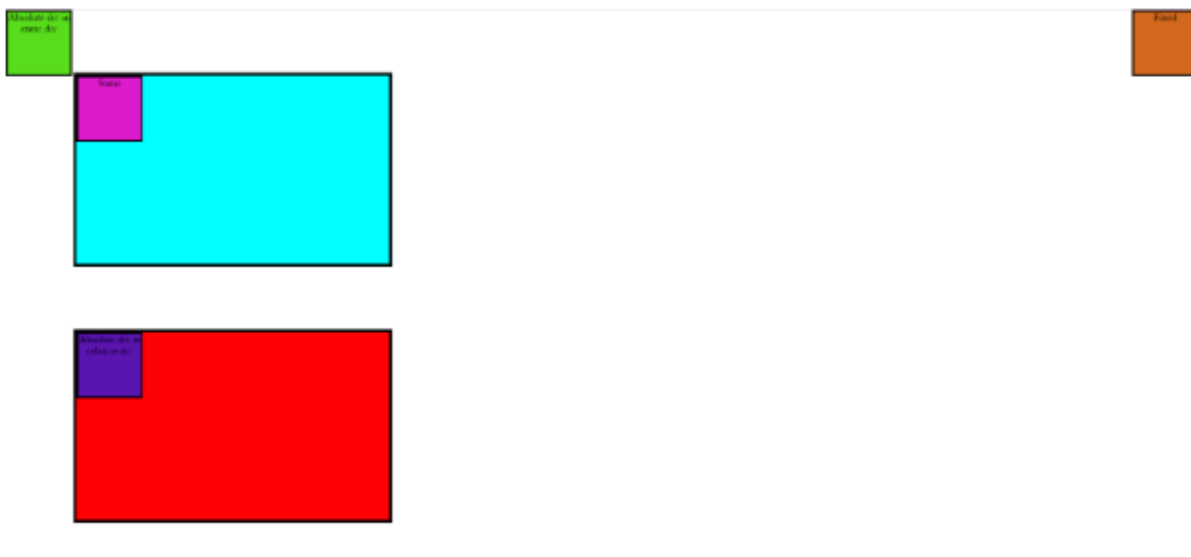## Code Challenge 1

Write proper code to generate the below given output :

1. Create a parent div (position should be static, provide some top and left margin) with the following child divs :
    a. Div with position : fixed; (Give top and right as 0px)
        i. Add more content or zoom in the page so that the page can be scrolled then observe the position of this div on scrolling.
    b. Div with position : absolute; (top and left should be 0px)
    c. Div with position : static; (Give some top/left margin)
2. Create another parent div (position should be relative, provide some top and left margin) with the following child divs :
    a. Div with position absolute (top and left should be 0px)

Expected Output :

Solution to Code Challenge 2: -

Code Challenge 2

```html
<!DOCTYPE html>
<html>
<head>
    <style>
        .parentDiv{
            background-color: aqua;
            width: 500px;
```

```css
        height: 300px;
        border: solid 5px black;
        position: static;
        margin: 100px;
        text-align: center;
    }
    .parentDiv1{
        background-color: rgb(255, 0, 4);
        width: 500px;
        height: 300px;
        border: solid 5px black;
        position: relative;
        margin-left: 100px;
        text-align: center;
    }
    .child1{
         background-color: chocolate;
         width: 100px;
         height: 100px;
         border-style: solid;
         position:fixed;
         top:0;
         right:0;

    }
    .child2{
        background-color: rgb(89, 220, 28);
        width: 100px;
        height: 100px;
        border-style: solid;
        position: absolute;
        top: 0px;
        left: 0px;
    }
    .child3{
        background-color: rgb(220, 28, 204);
        width: 100px;
        height: 100px;
        border-style: solid;
        position: static;
```

```
                top: 400px;
                left: 400px;
            }
            .child4{
                background-color: rgb(90, 21, 174);
                width: 100px;
                height: 100px;
                border-style: solid;
                position: absolute;
                top: 0px;
                left: 0px;
            }
        </style>
</head>
<body>
<div class="parentDiv">
    <div class="child1">Fixed</div>
    <div class="child2">Absolute div in static div</div>
    <div class="child3">Static</div>
</div>
<div class="parentDiv1">
    <div class="child4">Absolute div in relative div</div>
</div>
</body>
</html>
```
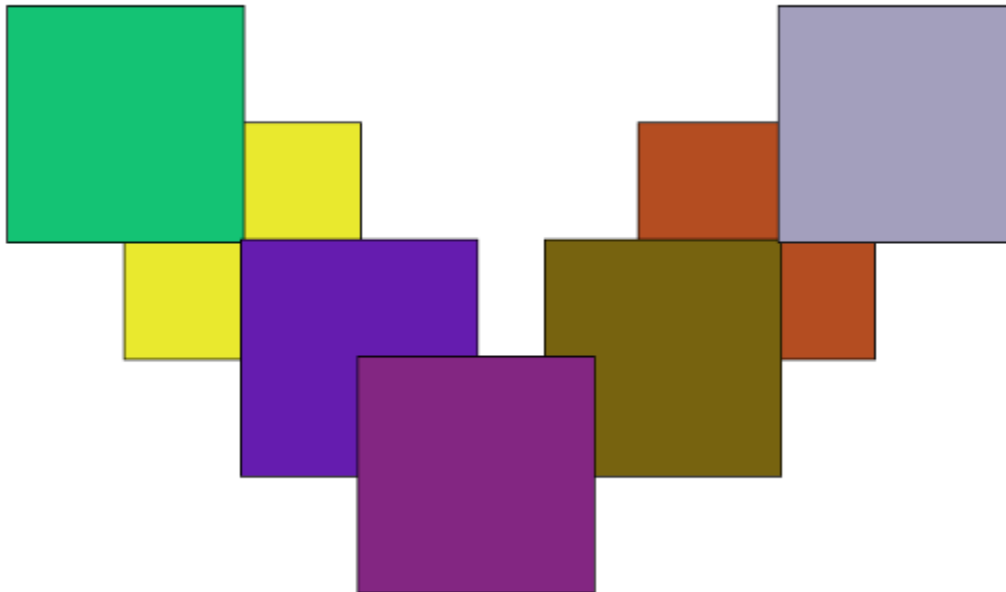
Code Challenge 2

Write proper code to generate the below-given output :

1. Create the below given layout, where the 3 divs overlap each other (using z-index)

Solution to Code Challenge 2: -

```html
<!DOCTYPE html>
<html>

<head>
    <title> CSS z-index</title>
    <style>
.child1 {
  border: 1px solid #000;
  width: 100px;
  height: 100px;
  background-color: rgb(20, 195, 116);
  color: #000;
  position: absolute;
  left: 50px;
  top: 50px;
  z-index: 4;
```

```css
}
.child2 {
  border: 1px solid #000;
  width: 100px;
  height: 100px;
  background-color: rgb(233, 233, 46);
  color: #000;
  position: absolute;
  left: 100px;
  top: 100px;
 z-index: 2;
}
.child3 {
  border: 1px solid #000;
  width: 100px;
  height: 100px;
  background-color: rgb(101, 28, 175);
  color: #000;
  position: absolute;
  left: 150px;
  top: 150px;
  z-index: 4;
}
.child4 {
  border: 1px solid #000;
  width: 100px;
  height: 100px;
  background-color: rgb(131, 38, 130);
  color: #000;
  position: absolute;
  left: 200px;
  top: 200px;
 z-index: 5;

}
.child5 {
  border: 1px solid #000;
  width: 100px;
  height: 100px;
  background-color: rgb(119, 99, 15);
```

```css
  color: #000;
  position: absolute;
  left: 280px;
  top: 150px;
  z-index: 4;


}
.child6 {
  border: 1px solid #000;
  width: 100px;
  height: 100px;
  background-color: rgb(180, 77, 33);
  color: #000;
  position: absolute;
  left: 320px;
  top: 100px;
 z-index: 2;
}
.child7 {
  border: 1px solid #000;
  width: 100px;
  height: 100px;
  background-color: rgb(163, 159, 189);
  color: #000;
  position: absolute;
  left: 380px;
  top: 50px;
  z-index: 4;


}
    </style>
</head>
<body>
        <div class="child1"></div>
        <div class="child2"></div>
        <div class="child3"></div>
        <div class="child4"></div>
        <div class="child5"></div>
        <div class="child6"></div>
        <div class="child7"></div>
```
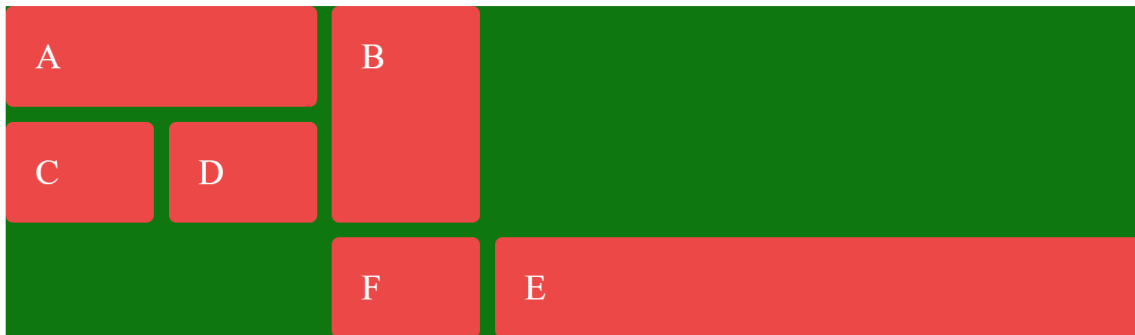
```html
</body>
</html>
```

Code Challenge 3

Write proper code to generate the below-given output :

1. Create the below given layout using Grids.

Solution to Code Challenge 3-

```html
<!DOCTYPE html>
<html>
<head>
<style>
body {margin: 40px;}

.wrapper {
    display: grid;
    grid-gap: 10px;
    grid-template-columns: 100px 100px 100px;
    background-color: rgb(15, 119, 15);
    color: rgb(235, 255, 79);
  }

  .box {
    background-color: rgb(237, 72, 72);
    color: #fff;
    border-radius: 5px;
    padding: 20px;
    font-size: 150%;
  }

  .a {
    grid-column: 1 / 3;
```
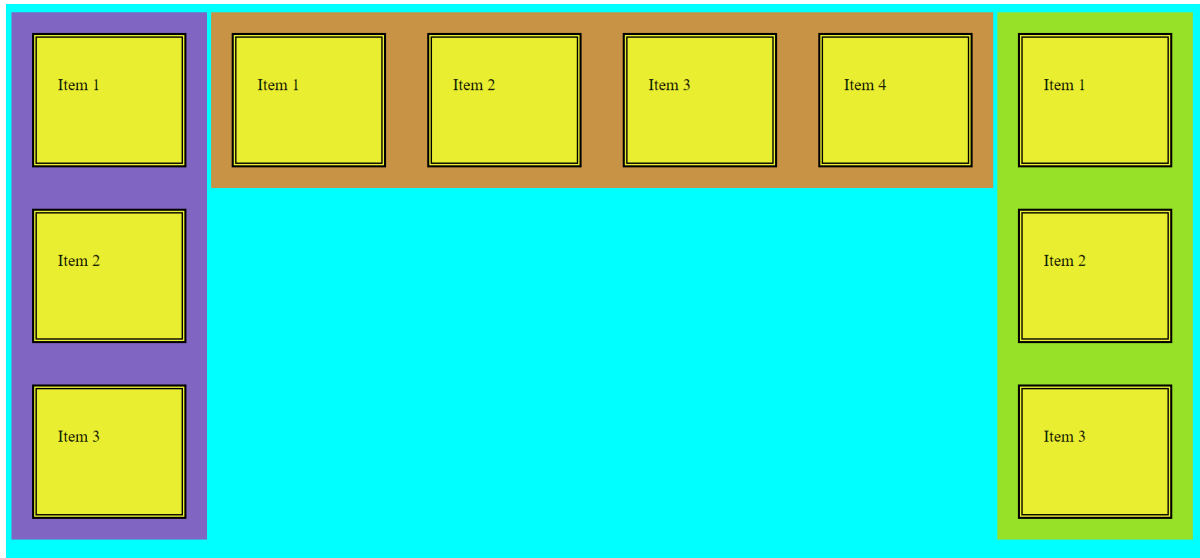
```
    grid-row: 1;
  }
  .b {
    grid-column: 3 ;
    grid-row: 1 / 3;
  }
  .c {
    grid-column: 1 ;
    grid-row: 2 ;
  }
  .d {
    grid-column: 2;
    grid-row: 2;
  }
  .e {
    grid-column: 4;
    grid-row: 3;
  }
  .f {
    grid-column: 3;
    grid-row: 3;
  }
    </style>
</head>
<body>
    <div class="wrapper">
        <div class="box a">A</div>
        <div class="box b">B</div>
        <div class="box c">C</div>
        <div class="box d">D</div>
        <div class="box e">E</div>
        <div class="box f">F</div>
    </div>
</body>
</html>
```

Code Challenge 4

Write proper code to generate the below-given output :

1. Create the below-given layout using Flex properties.

Solution to Code Challenge 4: -

```html
<html>
    <head>
        <title> Flex </title>
        <style>
            body
            {
                background-color: aqua;
            }
            .container {
                display: inline-flex;
                background-color: rgb(129, 101, 194);
                flex-direction: row;
                flex-wrap: wrap;
                justify-content: flex-start;
                flex-direction: column;

            }
            .container2 {
                display: inline-flex;
                background-color: rgb(200, 147, 68);
```

```css
            flex-direction: row;
            flex-wrap: wrap;
            justify-content: center;


        }
        .container3 {
            display: inline-flex;
            background-color: rgb(151, 226, 40);
            flex-direction: row;
            flex-wrap: wrap;
            justify-content: flex-end;


        }
        .item{
            background-color: rgb(233, 239, 48);
            border-width: 5px;
            border-style: double;
            width: 100px;
            height: 80px;
            padding: 20px;
            margin: 20px;
        }

    </style>
</head>
<body>
    <div class="container">
        <div class="item">
            <p>Item 1</p>
        </div>
        <div class="item">
            <p>Item 2</p>
        </div>
        <div class="item">
            <p>Item 3</p>
        </div>

    </div>
    <div class="container2">
        <div class="item">
            <p>Item 1</p>
        </div>
        <div class="item">
```

```html
        <p>Item 2</p>
    </div>
    <div class="item">
        <p>Item 3</p>
    </div>
    <div class="item">
        <p>Item 4</p>
    </div>
</div>
<div class="container3">
    <div class="item">
        <p>Item 1</p>
    </div>
    <div class="item">
        <p>Item 2</p>
    </div>
    <div class="item">
        <p>Item 3</p>
    </div>

</div>
</body>
</html>
```