

Wstęp teoretyczny

Operacje na tablicach

Podstawowe funkcje dostępne dla **Tablicy**:

- [forEach](#)
- [map](#)
- [find](#)
- [sort](#)

Pobieranie i wyświetlanie danych

Do pobierania danych można użyć funkcji [fetch](#).

Należy zwrócić uwagę, że jest to funkcja asynchroniczna. Aby poczekać na dane, możemy użyć słowa **await**.

W celu tworzenia dynamicznie elementów warto użyć funkcji [createElement](#).

```
document.createElement('div');
```

Należy pamiętać o tym, żeby wyczyścić stare elementy przy ponownym renderowaniu:

```
document.querySelector(selector).innerHTML = "";
```

Dodatkowo konieczne może być znalezienie interesującego nas elementu na stronie:

```
document.getElementById(elementId);  
document.getElementsByClassName(className);  
document.querySelector(selector);
```

Przy tworzeniu dynamicznych elementów można skorzystać z funkcji [appendChild](#):

```
document.appendChild(element);
```

Tworzenie gry przy użyciu canvas

Tutoriale z tworzenia gier:

- [Bouncing Ball](#)
- [Block Hopper](#)
- [Platform Game](#)

Do stworzenia pętli rozgrywki będzie potrzebna funkcja [requestAnimationFrame](#).

Dodatkowo warto zwrócić uwagę na możliwość dodawania [eventListenerów](#).

Pamiętamy o ich usunięciu!! [removeEventListener](#)

Lab 3 - Ćwiczenia [5.0 pkt]

[0.5 pkt] Ćwiczenie 1: Stoper

- Na stronie znajdują się przyciski "start", "stop" oraz "reset"
- Przyciski kontrolują stoper odmierzający czas w formacie "Xmin Xs" np. "3min 48s", dla czasu poniżej minuty - "51s".

[1.5 pkt] Ćwiczenie 2: Generator hasła

- Na stronie znajdują się następujące kontrolki/ustawienia:
- Minimalna długość hasła (pole tekstowe)
- Maksymalna długość hasła (pole tekstowe)
- Czy hasło ma zawierać wielkie litery? (checkbox)
- Czy hasło ma zawierać znaki specjalne? (checkbox)
- Przycisk "Wygeneruj hasło"
- Kliknięcie przycisku "Wygeneruj hasło" powoduje wygenerowanie hasła na podstawie zdefiniowanych w kontrolkach kryteriów i wyświetlenie go z wykorzystaniem funkcji `alert()`

Ćwiczenie 3: Pobranie i wyświetlenie danych z API

- **[1.5 pkt]** Pobrać dane spod adresu <https://dummyjson.com/products> i wyświetlić w formie tabelki pierwsze 30 elementów z kolumnami:
 - zdjęcie
 - tytuł
 - opis
- **[1.5 pkt]** Na górze strony dodać input pozwalający na filtrowanie danych po wpisanej frazie, obok input'a dodać [select](#) pozwalający na sortowanie danych:
 - rosnąco
 - malejąco
 - oryginalna kolejność

Lab 3 - Zadanie domowe [5.0 pkt]



Proszę zaimplementować grę z powyższego screenshot'a:

- Podstawowy wygląd gry
- Tło powinno przykrywać całą planszę
- Celownik jest zmieniony na odpowiednie zdjęcie
- W lewym górnym rogu znajdują się serca oznaczające aktualną ilość żyć.
- W prawym górnym rogu znajduje się aktualny wynik gracza
- Podstawowe działanie gry:
- Gracz zaczyna mając 3 życia
- Celujemy i strzelamy myszką (lewy click)
- Za strzał w zombie gracz otrzymuje 20 punktów
- Nie trafienie w zombie skutkuje stratą 5 punktów
- Gdy zombie dojdzie do lewej części ekranu, gracz traci życie
- Gdy ilość żyć spadnie do zera, gra kończy się
- Gdy gra się zakończy należy pokazać pop-up z przyciskiem umożliwiającym ponowne uruchomienie gry, a w tle powinna być odtworzona smutna muzyka
- Zachowanie Zombie:
- pojawiają się na prawej stronie planszy (dowolna wysokość i miejsce)
- poruszają się w lewo (losowa prędkość)
- mają losowy rozmiar (skale)
- zaimplementowana jest animacja chodu