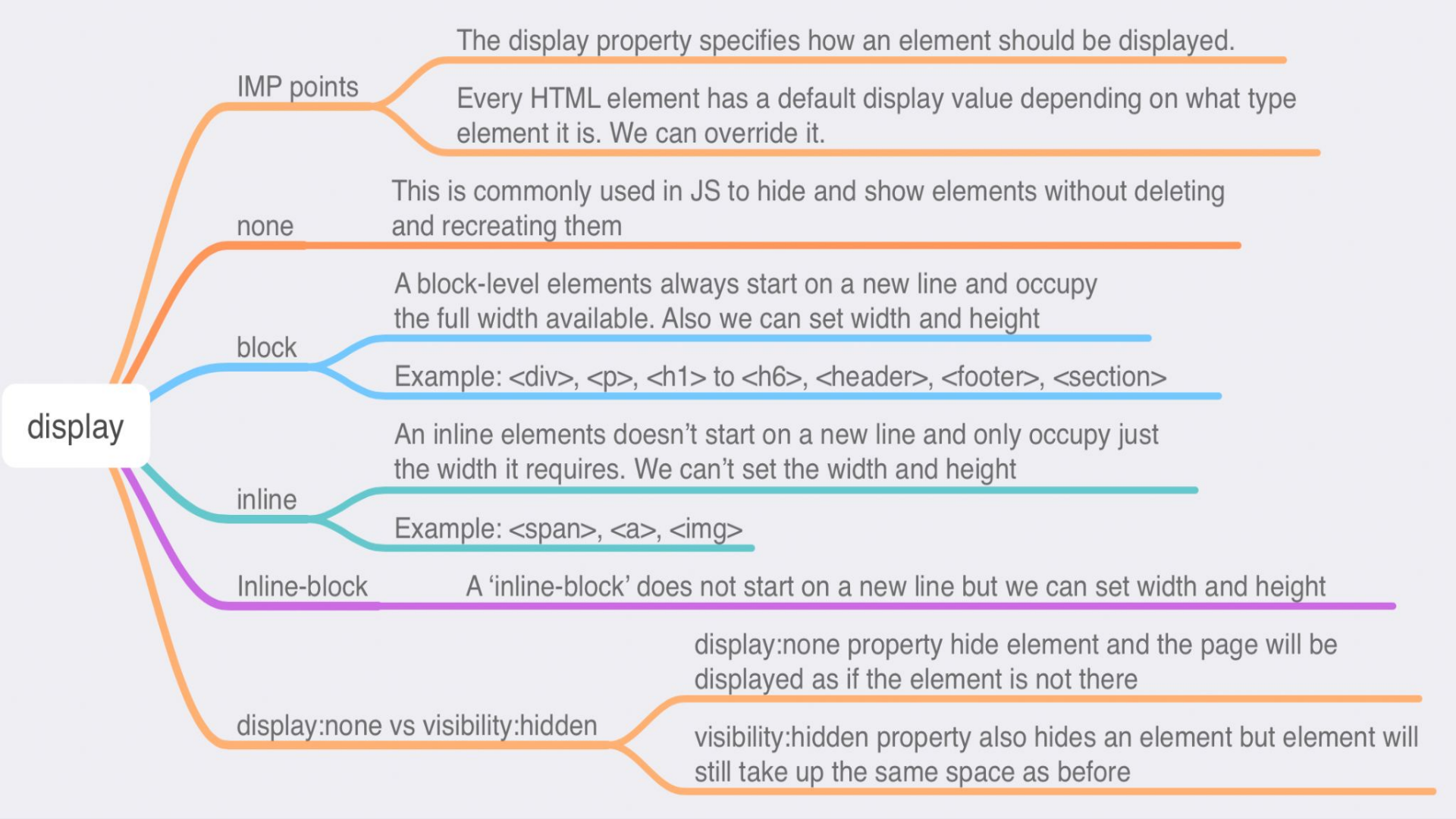
CSS - Display and Positioning

By CODEMIND Technology

Contact us – 966 50 44 698
966 50 44 598

The ‘display’ Property

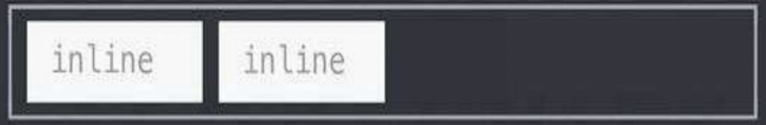
- Specifies how an element is displayed.
- Note: Every element has a default display value depending on what type of element it is.
    - none
    - block
    - inline
    - inline-block


display
IMP points
The display property specifies how an element should be displayed.
Every HTML element has a default display value depending on what type element it is. We can override it.
none
This is commonly used in JS to hide and show elements without deleting and recreating them
block
A block-level elements always start on a new line and occupy the full width available. Also we can set width and height
Example: , <p>, <h1> to <h6>, <header>, <footer>, <section>
inline
An inline elements doesn't start on a new line and only occupy just the width it requires. We can't set the width and height
Example: , <a>, <img>
Inline-block
A 'inline-block' does not start on a new line but we can set width and height
display:none vs visibility:hidden
display:none property hide element and the page will be displayed as if the element is not there
visibility:hidden property also hides an element but element will still take up the same space as before

Inline elements take up as little space as possible, flow horizontally, & can't have their width or height manually adjusted. eg: span

```
.element {
    display: inline;
}
```

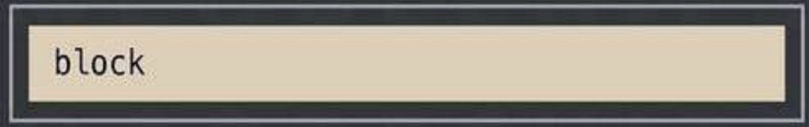
takes only needed space but width and height cannot be adjusted



Block-level elements take up the full width of their container with line breaks before and after, and can have their height and width manually adjusted. eg: div

```
.element {
    display: block;
}
```

takes full width



width and height can be adjusted

inline-block

Sometimes you might have extra stuff and may require extra space. inline-block are those adaptable guys who adjust their spacing and let others sit next in the available space

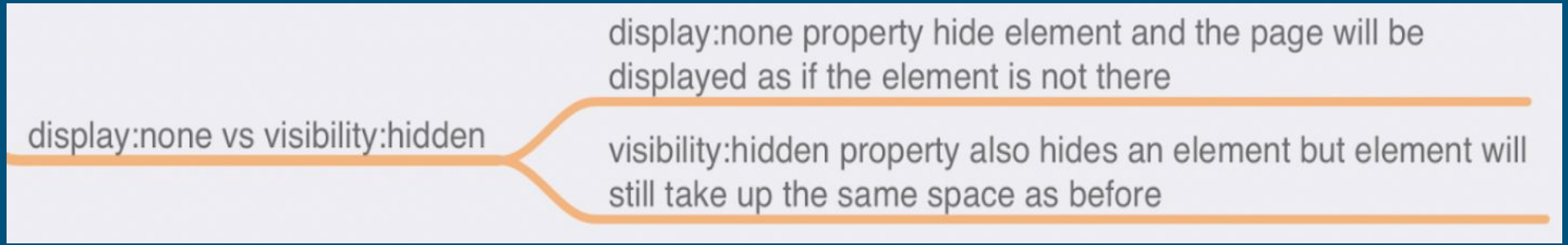
```
.element {
    display: inline-block;
}
```

Note: These elements appears next to each other and it's allowed to set the width and height

The visibility property

visibility:hidden

display:none



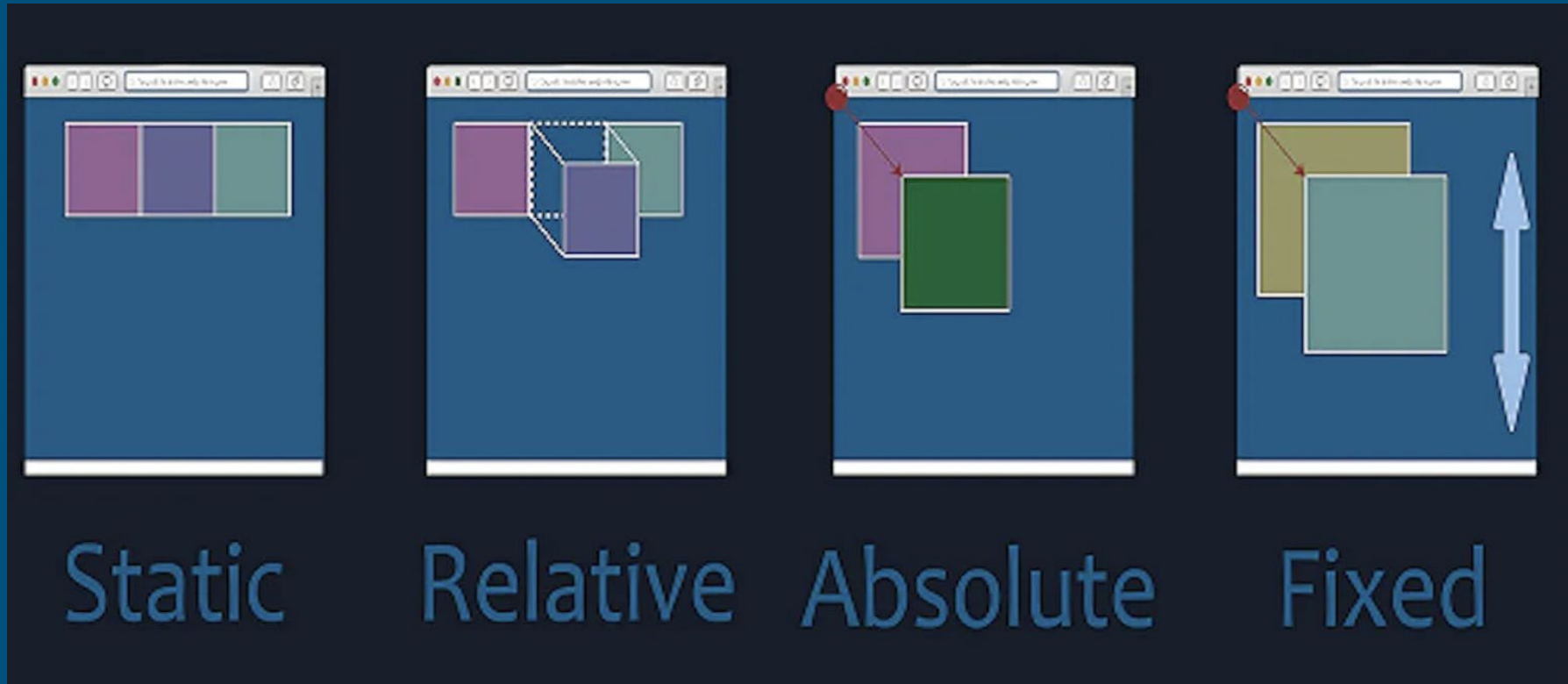
The ‘position’ Property

- static
- absolute
- relative
- fixed

The ‘position’ Property

The position property defines where we want to place an element on the page

position : static | absolute | relative | fixed

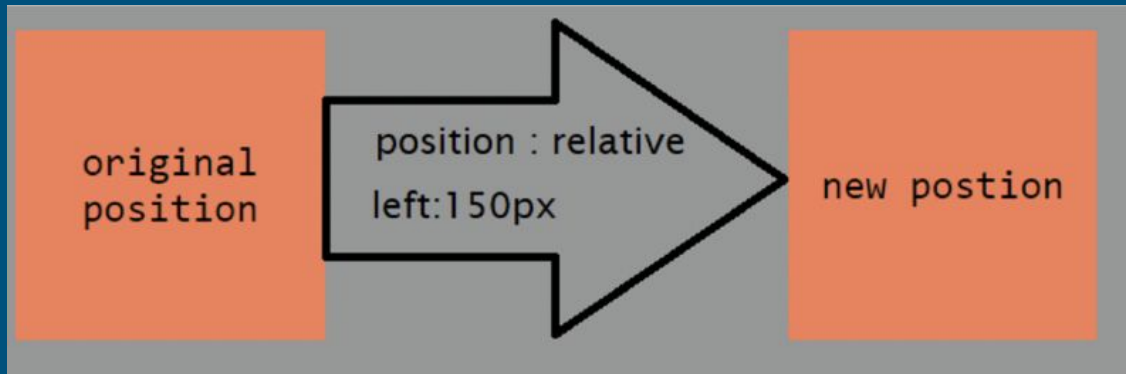


1. Static Position

- This is the default property of every element in the page
- Note: For the static positioned elements don’t have effect of additional properties like top, right, bottom, left and z-index

2. Relative Position

- Element with position: relative will be placed with respective to its original position in the page
- When position: relative applied to an element, It appears in the normal flow of the program document.
- Note: For the relative positioned elements we can set the properties like top, right, bottom, left and z-index



3. Absolute Position

- When “position: absolute” will be applied to an element, It is taken out of the normal flow of the document
- When position: absolute applied to an element, It positioned by default with respect to its parent element.

- Note: For the absolute positioned elements we can set the properties like top, right, bottom, left and z-index

4. Fixed Position

- A fixed element is rigid in terms of position. If we set the position as position: fixed; then it will always their no matter what we are doing on page i.e., it will stay on its fixed position even if you scroll the page.

- Note: For the fixed positioned elements we can set the properties like top, right, bottom, left and z-index

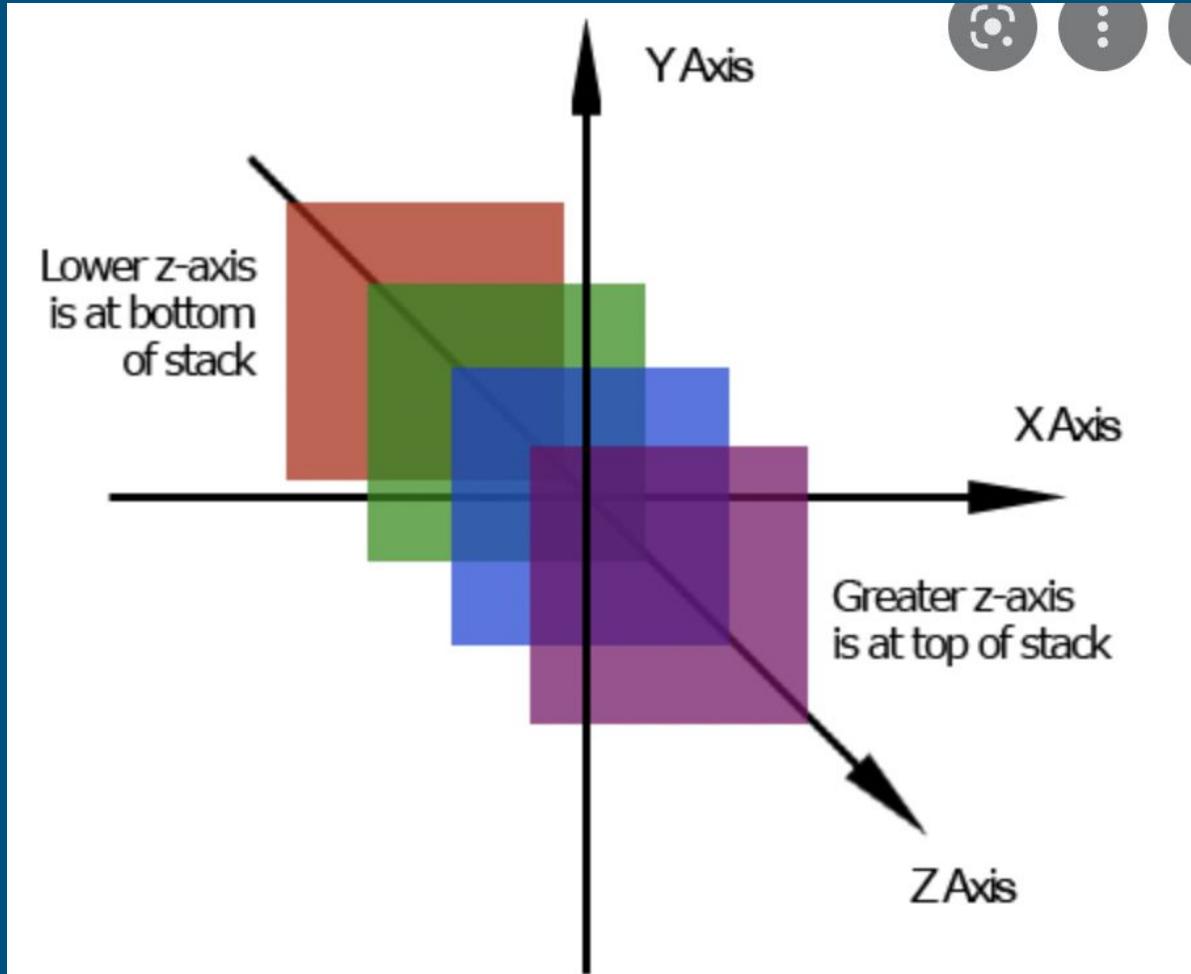
- The `top`, `right`, `bottom`, and `left`, properties specify how far away from the top/ right/ bottom/ left and element should be positioned.
  - By default all these properties have value *auto*, and the element is placed in a Static Position.
  - It accepts both negative and positive integer values and common units such as *px*, *rem*, *em*, *%* etc.

- `z-index` is used to specify the stack level, and layer elements.
  - The default value for `z-index` is 0, and will not work unless you apply position on the element (except static).
  - Elements with a larger `z-index` value overlaps elements with smaller `z-index` value.
  - It accepts positive and negative integers without any unit.

X, Y, Z - axis

```
.div1 {
        width: 150px;
        height: 150px;
        background-color: cadetblue;
        position: absolute;
        margin-top: 50px;
        z-index: -2;
      }

 .div2{
    width: 150px;
    height: 150px;
    background-color: rgb(1, 11, 11);
     z-index: 2;
  }
```



Z-index

z-index is used to specify the stack level, and layer elements.

- The default value for z-index is 0, and will not work unless you apply position on the element (except static).
- Elements with a larger z-index value overlaps elements with smaller z-index value.
- It accepts positive and negative integers without any unit.

This will not work for the default position that is static position, It will work for the absolute, relative, fixed and sticky positions

Overflow and Float Property

overflow: visible | hidden | scroll | auto

Note: The overflow property only works for block elements with a specified height

Float property: Specifies how an element should float

float: left | right | none