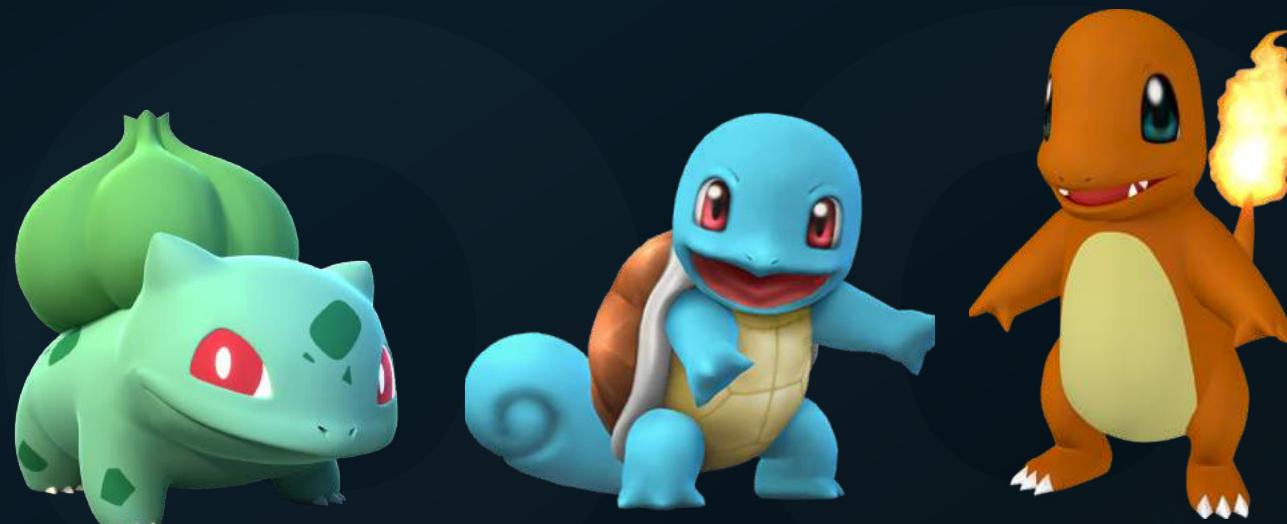




CSS Flexbox

Chéat Sheet



Part - 1



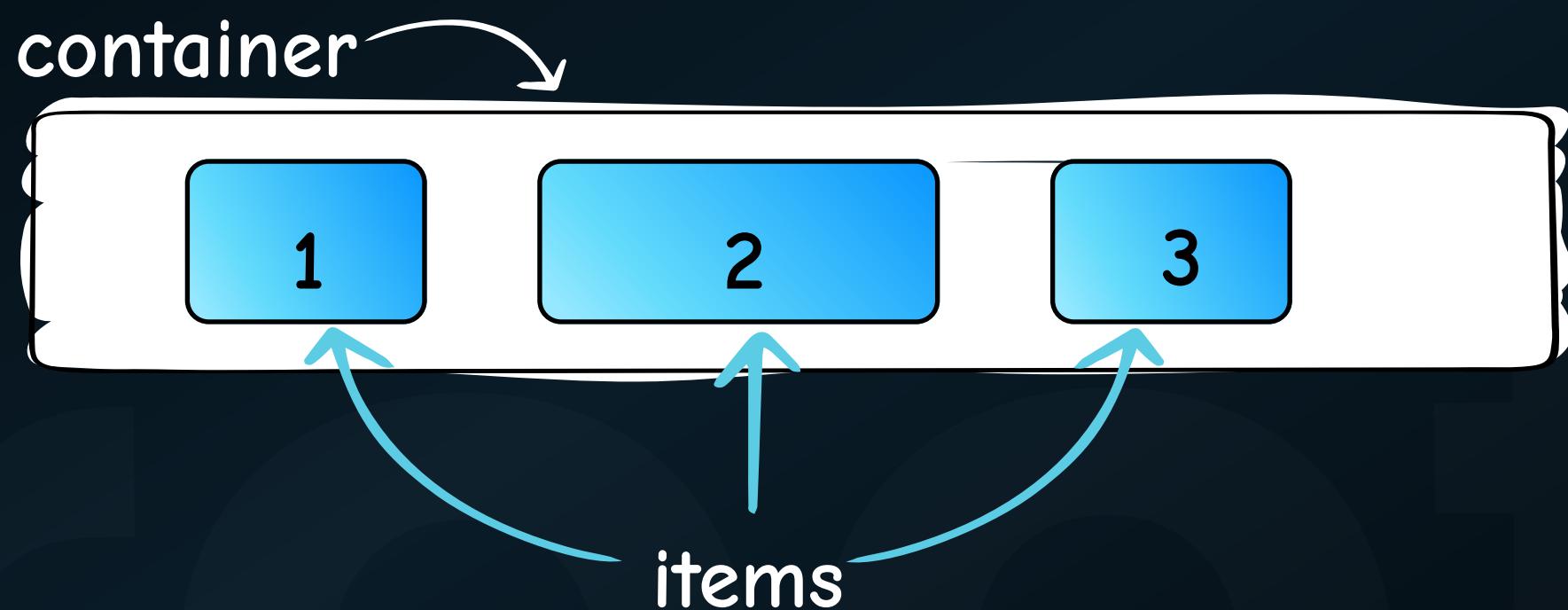
Manish Poduval





What is it ?

The Flexible Box Module (Flexbox) helps us lay out, align and distribute space amongst items in a container



Your first step would be to add this in your styles



```
css index.css
1 // Target your container
2 .container {
3   display: flex;
4 }
```



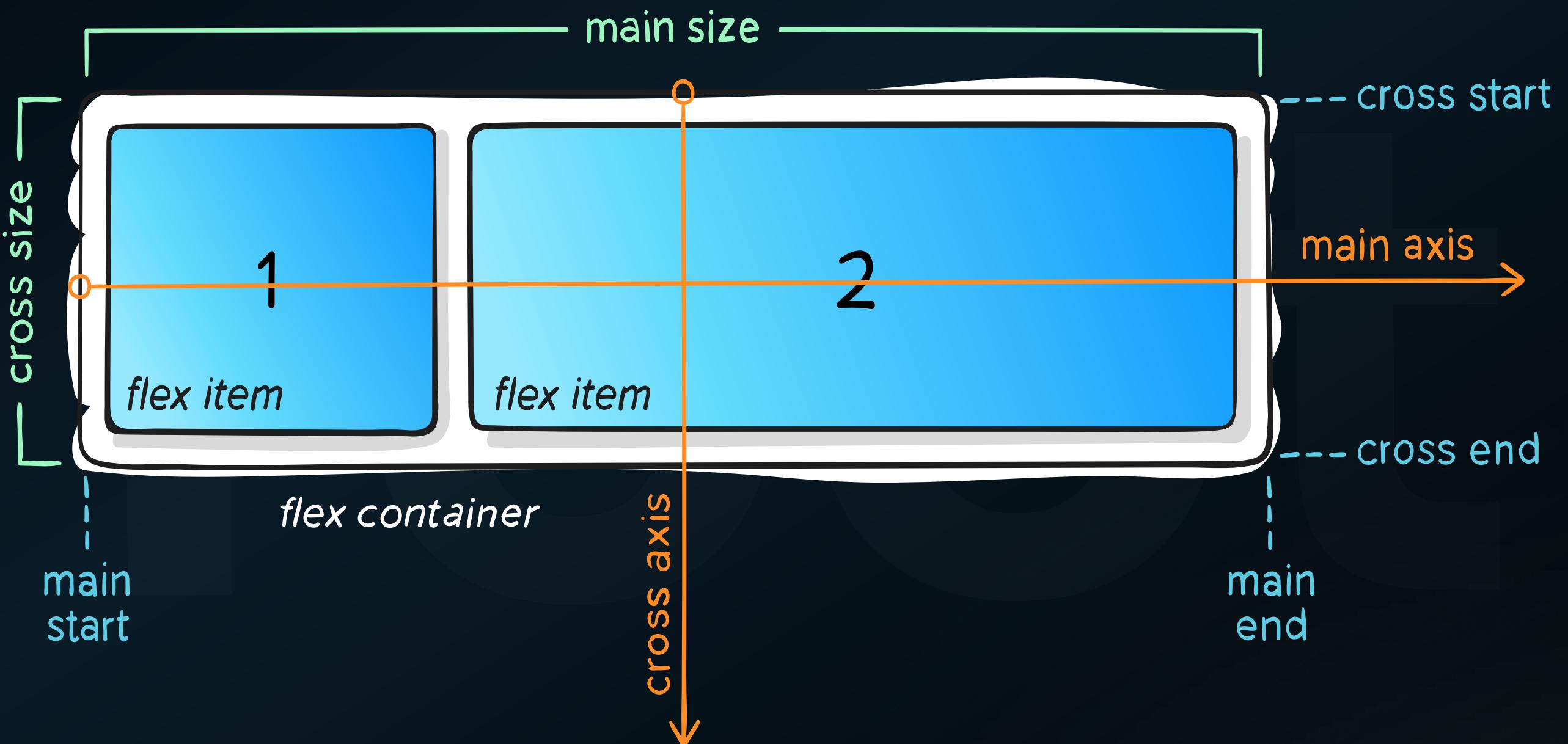
Manish Poduval





Flexbox architecture

The items are distributed along the main axis and cross axis. And, depending on the flex-direction property, the layout position changes between rows and columns.



Manish Poduval

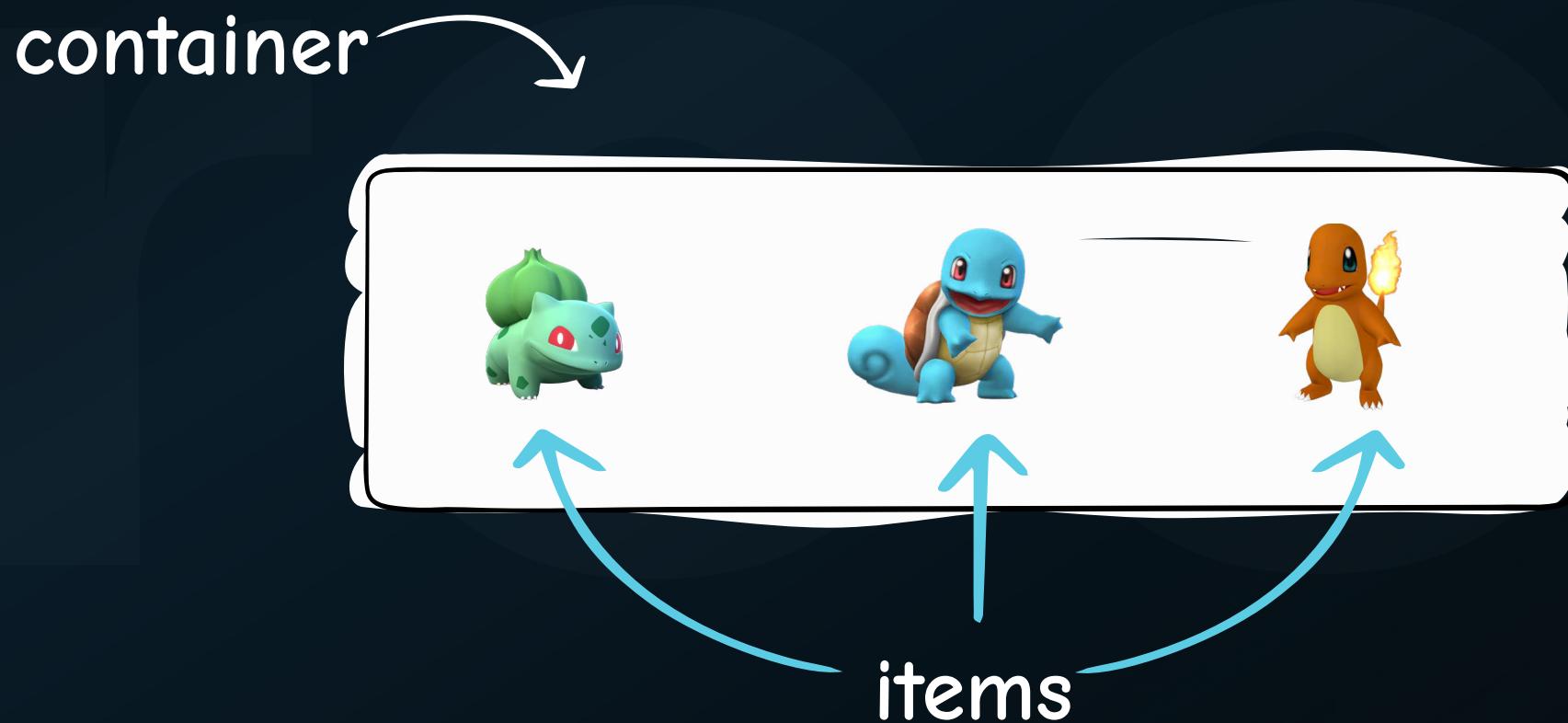




Flexbox properties 😊

We'll divide this cheatsheet in two parts

1. Properties for the container
2. Properties for the items



Manish Poduval





1. Properties for the container

What does this line do ?



index.css

```
1 // Target your container
2 .container {
3   display: flex;
4 }
```

Enables a flex context for all its direct children.



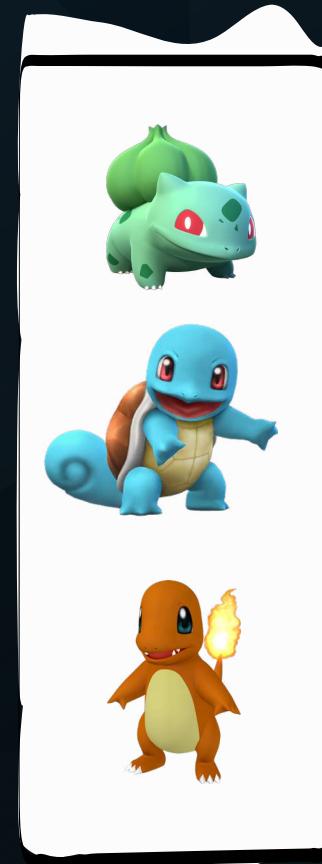
By default all items are aligned in a row



You can change the default direction to a column as well

index.css

```
1 // Target your container
2 .container {
3   display: flex;
4   flex-direction: column;
5 }
```



Manish Poduval



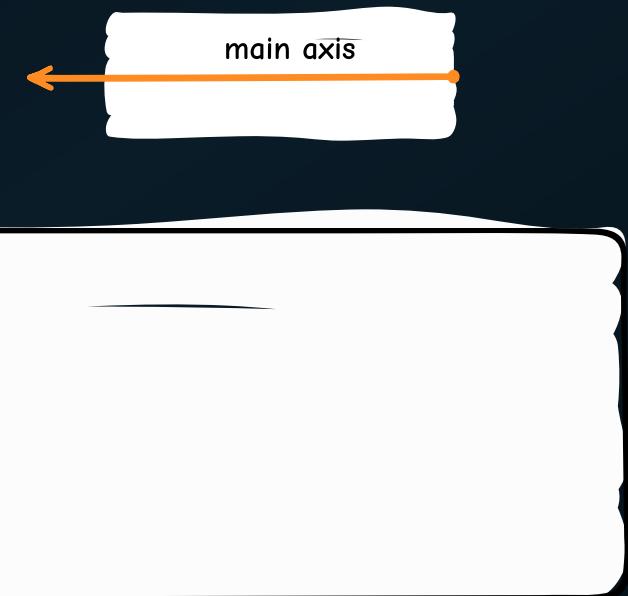


1. Properties for the container

flex-direction will help you establish the main-axis

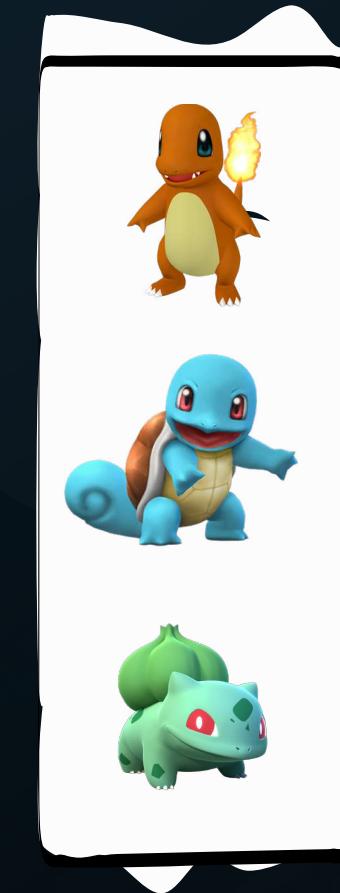
index.css

```
1 // Target your container
2 .container {
3   display: flex;
4   flex-direction: row-reverse;
5 }
```



index.css

```
1 // Target your container
2 .container {
3   display: flex;
4   flex-direction: column-reverse;
5 }
```



Manish Poduval

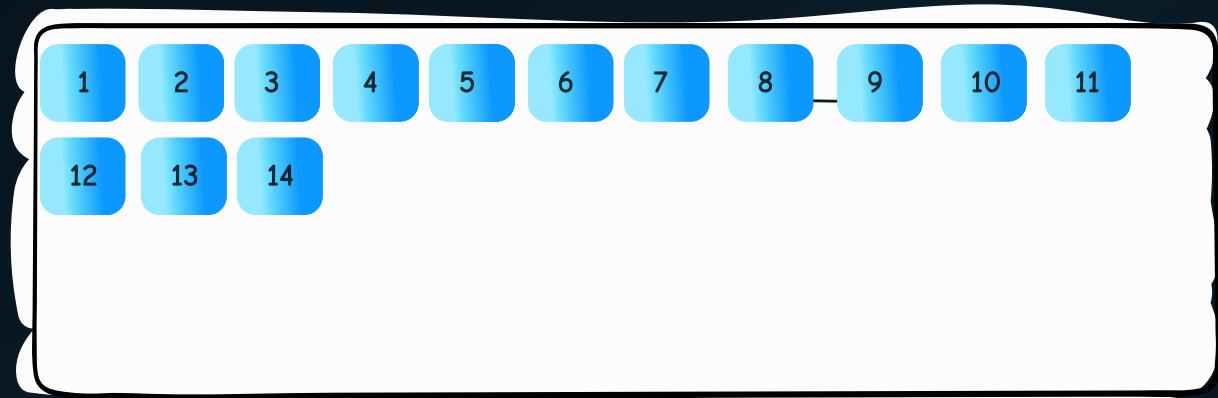




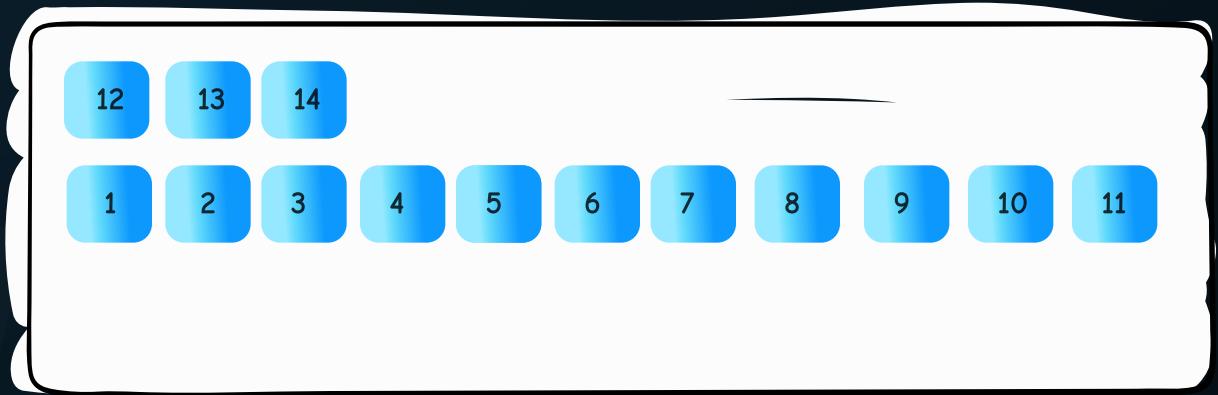
1. Properties for the container

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

```
css index.css
1 // Target your container
2 .container {
3   display: flex;
4   flex-wrap: wrap;
5 }
```

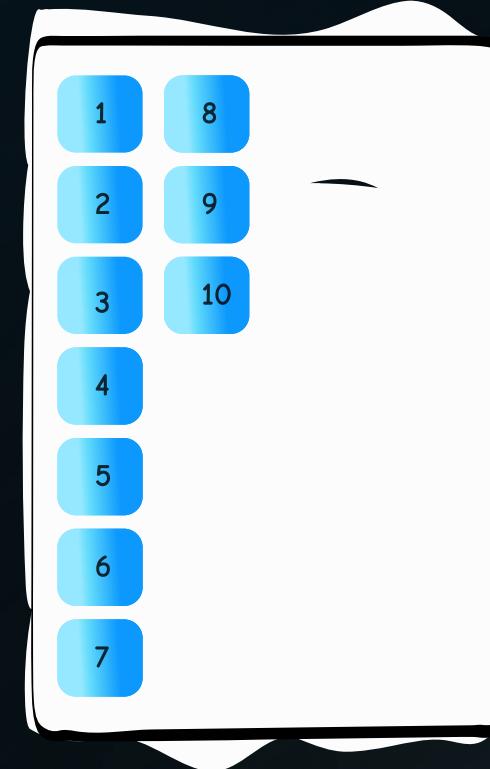


```
css index.css
1 // Target your container
2 .container {
3   display: flex;
4   flex-wrap: wrap-reverse;
5 }
```



flex-flow will combine **flex-direction** and **flex-wrap** into one.

```
css index.css
1 // Target your container
2 .container {
3   display: flex;
4   flex-flow: column wrap;
5 }
```



Manish Poduval





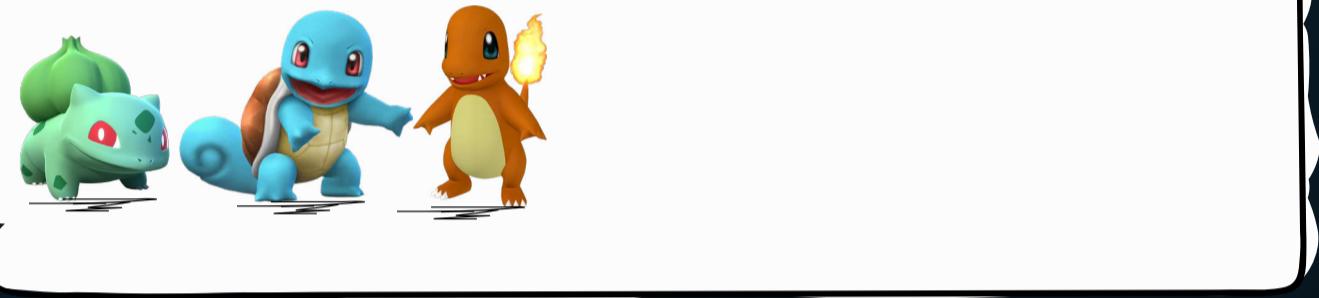
1. Properties for the container



justify-content will help you align the items along the main axis

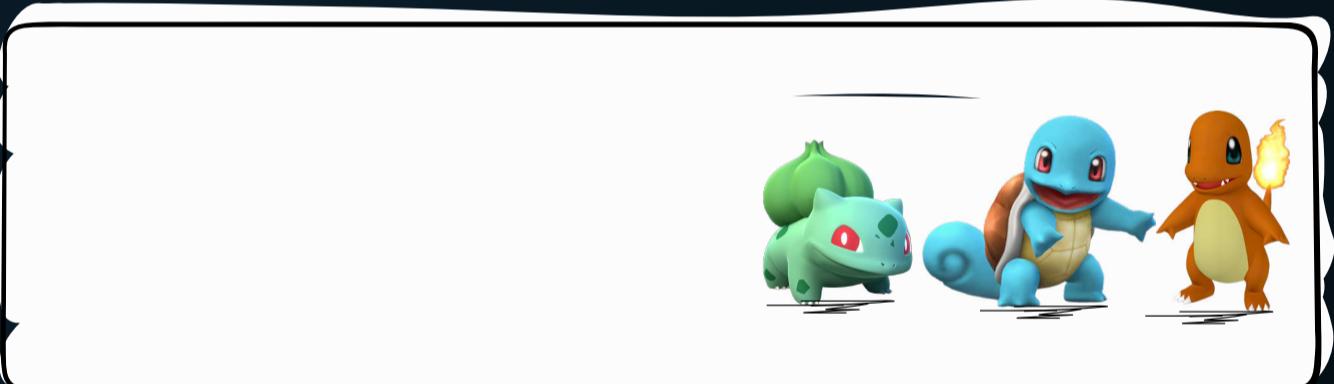
justify-content: flex-start

Items align to the left side of the container.



justify-content: flex-end

Items align to the right side of the container.



justify-content: center

Items align at the center side of the container.



Manish Poduval



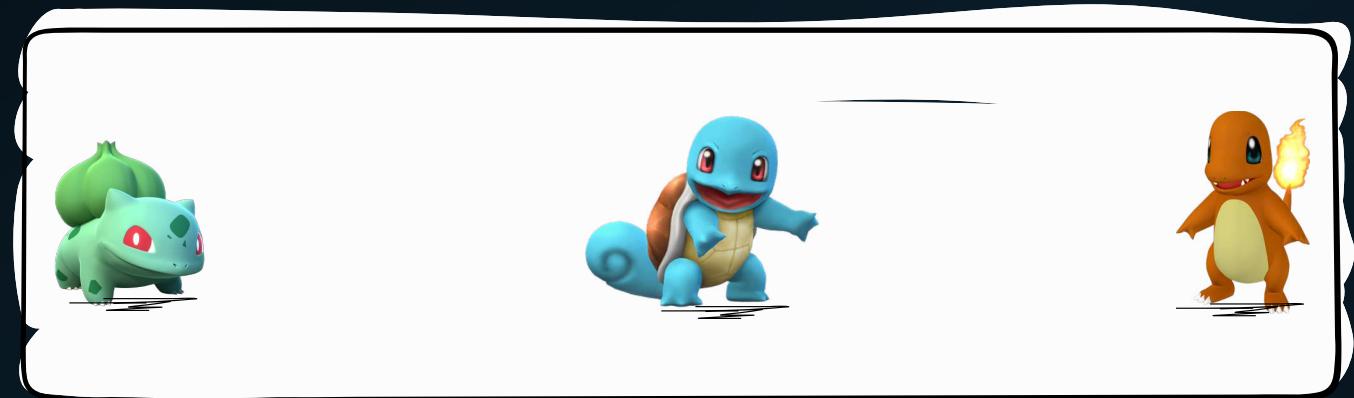


1. Properties for the container



justify-content will help you align the items along the main axis

justify-content: space-between



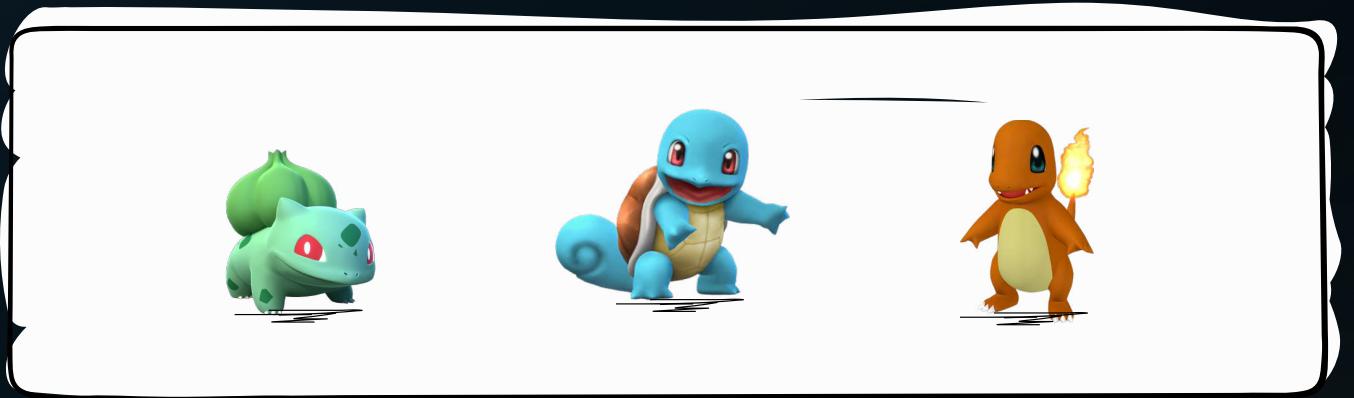
Items display with equal spacing between them

justify-content: space-around



Items display with equal spacing around inner items

justify-content: space-evenly



Items display with equal spacing around them on all sides



Manish Poduval





1. Properties for the container



align-items will help you align the items along the cross axis

align-items: flex-start

Items align to the top of the container.



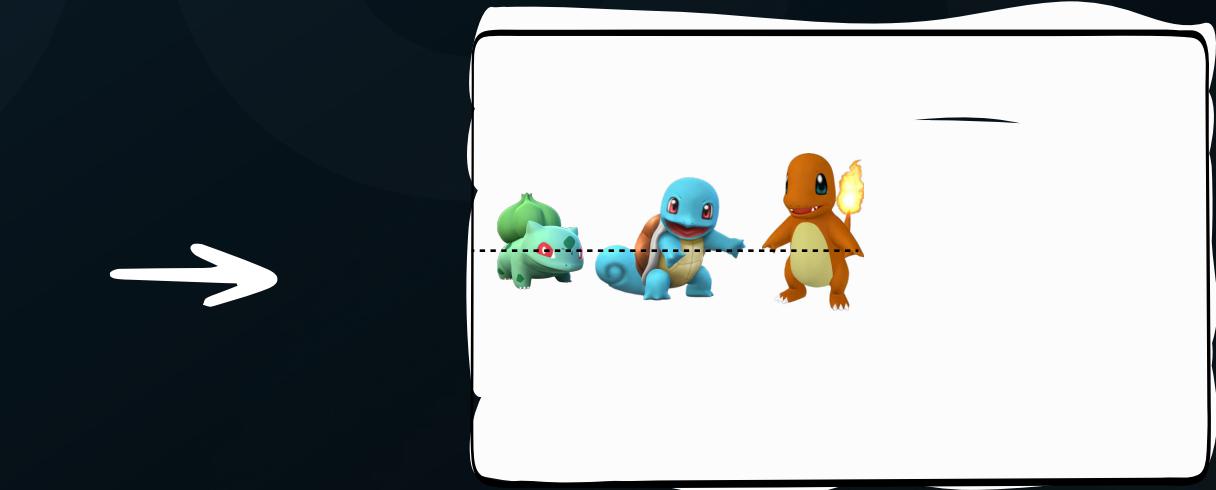
align-items: flex-end

Items align to the bottom of the container



align-items: center

Items align at the vertical center of the container.



Manish Poduval





1. Properties for the container



`align-content` will help you align multiple lines along the cross axis

This can be confusing 🤔

👉 `align-content` determines the spacing between multiple lines

👉 `align-items` determines how the items as a whole are aligned within the container.

💡 When there is only one line, `align-content` has no effect.



Manish Poduval





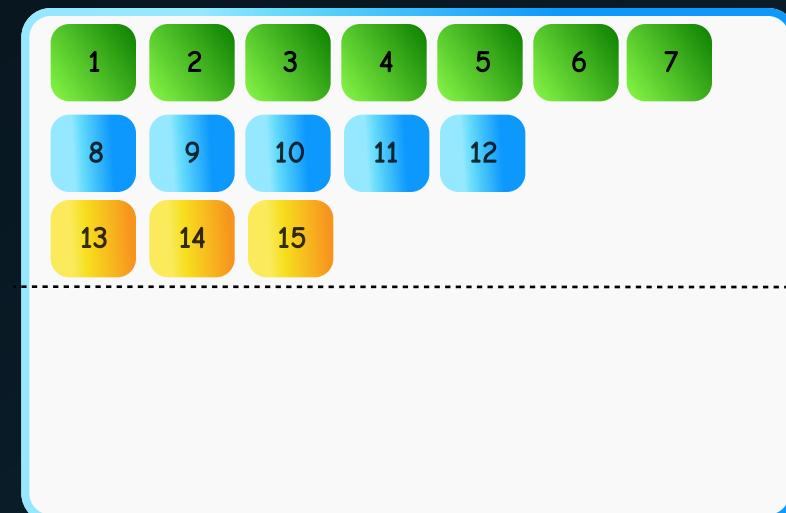
1. Properties for the container



align-content will help you align multiple lines along the cross axis

align-content: flex-start

Lines are packed at the top of the container



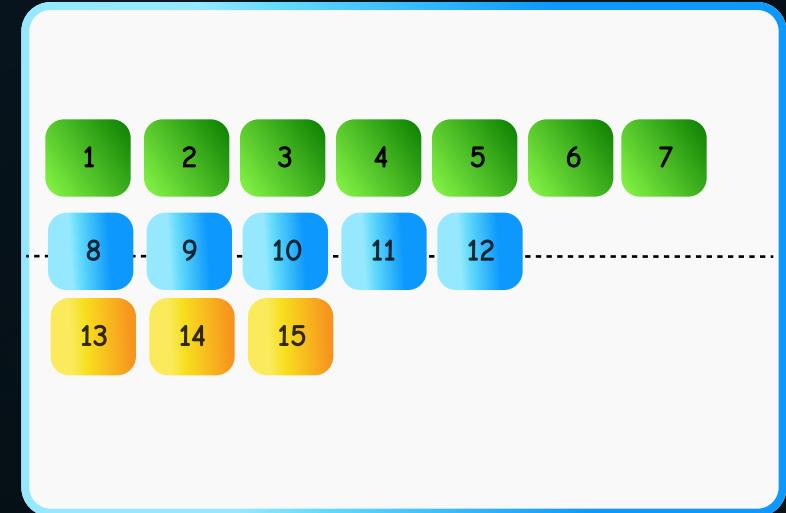
align-content: flex-end

Lines are packed at the bottom of the container



align-content: center

Lines are packed at the vertical center of the container.



Manish Poduval





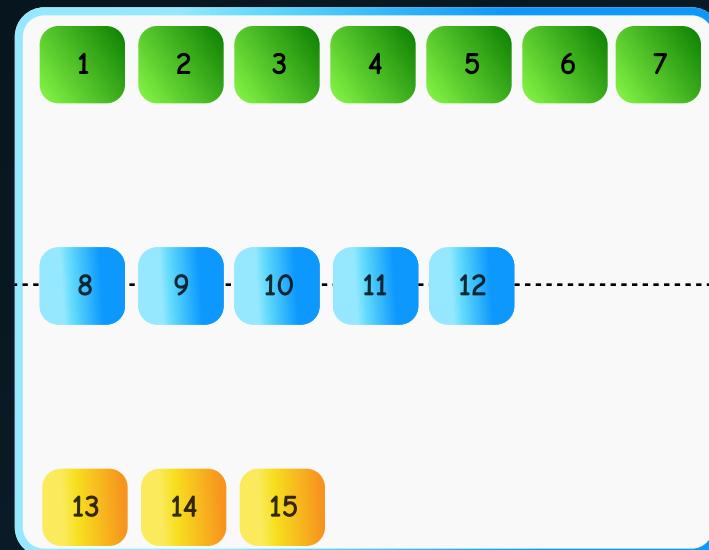
1. Properties for the container



align-content will help you align multiple lines along the cross axis

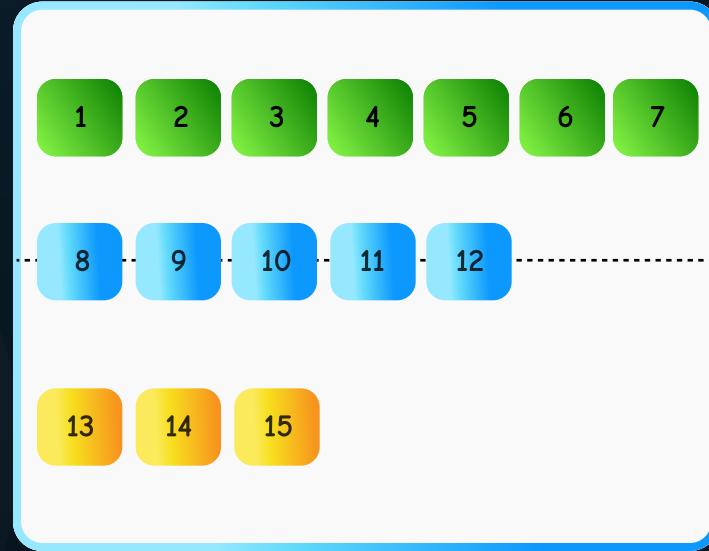
align-content: space-between

Lines display with equal spacing between them.



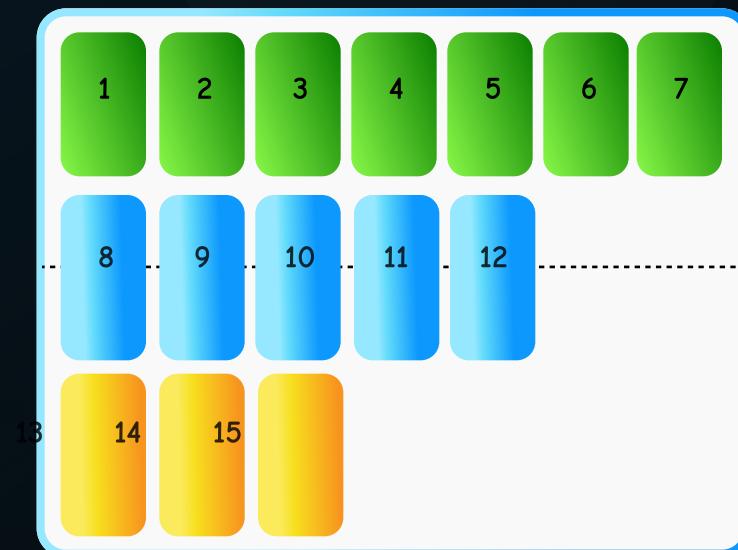
align-content: space-around

Lines display with equal spacing around them.



align-content: stretch

Lines are stretched to fit the container.



Manish Poduval





1. Properties for the container

`gap` explicitly controls the space between flex items.



It applies that spacing only between items not on the outer edges.

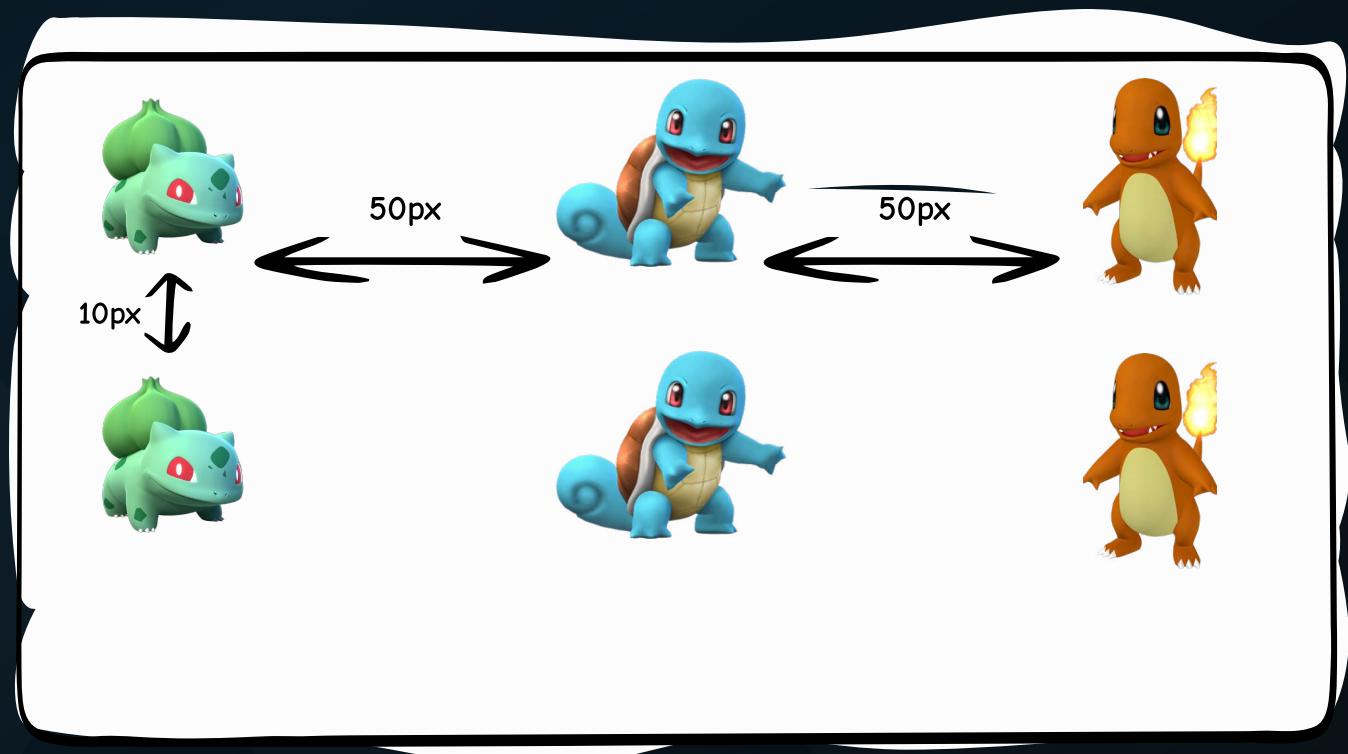
`gap: 10px;`

spacing between items



`gap: 10px 50px;`

spacing between items -> row column



Manish Poduval





Resources to practice



Flex Box Adventure

codingfantacy.com



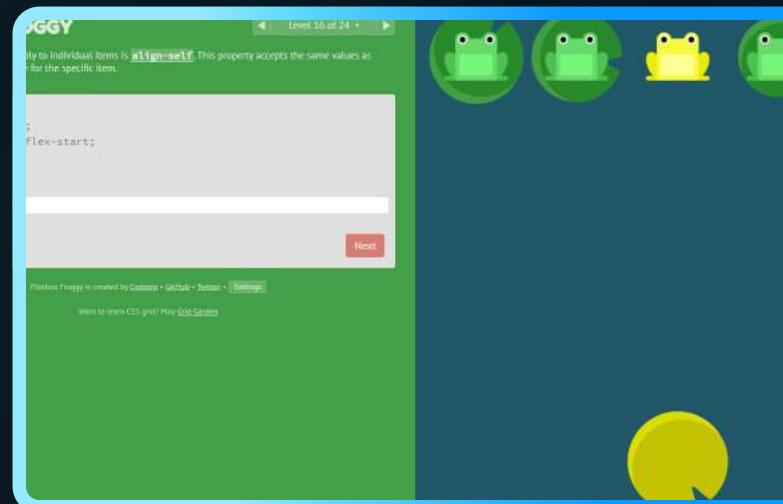
Flexbox Zombies

Chapter 1 The Crossbow flex-direction align-items: flex-start;	Chapter 2 The Justify Laser justify-content: space-around;	Chapter 3 The Alignment Laser align-items: center;
Chapter 4 One of These Things is Not Like the Other align-self: flex-end;	Chapter 5 Where the Wild Things Grow flex-grow: 1;	Chapter 6 That Shrinking Feeling flex-shrink: 1;

mastery.games



flexboxdefence.com



flexboxfroggy.com



Manish Poduval



Thank you!

Want to upskill yourself with our live cohort based courses ?

Scan this to know more

