



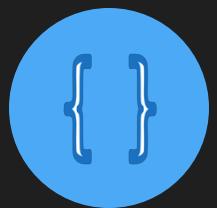
FUTURE FULLSTACK

# SUMMARY CARDS



CSS

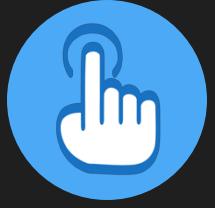
# CSS



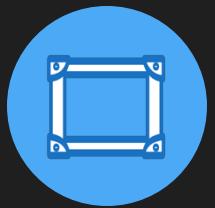
INTRODUCTION  
TO CSS



CSS TEXT



CSS SELECTORS



CSS BOX MODEL



CSS UNITS



CSS FUNCTIONS



FLEXBOX



CSS GRID



RESPONSIVE  
DESIGN



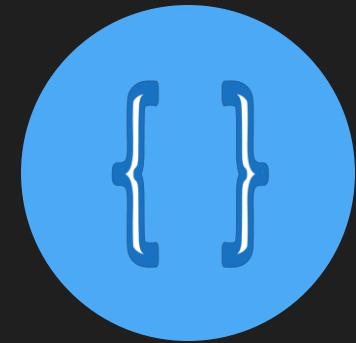
POSITIONING



SHADOWS &  
TRANSITIONS



CSS  
FRAMEWORKS



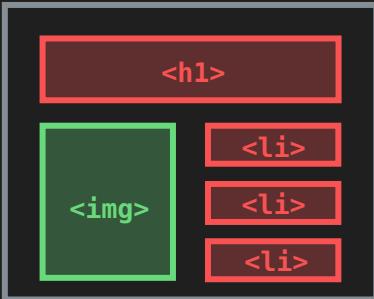
# INTRODUCTION TO CSS

## SUMMARY CARD

## INTRODUCTION TO CSS

## CSS PROPERTIES

INSIDE THE BOX  
WHERE THE BOX GOES



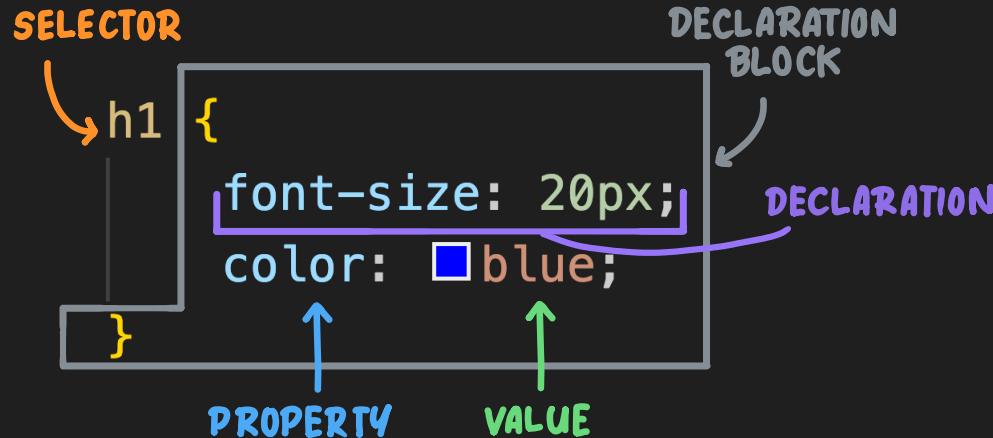
## INCLUDING CSS

## INLINE CSS ✗

ATTRIBUTE

`<h1 style="color: blue;">Welcome</h1>`

## CSS RULES



## INTERNAL CSS ✗

```
<head>
  <title>Document</title>
  <style>
    h1 {
      color: blue;
    }
  </style>
</head>
```

## EXTERNAL CSS ✓

index.html

```
<head>
  <link rel="stylesheet" href="app.css">
</head>
```

app.css

```
h1 {
  color: blue;
}
```



CSS TEXT



## SUMMARY CARD

## TEXT STYLING

## FONT WEIGHT

`font-weight: 700;`

100 200 300 **400** 500 600 **700** 800 900

HEADINGS 500-900 & OTHER TEXT 300-400

## FONT STYLE

`font-style: italic;`

**normal** italic oblique oblique 10deg

AVOID USING

## LIST STYLE

`font-style: disc;`

none **disc** circle square **decimal**

REMOVE LIST STYLING

## TEXT DECORATION

`text-decoration: underline blue dotted;`

**none** line-through underline overline  
named rgb hex  
double solid dotted dashed wavy

REMOVE UNDERLINE FROM ANCHOR TAGS

## TEXT TRANSFORM

`text-transform: uppercase;`

**none** uppercase lowercase capitalize

USE SENTENCE CASE IN HTML



## SUMMARY CARD

# TEXT SPACING AND SIZE

## TEXT ALIGN

```
font-align: center;
```

**left** right center justify

CAN USE CENTER BUT LEFT-ALIGN FOR BLOCKS OF TEXT

## FONT SIZE

```
font-size: 16px;
```

ABSOLUTE → pixels    rem ← RELATIVE

REGULAR TEXT SHOULD BE 16PX - 32PX  
AND HEADINGS CAN BE > 60PX

## LINE HEIGHT

```
line-height: 1.5;
```

**unitless** pixels percentages ems

HEADINGS <1.5 AND REGULAR TEXT 1.5-2

## TYPE SCALE

62px h1 WELCOME TO QUILL

48px h2 WELCOME TO QUILL

40px h3 WELCOME TO QUILL

32px h4 WELCOME TO QUILL

24px h5 WELCOME TO QUILL

x1.25 ↗ 20px p WELCOME TO QUILL  
16px a WELCOME TO QUILL

## LETTER SPACING

```
line-height: 8px;
```

**pixels** percentages ems

SMALL NEGATIVE VALUE APPLIED TO HEADINGS

## SUMMARY CARD

# FONT FAMILY AND COLOR

## TYPEFACE

Text    Text    Text    *Text*    Text  
SERIF   SANS-SERIF   MONOSPACE   CURSIVE   DISPLAY

## FONT FAMILY

font-family: 'Tahoma', sans-serif;

USE A POPULAR SANS-SERIF FONT

## COLOR

color: #9874F9;

rgb

hex

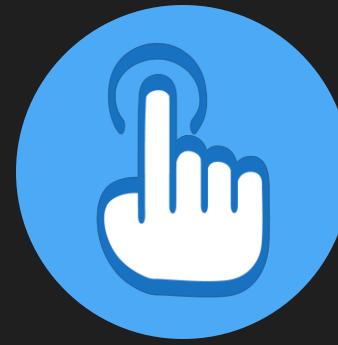
rgba

hsl

## COLOR PALETTE

PRIMARY	SECONDARY	TERTIARY	GREY
Arctic #d0ebff	Moonlight Iris #e5dbff	Dawn #ffe8cc	Pebble #f1f3f5
Sky Blue #339af0	Lavender #9775fa	Pumpkin #ff922b	Slate #495057
Deep Sea #1c7ed6	Velvet Night #7048e8	Lava #f76707	Graphite #212529

HAVE AT LEAST 2 COLORS, A PRIMARY AND A GREY WITH TINTS AND SHADES OF EACH



# CSS SELECTORS

## SUMMARY CARD

# BASIC SELECTORS & THE CASCADE

## TYPE SELECTOR

```
h1 {  
    font-size: 20px;  
    color: blue  
}
```

TYPE  
SELECTOR

USED FOR SETTING GLOBAL STYLES

## GROUPING SELECTORS

```
h1, h2 {  
    color: #495057;  
}  
  
h1 {  
    font-size: 62px;  
}
```

SELECTORS  
GROUPED WHEN  
SHARED STYLES

USE CASCADE TO ADD  
NEW STYLES TO  
PREVIOUS ONES

## THE CASCADE

```
h1 {  
    color: blue;  
}  
  
h1 {  
    color: green;  
}
```

```
<head>  
    <link rel="stylesheet" href="app.css">  
    <link rel="stylesheet" href="app2.css">  
</head>
```

app.css

app2.css

CONFICTING STYLES DECLARED  
LATER TAKE PRIORITY

CONFICTING STYLES IN LATER  
STYLESHEETS TAKE PRIORITY

## ID AND CLASS SELECTORS

```
<body>  
    <p id="red-text">Copper mug</p>  
    <p class="blue-text">Biodiesel tattooed</p>  
    <p class="blue-text">Coloring book</p>  
</body>
```

MORE  
COMMONLY  
USED

HASH

ID SELECTOR

```
#red-text {  
    color: red;  
}
```

FULL  
STOP

CLASS SELECTOR

```
.blue-text {  
    color: blue;  
}
```

CAN ONLY BE USED ONCE

CAN BE USED MULTIPLE TIMES

## SUMMARY CARD

# PSEUDO-CLASSES & SPECIFICITY

## PSEUDO-CLASSES

**IMPORTANT**  
→ STATE

STYLING BASED ON  
USER INTERACTION

```
.btn:link, .btn:visited {  
    color: blue;  
}  
  
.btn:hover, .btn:active {  
    color: darkblue  
}
```

SELECTOR →  
a : hover → PSEUDO-CLASS

CONDITION  
LESS IMPORTANT

STYLING BASED  
ON AN ELEMENTS  
RELATIVE POSITION

```
li:nth-child(n + 1) {  
    color: green;  
}
```

## COMBINATOR SELECTORS

DESCENDANT

DIRECT  
DESCENDANT

ADJACENT  
SIBLING

ADJACENT  
SIBLING

div p

div > p

h1 + p

h1 ~ p

## SPECIFICITY

DETERMINES PRIORITY OF A CSS RULES WHEN MULTIPLE  
CONFLICTING RULES ARE APPLIED TO THE SAME ELEMENT

```
<div>  
    <div class="textbox">  
        <h1 id="heading" class="large">Organic Green Juice</h1>  
    </div>  
</div>
```

```
#heading {  
    color: red;  
}  
  
.textbox .large {  
    color: yellow;  
}  
  
div div .large {  
    color: green;  
}  
  
div div h1 {  
    color: blue;  
}
```

HIGHEST SPECIFICITY → LOWER PRIORITY

Selector	ID	Class Pseudo-class	Type
#heading	1	0	0
.textbox.large	0	2	0
div div .large	0	1	2
div div h1	0	2	3

USE TYPE SELECTOR FOR GLOBAL STYLES & CLASS  
SELECTOR TO OVERRIDE FOR SPECIFIC STYLING



## SUMMARY CARD

# INHERITANCE & THE UNIVERSAL SELECTOR

## TEXT PROPERTIES

## INHERITANCE

PROPERTIES SET ON PARENT ELEMENTS ARE PASSED TO CHILD ELEMENTS BY DEFAULT

```
<body>
  <section>
    <h1>Asymmetrical beard</h1>
    <p>etro distillery banjo</p>
  </section>
  <footer>
    <p>Fashion cold-pressed</p>
  </footer>
</body>
```

TEXT PROPERTIES INHERITED

```
body {
  color: black;
  font-weight: 400;
  font-family: Arial, sans-serif;
}
```

GLOBAL FONT STYLES SET ON THE BODY ELEMENT

## THE UNIVERSAL SELECTOR

APPLIES STYLES TO ALL ELEMENTS

```
* {
  color: blue;
  border-top: 5px solid red;
}
```

USED FOR RESETTING DEFAULT STYLES

## INHERITANCE & TEXT-ALIGN

WHEN APPLYING TEXT-ALIGN TO A BLOCK-LEVEL PARENT BOTH BLOCK AND INLINE ELEMENTS ARE IMPACTED

```
text-align: center;
<div>
  <h1>Street art</h1>
  <p>Bespoke sustainable</p>
  <a href="#">meditation</a>
</div>
```

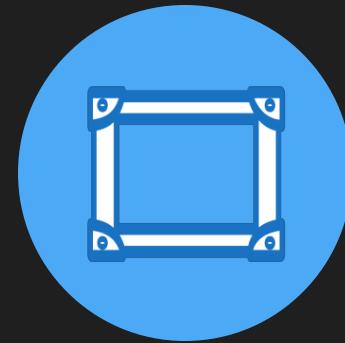
INLINE ELEMENTS HAVE SPACE TO MOVE INSIDE THE BLOCK CONTAINER

BLOCK ELEMENTS INHERIT TEXT-ALIGN PROPERTY

## PSEUDO-ELEMENTS

STYLES A SPECIFIC PART OF AN ELEMENT  
SELECTOR → PSEUDO-ELEMENT

```
h2 :: first-letter
```



# CSS BOX MODEL

## SUMMARY CARD

# THE CSS BOX MODEL

## PADDING

### LONGHAND

```
padding-top: 5px;  
padding-right: 10px;  
padding-bottom: 5px;  
padding-left: 10px;
```

### SHORTHAND

```
padding: 20px;  
padding: 10px 20px;  
padding: 10px 20px 5px;  
padding: 10px 20px 5px 15px;
```

## CONTENT

```
width: 500px; height: 100px;
```

```
background-color: #339af0;
```

APPLIES TO CONTENT & PADDING

## PADDING

## BORDER RADIUS

HEIGHT

WIDTH

## BORDER

## MARGIN

## BORDER

## SHORTHAND

```
border: 5px solid red;
```

## LONGHAND

```
border-width: 5px;  
border-style: solid;  
border-color: red;
```

## INDIVIDUAL

```
border-top: 6px solid purple;  
border-right: 2px dashed blue;  
border-bottom: 1px dotted red;  
border-left: 10px solid green;
```

## MARGIN

### LONGHAND

```
margin-top: 5px;  
margin-right: 10px;  
margin-bottom: 5px;  
margin-left: 10px;
```

### SHORTHAND

```
margin: 20px;  
margin: 10px 20px;  
margin: 10px 20px 5px;  
margin: 10px 20px 5px 15px;
```

## BORDER RADIUS

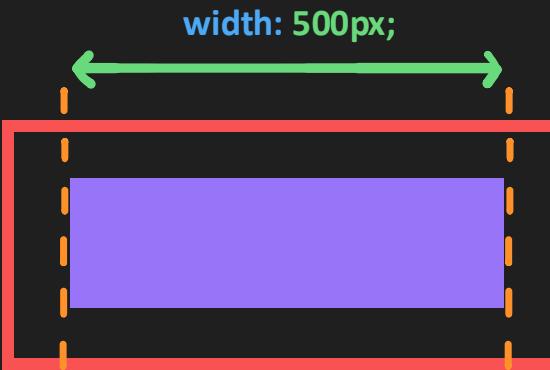
```
border-radius: 12px;
```

HALF ELEMENT'S HEIGHT GIVES  
FULLY ROUNDED CORNERS

## SUMMARY CARD

# SETTING GLOBAL RULES WITH THE CSS BOX MODEL

## BOX SIZING



### CONTENT BOX

box-sizing: content-box;

SET WIDTH IS WIDTH OF CONTENT ONLY

### BORDER BOX

box-sizing: border-box;

SET WIDTH WILL BE SUM OF CONTENT, PADDING & BORDER

## CSS RESET

```
* {  
    box-sizing: border-box;  
    margin: 0;  
    padding: 0;  
}
```

MAKES SETTING DIMENSIONS MORE PREDICTABLE

REMOTES ALL DEFAULT MARGIN & PADDING

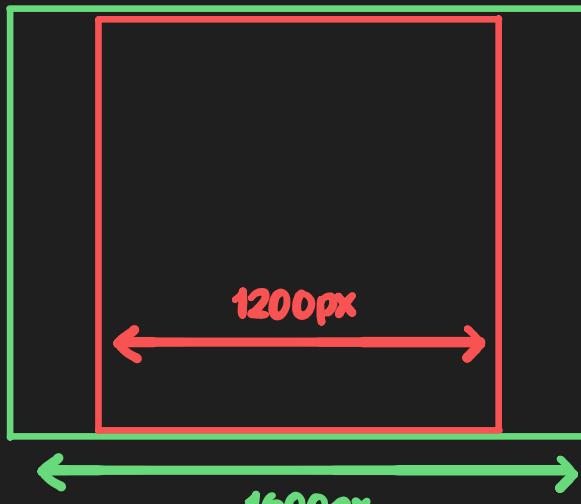
## DISPLAY

	BLOCK	INLINE	INLINE-BLOCK
STARTS ON A NEW LINE	✓	✗	✗
TAKES UP FULL WIDTH	✓	✗	✗
WIDTH & HEIGHT	✓	✗	✓
PADDING APPLIES	✓	✗	✓
BORDER APPLIES	✓	✗	✓
MARGIN APPLIES	✓	✗	✓

## SUMMARY CARD

# RESPONSIVE DESIGN WITH THE CSS BOX MODEL

## MAIN CONTAINER



```
.container {  
  width: 1200px;  
  margin-left: auto;  
  margin-right: auto;  
}
```

## MAX & MIN WIDTH

WIDTH	MAX WIDTH	MIN WIDTH
width: 500px;	max-width: 500px;	
ELEMENT HAS FIXED WIDTH	ELEMENT WON'T EXCEED WIDTH BUT CAN REDUCE	ELEMENT WON'T REDUCE WIDTH BUT CAN EXCEED
SMALL IMAGES & ICONS	CONTAINERS & LARGER IMAGES	GRIDS

## MAX & MIN HEIGHT

HEIGHT	MAX HEIGHT	MIN HEIGHT
height: 500px;	max-height: 500px;	
ELEMENT HAS FIXED HEIGHT	ELEMENT WON'T EXCEED HEIGHT BUT CAN REDUCE	ELEMENT WON'T REDUCE HEIGHT BUT CAN EXCEED
NAVBAR	MODALS	SECTION CONTAINERS



# CSS UNITS

SUMMARY CARD

# CSS UNITS

## MORE COMMONLY USED

- PIXELS
- BORDER RADIUS
- BORDER
- LETTER SPACING
- LOGO AND ICONS
- MAX-WIDTH
- SHADOWS

- PERCENTAGES
- MAIN CONTAINERS
- IMAGES
- BORDER RADIUS

- REMS
- TEXT
- MARGIN
- PADDING

## USED FOR SPECIFIC CASES

- EMS
- MARGIN
- PADDING

- VH
- SECTION HEIGHT

- VW
- TEXT

REMS ARE FINE

DESIGN DEPENDENT

NOT COMMON



# CSS FUNCTIONS

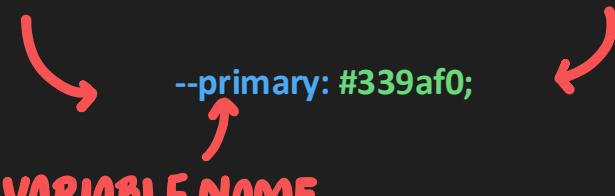
## SUMMARY CARD

# CSS FUNCTIONS

## CSS VARIABLES

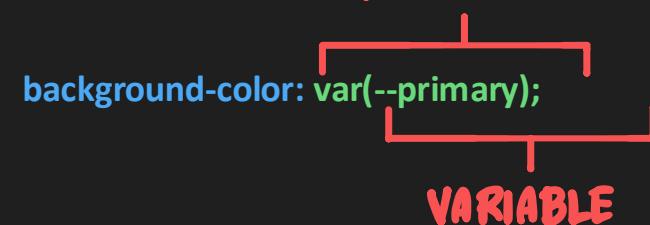
### DEFINING A VARIABLE

DOUBLE DASHES



### USING A VARIABLE

VAR FUNCTION



## ROOT PSEUDO-CLASS

### CSS VARIABLES SET ON ROOT PSEUDO-CLASS

```
:root {  
  --primary: #339af0;  
}
```

## CSS CALCULATIONS

CALC FUNCTION

```
calc(100%/3) - 50px)
```

MIXED UNITS

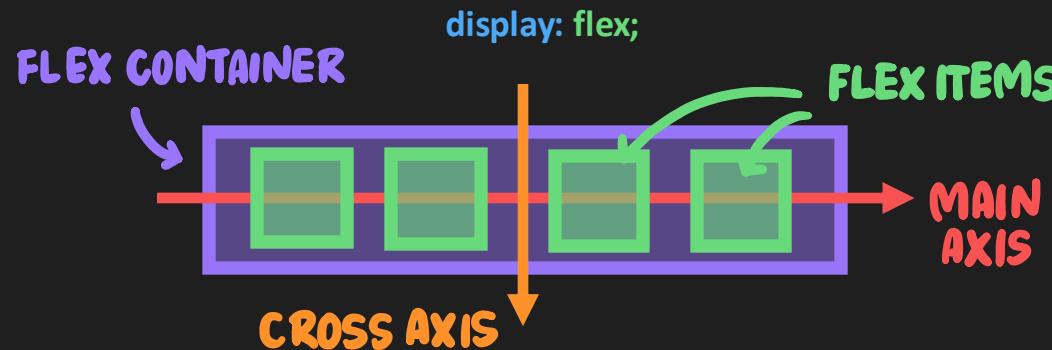


# FLEXBOX

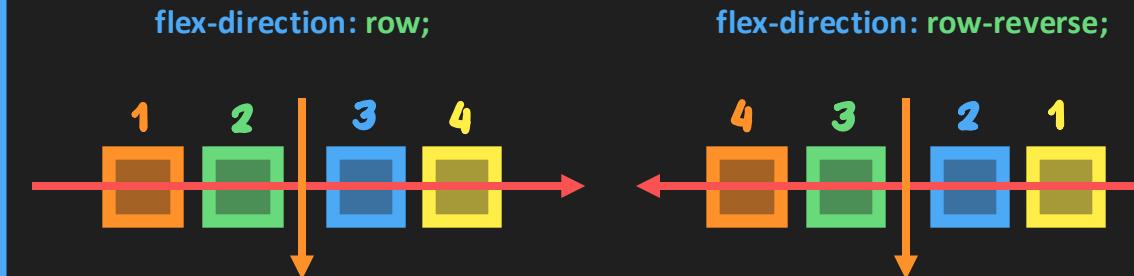
SUMMARY CARD

# INTRODUCTION TO FLEXBOX

## FLEX CONTAINER & ITEMS

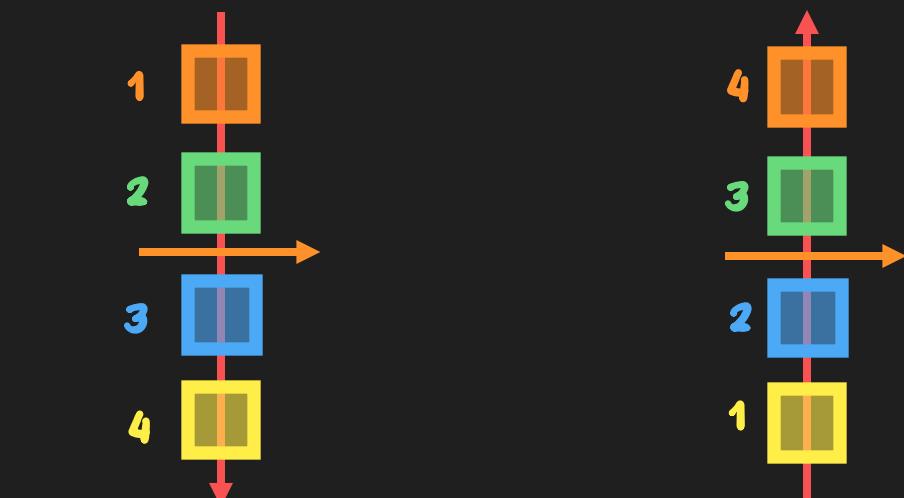


## FLEX DIRECTION



## NORMAL FLOW VS. FLEXBOX

FLEXBOX OVERRIDES BLOCK & INLINE BEHAVIOR

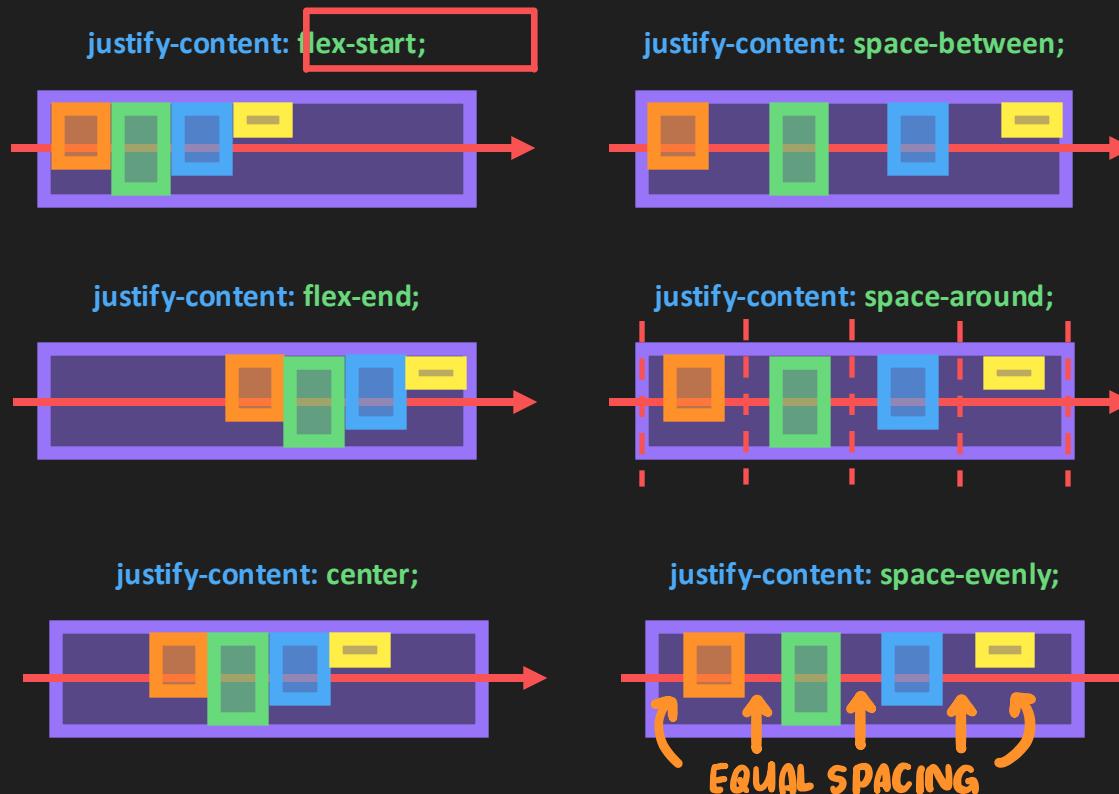


## SUMMARY CARD

# JUSTIFY CONTENT & ALIGN ITEMS

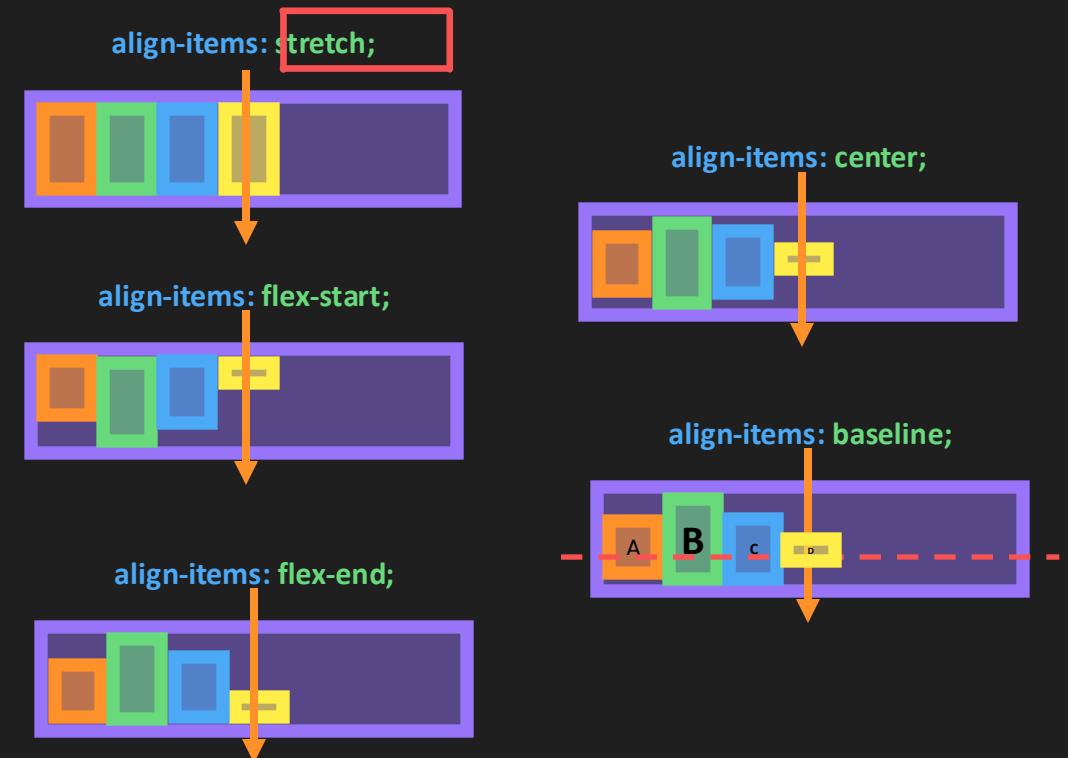
## JUSTIFY CONTENT

### POSITIONS FLEX ITEMS ALONG THE MAIN AXIS



## ALIGN ITEMS

### POSITIONS FLEX ITEMS ALONG THE CROSS AXIS

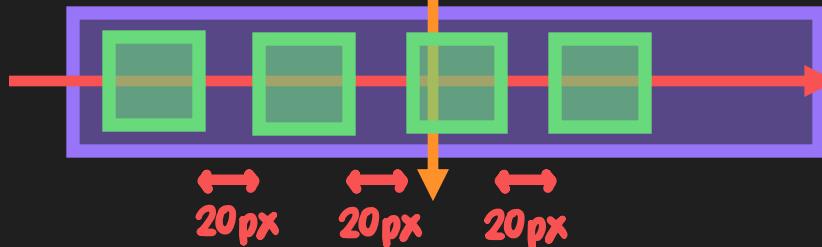


## SUMMARY CARD

# GAP, NESTED FLEXBOX AND CENTERING

## GAP

gap: 20px;



## NESTED FLEXBOX

FLEX CONTAINER  
→

NESTED FLEX CONTAINER  
→



## CENTERING

### TRADITIONAL METHODS

margin: 0 auto;



### HORIZONTAL

### VERTICAL WITH NO HEIGHT ON PARENT

MARGIN OR PADDING

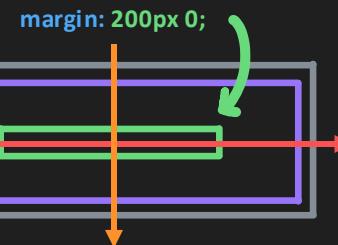
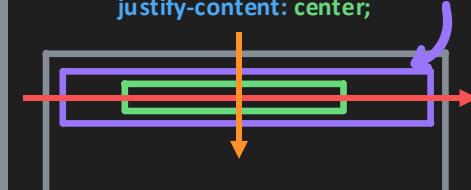


### VERTICAL WITH HEIGHT ON PARENT

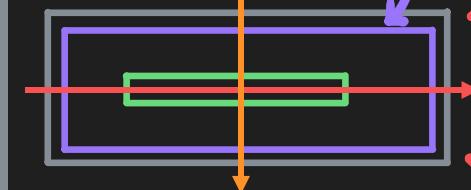
X  
NO EASY WAY

### FLEXBOX

justify-content: center;



margin: 200px 0;

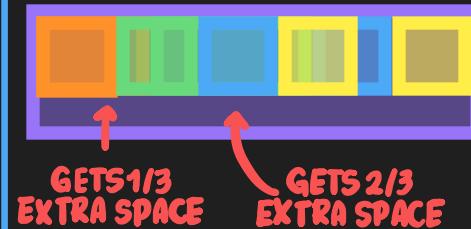


SUMMARY CARD

# FLEX CHILDREN

## FLEX GROW

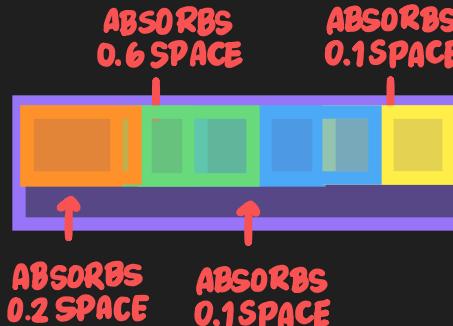
### HOW FLEX ITEMS GROW TO FILL EXTRASPACE



flex-grow: 1;      flex-grow: 0;  
flex-grow: 2;      flex-grow: 0;  
  
TOTAL FLEX GROW =  $1 + 2 + 0 + 0 = 3$

## FLEX SHRINK

### HOW FLEX ITEMS SHRINK WHERE THERE ISN'T SPACE

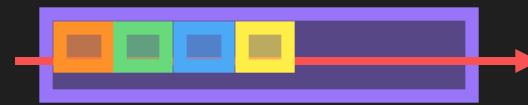


flex-shrink: 2;      flex-shrink: 1;  
flex-shrink: 6;      flex-shrink: 1;  
  
TOTAL FLEX SHRINK =  $2 + 6 + 1 + 1 = 10$

## ALIGN SELF

### SETS INDIVIDUAL FLEX ITEM ALIGNMENT

align-self: auto;



align-self: flex-end;



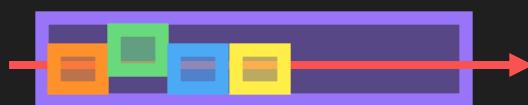
align-self: stretch;



align-self: center;



align-self: flex-start;



align-self: baseline;



## ORDER

order: 0;

LOWER VALUE



HIGHER VALUE

## SUMMARY CARD

# FLEX WRAP & ALIGN CONTENT

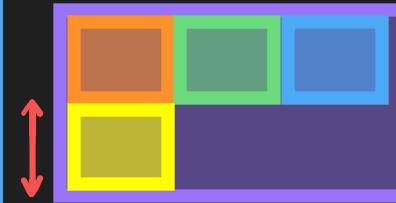
## FLEX-WRAP

FLEX ITEMS WRAP ON MULTIPLE LINES  
IF THEY EXCEED THE CONTAINER

`flex-wrap: wrap;`

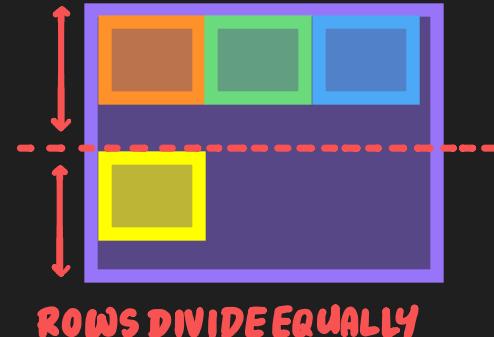
`no wrap` `wrap` `wrap-reverse`

CONTAINER WITH  
NO HEIGHT



CONTAINER  
EXPANDS

CONTAINER WITH  
SET HEIGHT

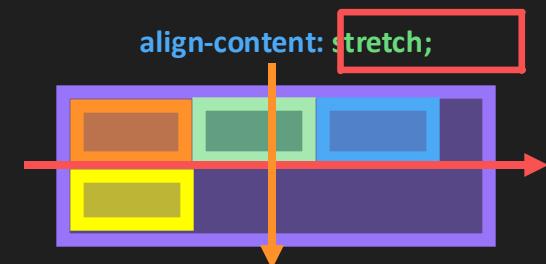


ROWS DIVIDE EQUALLY

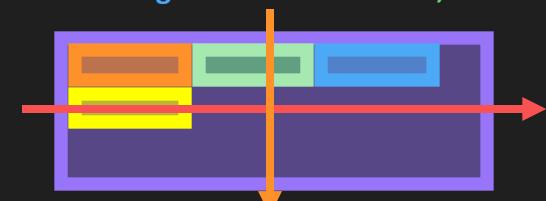
## ALIGN CONTENT

FLEX ITEM ALIGNMENT ALONG CROSS AXIS WHEN MULTIPLE ROWS

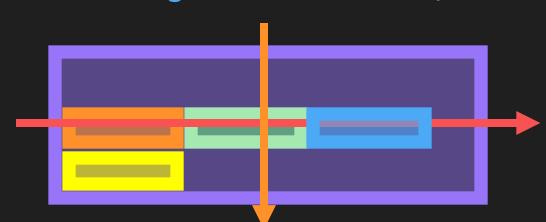
`align-content: stretch;`



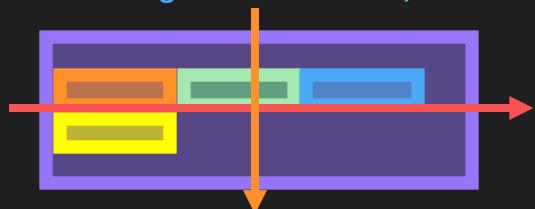
`align-content: flex-start;`



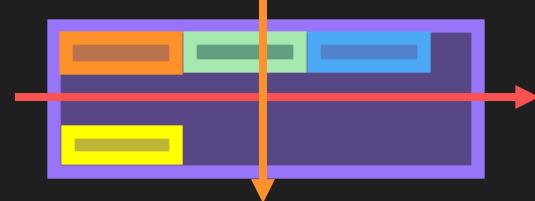
`align-content: flex-end;`



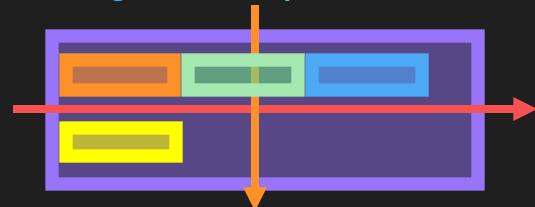
`align-content: center;`



`align-content: space-between;`



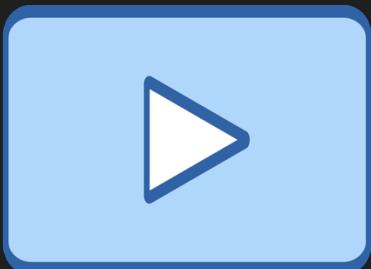
`align-content: space-around;`





# WANT THE REST?

TOPICS IN THIS COURSE ARE BEING DRIPPED  
RELEASED ON YOUTUBE AND THE SUMMARY CARDS  
ARE BEING UPDATED WITH EACH NEW RELEASE



CLICK HERE TO  
SUBSCRIBE ON YOUTUBE  
TO STAY IN THE LOOP