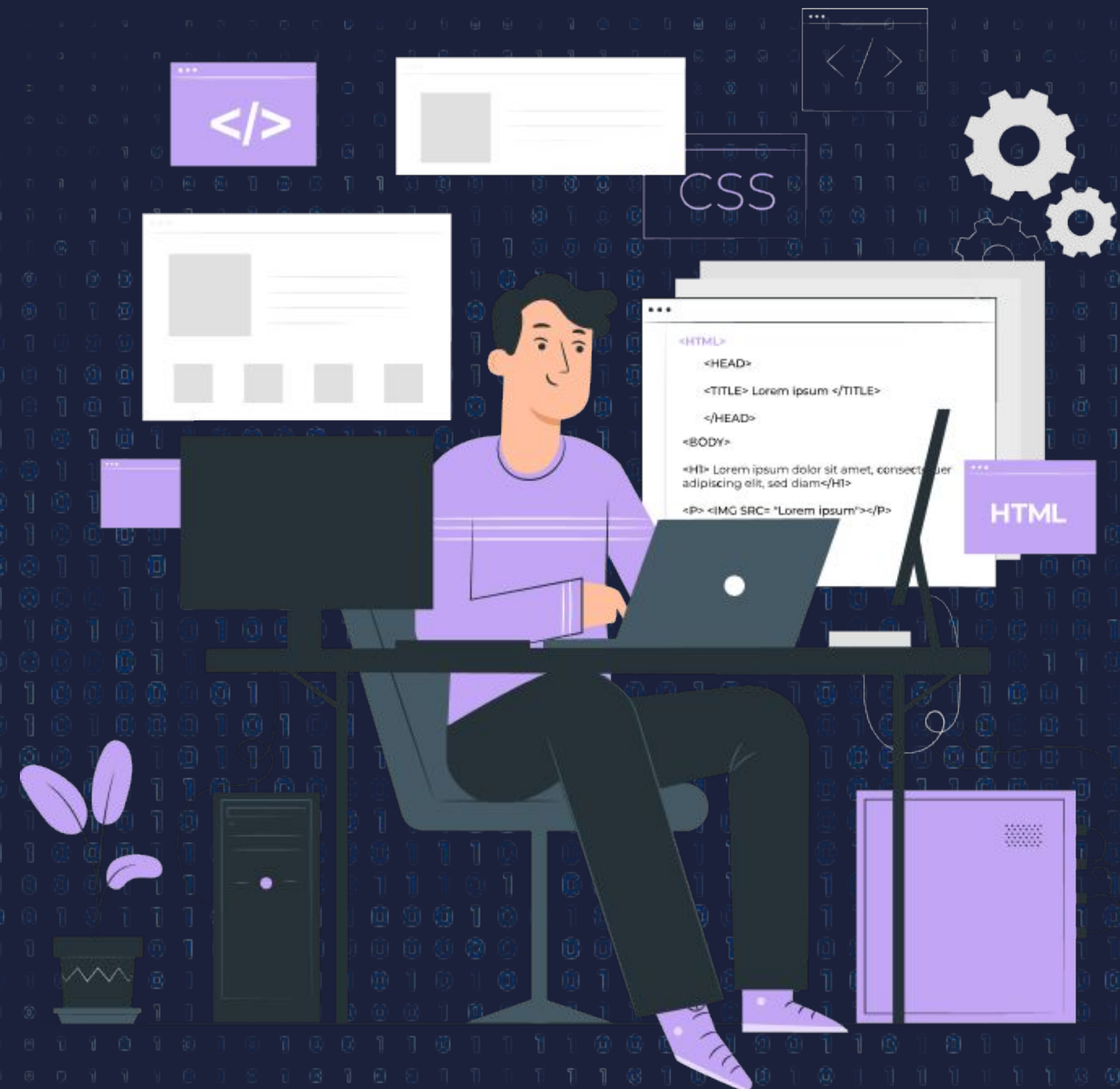




# CSS Positions (Part 1)





# Topics Covered

- What is CSS Position?
- Why to use CSS Position?
- Top, Bottom, Left and Right properties
- Static Positioning





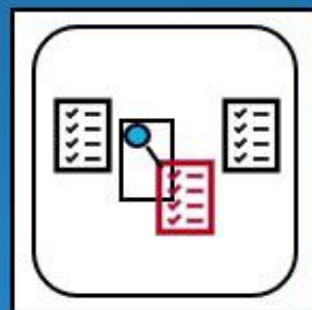
# What is CSS Position?

Process of controlling the **Placement of Elements** on a webpage.

Static



Relative



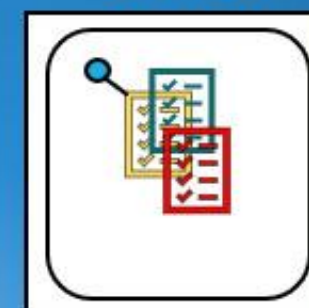
Fixed



Absolute



Z- index





# Position Properties – Top, Left, Right, Bottom

Used in conjunction with the **position** property to specify the exact offset location of an element on a webpage.

**Top** : Offset from top edge of the element.

**Bottom** : Offset from bottom edge of the element.

**Left** : Offset from left edge of the element.

**Right** : Offset from right edge of the element.





# Static Positioning

Default position for all HTML elements.

Elements are positioned based on the normal document flow,



# Example

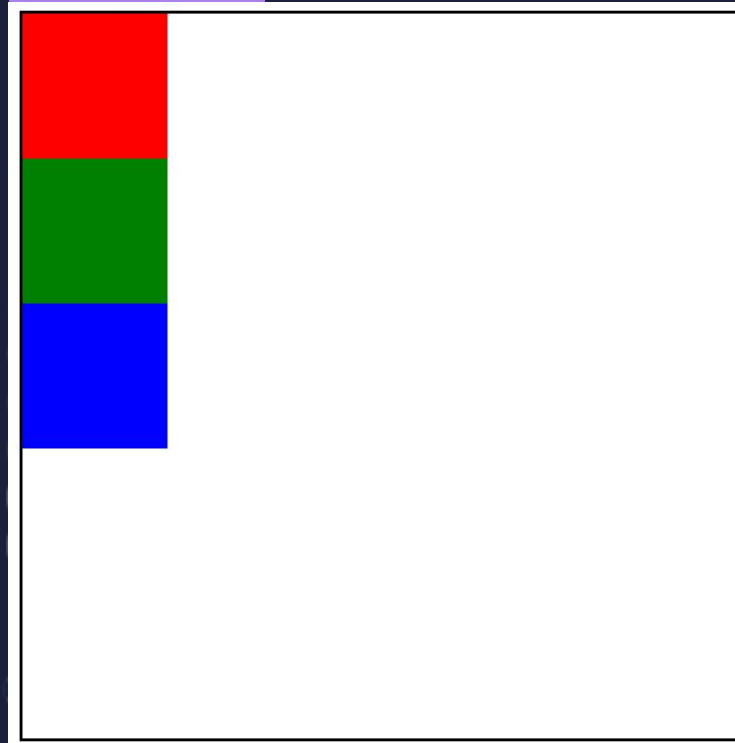
## HTML

```
<body>
<div class="container">
  <div class="box box1"></div>
  <div class="box box2"></div>
  <div class="box box3"></div>
</div>
</body>
```

## CSS

```
<style>
  .container {
    border: 2px solid black;
    height: 500px;
    width: 500px;
  }
  .box {
    width: 100px;
    height: 100px;
  }
  .box1 {
    background-color: red;
  }
  .box2 {
    background-color: green;
    position: static;
    top: 50px;
  }
  .box3 {
    background-color: blue;
  }
</style>
```

## Output







# Why to use CSS Position?

- Control over element position
- Positioning relative to other elements or viewport.
- Removing elements from document flow
- Create scroll effect
- Overlapping elements
- Accessibility



▶ THANK YOU ◀