

# Warsztat - HTML i CSS

v3.1



# Plan

- Wskazówki ogólne
- Wskazówki szczegółowe: od czego zacząć?

# Wskazówki do projektu Sit on Chair

# Który projekt mam zrobić?

Warsztat Sit On Chair jest obowiązkowy. Wzoruj się na pliku **warsztat1.jpg**.

Nie musisz robić podstron do tego projektu, wystarczy strona główna (ta którą widać w pliku jpg).

Strona główna powinna być odzwierciedlona przez Ciebie w HTML i CSS 1:1, powinna wyglądać tak samo jak projekt.

# Jednostki

- Używamy **pixeli** do ustawiania szerokości, odstępów i innych wielkości w projekcie (Ewentualnie **em**, **rem**). Tylko elementy rozciągnięte na całą szerokość strony powinny mieć ustawiony **width: 100%;**.

(Nie martwimy się, że strona będzie się źle wyświetlała na urządzeniach z mniejszą szerokością ekranu, tym zajmiemy się później, przy okazji omawiania RWD)

# Układ

- Pamiętajmy o korzystaniu z kontenera, który będzie odpowiadał za wyśrodkowanie strony np. `main_width`, wewnątrz elementów rozciągniętych na 100% szerokości okna.



# Czczionka

- Czczionka użyta w projekcie to **Ra<sup>l</sup>eway**. Możemy ją ściągnąć ze strony <http://www.fontsquirrel.com/fonts/raleway>.

Pamiętamy o wygenerowaniu różnych typów czczionki.

# Szczegóły

Pamiętamy, aby zadbać o każdy szczegół:

1. czcionki (w niektórych miejscach jest pogrubiona w innych normalna)
2. rozmiary (sprawdźmy rozmiary elementów, przed ich ustawianiem)
3. kolory
4. interlinia ( **line-height**) (pamiętamy o interlinii)
5. odstępy pomiędzy elementami

Wszystko powinno jak najlepiej odwzorowywać projekt.



## Pamiętamy o ...

- korzystaniu z tagów HTML5 ( **<header>**, **<nav>**, **<section>**, **<footer>** itp.)
- **pracowaniu na repozytorium**, a nie lokalnym projekcie. Po zakończeniu jednej części np. **headera**, robimy commit do swojego repozytorium w **git i push**
- że wykładowca, który prowadzi z nami warsztat, zawsze nam chętnie pomoże i doradzi. Chociaż umiemy korzystać z dobrodziejstw internetu np. ze **Stackoverflow**. Jest tam wiele przydatnych informacji, a umiejętność korzystania z nich na pewno zaowocuje w przyszłości.

**Sit on Chair**  
**Od czego**  
**zacząć?**

# Przygotowanie wstępne

- Upewnij się, że stworzyłeś repozytorium według wytycznych zawartych w pliku „Jak przygotować konto na githubie”
  - Zanim na dobre zasiądziecie do pisania kodu, spróbujcie sobie schematycznie opisać i naszkicować projekt – jest to bardzo pomocne w logicznym rozplanowaniu jego struktury. Warto przy tym zbadać z ilu sekcji głównych składa się strona i jakie cechy wspólne można wyodrębnić. Przykładowo, w omawianym projekcie możemy wyróżnić poszczególne sekcje, które opiszemy przy pomocy tagów HTML5:
- **<header></header>** - nagłówek strony, czarny pasek z logo i menu
  - **<section></section>** - sekcja z fotelem i opisem
  - **<section></section>** - sekcja ze zdjęciami foteli
  - **<section></section>** - sekcja z cennikiem
  - **<section></section>** - sekcja z formularzem
  - **<footer></footer>** - stopka strony, ponieważ ma takie samo tło jak sekcja wyżej, nie odcina się od niej w widoczny sposób, stanowi jednak odrębną część strony

## Opis sekcji

Spróbujmy wypisać najbardziej ogólne cechy każdej sekcji

- **Kolor** – każda sekcja ma swój kolor
- **Szerokość** – wszystkie sekcje są rozciągnięte na całą szerokość strony
- **Paddingi** – każdej sekcji można przypisać marginesy wewnętrzne u góry i u dołu
- **Zawartość sekcji** – zawartość każdej sekcji jest wyśrodkowana względem szerokości całej sekcji i posiada takie same paddingi z lewej i z prawej strony

Podsumowanie:

Pierwsze trzy punkty mówią nam jakie właściwości CSS powinniśmy przypisać każdej sekcji. Ostatni punkt informuje nas, że strona ma ustaloną szerokość i potrzebny jest **div**, który będzie odpowiedzialny za **wyśrodkowanie każdej sekcji**.

## Kontener wyśrodkowujący

W każdej sekcji potrzebny jest div, który przy pomocy przypisanych mu właściwości w CSS, będzie trzymał zawartość strony na środku względem całej szerokości sekcji. Dodaj mu klasę, której nazwa będzie adekwatna do spełnianej funkcji, np. „**wrapper**”, „**container**”, „**main-width**” itp. Właściwości, które powinna zawierać, to:

- **max-width** – pozwoli zdefiniować maksymalną szerokość strony
- **margin: 0 auto** – wyśrodkuje zawartość strony
- **padding-left i padding-right** o takiej samej wartości – zachowa margines wewnętrzny z lewej i prawej strony, aby treść strony nie stykała się z krawędzią okna przeglądarki

# Struktura sekcji

Przypatrz się z jakich elementów głównych składa się każda z wydzielonych przez nas sekcji. Np.:

- **Header** – logo SIT ON CHAIR oraz nawigacja
- **Sekcja z cennikiem** – tytuł sekcji z podkreśleniem oraz trzy takie same elementy z treścią oferty
- **Sekcja z formularzem** – tytuł sekcji z podkreśleniem oraz formularz

itd.

Następnie zwróć uwagę jak **elementy główne** są ustawione względem siebie (czy są obok siebie, pod sobą, po przeciwległych stronach okna)

- Zastanów się jakie właściwości odpowiedzialne za rozmieszczenie elementów na stronie (**display, float**), powinieneś przypisać divom w których umieścisz **elementy główne**.

# Elementy główne

Z całą pewnością nie każdy z elementów głównych potrzebuje swojego osobistego diva w którym będzie umieszczony jak w pudełku, jednak znacznie ułatwia to odpowiednie rozlokowanie elementów na stronie i określenie relacji między nimi. Zwłaszcza na początku pomoże Ci to zrozumieć strukturę projektu nad którym pracujesz.

Spróbuj teraz określić elementy składowe elementów głównych oraz ubrać je w odpowiednie tagi, np.:

- Logo SIT ON CHAIR - składa się tylko z tegoż napisu (np. **<p>** lub **<span>**)
- Nawigacja - składa się tylko z listy nieuporządkowanej ( **<ul>**)
- Tytuł sekcji - składa się z napisu (np. **<p>** lub **<span>**) oraz poziomej kreski ( **before** lub **after** dla napisu lub **<span>**)

itd.

Tak opisany projekt z pewnością będzie łatwiej przełożyć na kod.

## Warto pamiętać

- Jednemu elementowi możesz przypisać wiele klas.
- Nie nadużywaj id!!!
- Elementy klikalne powinny mieć hover lub cursor: pointer.
- Staraj się tworzyć klasy, które będziesz mógł wykorzystać wielokrotnie, dostrzegaj właściwości, które się powtarzają – jeśli każda z twoich sekcji ma taki sam padding górny i dolny, nadaj im wspólną klasę.
- Nazwy klas powinny być angielskie i ich znaczenie powinno się odnosić do funkcji jaką pełnią a nie do stylu jakie są w niej zdefiniowane.
- Trzymaj się zasady „od ogółu do szczegółu” – zacznij od najbardziej ogólnej struktury, pozycjonuj divy – pudełka względem całej strony by potem przejść do układu elementów w samym divie – to znacznie ułatwia pracę.