

# Instrukcja II - HTML, CSS

🕒 Created	@March 17, 2024 2:31 PM
🏷️ Tags	Wstęp

## Laboratorium II: HTML + CSS

### 0a. Przygotowanie środowiska do pracy:

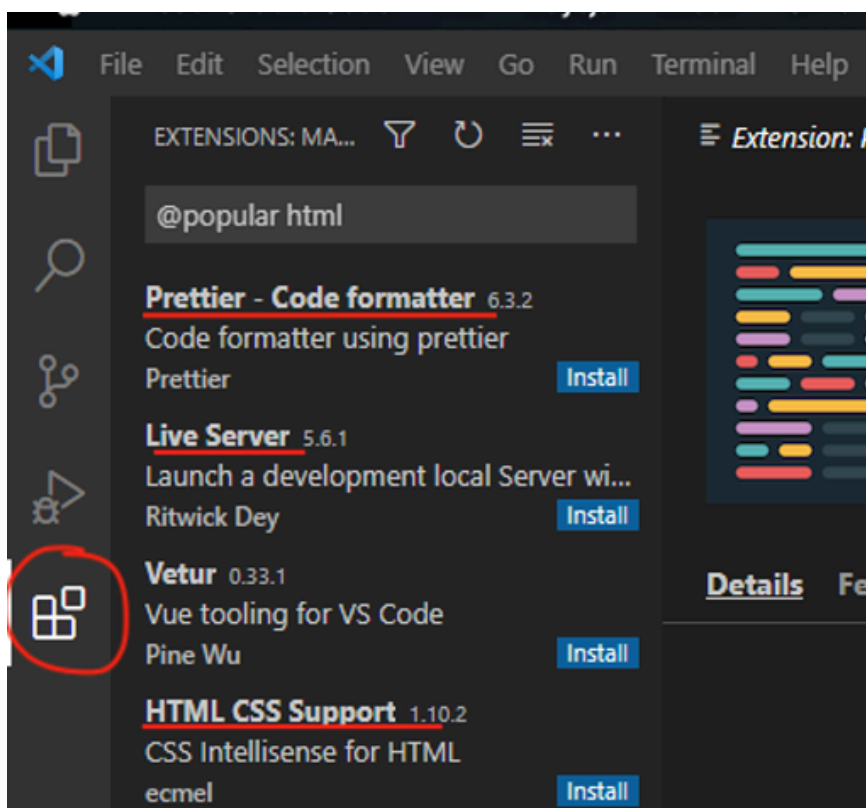
Proszę utworzyć nowe repozytorium w serwisie [GitHub.com](https://github.com) o nazwie "**Zajęcia 2 HTML CSS**", repozytorium powinno być publiczne. Następnie proszę sklonować je w programie Visual Studio Code. Proszę utworzyć branch "**develop**", do którego w trakcie pracy nad instrukcją proszę wykonywać systematycznie *commity*. Po wykonaniu wszystkich ćwiczeń proszę wykonać *pull request* do brancha **main**.

### 0b. Przygotowanie środowiska do pracy:

Frontend aplikacji webowych można w zasadzie tworzyć w dowolnym oprogramowaniu np. Windowsowym notatniku. Jednakże w celu zwiększenia wygody pracy (odpowiednie formatowanie oraz kolorowanie składni), zalecam używanie środowiska **Microsoft Visual Studio Code** (<https://code.visualstudio.com>) oraz przeglądarki internetowej **Google Chrome** ([https://www.google.com/intl/pl\\_pl/chrome/](https://www.google.com/intl/pl_pl/chrome/)) ewentualnie korzystanie ze

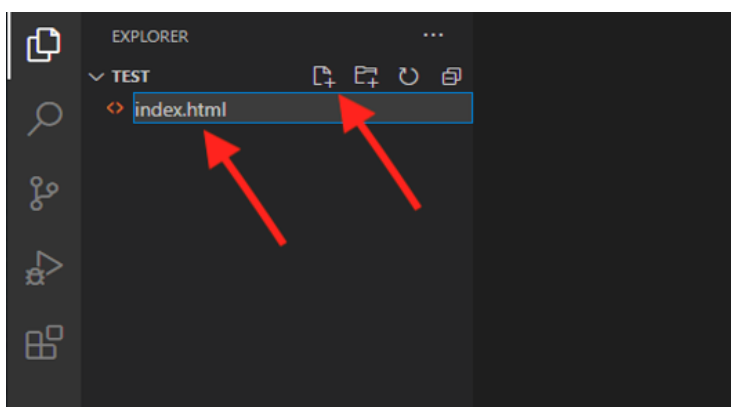
środowiska online np. <https://replit.com>. Oprogramowanie to wystarczy nam w zupełności do pracy zarówno z HTML, jak i JavaScript oraz Python.

Po instalacji oprogramowania konieczne będzie zainstalowanie dodatków: *LiveServer*, *HTML CSS Support* oraz *Prettier*. W tym celu należy kliknąć na ikonkę zaznaczoną na obrazku poniżej i wyszukać odpowiednie dodatki.



***Po instalacji dodatków należy ponownie uruchomić Visual Studio Code***

W kolejnym kroku należy wybrać z menu VSC File → Open Folder i wybrać folder stanowiący katalog roboczy. Następnie wybieramy przycisk „dodaj plik” i wpisujemy dowolną nazwę z rozszerzeniem *.html*

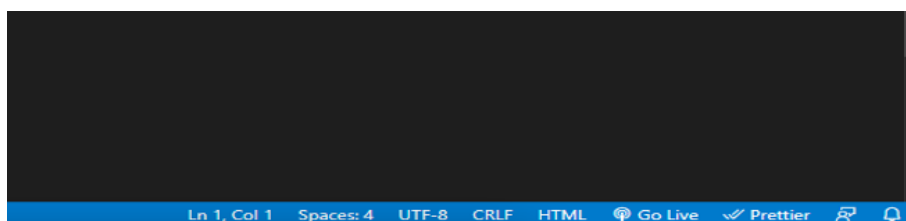


## 1. Podstawowe elementy języka HTML:

Proszę uzupełnić treść stworzonego pliku o podstawowy szablon strony w języku HTML jak na przykładzie poniżej.

```
<> index.html ●
<> index.html > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4  |   <title>Moja pierwsza strona internetowa</title>
5  </head>
6  <body>
7  |   To jest moja pierwsza strona internetowa w języku HTML.
8  </body>
9  </html>
```

Jeśli chcą Państwo przetestować przygotowany dokument w przeglądarce, należy z dolnego paska wybrać opcję "Go Live".



Powinno wyświetlić się okno przeglądarki Chrome z widokiem Państwa dokumentu.

Każdorazowa zmiana kodu w dokumencie i zapisanie go skrótem klawiszowym **ctrl+s**, spowoduje aktualizację wyglądu strony w przeglądarce.



W celu przyspieszenia pisania kodu można zastosować tzw. Emmety: <https://docs.emmet.io/cheat-sheet/>

### Zadanie 1.

Proszę zapoznać się ze ściągawką znaczników używanych w języku HTML5 dostępną pod adresem: <https://websitesetup.org/HTML5-cheat-sheet.pdf>.



## Tutorial

<https://www.w3schools.com/html/>

Można zauważyć, że ze specyfikacji języka HTML wycofano znaczniki formatujące czcionki, tło dokumentu czy chociażby ustawienie obrazka na stronie. Znaczniki te zostały zastąpione kaskadowym arkuszem stylów **CSS3**.

Do utworzonego wcześniej pliku proszę dodać następujące elementy:

- > Koniec linii
- > Akapit
- > Tekst pogrubiony
- > Tekst pochylony
- > Tekst podkreślony
- > Rodzaj czcionki
- > Łączenie parametrów
- > Wstawienie obrazka
- > Ustawienie parametrów obrazka
- > Odsyłacz do podstrony
- > Odsyłacz do adresu internetowego
- > Odsyłacz pocztowy
- > Odsyłacz obrazkowy
- > Przycisk
- > Cytat
- > Indeksy dolne i górne

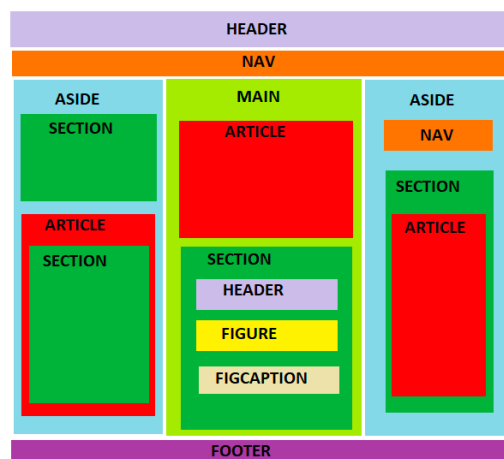
### Zadanie 2.

Podstawowym sposobem blokowania elementów witryny WWW są znaczniki sekcji `<div>` - będą umożliwiały one późniejsze formatowanie zawartych w nich elementów za pomocą arkusza stylów CSS.

W języku HTML5 oprócz sekcji, można także podzielić layout strony za pomocą znaczników blokowych:

- > *article*

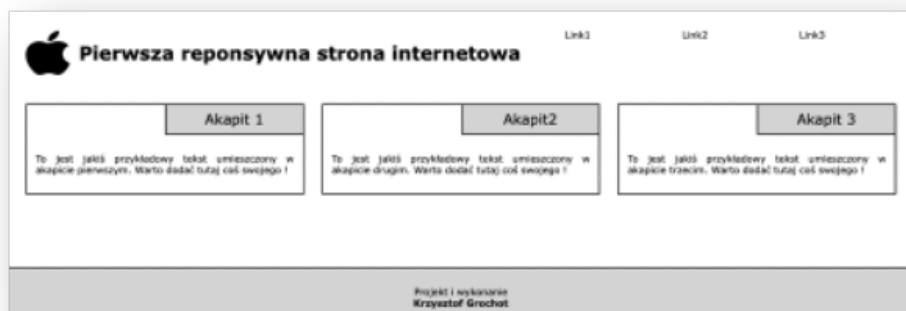
- > *aside*
- > footer
- > header
- > main
- > nav
- > section

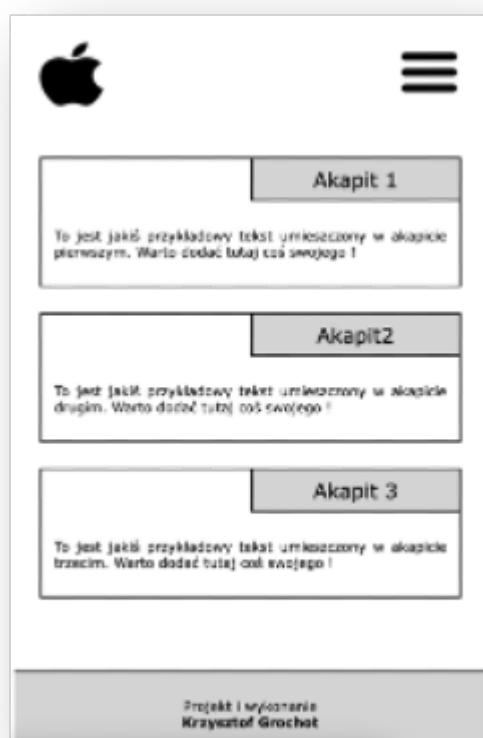


źródło: <https://kamakaczmarek.net/html-5-znaczники-section-article-aside-hgroup-header-footer-nav-main-details-summary/>

Proszę odpowiednie fragmenty przygotowywanego dokumentu umieścić w znacznikach `<div>` lub znacznikach blokowych stosownie do ich przeznaczenia.

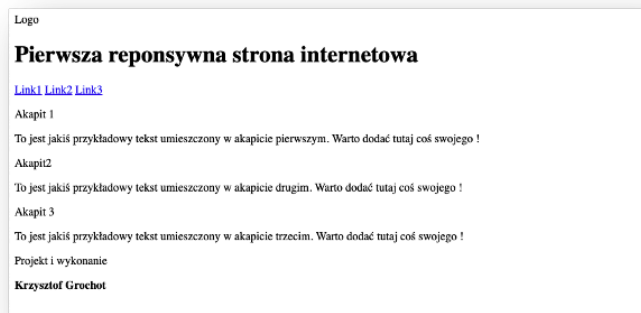
Głównym celem instrukcji jest stworzenie responsywnej strony/interfejsu webowego dostosowującego swój wygląd do urządzeń desktopowych oraz mobilnych. Gotowa strona w dwóch wariantach będzie wyglądała jak poniżej:



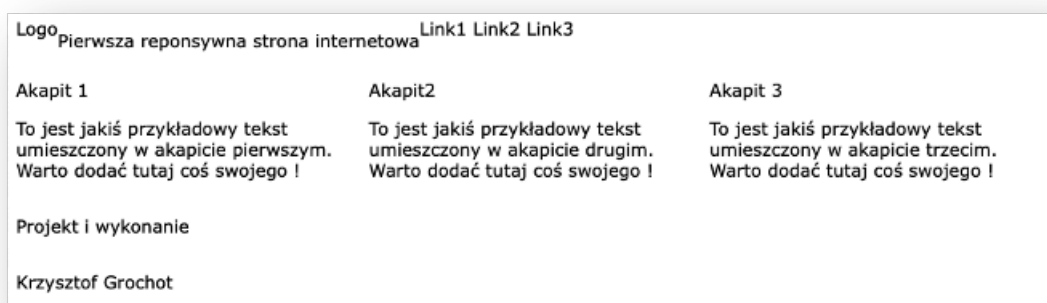


1. Przygotuj strukturę strony wraz z przykładową treścią. Strona jest podzielona na trzy główne sekcje: nagłówek (header), zawartość (main) oraz stopkę (footer). W nagłówku znajduje się pole z napisem „Logo”, które posłuży jako logo. Poniżej znajduje się tytuł strony oraz sekcja nawigacyjna (nav). Sekcja główna składa się z trzech artykułów (article), z których każdy ma nagłówek oraz treść. Stopka składa się z dwóch akapitów.

Po uruchomieniu strona bez żadnego stylu CSS (zobacz: [https://www.w3schools.com/css/css\\_syntax.asp](https://www.w3schools.com/css/css_syntax.asp)) powinna wyglądać podobnie jak poniżej.



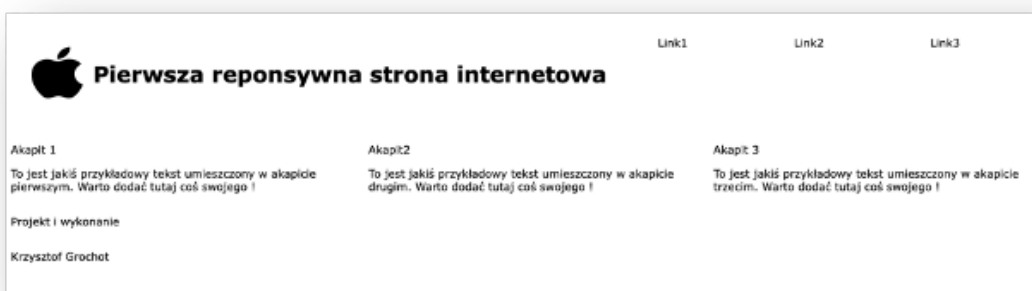
2. Proszę zresetować domyślne marginesy (margin) i wypełnienia (padding) elementów *html*, *body* oraz *p* ustawiając parametr na 0. Utwórz osobny plik CSS, dołącz go do dokumentu HTML wprowadzając odpowiednią regułę ([https://www.w3schools.com/css/css\\_howto.asp](https://www.w3schools.com/css/css_howto.asp)).
3. Proszę wybrać sposób definicji wielkości znajdujących się na stronie. Odpowiadać za to będzie właściwość *box-sizing*. Na podstawie strony: <https://css-tricks.com/box-sizing/> proszę wybrać jedną z wartości.
4. Wszystkim elementom na stronie proszę ustawić czcionkę Verdana o rozmiarze 16px oraz wyłączyć domyślne pogrubienie (*font-weight*) i podkreślenie (*text-decoration*) a także ustawić kolor wszystkich elementów na czarny.
5. Odpowiednie rozmieszczenie elementów na stronie zapewni mechanizm układu CSS zwany flexboxem (<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>). Korzystając z właściwości *display* oraz *flex-direction* proszę ustawić parametry każdej z głównych sekcji. Po tym zabiegu strona powinna wyglądać mniej więcej jak poniżej:



6. Dodaj wypełnienie nagłówka w wielkości **30px** oraz wyrównania elementów w pionie (korzystając z jednej z właściwości flexbox'a). Dla tytułu strony w nagłówku ustaw lewy margines wielkości **15px** oraz pogrubioną czcionkę w rozmiarze dwa razy większym niż w reszcie dokumentu.
7. Ustaw wielkość sekcji z linkami (nav) na 2 razy większą niż sekcje z logo oraz tytułem strony (flex-grow) oraz przesun wszystkie linki w tej sekcji do górnej krawędzi sekcji (align-self). W sekcji nav włącz mechanizm flexbox'a (display) a następnie przesun wszystkie odnośniki do prawej krawędzi (justify-content).
8. Ustaw szerokość każdego odnośnika na **75px**.

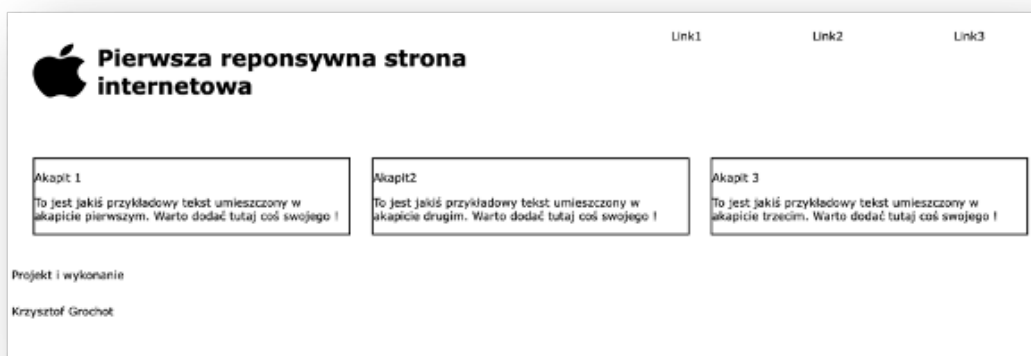


9. W miejscu logo wstaw obrazek o wymiarach 75px na 75px.
10. Strona powinna wyglądać podobnie jak poniżej:

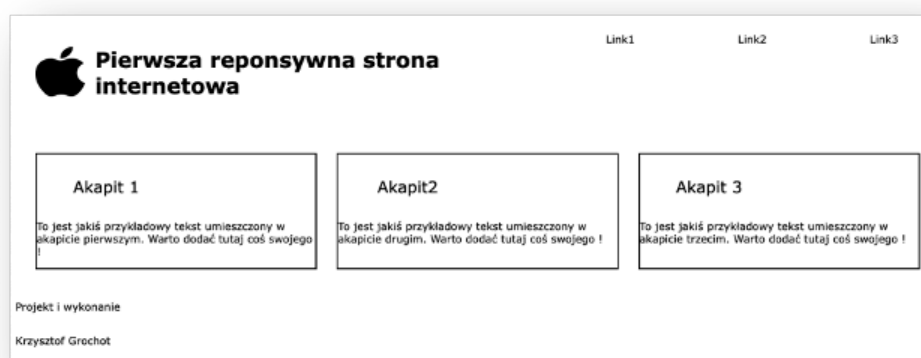


11. Proszę ustawić wypełnienie sekcji głównej (main) oraz marginesy artykułów na wartość 15 px. Proszę ustawić szerokość każdego artykułu na 100% (aby zawsze zajmował tyle samo miejsca, niezależnie od ilości zawartego w nim tekstu). Następnie do artykułów dodaj czarną ramkę o grubości 2px

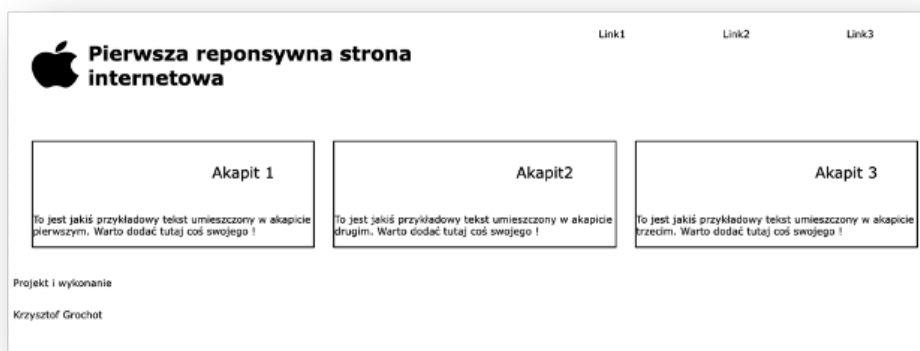




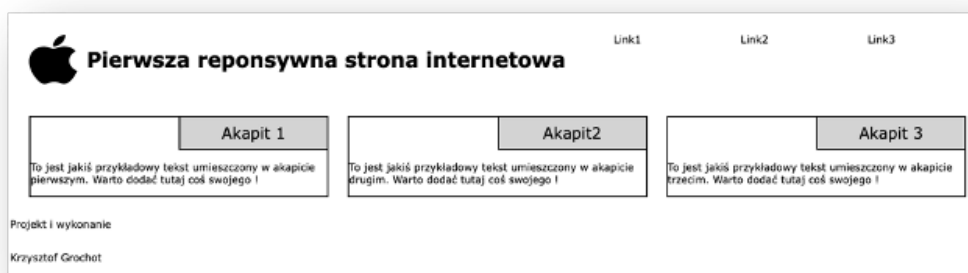
12. Ustaw każdemu nagłówkowi wysokość **50px**, szerokość **50%** oraz rozmiaru czcionki 1.5em, a następnie wycentruj tekst w każdym nagłówku (display, justify-content, align-items).



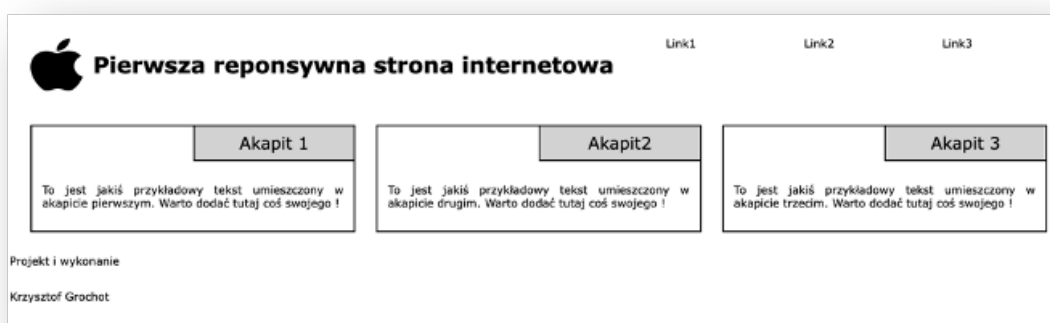
13. Wyrównaj nagłówki każdego z artykułów do prawej strony za pomocą flexbox'a. W artykułach włącz układ flexbox i ustaw kierunek kolumnowy. W nagłówkach ustaw odpowiednie wyrównanie (align-self).



14. W nagłówkach ustaw jasnoszare tło oraz lewą i dolną czarną ramkę o grubości **2px**.

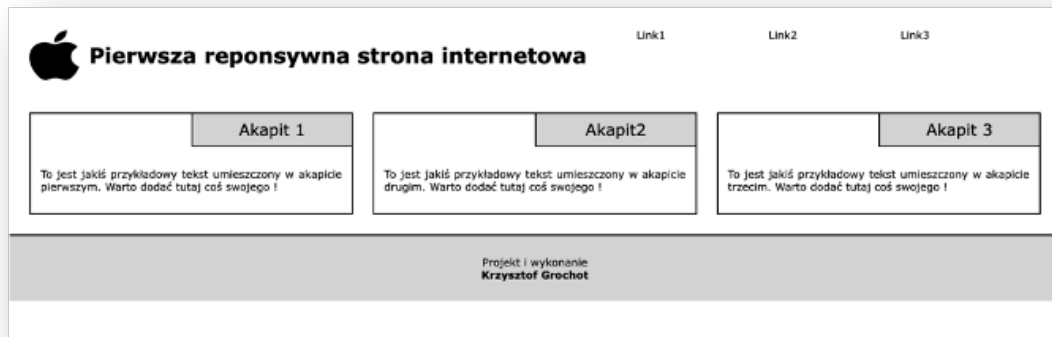


15. Wyjustuj akapity w artykułach (text-align) oraz ustaw wypełnienie na **15px**.

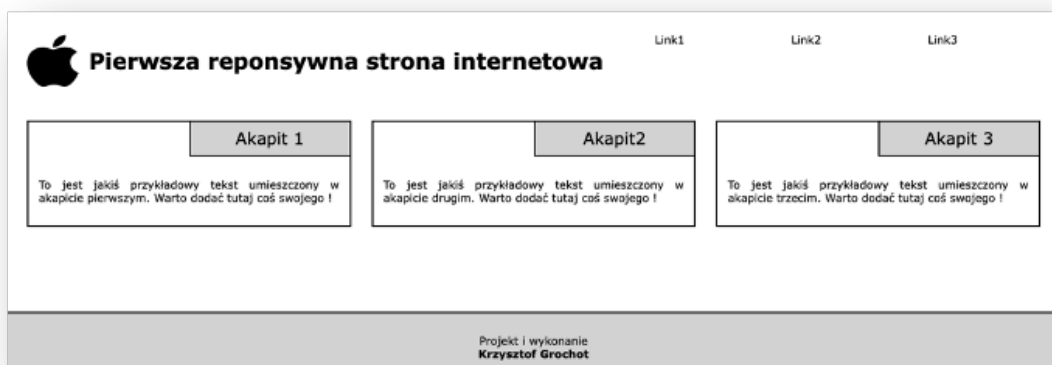


16. Ustaw stopce stałą wysokość **100px**, jasnoszare tło, czarną górną ramkę o grubości **2px** i górny margines o wysokości **15px**, a następnie wycentruj

zawartość w pionie i poziomie (`align-items`, `justify-content`). Ustaw drugiemu akapitowi w stopce pogrubioną czcionkę.



17. Za pomocą flexbox'a (<https://css-tricks.com/couple-takes-sticky-footer/>) zaimplementuj mechanizm przyklejonej stopki (*sticky footer*).

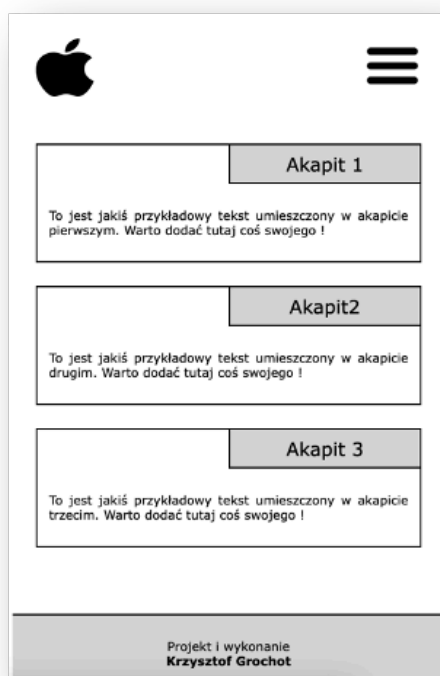


18. Aby umożliwić obsługę małych ekranów i definiować style w zależności od parametrów przeglądarki trzeba posłużyć się zapytaniami media. Na końcu arkusza CSS dodaj poniższą regułę i zobacz co się dzieje ze stroną w momencie przekroczenia progu 900px szerokości okna.

```
@media all and (max-width: 900px) {  
  main {  
    flex-wrap: wrap;  
  }  
}
```

```
}  
}
```

19. Do reguły media dodaj zapytania, które spowodują zniknięcie tytułu oraz sekcji nawigacyjnej na mniejszych urządzeniach (`display:none`), a w ich miejsce pojawi się tzw. hamburger. Dodaj do sekcji nagłówkowej obrazek „hamburgera” i ustaw go domyślnie jako niewyświetlany. Element ten powinien pojawiać się dopiero na mniejszych urządzeniach (`display:block`). Zmodyfikuj w sekcji nagłówkowej rozmieszczenie elementów tak, aby na mniejszych urządzeniach logo pozostało po lewej stronie, natomiast hamburger po prawej (`justify-content`).



20. Wprowadzeniem elementów interaktywnych na stronie z użyciem języka JavaScript będziemy zajmowali się na kilku kolejnych zajęciach.