

Wprowadzenie do aplikacji Internetowych

laboratorium 3

Cel zajęć:

Celem laboratorium jest przeciwiczenie zagadnień związanych z programowanie z wykorzystaniem JS. Skoncentrujemy się na aspektach zarządzania zdarzeniami oraz modyfikacji modelu DOM.

Zadania na rozgrzewkę (bez oceniania).

Zadanie 1. Po naciśnięciu na przycisk wyświetl okno dialogowe typu prompt pozwalające na wprowadzenie swojego imienia. Podane imię ma być wstawione do przygotowanej sekcji.

Zadanie 2. Przełączanie pomiędzy trzema (lub więcej) zdjęciami. Niech na stronie wyświetlone jest zdjęcie z gór w czerwonym obramowaniu. Po naciśnięciu przycisku zdjęcie zamieni się na zdjęcie morze w niebieskim obramowaniu. Kolejne naciśnięcie to załadowanie np zdjęcia Lasu i inny kolor obramowania. Kolejne naciśnięcie to powrót do pozycji pierwszej (góry).

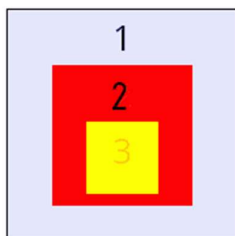
Zadanie 3. Napisz skrypt, w którym za pomocą 2 przycisków modyfikujesz zawartość listy elementów. Jeden przycisk dodaje a drugi usuwa elementy z listy. Niech usuwane będzie zawsze pierwszy element listy.

Zadanie 4. Napisz skrypt, który dodaje lub usuwa obsługę przycisku testowego. Naciśnięcie przycisku inkrementuje licznik przyciśnięć. Odpięcie zdarzenia resetuje licznik

Zadania punktowane.

Sekcja podstawowa (max 10 pkt)

Zadanie 5. Stwórz stronę zawierającą 3 elementy np. divy/zdjęcia spozycjonowane tak jak na rysunku poniżej. (2 pkt)



Niech naciśnięcie któregokolwiek z nich wyświetla odpowiedni komunikat w odpowiedniej sekcji na stronie.

Żółtego - „naciśnąłeś żółty o wartości 5”
Czerwonego – „naciśnąłeś czerwony o wartości 2”
Niebieski – „naciśnąłeś niebieski o wartości 1”

Jeśli suma naciśniętych wartości przekroczy 30 należy wyłączyć możliwość obsługi zdarzenia click przez obiekt 2. Jeśli 50 usuwamy dodatkowo obsługę zdarzenia przez obiekt 3.

Niech na stronie będą jeszcze 3 przyciski: Stop/StartPropagation, Reset, Zmiana kolejności wywoływania zdarzeń.

Stop/StartPropagation – włącza i wyłącza propagację (niech tekst na przycisku odpowiada stanowi przycisku)

Reset – wraca do stanu początkowego – reset licznika punktów oraz możliwość obsługi zdarzeń typu click.

Zadanie 6. Napisz skrypt, w którym za pomocą przycisk Dodaj będzie dodawał nową pozycję do książki telefonicznej (imię, nr telefonu). Dane podajemy za pomocą formularza. Dodawana pozycja zawiera również przycisk (w formie ikony – kosz) pozwalający na usuwanie elementów z książki adresowej. Wymagana jest walidacja wprowadzanych wartości tak aby numer telefonu to były tylko cyfry w formacie 9 lub 12 cyfr ewentualnie zaczynające się znakiem + np. +048 123456789. Znaki mogą być poroździelane spacjami lub innymi białymi znakami – format 12 34 56 78 9 , lub 1234 56789 jest poprawny. Imię i Nazwisko musi rozpoczynać się od dużej litery i zawierać tylko litery w dowolnym alfabecie + ‘-’ (cyfry i inne znaki specjalne wykluczone). Dozwolone są nazwiska dwuczłonowe. Przykład poniżej. **(2pkt)**

The screenshot shows a web application interface for managing a contact list. It features a list of existing contacts and a form to add new ones.

Imię i Nazwisko	Telefon	Akcja
Grzegorz Rogs	500 600 700	
Jacek Dajda	500 600 700	

IMIĘ I NAZWISKO

TELEFON

Dodaj

Zadanie 7. W katalogu Zadanie 7 znajdziesz plik city.json zawierający kolekcję miast Polski.

Celem ćwiczenia jest weryfikacja umiejętności przetwarzania plików zewnętrznych z wykorzystaniem Fetch API. Aby można było przetwarzać pliki zewnętrzne (pliki json-owe) potrzebujemy serwera który będzie je hostował. Proponuje użycie json-server. W celu jego zainstalowania potrzeba posiadania jakiegoś menedżera pakietów: npm lub yarn.

Aby zainstalować JSON Server, należy otworzyć konsolę i wpisać poniższe polecenie:

```
npm install -g json-server
```

Flaga -g spowoduje, że serwer zostanie zainstalowany globalnie w systemie, co pozwoli uruchomić go z każdego miejsca.

Aby uruchomić JSON Server, proszę otworzyć wiersz poleceń i wpisać:

```
json-server db.json
```

W tym momencie serwer udostępnia pliki db.json i będzie można przy użyciu Fetch API przeczytać jego zawartość.

Napisz funkcje które:

- a). wyświetli na stronie tylko miasta z województwa małopolskiego
- b). wyświetli miasta które w swojej nazwie posiadają dwa znaki 'a'
- c). wyświetli piąte pod kątem gęstości zaludnienia miasto w Polsce.
- d). dla wszystkich miast powyżej 100000 dodać (na końcu) city do nazwy.
- e) wyliczyć czy więcej jest miast powyżej 80000 mieszkańców czy poniżej wraz z informacją o ich liczbie.
- f). jaka jest średnia powierzchnia miast z powiatów zaczynających się na literkę „P”
- g) odpowiedz na pytanie czy wszystkie miasta z województwa pomorskiego są większe od 5000 osób i ile jest takich miast.

Wyniki wyświetlić na dedykowanych sekcjach na stronie. **(2pkt)**

Zadanie 8. Zaimplementuj formularz zmiany hasła, który pozwala na ocenę siły hasła. **(2pkt)**

Formularz wygląda jak poniżej:





New password

Ala1 

Repeat new password

**** 

Password requirements

-  at least 8 characters
-  at least one special character
-  at least one capital letter
-  at least one digit

Wizualna ocena siły hasła dotyczy pierwszego pola – new password. Weryfikacji podlega również na zgodność podanych haseł. Gdy jej brak powinien wyświetlić się odpowiedni komunikat. Możliwe jest również podglądanie wprowadzanego hasła.

Zadanie 9. Napisz slider wizytówek. Wizytówka składa się ze zdjęcia, stanowiska i przykładowego opisu. Przyciski < i > pozwalają na przeglądanie naszych pracowników. Dodatkowo istnieje przycisk do losowego wyboru wizytówki do wyświetlenia

Podczas przeglądania zastosować interesujące efekty przejścia – takie jak w pokazanych przykładach. **(1pkt)**



Zadanie 10. Napisz prosta app, w której masz wydzielony obszar (prostokąt). Naciśnięcie w dowolnym miejscu tego obszaru powinno skutkować płynnym przesunięciem czerwonego kółka do tego miejsca. Naciśnięcie poza obszarem zaznaczonym – wyświetlenie komunikatu z odpowiednią informacją. **(1pkt)**



Sekcja rozszerzona (max 10 pkt)

Zad 11. Dynamiczna tablica (5pkt)

Stworzyć tablice zasilaną danymi zewnętrznymi w formacie json. Tablica powinna umożliwić filtrowanie danych oraz jej sortowanie – po każdej wskazanej kolumnie. Możliwe jest jednocześnie filtrowanie kilku kolumn (1pkt)

Lista kolumn ustalana dynamicznie na podstawie formatu json. Tablica na jednym ekranie wyświetla tylko ustaloną liczbę pozycji. Wymagana jest więc obsługa paginacji. (1pkt)







Pod adresem <https://restcountries.com/v3.1/all> znajdziesz listę wszystkich krajów świata. Wykorzystując znajdujące się tam dane zasili nimi tablice.

Wyświetlane powinny być następujące dane: Name, Capital, population, area

Niech kraje będą grupowane pod kątem Subregionu do którego należą. Na poziomie subregionu wyświetlane są tylko nazwa subregionu, sumaryczna ilość ludzi zamieszkujących ten subregion oraz całkowita jego powierzchnia. Oczywiście jest możliwe rozwijanie subregionu aby zobaczyć poszczególne kraje należące do niego. Wizualnie to może wyglądać tak jak przykład poniżej (uwaga! to jest tylko propozycja nie związana z tematem zadania) (2pkt)

Oceniana jest również jakość kodu (1pkt)

Name	Country	Company	Status	Date
>	 Amy Elsner			
>	 Anna Fali			
>	 Asiya Javayant			
>	 Bernardo Dominic			
>	 Elwin Sharvill			

Name	Country	Company	Status	Date
>	 Amy Elsner			
▼	 Anna Fali			
Minna Amigon	 Romania	Dorl, James J Esq	QUALIFIED	2018-11-07
Mattie Poquette	 Venezuela	Century Communications	NEGOTIATION	2017-12-12
Roxane Campain	 France	Rapid Trading Intl	UNQUALIFIED	2018-12-25
Total Customers			3	
>	 Asiya Javayant			

Zadanie 12. Farma Zoombie (5 punktów)

Twoim zadaniem jest implementacja gry widocznej na filmie (dołączonym w materiałach) i screenshocie poniżej.



W zakładce zadanie11 znajdziesz grafiki przydatne do realizacji gry.

Poniżej znajdziesz założenia gry i punktację

Wersja podstawowa, zawiera (3 punkty)

- podstawowy mechanizm strzelania do zoombie – strzelamy za pomocą myszki (lewym klawiszem)
- zoombie pojawiają się z prawej strony ekranu i poruszają się w stronę lewą
- zoombie może pojawić się na różnej wysokości ekranu
- pojawienie się zoombie jest generowane losowo
- szybkość poruszania jest również generowana losowo w pewnych przedziałach np. 1 – 5, gdzie 1 to standardowa szybkość a 5 to tryb turbo.
- wielkość zoombie jest również losowa
- każdy zestrzelony zoombie to 12 punktów wyświetlanych na ekranie w trybie on-line
- gra kończy się gdy trójka zoombie dotrze do końca ekranu
- punktacja statyczna - za każdego zoombie naliczane punkty (jak wyżej), za niecelny strzał ujemne punkty (-6pkt)

- jakość kodu (1 punkt)

Ranking (highscores) (1 punkt)

- na wstępie gra prosi o podanie nicka (do skutku) - potem ten nick jest wyświetlany nad planszą do gry
- po zakończeniu gry pobieramy zapisane na serwerze wyniki. Można do tego celu użyć:
// highscore json blob: <https://jsonblob.com/> WSTAW_SWOJ_HASH_JSON")
- dodajemy swój wynik, sortujemy, odcinamy pierwsze 7 wyników i zapisujemy znowu na serwerze
- ranking zawiera pozycję, nicka, ilość punktów i datę wpisu
- przy prezentacji highscores kursor zmienia się na zwykły
- pojawia się guzik umożliwiający powrót do nowej rozgrywki