.spec.whatwg.org/#sync-warning> [zadania-xhr.js:7:5](#)



<https://xhr.spec.whatwg.org/#sync-warning>

Synchronous XMLHttpRequest outside of workers is in the process of being removed from the web platform as it has detrimental effects to the end user's experience. (This is a long process that takes many years.) Developers must not pass false for the `async` argument when the `current global object` is a `Window` object. User agents are strongly encouraged to warn about such usage in developer tools and may experiment with throwing an `"InvalidAccessError"` DOMException when it occurs.

Zadanie 14.9:

Sprawdzić działanie wykonania żądania *asynchronicznego* pobierającego wszystkie kraje. Zwrócić uwagę na komunikaty „Przed” i „Po”.
Zwrócić uwagę na stany, w których znalazł się obiekt *XHR*.

–

Zadanie 14.10:

Napisać wykonywanie żądania *HTTP* za pomocą *XHR* pobierającego kraj o danym *id*. Przewidzieć możliwe odpowiedzi, obsłużyć je wyświetlając komunikaty na konsoli. Sprawdzić działanie dla różnych przypadków.

–

Zadanie 14.11:

Napisać wykonywanie żądania *HTTP* za pomocą *XHR* dodającego nowy kraj do bazy. Wykorzystać obiekt kraju „Czechy”.
Przewidzieć możliwe odpowiedzi, obsłużyć je wyświetlając komunikaty na konsoli. Sprawdzić działanie dla różnych przypadków.

`xhr.setRequestHeader`

Zadanie 14.12: *

Napisać wykonywanie żądania *HTTP* za pomocą *XHR* aktualizującego kraj o danym *id*. Wykorzystać obiekt kraju „Australia” o *id* 6.
Przewidzieć możliwe odpowiedzi, obsłużyć je wyświetlając komunikaty na konsoli. Sprawdzić działanie dla różnych przypadków.

–

Zadanie 14.13:

Napisać wykonywanie żądania *HTTP* za pomocą *XHR* usuwającego kraj o danym *id*. Na początku wykorzystać *id* kraju „Czechy” czyli np. 7.
Przewidzieć możliwe odpowiedzi, obsłużyć je wyświetlając komunikaty na konsoli. Sprawdzić działanie dla różnych przypadków.

–

Zadania (JavaScript, Fetch API):

Zadanie 14.14: *

Zapoznać się z następującymi zagadnieniami:

- *fetch* (Fetch API),
- *Promise*,
- słowa *async* i *await*, *
- *XHR* vs *fetch*. *

https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Promise

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/async_function

<https://blog.openreplay.com/ajax-battle-xmlhttprequest-vs-fetch/>

Zadanie 14.15: *

Zapoznać się z przykładem:

- tworzeniem obiektu „*obietnicy*”, z czego się składa,
- pozytywne i negatywne rozpatrzenie *obietnicy*,
- wykorzystaniem *obietnicy* za pomocą trzech funkcji,
- inny zapis metody *then*,
- praca z *obietnicami* z użyciem *async* i *await*.

<http://localhost:8000/lab14/zadania-fetch.html>

Zadanie 14.16: *

Sprawdzić działanie wykonania żądania *HTTP* za pomocą *fetch* pobierającego wszystkie kraje. Zwrócić uwagę na komunikaty „Przed” i „Po”.

–

Zadanie 14.17: *

Napisać wykonywanie żądania *HTTP* za pomocą *fetch* pobierającego kraj o danym *id*. Przewidzieć możliwe odpowiedzi, obsłużyć je wyświetlając komunikaty na konsoli. Sprawdzić działanie dla różnych przypadków.

–

Zadanie 14.18: *

Napisać wykonywanie żądania *HTTP* za pomocą *fetch* dodającego nowy kraj do bazy. Wykorzystać obiekt kraju „Słowacja”. Przewidzieć możliwe odpowiedzi, obsłużyć je wyświetlając komunikaty na konsoli. Sprawdzić działanie dla różnych przypadków.

–

Zadanie 14.19: *

Napisać wykonywanie żądania *HTTP* za pomocą *fetch* aktualizującego kraj o danym *id*. Wykorzystać obiekt kraju „Australia” o *id* 6. Przewidzieć możliwe odpowiedzi, obsłużyć je wyświetlając komunikaty na konsoli. Sprawdzić działanie dla różnych przypadków.

–

Zadanie 14.20: *

Napisać wykonywanie żądania *HTTP* za pomocą *fetch* usuwającego kraj o danym *id*. Na początku wykorzystać *id* kraju „Słowacja” czyli np. 8. Przewidzieć możliwe odpowiedzi, obsłużyć je wyświetlając komunikaty na konsoli. Sprawdzić działanie dla różnych przypadków.

–

Zadania (JavaScript, axios):

Zadanie 14.21: *

Wykonać analogicznie zadania 14.16 – 14.20 z użyciem biblioteki *axios*.

<https://github.com/axios/axios>

<https://github.com/axios/axios?tab=readme-ov-file#request-method-aliases>

<http://localhost:8000/lab14/zadania-axios.html>

- * – zadania/podpunkty do samodzielnego dokończenia/wykonania,
- * – zadania/podpunkty dla zainteresowanych.

Po zakończonym laboratorium należy skasować wszystkie pobrane oraz utworzone przez siebie pliki z komputera w sali laboratoryjnej.

Wersja pliku: v1.0