
CSS + HTML

Agenda

- HTML & CSS coming together
- CSS Display, Flexbox model
- Exercises within class
- Quiz
- Assignment

What Is HTML & CSS?

(we already know – little recap)



CSS Display

```
/* legacy values */  
display: block;  
display: inline;  
display: inline-block;  
display: flex;  
display: inline-flex;  
display: grid;  
display: inline-grid;  
display: flow-root;
```

```
/* box generation */  
display: none;  
display: contents;
```

```
/* other values */  
display: table;  
display: table-row;  
/* all table elements have an equivalent CSS display value */  
display: list-item;
```

```
/* Global values */  
display: inherit;  
display: initial;  
display: revert;  
display: unset;
```

CSS Display - Inline

`display: inline;`

The default value for elements. Think of elements like ``, ``, or `` and how wrapping text in those elements within a string of text doesn't break the flow of the text.

Pellentesque `inline element` morbi tristique senectu
Donec eu libero sit amet quam egestas semper. Ac

CSS Display - Inline

An inline element will accept margin and padding, but the element still sits inline as you might expect. Margin and padding will only push other elements horizontally away, not vertically.

Pellentesque *inline element* morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum tortor quam, feugiat vitae, ultricies eget, tempor sit amet, ante. Donec eu libero sit amet quam egestas semper. Aenean ultricies mi vitae est. Mauris placerat eleifend leo.

An inline element will not accept height and width. It will just ignore it.

CSS Display – Inline-block

display: inline-block;

The default value for An element set to inline-block is very similar to inline in that it will set inline with the natural flow of text (on the “baseline”).

Pellentesque



*inline
block*



*inline
block*

morbi tristique

senectus et netus et malesuada fames ac turpis egestas. Vestibulum tortor quam, feugiat vitae, ultricies eget, tempor sit amet, ante. Donec eu libero sit amet quam egestas semper. Aenean ultricies mi vitae est. Mauris placerat eleifend leo.

The difference is that you are able to set a width and height which will be respected.

CSS Display – Block

`display: block;`
A number of elements are set to block by the browser UA stylesheet. They are usually container elements, like `<div>`, `<section>`, and ``. Also text “blocks” like `<p>` and `<h1>`. Block level elements do not sit inline but break past them. By default (without setting a width) they take up as much horizontal space as they can.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum tortor quam, feugiat vitae, ultricies eget, tempor sit amet, ante. Donec eu libero sit amet quam egestas semper. Aenean ultricies mi vitae est. Mauris placerat eleifend leo.

hi

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum tortor quam, feugiat vitae, ultricies eget, tempor sit amet, ante. Donec eu libero sit amet quam egestas semper. Aenean ultricies mi vitae est. Mauris placerat eleifend leo.

CSS Before Flexbox – Float

- float: none;
- float: left;
- float: right;

CSS Before Flexbox – Float

Float
me

As much mud in the streets as if the waters had but newly retired from the face of the earth, and it would not be wonderful to meet a Megalosaurus, forty feet long or so, waddling like an elephantine lizard up Holborn Hill.

Float
me

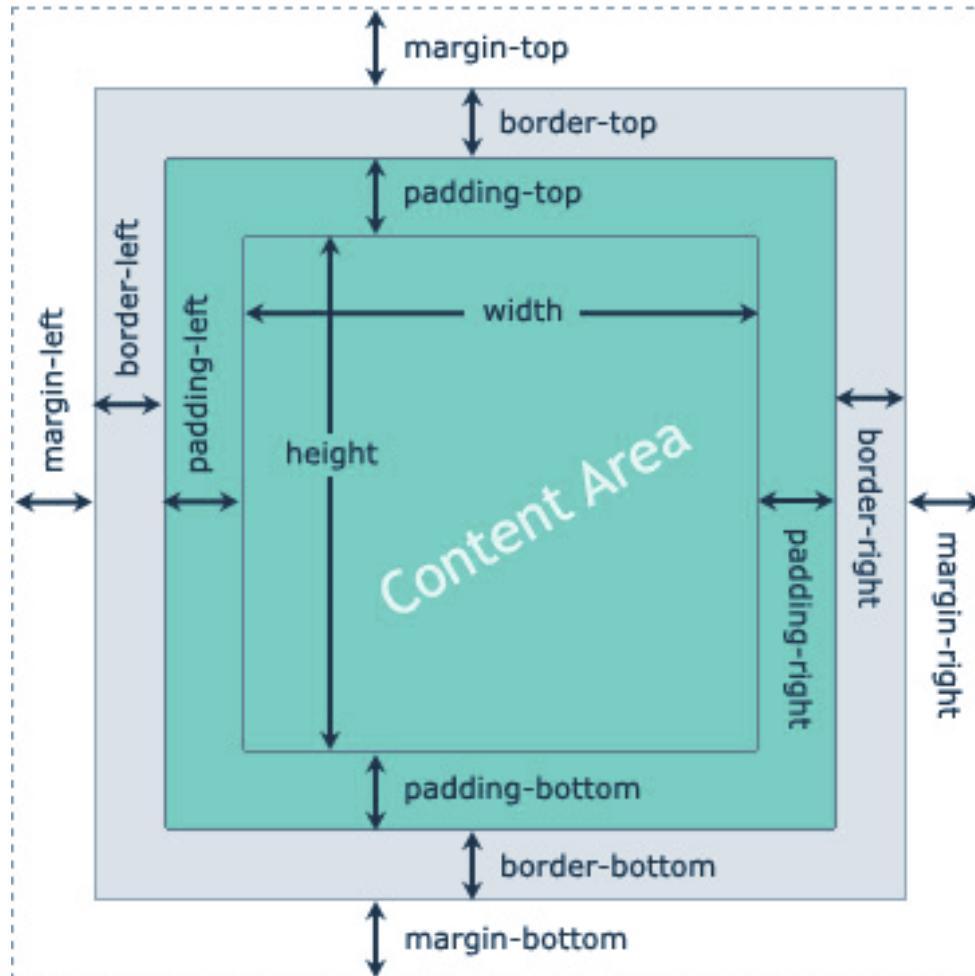
As much mud in the streets as if the waters had but newly retired from the face of the earth, and it would not be wonderful to meet a Megalosaurus, forty feet long or so, waddling like an elephantine lizard up Holborn Hill.

Float
me

As much mud in the streets as if the waters had but newly retired from the face of the earth, and it would not be wonderful to meet a Megalosaurus, forty feet long or so, waddling like an elephantine lizard up Holborn Hill.

CSS Structure – Box Model

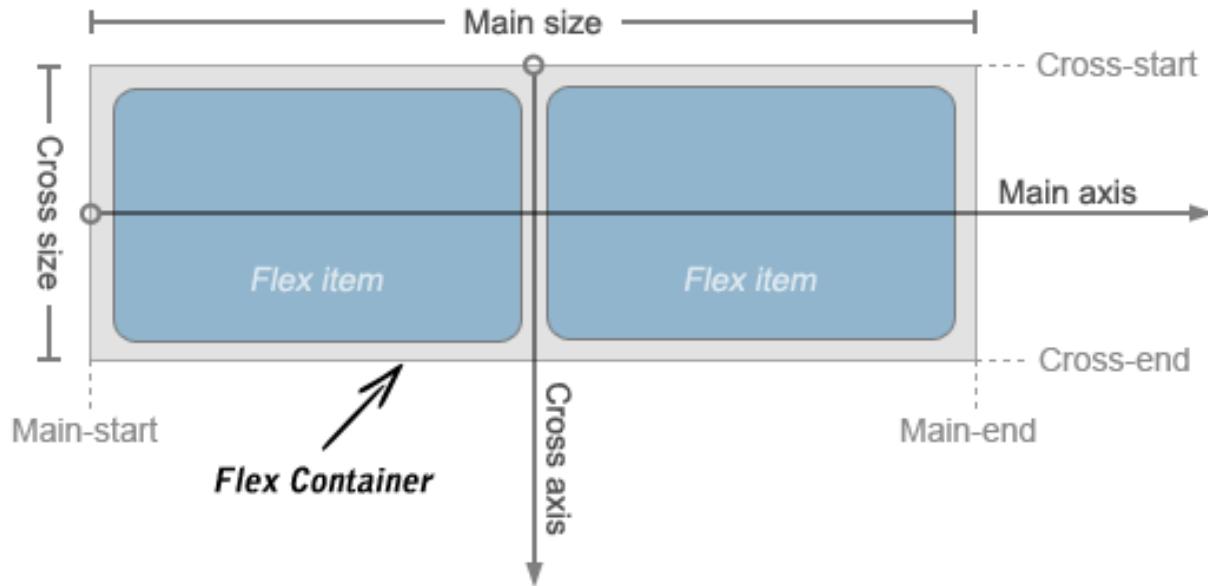
Margin



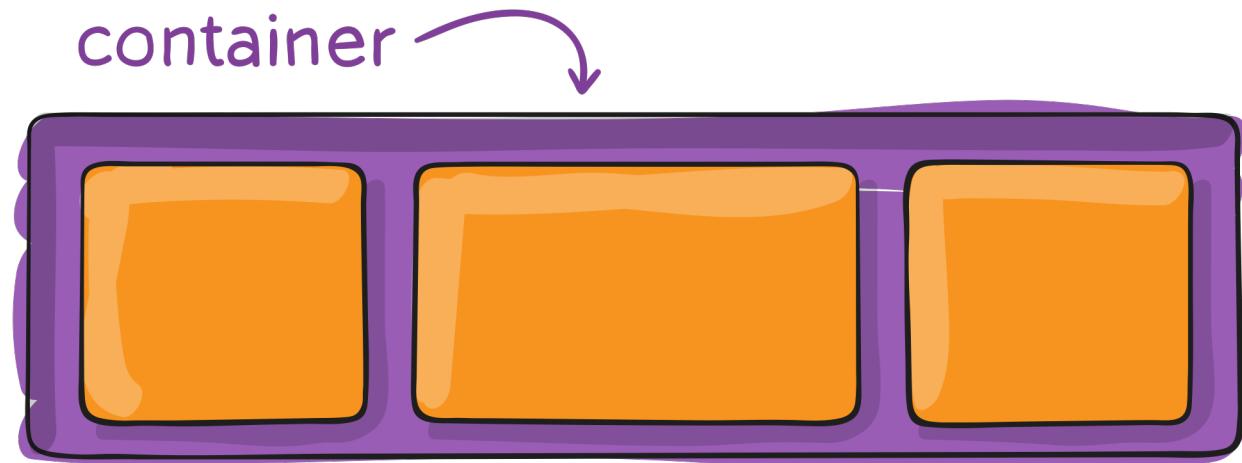
CSS Flexbox

- Flex Containers
- Flex items
- Building something out of them

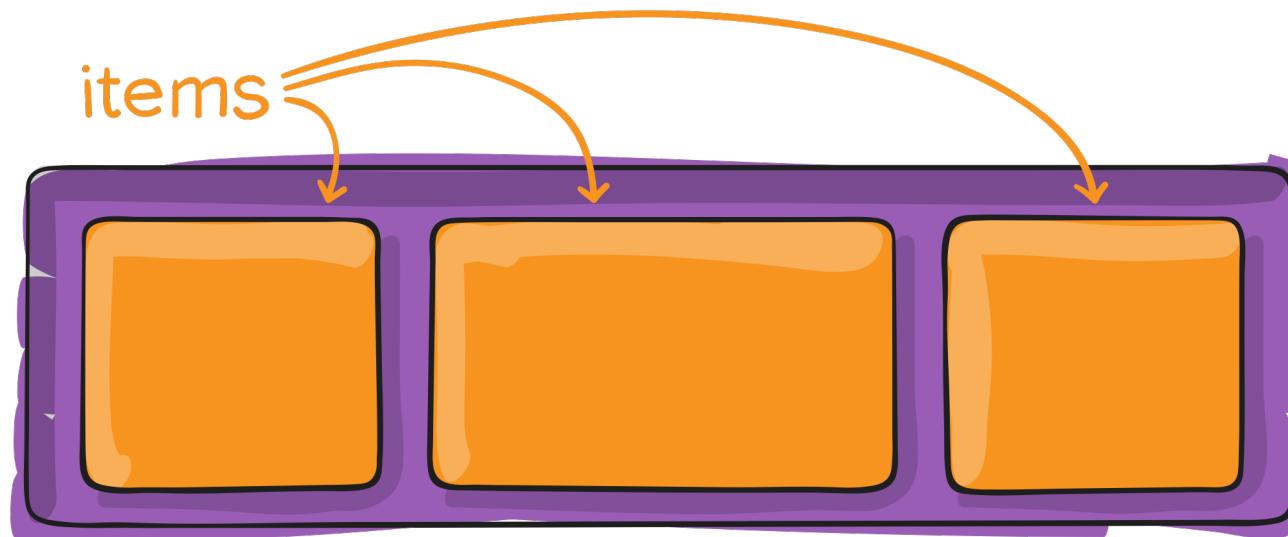
CSS Flexbox



CSS Flexbox - Containers



CSS Flexbox – Items



CSS Flexbox – Containers (Properties)

flex-direction

This establishes the main-axis, thus defining the direction flex items are placed in the flex container. Flexbox is (aside from optional wrapping) a single-direction layout concept. Think of flex items as primarily laying out either in horizontal rows or vertical columns.

`row` (default): left to right in ltr; right to left in rtl

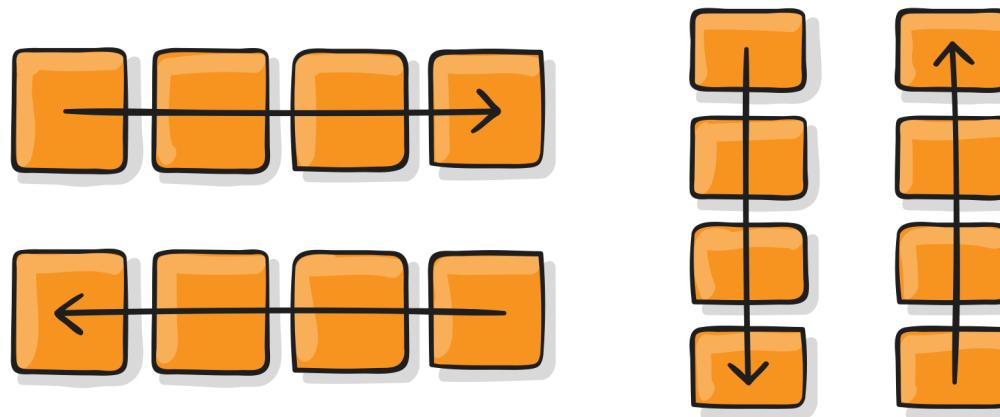
`row-reverse`: right to left in ltr; left to right in rtl

`column`: same as row but top to bottom

`column-reverse`: same as row-reverse but bottom to top

CSS Flexbox – Containers (Properties)

`flex-direction`



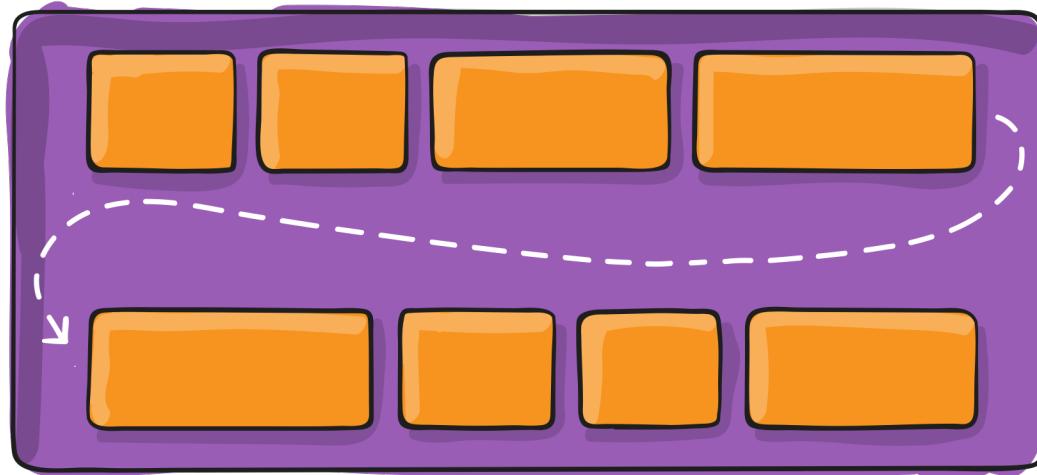
CSS Flexbox – Containers (Properties)

flex-wrap

nowrap (default): all flex items will be on one line

wrap: flex items will wrap onto multiple lines, from top to bottom.

wrap-reverse: flex items will wrap onto multiple lines from bottom to top.



CSS Flexbox – Containers (Properties)

flex-flow

This is a shorthand for the flex-direction and flex-wrap properties, which together define the flex container's main and cross axes. The default value is **row nowrap**.

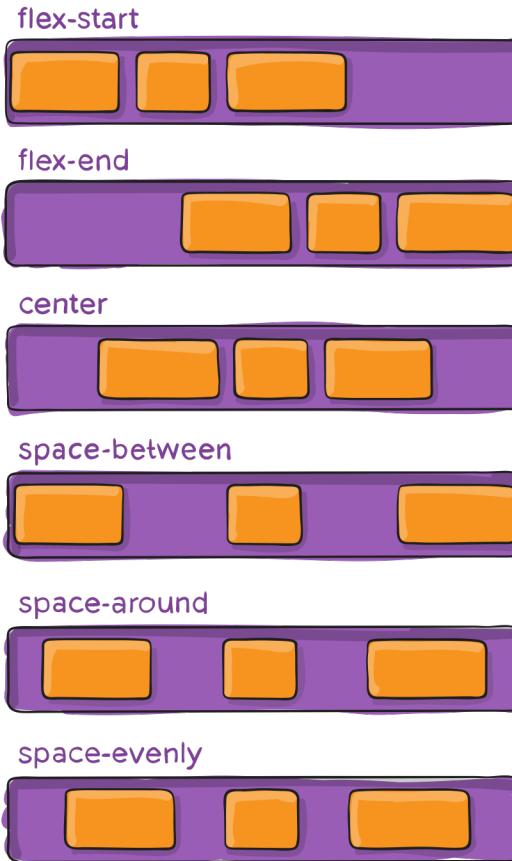
CSS Flexbox – Containers (Properties)

Justify-content

This defines the alignment along the main axis. It helps distribute extra free space leftover when either all the flex items on a line are inflexible, or are flexible but have reached their maximum size. It also exerts some control over the alignment of items when they overflow the line.

CSS Flexbox

Justify-content





CSS Flexbox

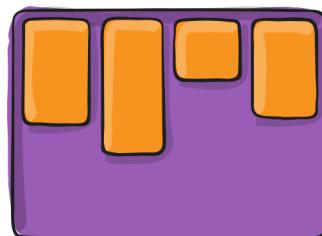
align-items

This defines the default behavior for how flex items are laid out along the **cross axis** on the current line. Think of it as the justify-content version for the cross-axis (perpendicular to the main-axis).

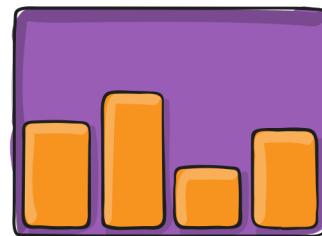
CSS Flexbox

align-items

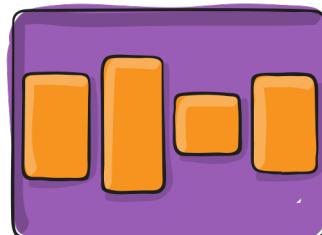
flex-start



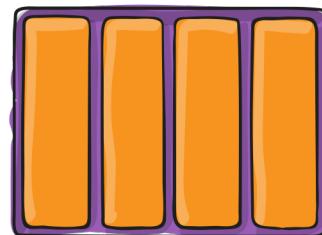
flex-end



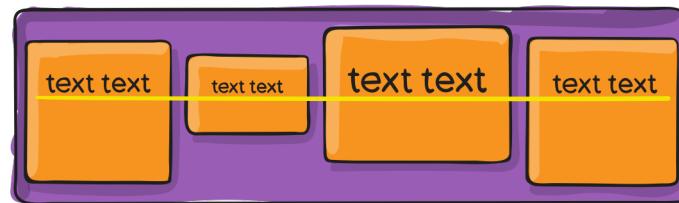
center



stretch



baseline



CSS Flexbox – Items (Properties)

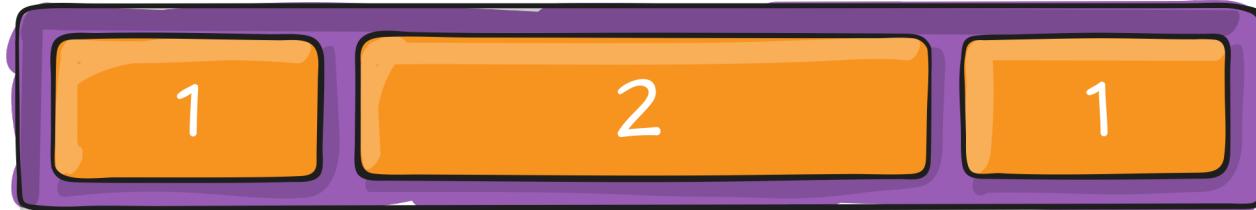
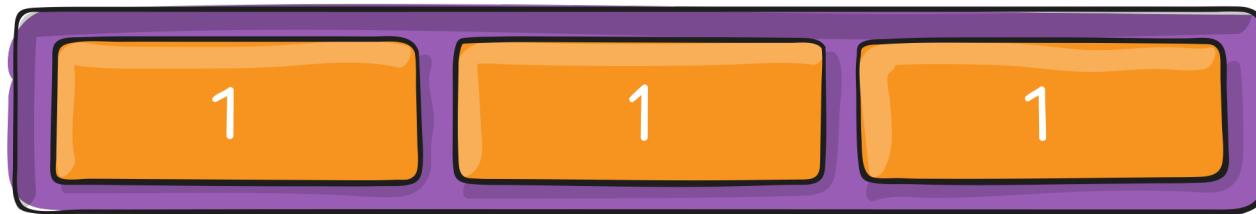
Flex-grow

This defines the ability for a flex item to grow if necessary. It accepts a unitless value that serves as a proportion. It dictates what amount of the available space inside the flex container the item should take up.

If all items have flex-grow set to 1, the remaining space in the container will be distributed equally to all children. If one of the children has a value of 2, that child would take up twice as much of the space either one of the others (or it will try, at least).

CSS Flexbox – Items (Properties)

Flex-grow



CSS Flexbox – Items (Properties)

Flex-basis

This defines the default size of an element before the remaining space is distributed. It can be a length (e.g. 20%, 5rem, etc.) or a keyword.

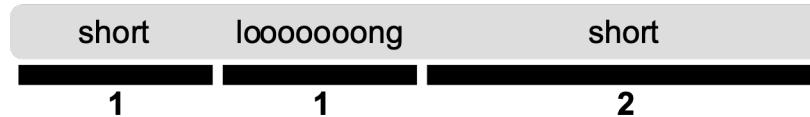
The auto keyword means “look at my width or height property” (which was temporarily done by the main-size keyword until deprecated).

The content keyword means “size it based on the item’s content” - this keyword isn’t well supported yet, so it’s hard to test and harder to know what its brethren max-content, min-content, and fit-content do.

CSS Flexbox – Items (Properties)

Flex-basis

All Space Distributed
(`flex-basis:0`)

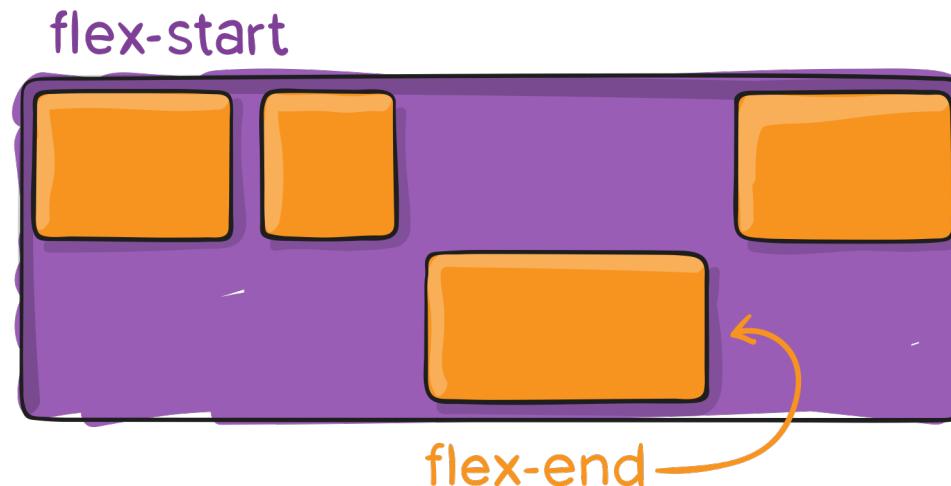


Extra Space Distributed
(`flex-basis:auto`)



CSS Flexbox – Items (Properties)

align-self



CSS Flexbox – Items (Properties)

Note that float, clear and vertical-align have no effect on a flex item.

Exercises

Exercises

- <https://froggyfasttrack-ddvscrgkr-alexghi.vercel.app/>
- <https://kaitlynbro.github.io/flexbox-quiz/>
- https://developer.mozilla.org/en-US/docs/Learn/CSS/CSS_layout/Flexbox_skills

-CSS Flexbox – References Resources

- https://www.w3schools.com/css/css3_flexbox.asp
- https://developer.mozilla.org/en-US/docs/Learn/CSS/CSS_layout/Flexbox
- <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>
- <https://www.tutorialrepublic.com/css-tutorial/css3-flexible-box-layouts.php>
- https://www.youtube.com/watch?v=u044iM9xsWU&ab_channel=KevinPowell
- [https://cssbattle.dev/ - CSS Battles & Challenges](https://cssbattle.dev/)

Assignment



Assignment

- Using the knowledge gained so far build the following mockup (HTML & CSS)
- You have the assets and a small example in index.html on how to include material google fonts and how to use them. You can also check the documentation for further reference (regarding fonts - https://developers.google.com/fonts/docs/material_icons)
- For contact page on how to style the inputs: https://developer.mozilla.org/en-US/docs/Learn/Forms/Styling_web_forms (!)
<https://blog.logrocket.com/how-to-style-forms-with-css-a-beginners-guide/>
<https://medium.com/codex/how-to-style-an-input-field-with-css-only-tips-and-techniques-e6a00e9dcc50>
- On portfolio you should use `display: flex` whenever possible (it is always possible)
- All the resources are in the archive with a little starter



Hello, I am Robert. A Product Designer.

Lorem ipsum dolor sit amet, consectetur adipiscing elit
dapibus porta lorem morbi hendrerit.

[Contact me ↗](#)

About Me

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Neque tempor, malesuada adipiscing congue diam. Quis orci amet porttitor blandit amet nullam sit. Elit, purus blandit non ut non quam curabitur.

Quis orci amet porttitor blandit amet nullam sit. Elit, purus blandit non ut non quam curabitur. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Neque tempor, malesuada adipiscing congue diam. Lorem ipsum dolor sit amet, consectetur.

[More about me ↗](#)

Past experience

VP of Design
Google, LLC.
Jan 2022 — Present

Senior Product Designer
Meta, Inc.
2018 — 2022

UX Researcher
Twitter, Inc.
2015 — 2018

Portfolio

Lorem ipsum dolor sit amet, consectetur adipiscing.

[All projects ↗](#)

Techplus X

UI Product Explorations

Lorem ipsum dolor sit amet, consectetur
asul molar adipiscing elit eu gravida etiam.

[See project ↗](#)

Techplus X

Presenting the next-gen CRM platform

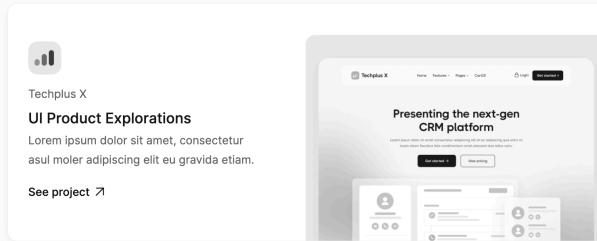
Present the next-generation CRM platform to your clients with our new design.

[View project ↗](#)

Portfolio

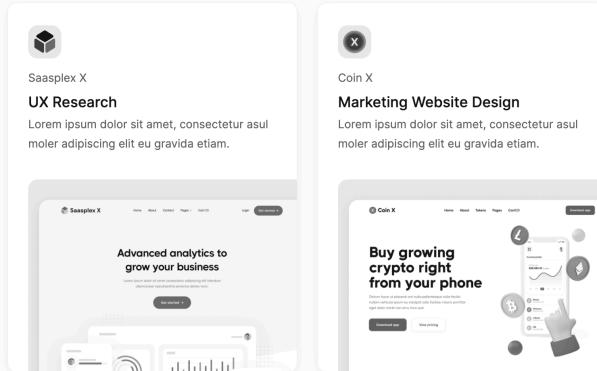
Lorem ipsum dolor sit amet, consectetur adipisci elit
dipibus porta lorem morbi hendrerit.

Contact me ↗

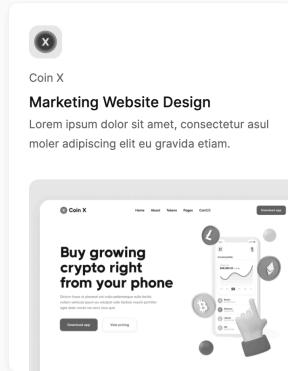


Techplus X
UI Product Explorations
Lorem ipsum dolor sit amet, consectetur
asul moler adipiscing elit eu gravida etiam.
[See project ↗](#)

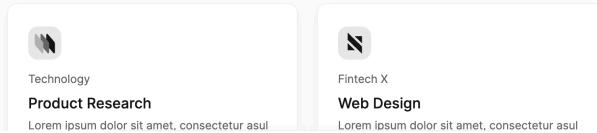




Saasplex X
UX Research
Lorem ipsum dolor sit amet, consectetur asul
moler adipiscing elit eu gravida etiam.

Coin X
Marketing Website Design
Lorem ipsum dolor sit amet, consectetur asul
moler adipiscing elit eu gravida etiam.

Technology
Product Research
Lorem ipsum dolor sit amet, consectetur asul



Fintech X
Web Design
Lorem ipsum dolor sit amet, consectetur asul

Contact

Lorem ipsum dolor sit amet consectetur adipiscing elit tempor
pharetra eget consequat se.

Name	Email
What's your name?	What's your email?
Phone	Subject
What's your phone?	How can I help?
Leave us a message	
Hi there! I would like to get in touch with you to inquiry about a project.	
<hr/> Send message ↗	

Follow me on social media



Some guiding lines



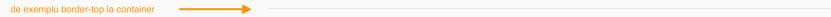
border-radius

Hello, I am Robert.
A Product Designer.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Neque tempor, malesuada adipiscing congue diam. Quis orci amet porttitor blandit amet nullam sit. Elit, purus blandit non ut non quam curabitur.

[Contact me ↗](#)

de exemplu border-top la container



About Me

Lore ipsum dolor sit amet, consectetur adipiscing elit. Neque tempor, malesuada adipiscing congue diam. Quis orci amet porttitor blandit amet nullam sit. Elit, purus blandit non ut non quam curabitur.

Quis orci amet porttitor blandit amet nullam sit. Elit, purus blandit non ut non quam curabitur. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Neque tempor, malesuada adipiscing congue diam. Lorem ipsum dolor sit amet, consectetur.

[More about me ↗](#)

Past experience

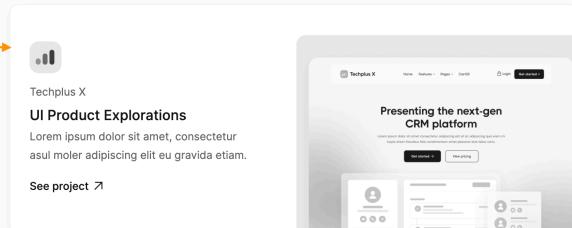
VP of Design
Google, LLC.
Jan 2022 — Present

Senior Product Designer
Meta, Inc.
2018 — 2022

UX Researcher
Twitter, Inc.
2015 — 2018

Portfolio

border cu border-radius



Techplus X
UI Product Explorations
Lore ipsum dolor sit amet, consectetur adipiscing.

[See project ↗](#)



All projects ↗

Some guiding lines

Contact

Lorem ipsum dolor sit amet consectetur adipiscing elit tempor
pharetra eget consequat se.

CSS regarding inputs and labels, placeholders

The contact form consists of several input fields and labels:

- Name**: What's your name?
- Email**: What's your email?
- Phone**: What's your phone?
- Subject**: How can I help?

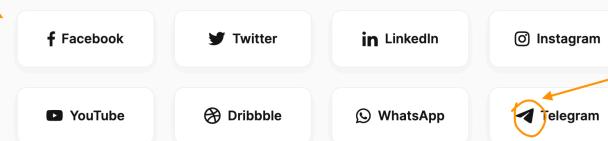
Leave us a message

Hi there! I would like to get in touch with you to inquiry about a project.

Send message ↗

display: flex
flex-wrap: wrap
sau display flex simplu in doua containere diferite

Follow me on social media



Pentru orice icoane puteti folosi font-ul de google,
cu orice icoana vreti din lista disponibila

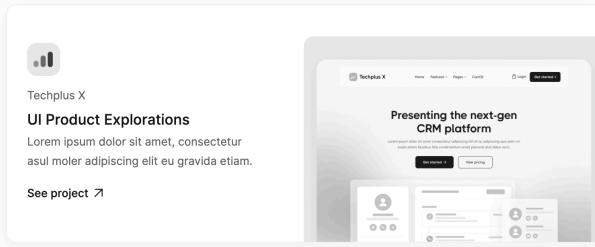


Portfolio

Lorem ipsum dolor sit amet, consectetur adipisci elit
dapibus porta lorem morbi hendrerit.

Contact me ↗

Incercati sa vizualizati intai, poate chiar cu o foaie
si un creion in mana cum as face impartirea containerelor
ex:

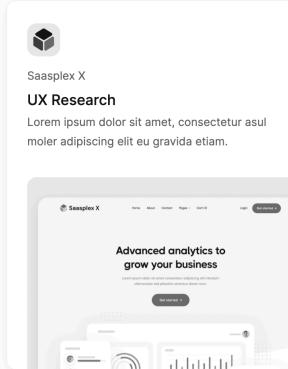


Techplus X

UI Product Explorations

Lorem ipsum dolor sit amet, consectetur
asul moler adipiscing elit eu gravida etiam.

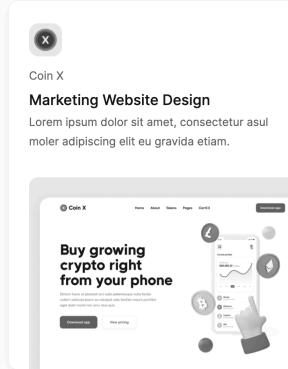
[See project ↗](#)



Saasplex X

UX Research

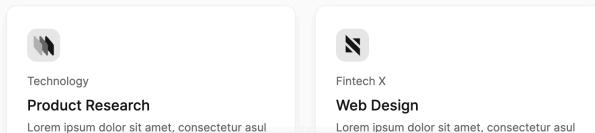
Lorem ipsum dolor sit amet, consectetur asul
moler adipiscing elit eu gravida etiam.



Coin X

Marketing Website Design

Lorem ipsum dolor sit amet, consectetur asul
moler adipiscing elit eu gravida etiam.



Technology

Product Research

Lorem ipsum dolor sit amet, consectetur asul

