AI1 **LAB D**	Kowieska Martyna Album 53882 Grupa 7	**Wersja 1**

REST API CLIENT

SPIS TREŚCI

CEL ZAJĘĆ

Celem głównym zajęć jest zdobycie następujących umiejętności:

- pobieranie danych z zewnętrznych zasobów za pomocą REST API
- zdobywanie wiedzy na temat zewnętrznych API za pomocą dokumentacji typu Swagger
- wysyłanie asynchronicznych żądań z wykorzystaniem XMLHttpRequest i Fetch API

W praktycznym wymiarze uczestnicy stworzą dynamiczną stronę HTML pozwalającą na wyświetlanie bieżącej informacji pogodowej oraz prognoz dla zadanej przez użytkownika miejscowości.

ROZPOCZĘCIE

Rozpoczęcie zajęć. Powtórzenie wykonywania połączeń synchronicznych i asynchronicznych z poziomu JS na stornie.

Wejściówka?

UWAGA

Ten dokument aktywnie wykorzystuje niestandardowe właściwości. Podobnie jak w LAB A wejdź do `Plik` -> `Informacje` -> `Właściwości` -> `Właściwości zaawansowane` -> `Niestandardowe` i zaktualizuj pola. Następnie uruchom ten dokument ponownie lub `Ctrl+A` -> `F9`.

Wymagania

W ramach LAB D przygotowane powinny zostać:

- pojedyncza strona HTML ze skryptem ładowanym z zewnętrznego pliku JS
- pole tekstowe (input typu „text”) do wprowadzania adresu
- przycisk „Pogoda”, po kliknięciu którego wykonywane jest zapytanie asynchroniczne:
  - do API Current Weather: https://openweathermap.org/current za pomocą XMLHttpRequest
  - do API 5 day forecast: https://openweathermap.org/forecast5 za pomocą Fetch API
- obsługa zwrotki z obu API – wypisanie pogody bieżącej oraz prognoz poniżej pola wyszukiwania.

Wygeneruj klucz do API. Ponieważ aktywacja może chwilę potrwać, na czas trwania laboratorium możesz wykorzystać „służbowy” klucz: `7ded80d91f2b280ec979100cc8bbba94`. **UWAGA!** Klucz zostanie dezaktywowany niedługo po zajęciach. Musisz wygenerować swój własny.

W przypadku blokady twórczej można posiłkować się filmem: https://www.youtube.com/watch?v=WoKp2qDFxKk jednakże spróbuj rozwiązać ten problem samodzielnie!

Prowadzący omówi powyższe wymagania. Upewnij się, czy wszystko rozumiesz.

Tu umieść swoje notatki:

…notatki…

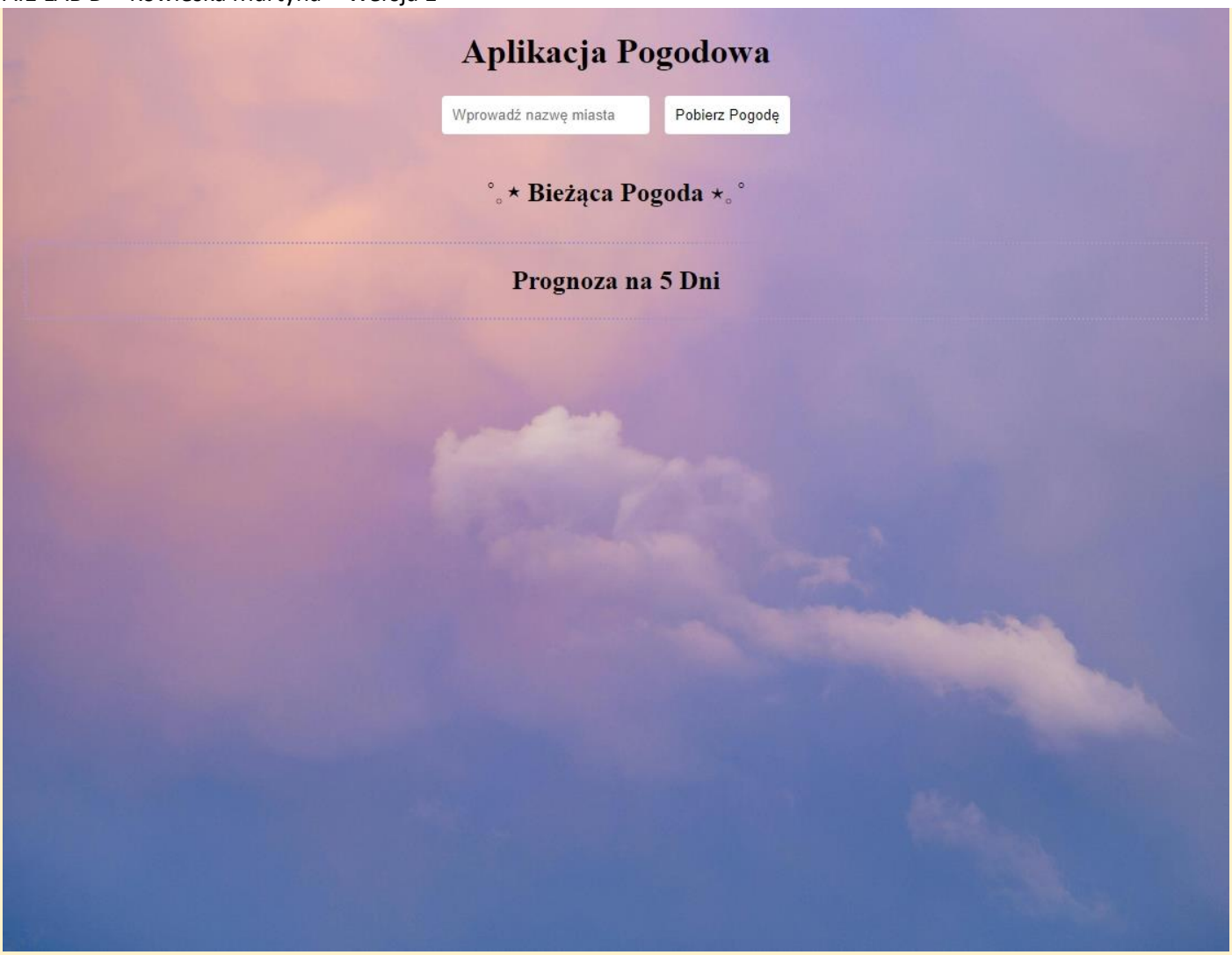
Badanie API

Poświęć kilka minut na wykonanie przykładowych zapytań do API z poziomu pasku adresu przeglądarki. Podaj wymagane parametry dla osiągnięcia różnych wyników. Zbadaj odpowiedzi API, aby uzyskać pełen obraz wymagań i możliwości API.

Implementacja

Tradycyjnie implementację należy zacząć od zbudowania w HTML + CSS wszystkich wymaganych elementów / placeholderów na te elementy. Następnie krok po kroku należy implementować poszczególne zachowania.

Wstaw zrzut ekranu zawierającego stronę ze wszystkimi elementami, tj. pole tekstowe, przycisk, miejsce do wyświetlenia pogody i prognozy:

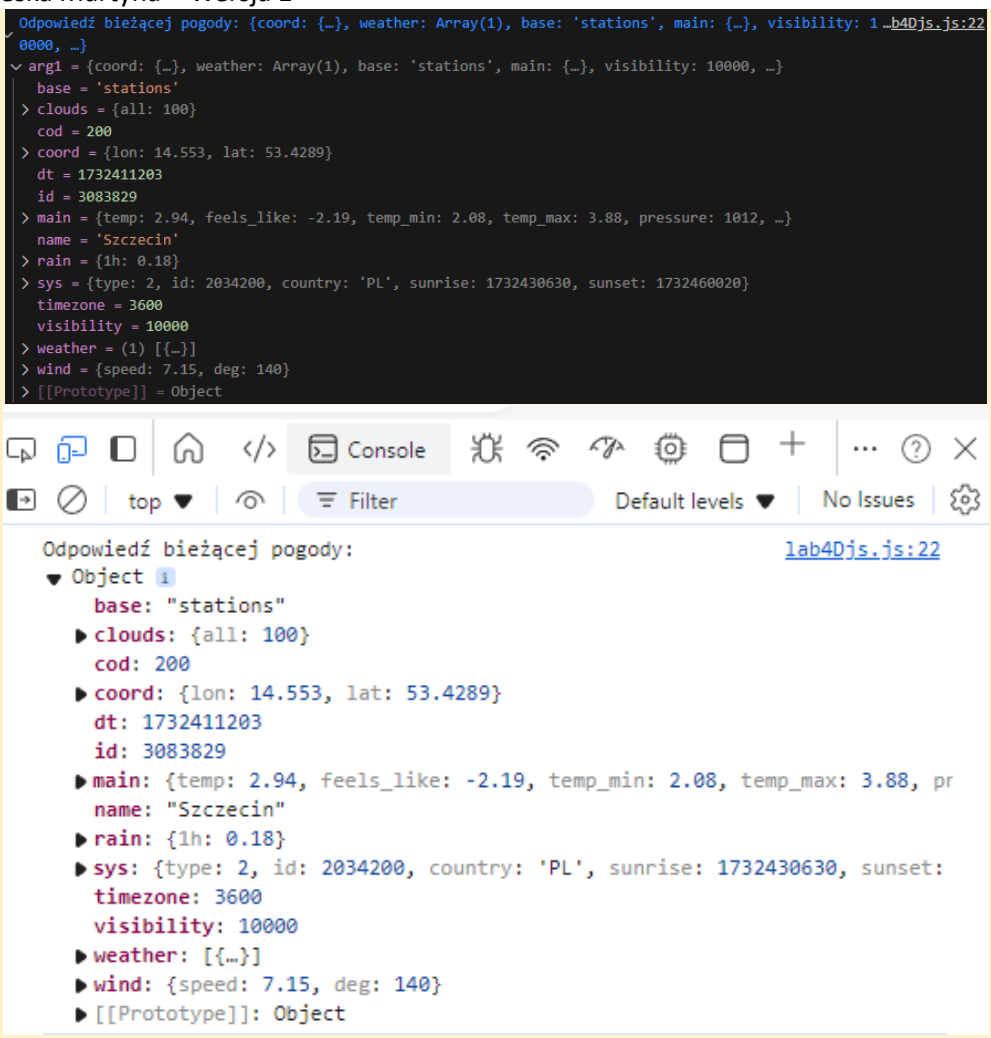


Punkty:	0	1

Wstaw zrzut ekranu kodu odpowiedzialnego za wysyłanie żądania do current za pomocą XMLHttpRequest:

```
function pobierzAktualnaPogode(miasto) {
    const xhr = new XMLHttpRequest();
    xhr.open('GET', `${urlAktualnaPogoda}?q=${miasto}&appid=${kluczApi}&units=metric`);
    xhr.onload = function () {
        if (xhr.status === 200) {
            const dane = JSON.parse(xhr.responseText);
            console.log('Odpowiedź bieżącej pogody:', dane);
            wyswietlAktualnaPogode(dane);
        } else {
            console.error('Błąd podczas pobierania bieżącej pogody:', xhr.responseText);
        }
    };
    xhr.onerror = function () {
        console.error('Wystąpił błąd sieci podczas pobierania bieżącej pogody.');
    };
    xhr.send();
}
```

Wstaw zrzut ekranu pokazujący otrzymaną odpowiedź za pomocą `console.log()` w przeglądarce.

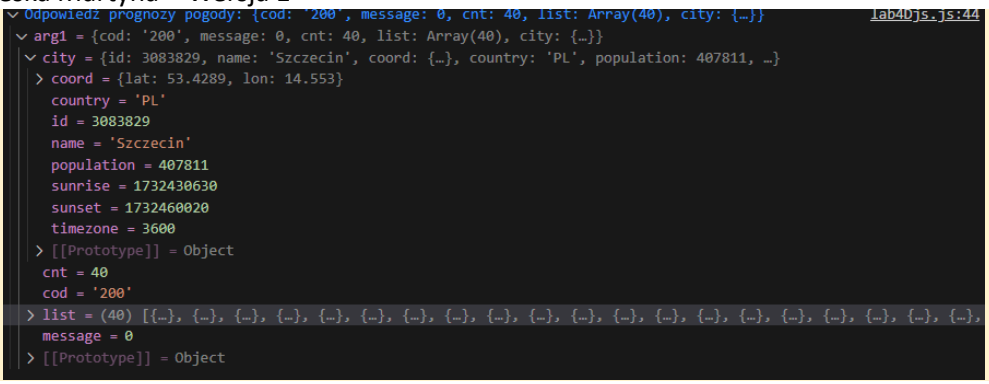


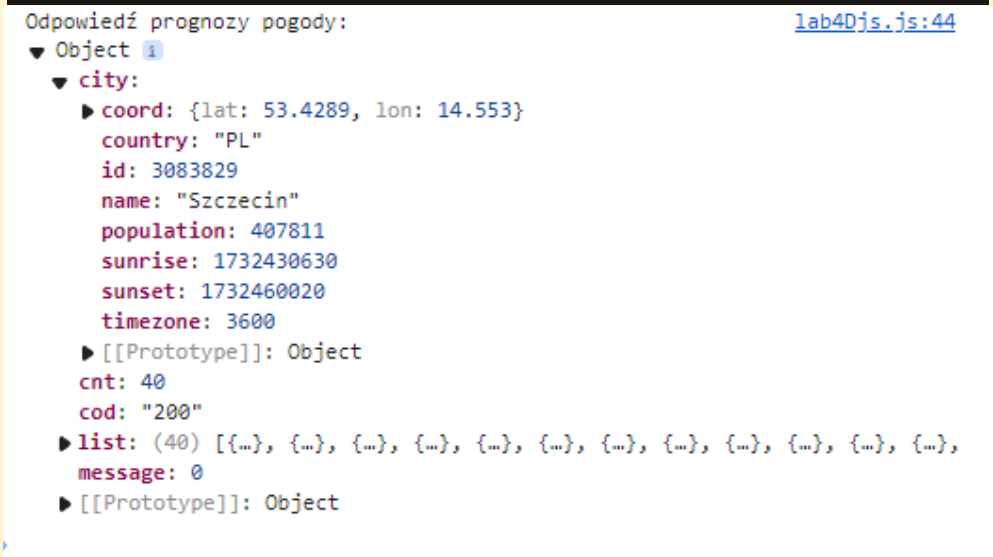
Punkty:	0	1

Wstaw zrzut ekranu kodu odpowiedzialnego za wysyłanie żądania do forecast za pomocą Fetch:

```
function pobierzPrognozePogody(miasto) {
    fetch(`${urlPrognozaPogoda}?q=${miasto}&appid=${kluczApi}&units=metric`)
        .then(odpowiedz => {
            if (!odpowiedz.ok) {
                throw new Error('Błąd podczas pobierania prognozy pogody');
            }
            return odpowiedz.json();
        })
        .then(dane => {
            console.log('Odpowiedź prognozy pogody:', dane);
            wyswietlPrognoze(dane);
        })
        .catch(blad => {
            console.error('Błąd:', blad);
        });
}
```

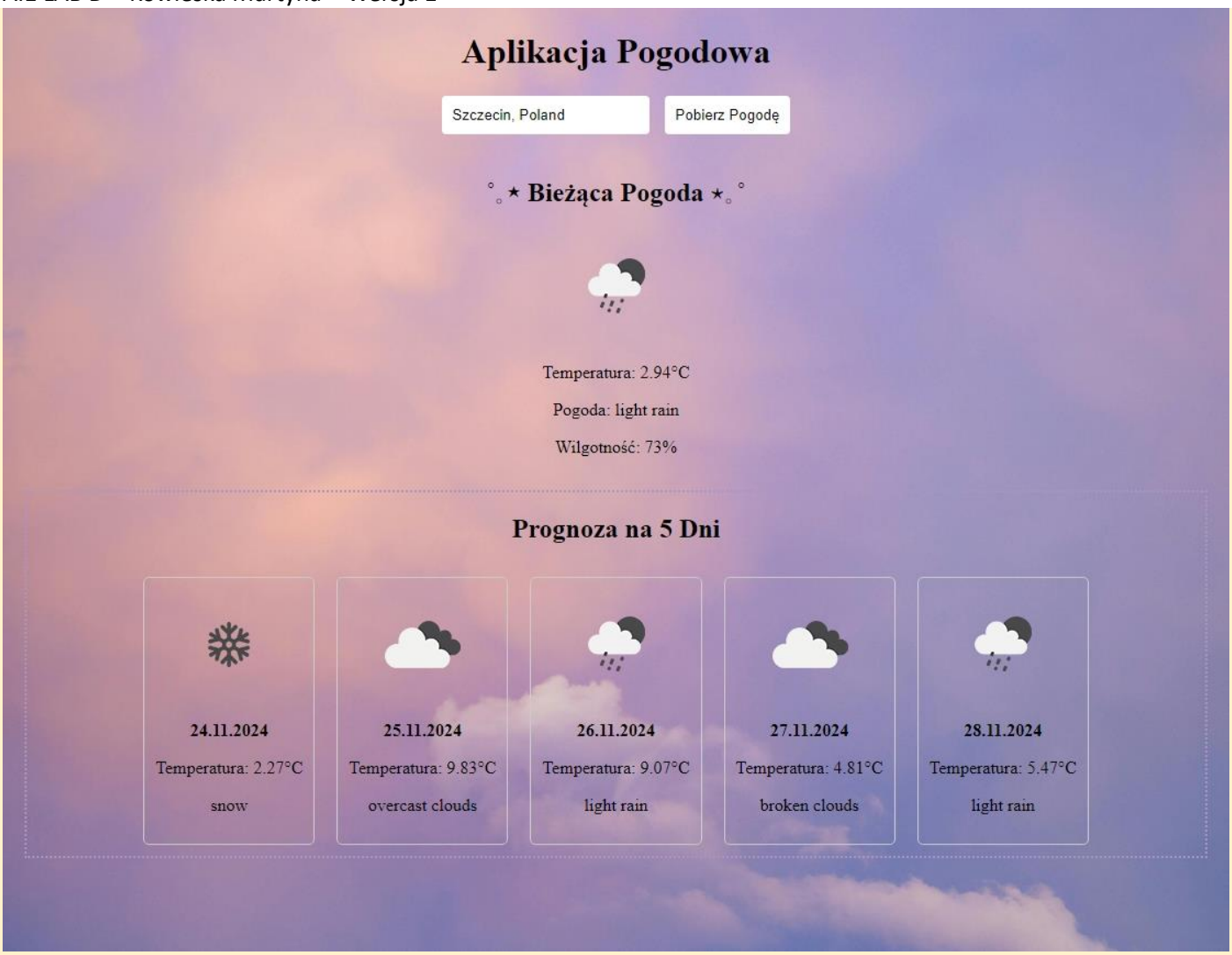
Wstaw zrzut ekranu pokazujący otrzymaną odpowiedź za pomocą `console.log()` w przeglądarce.





Punkty:	0	1

Wstaw zrzut ekranu przedstawiającego wizualizację prognoz pogody:



Upewnij się, że widoczne są pasek wyszukiwania ze wskazaną miejscowością, a także zarówno pogoda bieżąca jak i prognozy pogody.

Punkty:	0	1

COMMIT PROJEKTU DO GIT

Zacommituj i pushnij swoje rozwiązanie do repozytorium GIT.

Upewnij się, czy wszystko dobrze się wysłało. Jeśli tak, to z poziomu przeglądarki utwórz branch o nazwie `lab-d` na podstawie głównej gałęzi kodu.

Podaj link do brancha `lab-d` w swoim repozytorium:
https://github.com/mkowieska/Computer_Science/tree/master/year3/AI-1/lab/lab4D

…link, np. https://github.com/inazwisko/ai1-lab/tree/lab-d…

PODSUMOWANIE

W kilku zdaniach podsumuj zdobyte podczas tego laboratorium umiejętności.

Nauczyłam się tworzyć pogodynkę.

…podsumowanie…

Zweryfikuj kompletność sprawozdania. Utwórz PDF i wyślij w terminie.