CSS'S POSITIONING & DISPLAY

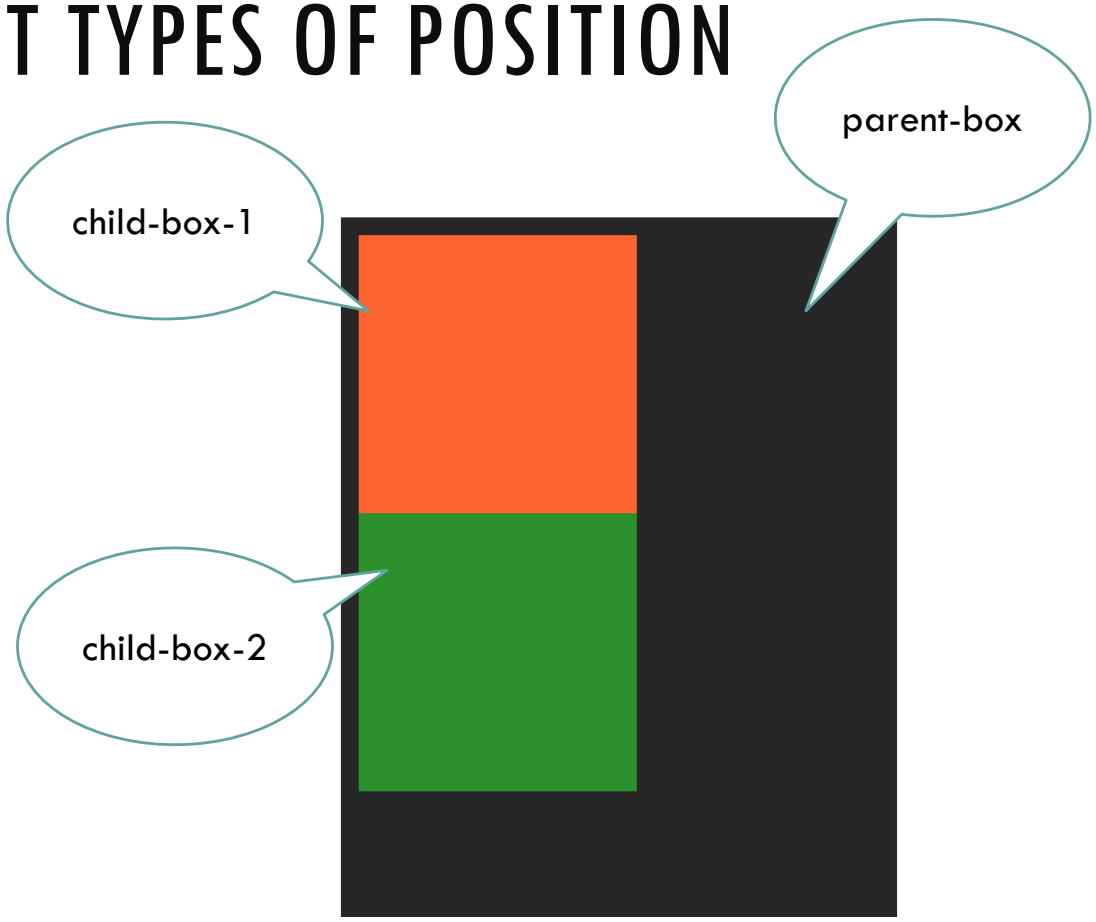
I4GIC
By Thavorac

THERE ARE 4 DIFFERENT TYPES OF POSITION

Static

The default positioning. Elements flow along with HTML document

```
<div class="parent-box">
  <div class="child-box-1"></div>
  <div class="child-box-2"></div>
</div>
```

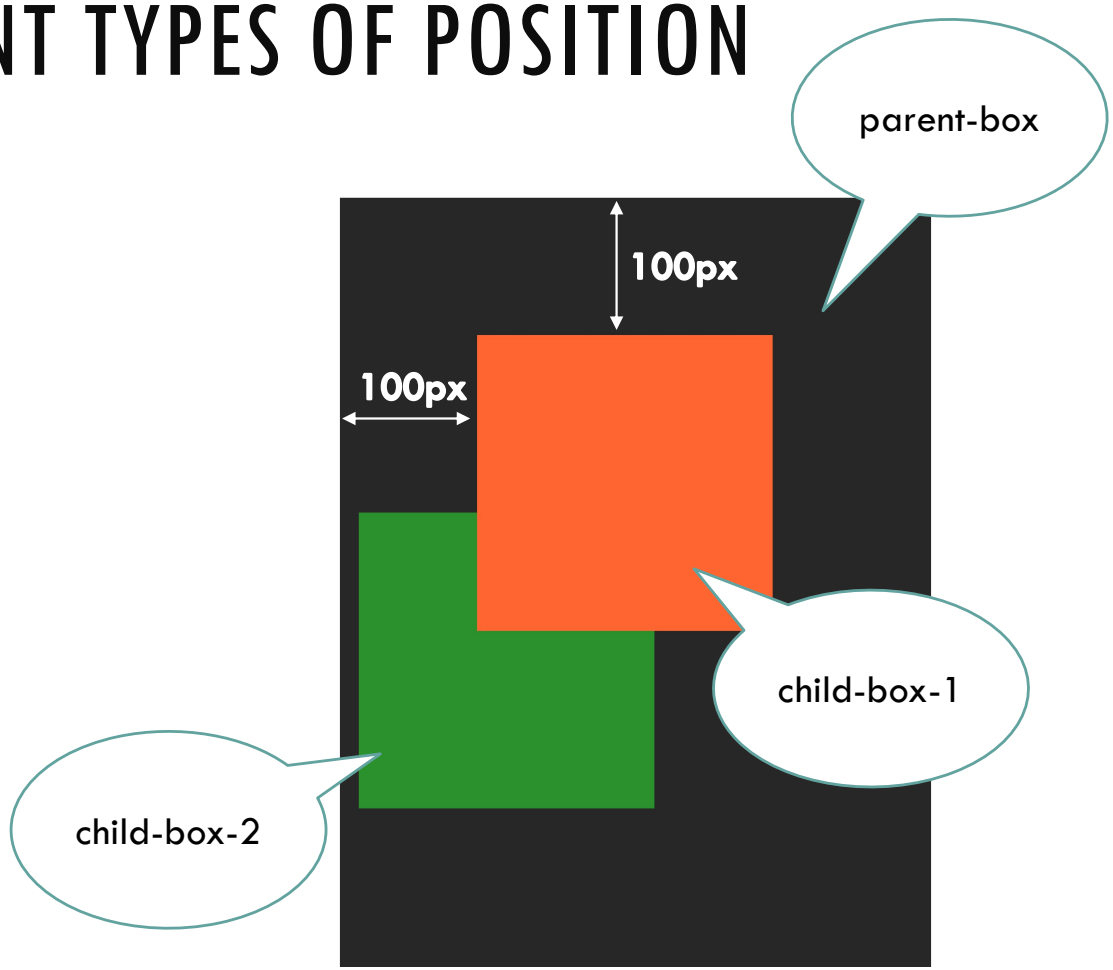


THERE ARE 4 DIFFERENT TYPES OF POSITION

Relative

Similar to static but relative elements can change its top/right/left/bottom attribute and then it will move relative to its original position.

```
.child-box-1 {
  position: relative;
  top: 100px;
  left: 100px;
  width: 250px;
  height: 250px;
  background: #ff652f;
}
```

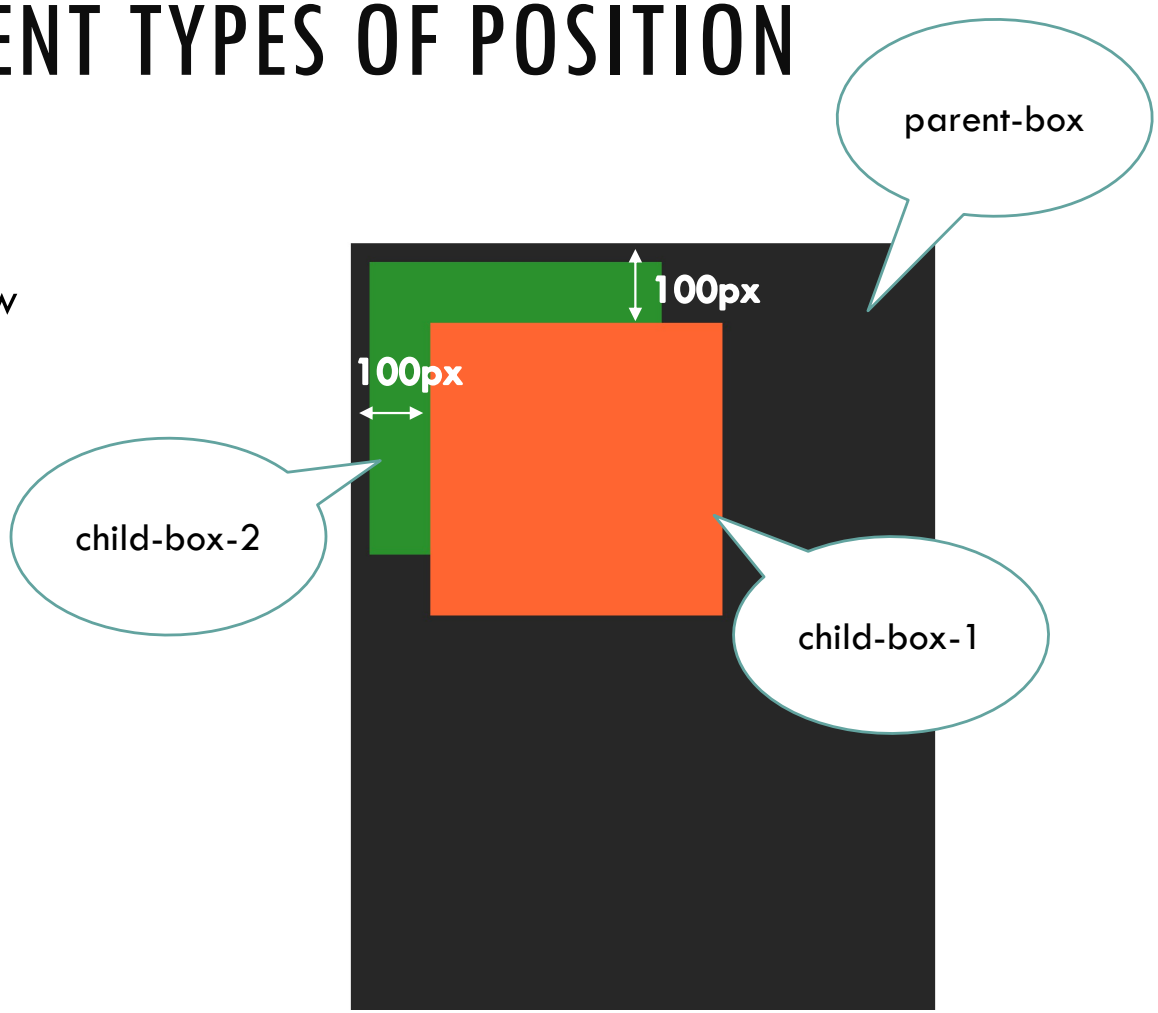


THERE ARE 4 DIFFERENT TYPES OF POSITION

Absolute

Remove element from document flow and position itself in reference to a positionted container.

```
.child-box-1 {
  position: absolute;
  top: 100px;
  left: 100px;
  width: 250px;
  height: 250px;
  background: #ff652f;
}
```



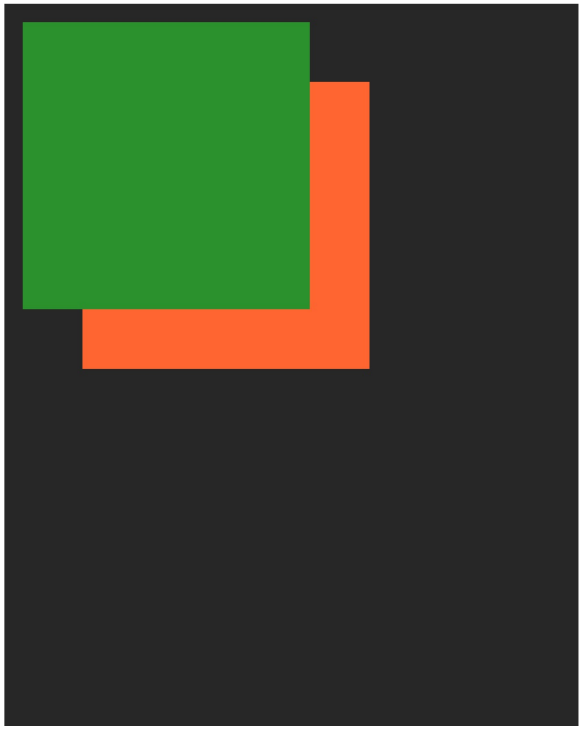
THERE ARE 4 DIFFERENT TYPES OF POSITION

Absolute

To move the element on top of another element, use z-index and assign it a positioning.

```
.child-box-1 {
  position: absolute;
  top: 100px;
  left: 100px;
  width: 250px;
  height: 250px;
  background: #ff652f;
}
```

```
.child-box-2 {
  width: 250px;
  height: 250px;
  position: relative;
  z-index: 1;
  background: #2c912c;
}
```

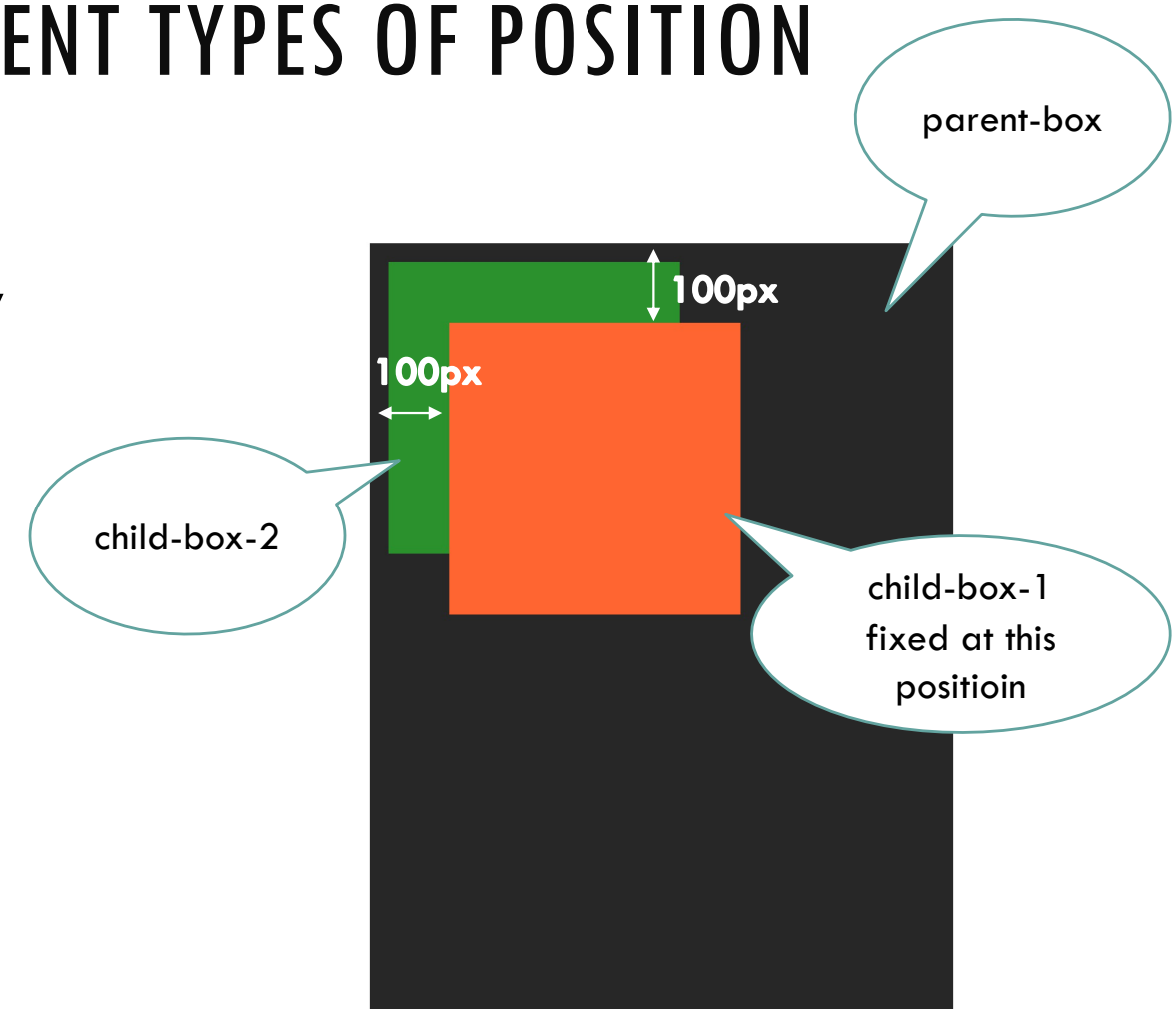


THERE ARE 4 DIFFERENT TYPES OF POSITION

Fixed

Fixed itseslf to a screen, not to any specific container. While scrolling, fixed element doesn't move along.

```
.child-box-1 {
  position: fixed;
  top: 100px;
  left: 100px;
  width: 250px;
  height: 250px;
  background: #ff652f;
}
```

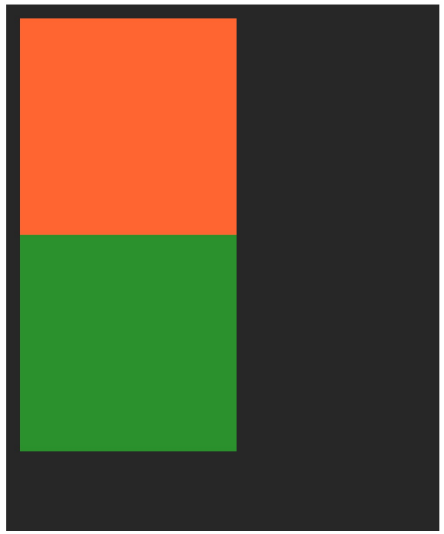


THERE ARE 4 DIFFERENT TYPES OF POSITION

Sticky

Combination of relative and fixed. Originaly, it stays relatively to its original position but when scroll out of bound, it change to absolute positioning.

```
.child-box-2 {
  position: sticky;
  top: 0;
  width: 250px;
  height: 250px;
  background: #2c912c;
}
```

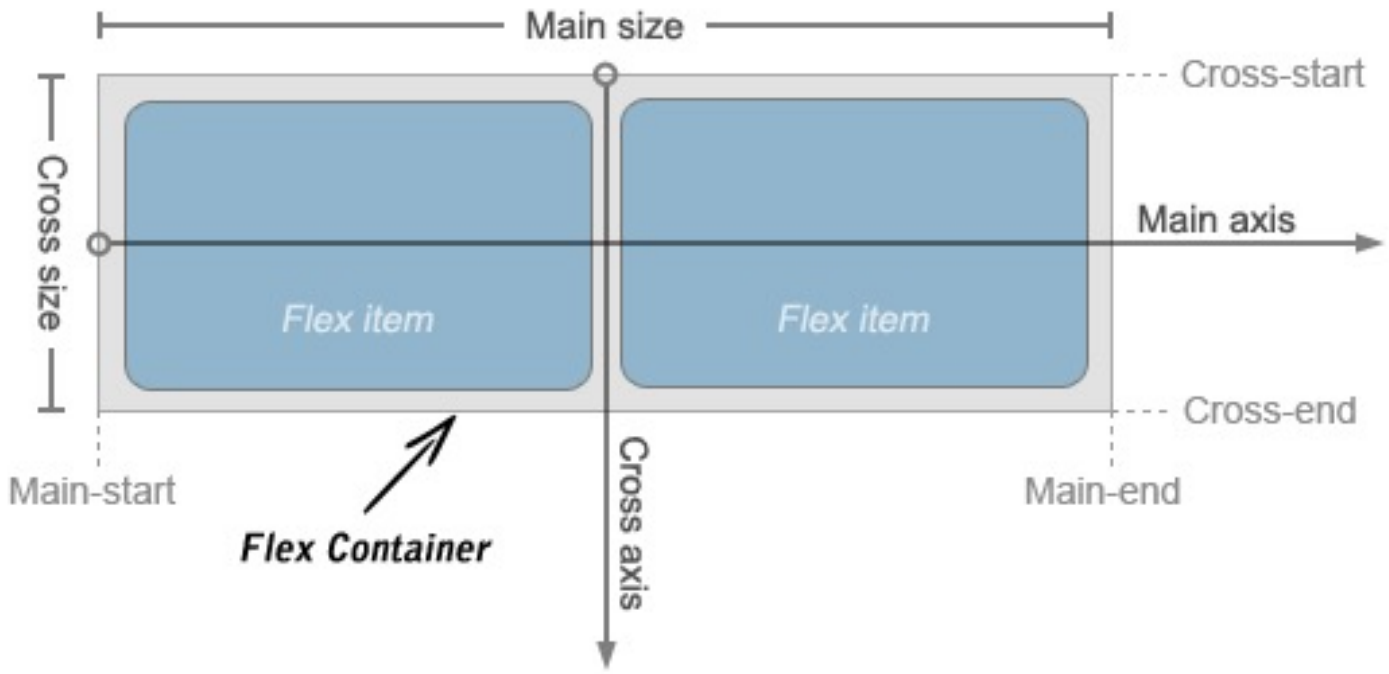


FLEXBOX

Before the Flexbox Layout module, there were four layout modes:

- Block, for sections in a webpage
- Inline, for text
- Table, for two-dimensional table data
- Positioned, for explicit position of an element

The Flexible Box Layout Module, makes it easier to design flexible responsive layout structure without using float or positioning.


Main size
Cross-start
Cross size
Main axis
Flex item
Flex item
Cross-end
Main-start
Main-end
Flex Container
Cross axis

DEFINE A FLEXBOX CONTAINER

```html
<div class="flex-container">
  <div>1</div>
  <div>2</div>
  <div>3</div>
</div>
```

```css
.flex-container {
  display: flex;
}
```

FLEX CONTAINER PROPERTIES

The flex container properties are:

- flex-direction : defines in which direction the container wants to stack the flex items.
- flex-wrap : specifies whether the flex items should wrap or not
- flex-flow : is a shorthand property for setting both the flex-direction and flex-wrap properties
- justify-content : is used to align the flex items.
- align-items
- align-content