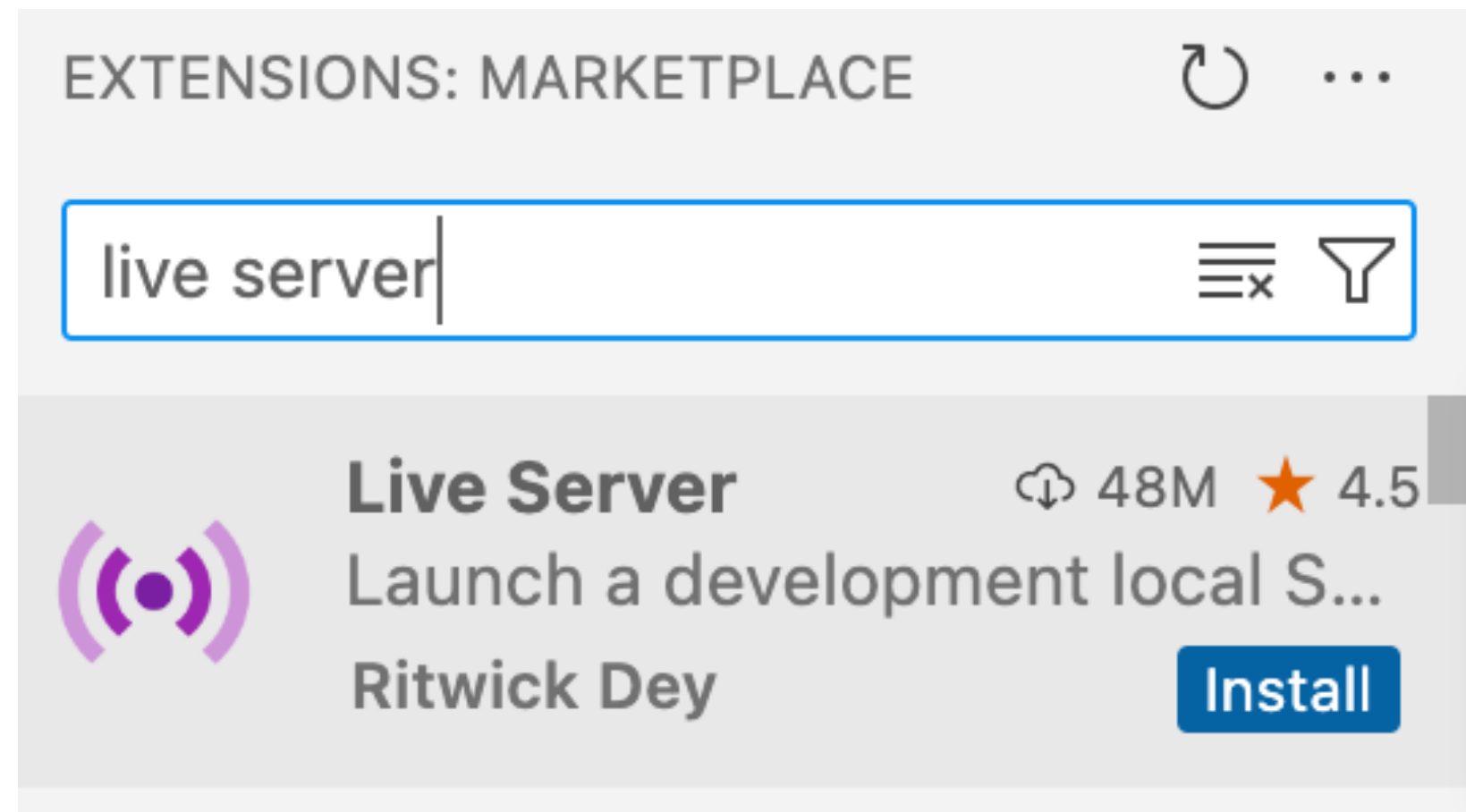
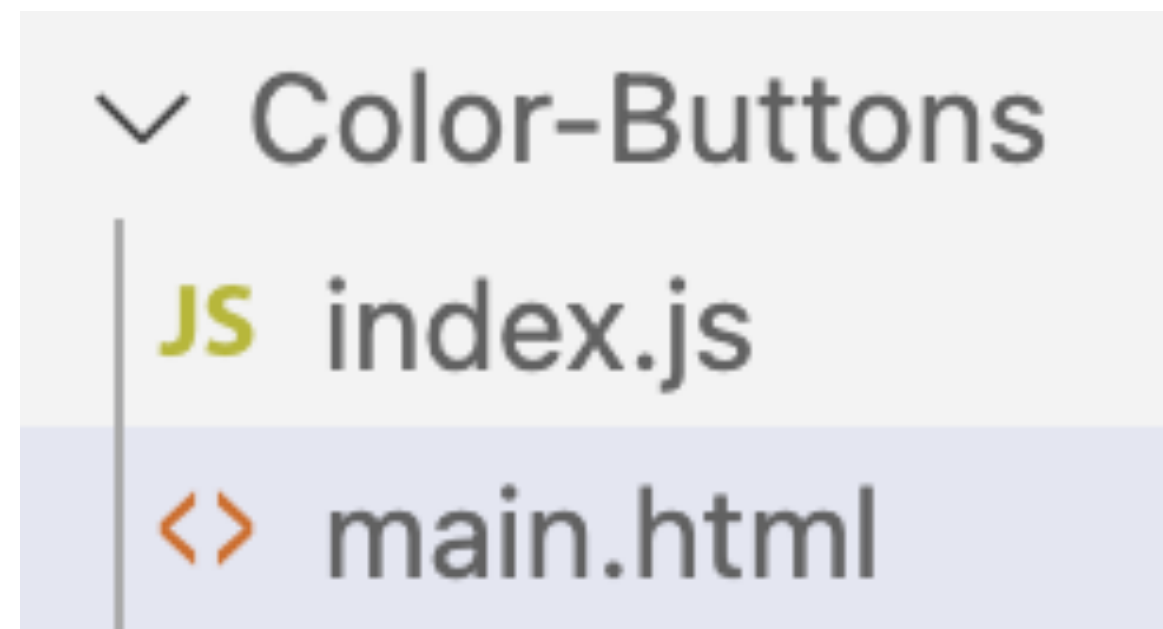


# Basic Learning on HTML, Style, and JavaScript

# Installation (make sure you Visual Code and Live Server extension installed)



Make sure you have visual source code installed and extension called "live server" also installed



Create an empty project like this and create two empty files inside the folder.



# Build the HTML Page

<> main.html ●

Color-Buttons > <> main.html

1

ht

- html
- html:5
- html:xml



Inside main.html file, use a tag `html:5`, and this will create a basic HTML page as shown below.

<> main.html ×

Color-Buttons > <> main.html > html

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Document</title>
7 </head>
8 <body>
9
10 </body>
11 </html>
```

# Update the page

<> main.html X

Color-Buttons > <> main.html > html

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5 |   <title>Color Button Palette</title>
6 </head>
7
8 <body>
9 |   <h3>Color Button Palette</h3>
10 </body>
11
12 </html>
```

Update the HTML code as shown with head and body tag to keep things simple.

Next press "Go Live" button at the bottom of your VS Code, this will open a new chrome web page for you as shown below.



## Color Button Palette

# Button and Style

<> main.html ×

Color-Buttons > <> main.html > html

```
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <title>Color Button Palette</title>
6      <style>
7          #my-red-button{
8              background-color: red;
9          }
10         #my-blue-button{
11             background-color: blue;
12         }
13         #my-orange-button{
14             background-color: orange;
15         }
16     </style>
17 </head>
18
19 <body>
20     <h3>Color Button Palette</h3>
21     <button id="my-red-button">Red</button>
22     <button id="my-blue-button">Blue</button>
23     <button id="my-orange-button">Orange</button>
24 </body>
25
26 </html>
```

Update the header with color button style.

Also, include button and id for the style inside the body tag of the html page.

We can use either class or id attribute to call a specific property of the element, for simplicity, we are using id.

## Color Button Palette



# Button and Style

Update the button colors, and add one extra button as random.

Also, lets' update button overall properties:

border-radius => this will make button round, more the pixel radius , rounder the button.

width => we increase the width to 100 pixels

height => we increase the height to 40 pixels

margin => the margin between the button

```
<head>
  <title>Color Button Palette</title>
  <style>
    #my-red-button{
      background-color: #lightpink;
    }
    #my-blue-button{
      background-color: #lightblue;
    }
    #my-orange-button{
      background-color: #orange;
    }
    button{
      border-radius: 5px;
      width: 100px;
      height: 40px;
      margin: 5px;
    }
  </style>
</head>

<body>
  <h3>Color Button Palette</h3>
  <button id="my-red-button">Red</button>
  <button id="my-blue-button">Blue</button>
  <button id="my-orange-button">Orange</button>
  <button>Random</button>
</body>
```

## Color Button Palette



# JavaScript and Properties

<> main.html JS index.js ×

Color-Buttons > JS index.js > ...

```
1  const mybodytag = document.getElementsByTagName("body")[0];
2  mybodytag.style.backgroundColor = "lightgrey"
```

```
<body>
  <h3>Color Button Palette</h3>
  <button id="my-red-button">Red</button>
  <button id="my-blue-button">Blue</button>
  <button id="my-orange-button">Orange</button>
  <button>Random</button>
  <script src="index.js"></script>
</body>
```

JavaScript file , index.js, will have our logical code. In this, we will have a custom tag for our HTML <body> tag. We want to change background color of our page.

For this, we first grab the "body" tag by tag name, and index it using [0]. However we only have one body tag in our HTML page, still we need to index it using [0].

And now we have our own body tag variable defined, using which we can modify the style of body, first lets change the background color to light grey.

Finally in our body, we include/link this index.js file.

## Color Button Palette



# JavaScript and Properties

JS index.js ×

Color-Buttons > JS index.js > ...

```
1  const mybodytag = document.getElementsByTagName("body")[0];
2
3  function setColor(name){
4      mybodytag.style.backgroundColor = name
5  }
6
7  setColor("lightgrey")
```

We can create a function called setColor, and when called, it will change or the background color of our web page.

For this, we pass name of color as an argument to the function, and whatever name we give, that will be become the background color.

## Color Button Palette





# JavaScript and Properties

```
<body>
  <h3>Color Button Palette</h3>
  <button id="my-red-button" onclick="setColor('lightpink')">Red</button>
  <button id="my-blue-button">Blue</button>
  <button id="my-orange-button">Orange</button>
  <button>Random</button>
  <script src="index.js"></script>
</body>
```

```
<body>
  <h3>Color Button Palette</h3>
  <button id="my-red-button" onclick="setColor('lightpink')">Red</button>
  <button id="my-blue-button" onclick="setColor('lightblue')">Blue</button>
  <button id="my-orange-button" onclick="setColor('orange')">Orange</button>
  <button>Random</button>
  <script src="index.js"></script>
</body>
```

Now we can call setColor function using button's onclick functionality. We are changing color to light pink when red button is clicked.

You can further modify the colors fedora all button, and play with this function of setColor.

Notice, how color is mentioned in single quotes. Reason is inside "" double quotes, we cannot have another double quotes. For this, we use ` ` or single quotes for the colors.

## Color Button Palette



## Color Button Palette



# JavaScript and Properties

```
function randomColor(){
  // R G B values of 0, 0 , 0 is black
  // R G B values of 255, 255 , 255 is white

  // we need three colors for our RGB values
  // we randomly generate them
  // Math.random() will give us a decimal value between 0 and 1
  // we multiply that random value with 255 because our color
  // values has to be between 0 and 255, and the we round
  // to get an absolute number value.

  const red = Math.round(Math.random() * 255)
  const blue = Math.round(Math.random() * 255)
  const green = Math.round(Math.random() * 255)

  // we use back ticks (``) , also called as tilde, to convert our above
  // randomly generated three number and convert them
  // into a RGB string. So back ticks is essentially a string manipulation trick
  const color = `rgb(${red}, ${green}, ${blue})`

  setColor(color)
}
```

Finally we need a randomColor function in our js file. What we want is if user clicks on random button, a random color is generated as background.

Red, Green, and Blue are primary colors whose values are between 0 and 255. If all three colors values are 0 we get a black color. And if all three of them is 255, we get the white color.

We randomly generate our own version of red, blue, and green as numbers between 0 and 255

Then we build a custom color variable using three values as shown.

We pass the custom color to our earlier setColor function.

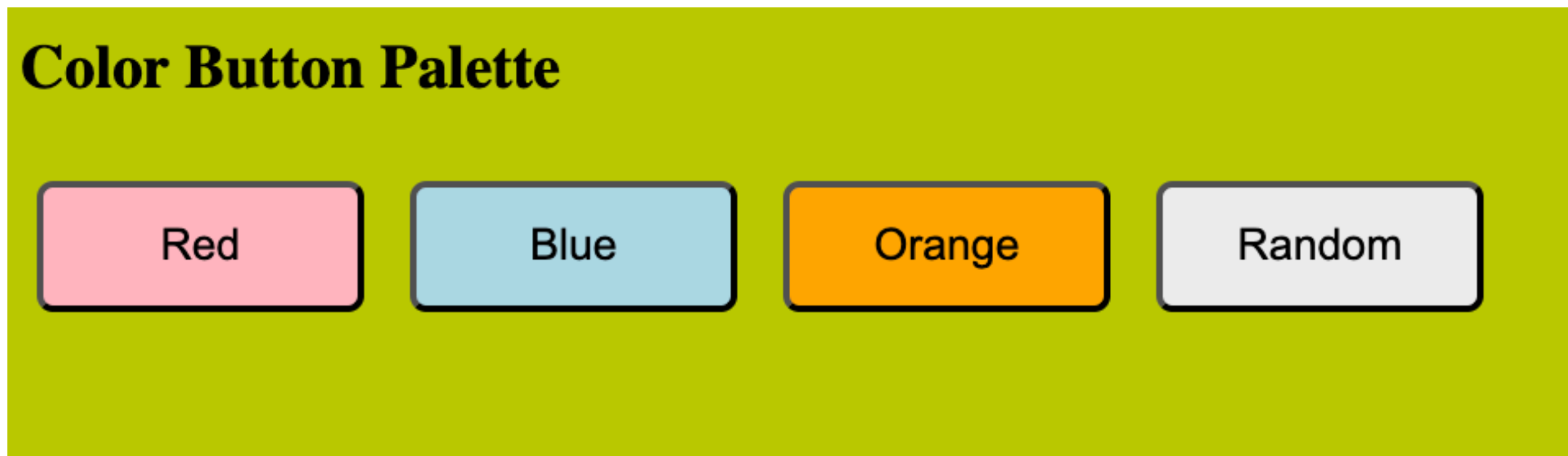
# JavaScript and Properties

```
<body>
  <h3>Color Button Palette</h3>
  <button id="my-red-button" onclick="setColor('lightpink')">Red</button>
  <button id="my-blue-button" onclick="setColor('lightblue')">Blue</button>
  <button id="my-orange-button" onclick="setColor('orange')">Orange</button>
  <button onclick="randomColor()">Random</button>
  <script src="index.js"></script>
</body>
```

Finally we call the randomColor function in our html page and test by clicking on random button.

Every time you click you will get different colors.

This is how you build web pages, links, use buttons, styles and interact with the JavaScript.



# Full Code

```
<> main.html ×
Color-Buttons > <> main.html > html > head > style > #my-orange-button
 2  <html lang="en">
 4  <head>
 6      <style>
 7          /
 9          }
10          #my-blue-button{
11              background-color: lightblue;
12          }
13          #my-orange-button{
14              background-color: orange;
15          }
16          button{
17              border-radius: 5px;
18              width: 100px;
19              height: 40px;
20              margin: 5px;
21          }
22      </style>
23  </head>
24
25  <body>
26      <h3>Color Button Palette</h3>
27      <button id="my-red-button" onclick="setColor('lightpink')">Red</button>
28      <button id="my-blue-button" onclick="setColor('lightblue')">Blue</button>
29      <button id="my-orange-button" onclick="setColor('orange')">Orange</button>
30      <button onclick="randomColor()">Random</button>
31      <script src="index.js"></script>
32  </body>
33
34  </html>
```

# Full Code

JS index.js ×

Color-Buttons > JS index.js > ...

```
1  const mybodytag = document.getElementsByTagName("body")[0];
2
3  function setColor(name){
4      mybodytag.style.backgroundColor = name
5  }
6
7  function randomColor(){
8      // R G B values of 0, 0 , 0 is black
9      // R G B values of 255, 255 , 255 is white
10
11     // we need three colors for our RGB values
12     // we randomly generate them
13     // Math.random() will give us a decimal value between 0 and 1
14     // we multiply that random value with 255 because our color
15     // values has to be between 0 and 255, and the we round
16     // to get an absolute number value.
17
18
19     const red = Math.round(Math.random() * 255)
20     const blue = Math.round(Math.random() * 255)
21     const green = Math.round(Math.random() * 255)
22
23     // we use back ticks (`) , also called as tilde, to convert our above
24     // randomly generated three number and convert them
25     // into a RGB string. So back ticks is essentially a string manipulation trick
26     const color = `rgb(${red}, ${green}, ${blue})`
27
28     setColor(color) //set color is called to change background color
29 }
30
31 setColor("lightgrey") // default
```

**Thank you !!!**