

STILL USING **Z-INDEX : 9999 ???**

LEARN **Z-INDEX**
IN 5 MINUTES !



Follow @PriteshKiri

CSS Z-index

→ What does preload do?

=> In simple terms, It helps the browser know about important resources that will be needed by the page, without delaying the page's "onload" event.

Syntax:

```
<link rel="preload" href="http://font.com" as="font" />
```

CSS Z-index

→ What is z-index?

=> It defines the order of the elements on the z-axis or in simple terms, It specifies the stack order of an element.

=> z-index only works only on positioned elements except for static position (position - absolute, relative, fixed, sticky) and flex items.

CSS Z-index

→ Why use z-index?

=> We all face issues like overlapping when positioning elements.

=> Z-index helps us stack the elements or in simple words, helps us position elements along the axis perpendicular to your screen.

CSS Z-index

→ How to use z-index?

Always remember the following points while using z-index :

CSS Z-index

1

- ◆ Without any z-index value, elements stack in the order that they appear in the DOM.
- ◆ In simple words, the lowest one down at the same hierarchy level appears on top.

CSS Z-index

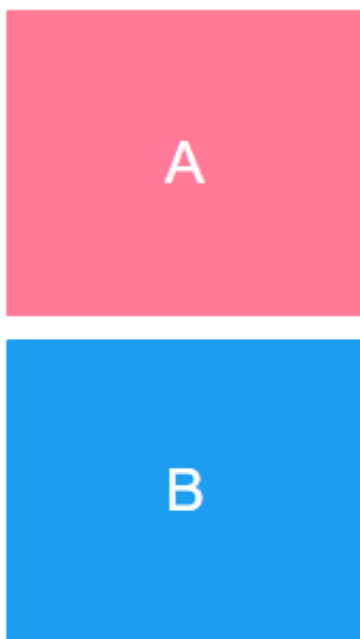


OUTPUT

<-- HTML -->

```
<div class="container">
  <div class="itemA">A</div>
  <div class="itemB">B</div>
</div>
```

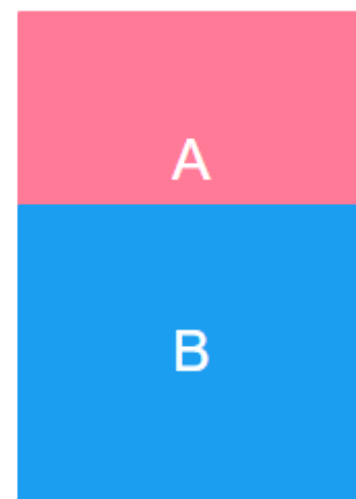
Before



<-- CSS -->

```
.itemA {
  background-color: pink;
}
.itemB {
  background-color: blue;
  margin-top: -50px;
}
```

After



CSS Z-index

2

- ◆ Positioned elements appear in front of non-positioned ones.
- ◆ This means if two elements overlapping and you give a position property to the bottom one then it will be shown in front of the other.

CSS Z-index



OUTPUT

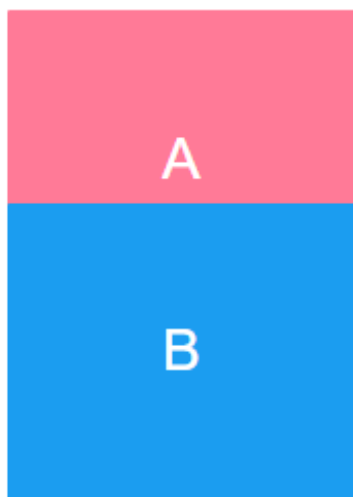
<-- HTML -->

```
<div class="container">
  <div class="itemA">A</div>
  <div class="itemB">B</div>
</div>
```

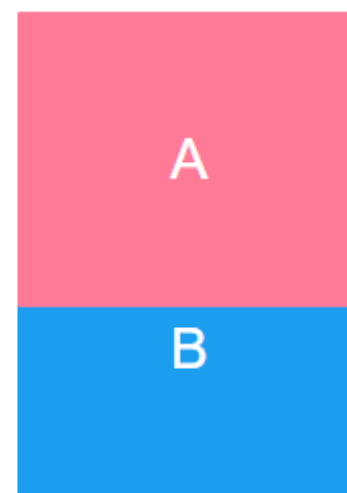
<-- CSS -->

```
.itemA {
  background-color: pink;
  position: relative;
}
.itemB {
  background-color: blue;
  margin-top: -50px;
}
```

Before



After



CSS Z-index

3

- ◆ Z-index property specifies the stack order of an element. One with a higher z-index will appear in front only if are under the same parent element.
- ◆ An element can have a positive or negative stack order.

CSS Z-index



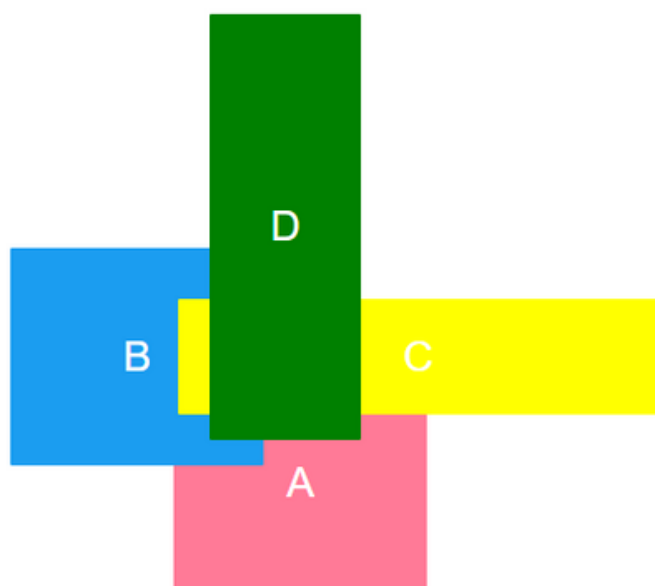
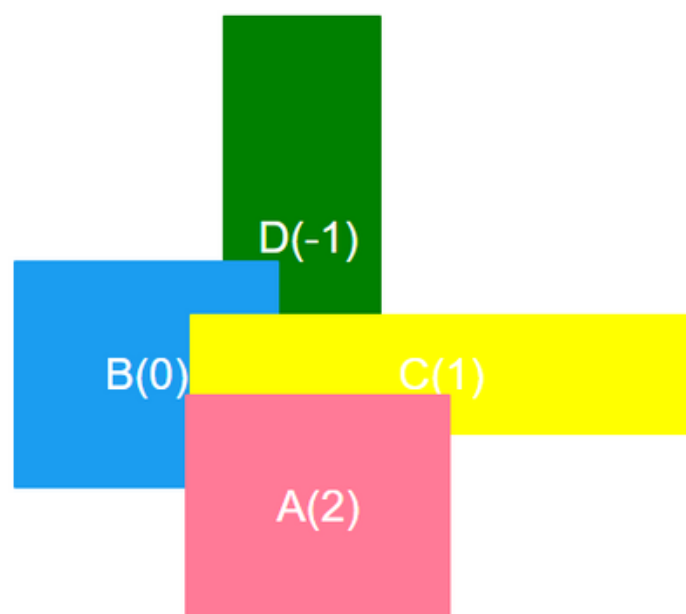
OUTPUT

<-- HTML -->

```
<div class="container">
  <div class="itemA">A</div>
  <div class="itemB">B</div>
  <div class="itemC">C</div>
  <div class="itemD">D</div>
</div>
```

<-- CSS -->

```
.itemA {
  background-color: pink;
  z-index: 2;
}
.itemB {
  background-color: blue;
  z-index: 0;
}
.itemC {
  background-color: yellow;
  z-index: 1;
}
.itemD {
  background-color: green;
  z-index: -1;
}
```

BeforeAfter

CSS Z-index

4

- ◆ Nesting plays a big role in z-index concept.
- ◆ If there are two parent elements A and B. And if z-index of A is higher than B. Then, the children of B can't overlap the children of A at any cost.
- ◆ Even if you put z-index: 9999 🤔

CSS Z-index



OUTPUT

<-- HTML -->

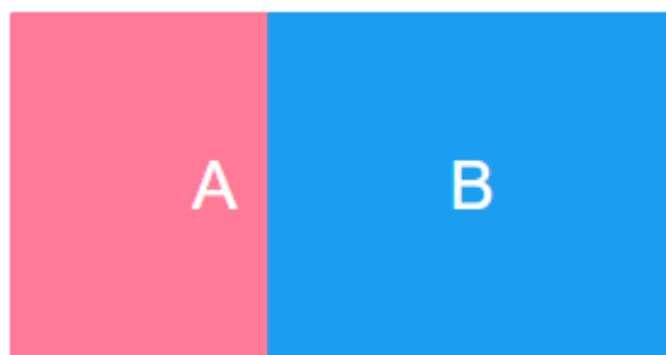
```
<div class="container1">
  <div class="itemA">A</div>
</div>

<div class="container2">
  <div class="itemB">B</div>
</div>
```

<-- CSS -->

```
.container1 {
  z-index: 1;
}
.container2 {
  z-index: 2;
}
.itemA {
  background-color: pink;
  z-index: 9999;
}
.itemB {
  background-color: blue;
  z-index: -1;
}
```

Output





FOLLOW ME FOR MORE **WEB DEVELOPMENT**
TIPS AND TRICKS



SHARE



LIKE



COMMENT



REPOST



SAVE



Follow @PriteshKiri