

FLEXBOX

A. Definition

Flexbox is a one-dimensional CSS layout that can control the way items are spaced out(ditempatkan) and aligned(disejajarkan) within a container(wadah).

[display: flex;]

This will make the element a flex container(wadah fleksibel). Any direct children(turunan lgsg) of a flex container are called flex items(item fleksibel).

display: inline-flex vs flex

>flex: container will behave like block-element

>inline-flex: container will behave like inline-element

B. Flex Properties

Source: <https://www.freecodecamp.org/news/the-css-flexbox-handbook/>

Simple explanation (these images below):

1) Flex Container Properties

[justify-content :]

mengatur posisi sepanjang garis horizontal

- flex-start: elemen berada paling kiri
- flex-end: elemen berada paling kanan
- center: element berada di tengah
- space-evenly: elemen akan memiliki jarak yg sama(baik thd elemen lain maupun thd container)
- space-between: elemen akan memiliki jarak yg sama dengan elemen lain(dan cenderung menempel thd container)
- space-around: 2 elemen memiliki jarak sebelum, diantara, dan sesudah mereka. jarak diantara keduanya lebih panjang ketimbang jarak sebelum&sesudah keduanya

18.12 ✓

[align-items :]

mengatur posisi sepanjang garis vertikal

- flex-start: elemen berada paling atas
- flex-end: elemen berada paling bawah
- center: elemen berada di tengah
- baseline: elemen ditampilkan di paling atas (tanpa terpotong)
- stretch: elemen diregangkan agar tingginya sama dengan tinggi wadah

18.17 ✓

[align-content :]

mengatur jarak antar baris (scr vertikal).

bedanya dg align-items adlh jika align-items mengatur jarak antar element, tapi align-content mengatur jarak antar baris.

- flex-start: baris berkumpul di paling atas
- flex-end: baris berkumpul di paling bawah
- center: baris berkumpul di tengah
- space-between: baris akan memiliki jarak yg sama dg baris lain(dan cenderung menempel thd container)
- space-around: 2 baris memiliki jarak sebelum, diantara, dan sesudah mereka. jarak diantara keduanya lebih panjang ketimbang jarak sebelum&sesudah keduanya
- stretch: baris ditarik utk menyesuaikan jarak wadah??

09.02 ✓

2) Flex Items Properties

[order: 0(default);]

menentukan urutan layout dalam grid/flex container, semakin besar nilainya maka akan semakin di akhir posisinya.

08.45 ✓

[align-self: ;] = align-items

align-self menggantikan align-items pada grid/flex container

08.49 ✓

You use flex-grow when you want to increase the flex item's size if there's extra space. But, you use flex-shrink when you want to decrease the flex-item's size if there's not enough space in the flex container.

18.49 ✓

flex-shrink: menentukan sebrp besar penyusutan yang dialami element terhadap selemen lain.

Note: jika wadah 400px dan ada 4 elemen masing2 150px maka semua elemen akan otomatis menyusut, supaya muat dlm wadah. baru ketika di beri flex-shrink: 2; suatu elemen akan menyusut 2 kali lebih kecil ketimbang elemen lainnya

Note: Pelajari flex grow; shrink; basis dari Scrimba

[flex: 1; = flex: 1 1 0;]

[flex: auto; = flex: 1 1 auto;]

flex-basis: auto; artinya lebar element tersebut sama dengan lebar textnya.

[flex: 1]

item will stretch across the entire container, also grow & shrink following the container.

item akan merenggang ke seluruh kontainer serta melebar & menyusut mengikuti kontainer.

18.31 ✓

[flex-basis: ;]

Menentukan ukuran awal dari elemen flex sebelum ruang dibagi di antara elemen-elemen flex.

[flex-grow: 5 VS flex-grow: 1]

the (flex-grow: 5) item will growing 5 times faster (than flex-grow: 1)

[flex-shrink: 5 VS flex-shrink: 1]

the (flex-shrink: 5) item will shrinking 5 times faster (than flex-shrink: 1)

3) Additional Note

[flex-direction: ;]

Note: untuk kombinasi flex-direction&justify-content, jika kamu mengatur arah baris terbalik, mulai dan akhir juga terbalik.

contoh:

flex-direction: row-reverse;

jus-content: flex-end;

maka flex end nya berada di sebelah kiri karena ketika row-reverse elemen startnya di kanan

Diedit 08.34

↓ 33 kB

mengapa perlu [`.container > .search`] tidak [`.search`] saja ?

karena styling di atasnya (yaitu [`.container > div`]) menggunakan selektor spesifik. sehingga jika hanya menggunakan [`.search`] saja tidak akan work.

Hal ini disebabkan CSS akan memprioritaskan selektor yang lebih spesifik.

18.35 ✓

[`flex-basis: value`]

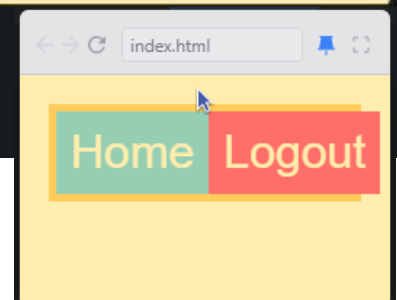
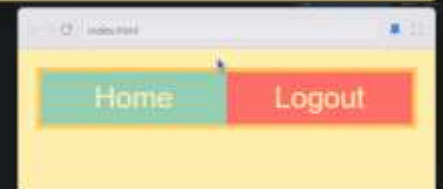
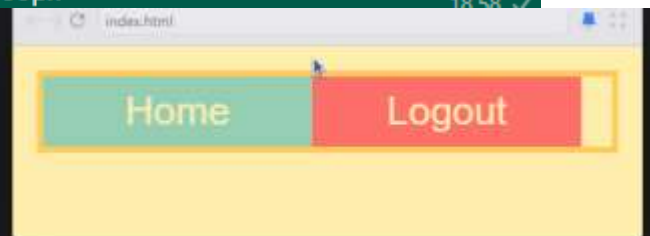
efek: membuat lebar elemen sesuai value dan menyusut ketika mentok kontainer (akan terus menyusut sampai lebar teks dalam item, setelah melewati lebar text maka item akan keluar kontainer)

18.54

jadi flex-basis set base-width of the element, sehingga (dalam gambar) lebar item akan 200px ketika lebar kontainer $\geq 400px$

18.58 ✓

```
.container {  
  border: 5px solid #ffc05e;  
  display: flex;  
}  
  
.home {  
  /*  
  flex-grow: 1;  
  flex-shrink: 1;  
  */  
  flex-basis: 200px;  
}  
  
.logout {  
  /*  
  flex-grow: 1;  
  flex-shrink: 1;  
  */  
  flex-basis: 200px;  
}
```



[`flex: 50%;`] lebih baik dari [`flex-grow: 50%`]

karena `flex: 50%` mengandung `flex-basis auto` yang bisa menhandel beberapa baris elemen dengan `flex: wrap`{scrimba bonus respnsive navbar}

16.41 ✓

C. Assignment (Latihan)

1. Theodinproject > Foundations > Project > Alignment > Assignment (yang terakhir dari github)
2. Scrimba > Flexbox > Bonus
3. www.developer.mozilla.org/en-US/docs/Learn/CSS/CSS_layout/Flexbox_skills

