

CSS Gradients

Html part:

1.Linear Gradient

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/css2_1.css">
</head>
<body>
  <div class="box1">box1 Linear Gradient by default</div>
  <hr>
  <div class="box2">box2 Linear Gradient =to bottom</div>
  <hr>
  <div class="box3">box3 Linear Gradient =to top</div>
  <hr>
  <div class="box4">box4 Linear Gradient =to left</div>
  <hr>
  <div class="box5">box5 Linear Gradient =to right</div>
  <hr>
  <div class="box6">box6 Linear Gradient = to bottom right</div>
  <hr>
  <div class="box7">box7 Linear Gradient = to top right</div>
  <hr>
  <div class="box8">box8 Linear Gradient = using tranparancy</div>
</body>
</html>
```

CSS part :

```
.box1{
  border: 1px solid black;
  height: 100px;
  width: 100px;
  background-image: linear-gradient(red,green);
}

.box2{
  border: 1px solid black;
  height: 100px;
  width: 100px;
  background-image: linear-gradient(to bottom,red,blue);
}

.box3{
  border: 1px solid black;
  height: 100px;
  width: 100px;
  background-image: linear-gradient(to top,red,blue);
}

.box4{
  border: 1px solid black;
  height: 100px;
  width: 100px;
  background-image: linear-gradient(to left,red,blue);
}

.box5{
  border: 1px solid black;
  height: 100px;
  width: 100px;
  background-image: linear-gradient(to right,red,blue);
}

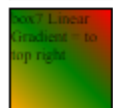
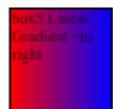
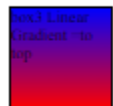
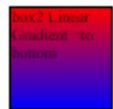
.box6{
  border: 1px solid black;
  height: 100px;
  width: 100px;
  background-image: linear-gradient(to bottom right,orange,green,red);
}
```

```

.box7{
  border: 1px solid black;
  height: 100px;
  width: 100px;
  background-image: linear-gradient(to top right,orange,green,red);
}
.box8{
  border: 1px solid black;
  height: 100px;
  width: 100px;
  background-image: linear-
gradient(to bottom,rgba(255,0,0,0),rgba(0,0,255,1));
}

```

Output :



2.Radial Gradient

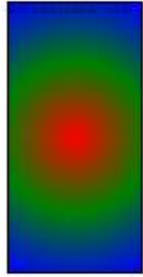
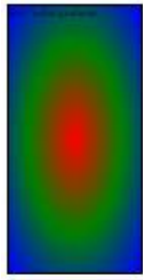
Html part :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/html2_2.css">
</head>
<body>
  <div class="box1">box1 radial gradients </div>
  <hr>
  <div class="box2">box2 radial gradients =circle</div>
  <hr>
  <div class="box3">box2 radial gradients </div>
</body>
</html>
```

CSS part :

```
.box1{
  border: 1px solid black;
  height: 400px;
  width: 200px;
  background-image: radial-gradient(red,green,blue);
}
.box2{
  border: 1px solid black;
  height: 400px;
  width: 200px;
  background-image: radial-gradient(circle,red,green,blue);
}
.box3{
  border: 1px solid black;
  height: 400px;
  width: 200px;
  background-image: radial-gradient(red 5%,yellow 10%,#33CAFF 50%);
}
```

Output :



3. Conic Gradient (Home work)

Shadow

Html :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/html2_2.css">
</head>
<body>
  <div class="box4">this is box4 SHADOW</div>
  <hr>
  <div class="box5">this is box5 SHADOW</div>
  <hr>
  <div class="box6">this is box6 SHADOW</div>
  <hr>
  <div class="box7">this is box7 SHADOW</div>
  <hr>
  <div class="box8">this is box8 SHADOW</div>
  <hr>
</body>
</html>
```

Css :

```
.shadow-effect{
  text-shadow: 3px 3px 3px red,6px 6px 6px green;
}
.box4{
  background-color: #33CAFF;
  color: red;
  height: 200px;
  width: 200px;
  box-shadow: 10px 10px;
}
.box5{
  background-color: #33CAFF;
  color: red;
  height: 200px;
  width: 200px;
  box-shadow: 10px 10px green;
}
```

```
.box6{
  background-color: #33CAFF;
  color: red;
  height: 200px;
  width: 200px;
  box-shadow: 10px 10px 10px green;
}
.box7{
  background-color: #33CAFF;
  color: red;
  height: 200px;
  width: 200px;
  box-shadow: 10px 10px 10px green,15px 15px 15px orangered;
}
.box8{
  background-color: #33CAFF;
  color: red;
  height: 200px;
  width: 200px;
  box-shadow: -10px -10px 10px violet,15px 15px 15px orangered;
}
```

Output :

Hello Trisha

this is box4 SHADOW

this is box5 SHADOW

this is box6 SHADOW

this is box7 SHADOW

this is box8 SHADOW

CSS Dimension properties

1.relative

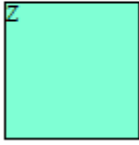
Html :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/html3_1.css">
</head>
<body>
  <div class="box">Z</div>
</body>
</html>
```

Css :

```
div{
  width: 100px;
  height: 100px;
  background-color: aquamarine;
  border:1px solid black;
  display: inline-block;
}
.box{
  position: relative;
  top: 100px;
  left: 50px;
}
```

Output :



Position: relative work karta hai top, bottom, left aur right pe.

Html

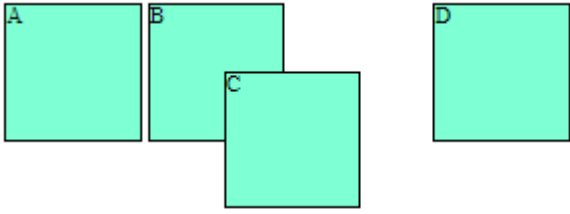
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/html3_1.css">
</head>
<body>

  <div class="box" id="box1">A</div>
  <div class="box" id="box2">B</div>
  <div class="box" id="box3">C</div>
  <div class="box" id="box4">D</div>

</body>
</html>
```

Css

```
div{
  width: 100px;
  height: 100px;
  background-color: aquamarine;
  border:1px solid black;
  display: inline-block;
}
#box3{
  position: relative;
  top: 50px;
  right: 50px;
}
```



Fixed

Html

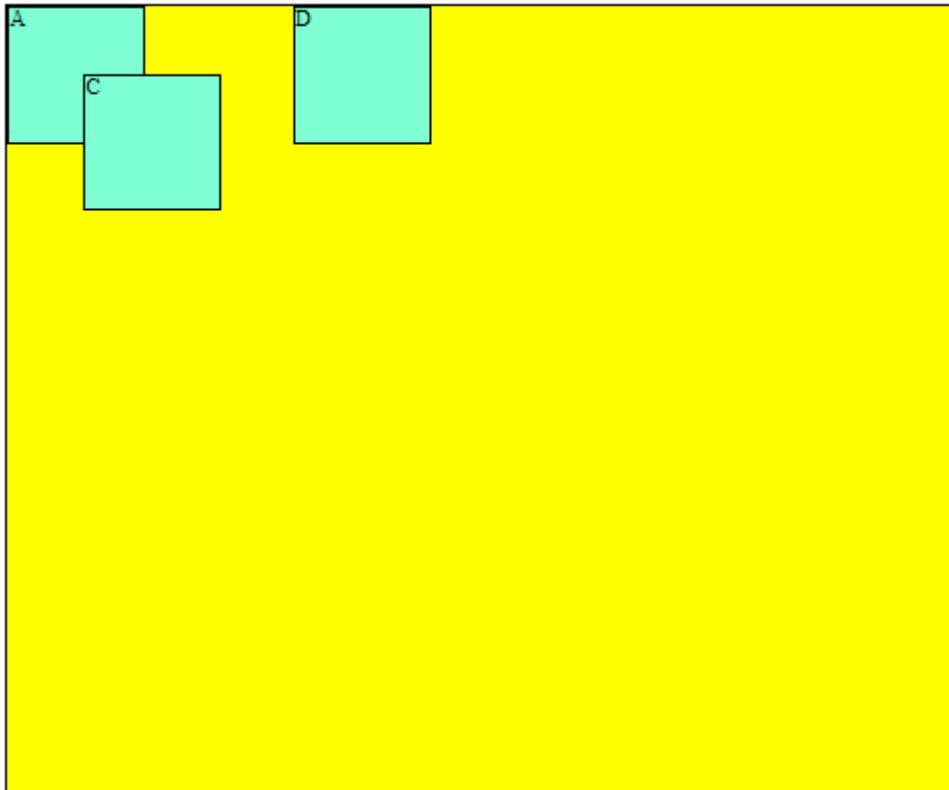
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/html3_1.css">
</head>
<body>

  <div class="wrapper">
    <div class="box" id="box1">A</div>
    <div class="box" id="box2">B</div>
    <div class="box" id="box3">C</div>
    <div class="box" id="box4">D</div>
  </div>
</body>
</html>
```

Css

```
div{
  width: 100px;
  height: 100px;
  background-color: aquamarine;
  border:1px solid black;
  display: inline-block;
}
#box3{
  position: relative;
  top: 50px;
  right: 50px;
}
#box1{
  position:fixed;
}
.wrapper{
  background-color: yellow;
  height: 20000px;
  width: 100vw;
}
```

Output



Fixed mai A ki position fixed ho jayegi

CSS Basics – III [HD]

CSS FlexBox

Html :

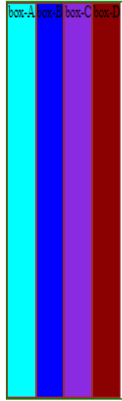
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/html3_2.css">
</head>
<body>
  <div class="container">
    <div class="box" id="boxA">box-A</div>
    <div class="box" id="boxB">box-B</div>
    <div class="box" id="boxC">box-C</div>
    <div class="box" id="boxD">box-D</div>
  </div>
</body>
</html>
```

Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
}
.box{
  border:1px solid brown;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color:blue;
}
#boxC{
  background-color:blueviolet;
}
#boxD{
```

```
background-color:darkred;
}
```

Output :



//Html code part is same but change in css part

Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-direction: column;
}
.box{
  border:1px solid brown;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color:blue;
}
#boxC{
  background-color:blueviolet;
}
#boxD{
  background-color:darkred;
}
```


Output

A diagram showing a vertical stack of four colored rectangular layers. The layers are labeled on the left side as follows: 'box-A' for the top cyan layer, 'box-B' for the blue layer, 'box-C' for the purple layer, and 'box-D' for the bottom dark red layer. The layers are separated by thin white lines.

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-direction: column-reverse;
}
.box{
  border:1px solid brown;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color:blue;
}
#boxC{
  background-color:blueviolet;
}
#boxD{
  background-color:darkred;
}
```

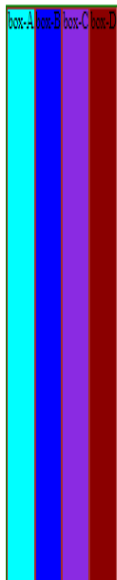
Output



Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-direction: row;
}
.box{
  border:1px solid brown;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color:blue;
}
#boxC{
  background-color:blueviolet;
}
#boxD{
  background-color:darkred;
}
```

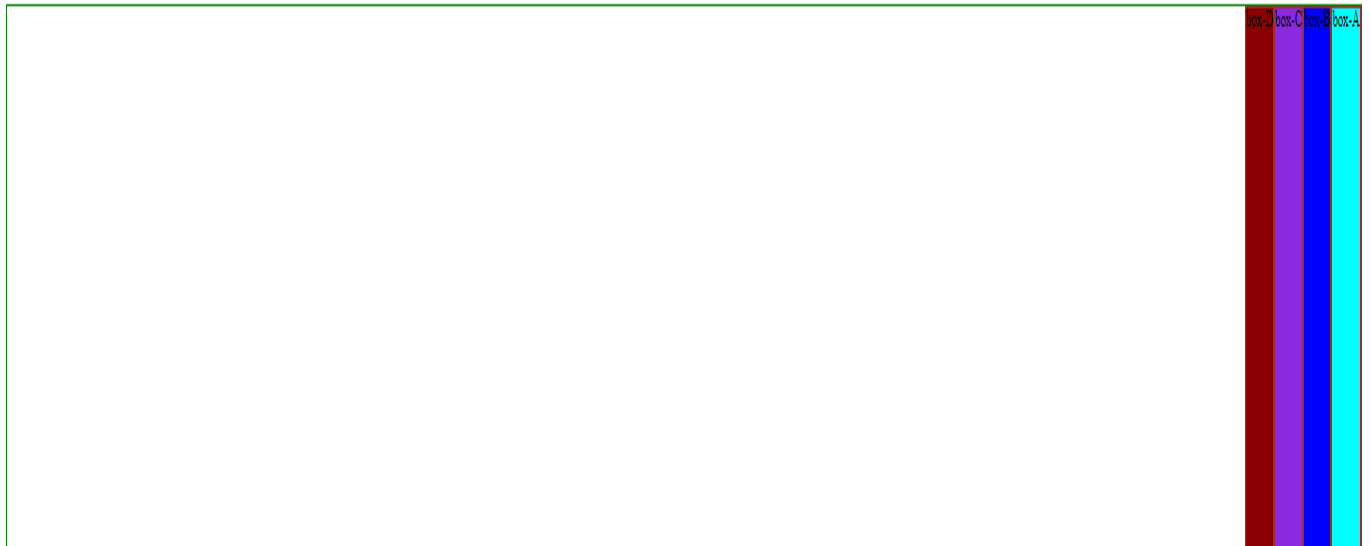
Output



Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-direction: row-reverse;
}
.box{
  border:1px solid brown;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color:blue;
}
#boxC{
  background-color:blueviolet;
}
#boxD{
  background-color:darkred;
}
```

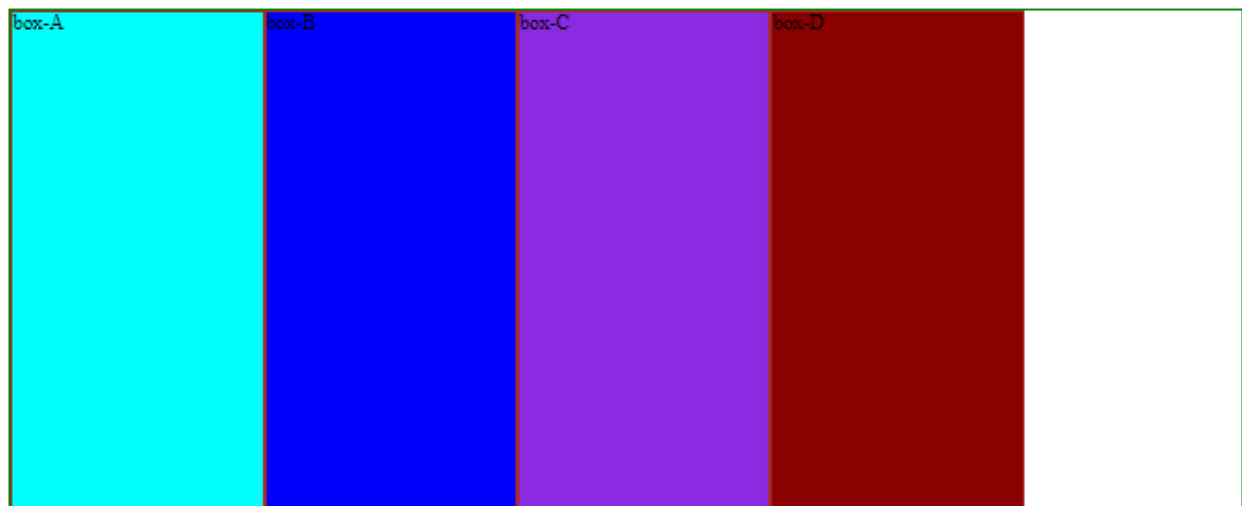
Output

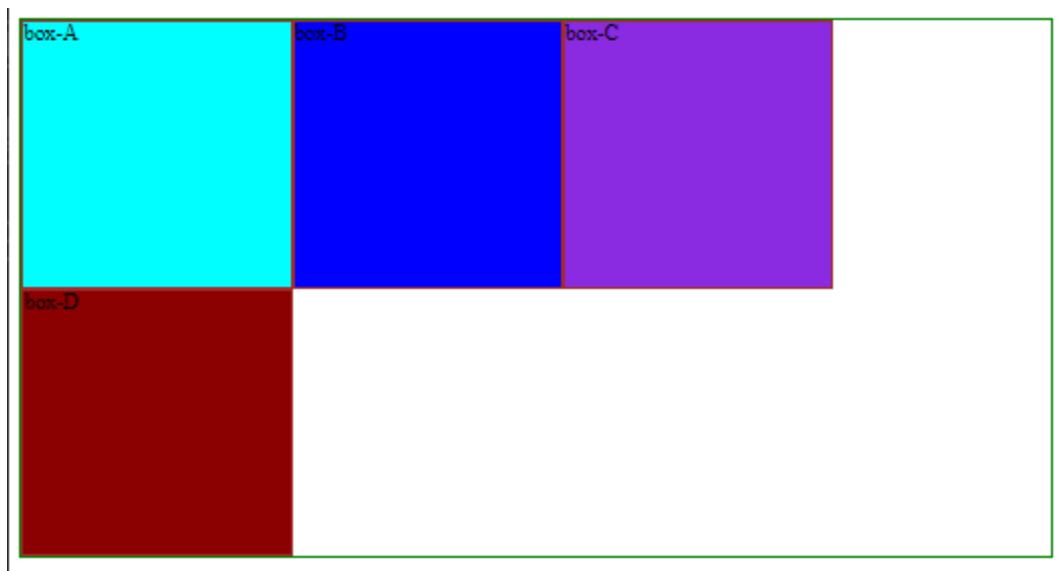


Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
}
.box{
  border:1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color:blue;
}
#boxC{
  background-color:blueviolet;
}
#boxD{
  background-color:darkred;
}
```

Output

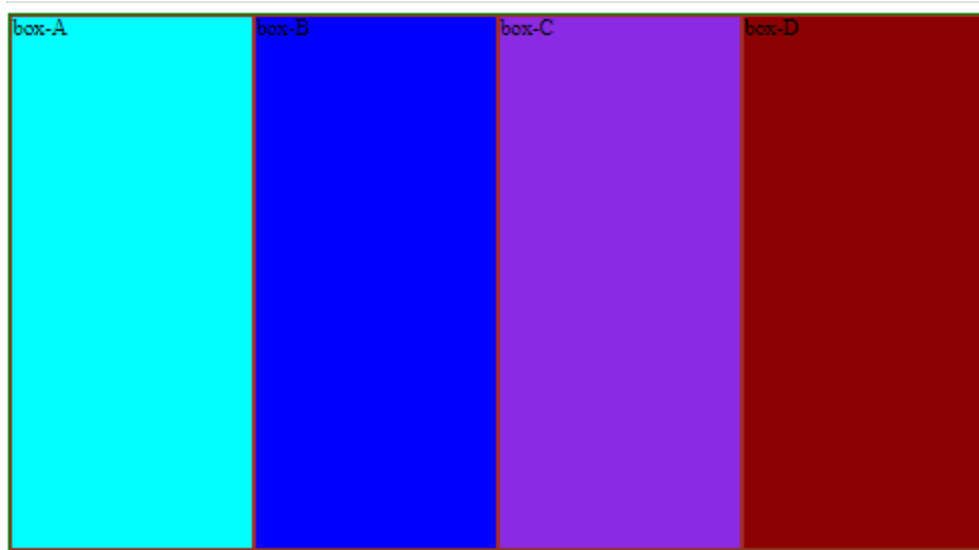




Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  /* flex-direction: row-reverse; */
  flex-wrap: nowrap;
}
.box{
  border:1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color:blue;
}
#boxC{
  background-color:blueviolet;
}
#boxD{
  background-color:darkred;
}
```

Output



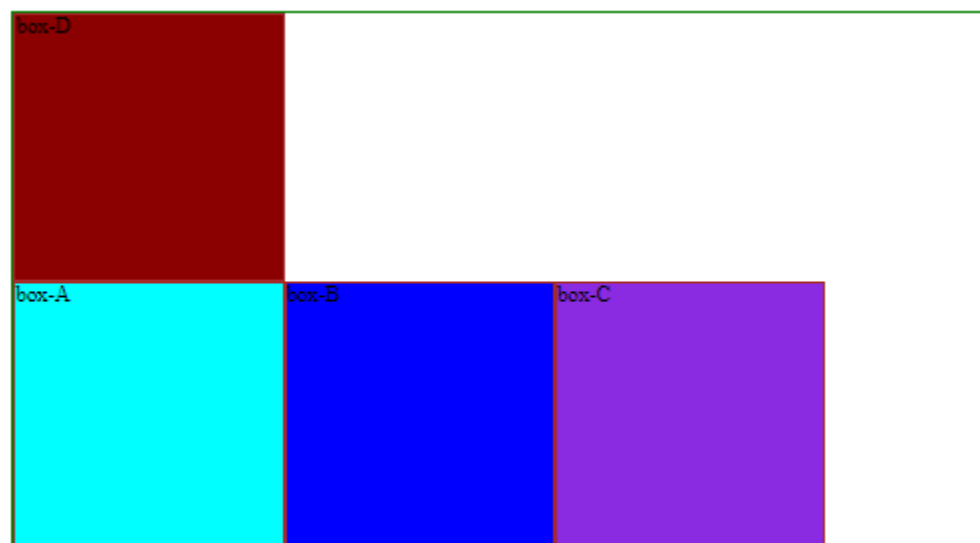
flex item ki width ko preserve karne ke liye flex-wrap ka use karte hai



```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  /* flex-direction: row-reverse; */
  flex-wrap: wrap-reverse;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```

Output :





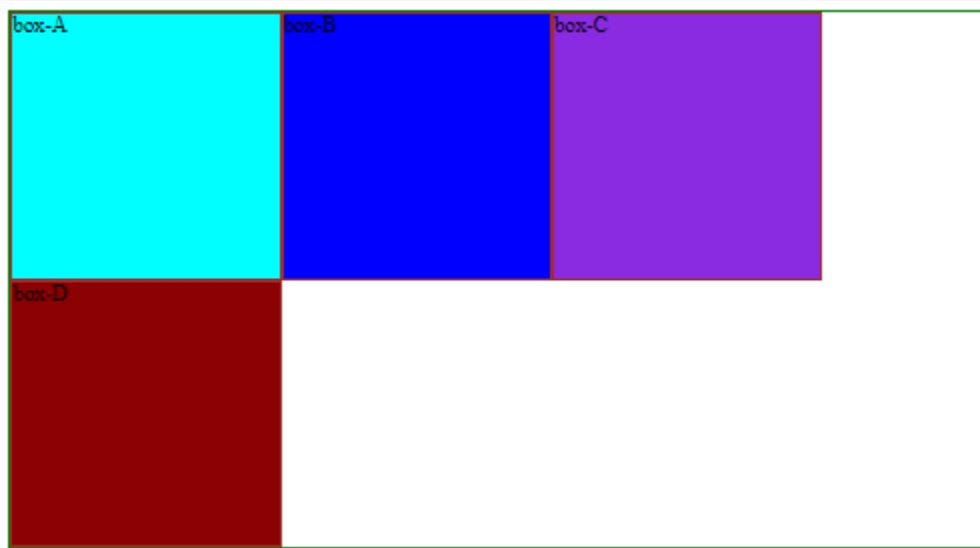
Using flex-flow is combination of `flex direction` and `flex-wrap`

```
flex-flow: row wrap;
```

Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-flow: row wrap;
}
.box{
  border:1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color:blue;
}
#boxC{
  background-color:blueviolet;
}
#boxD{
  background-color:darkred;
}
```

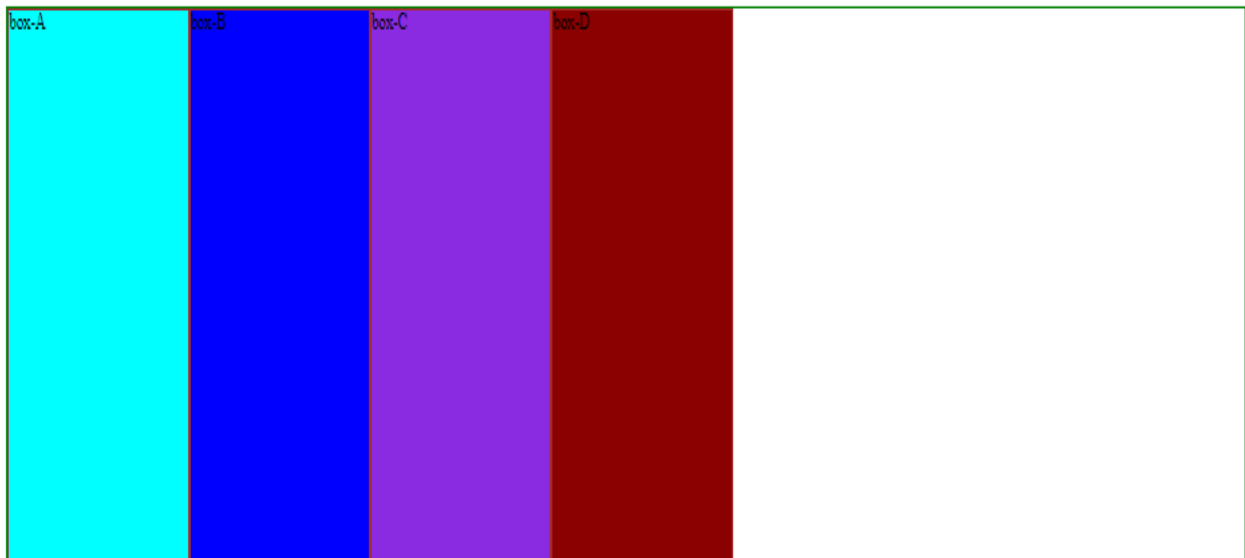
Output :



Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: start;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```

Output

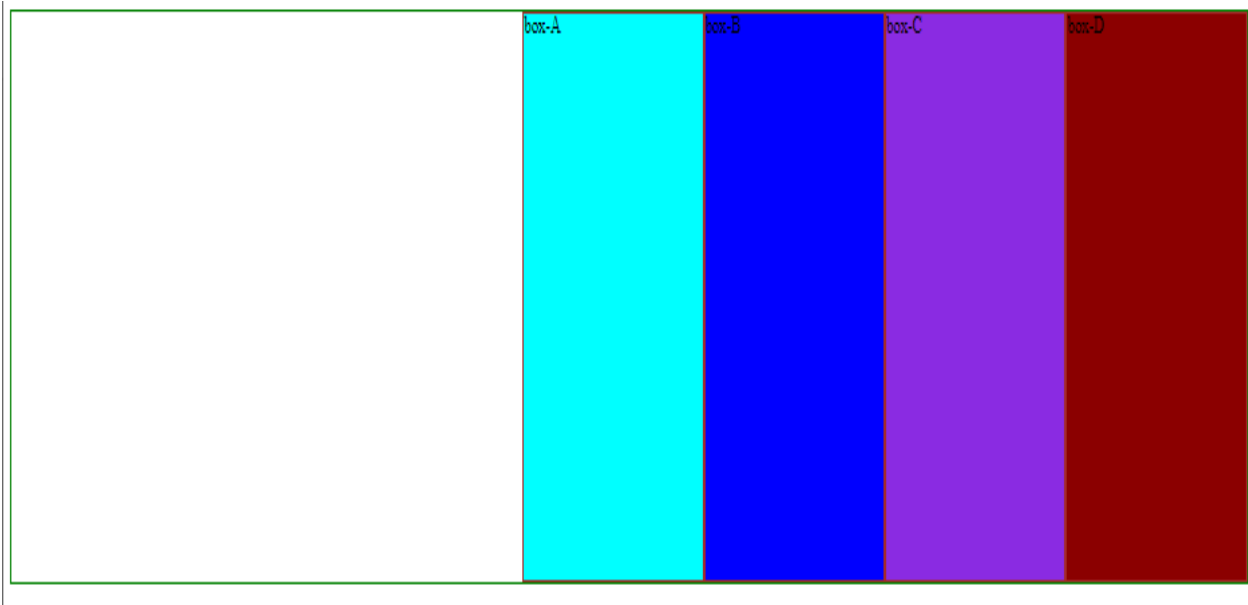


Note: aap ke main axis ke upar items ko align karta hai ,by default main axis horizontal hota hai.

Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: end;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```

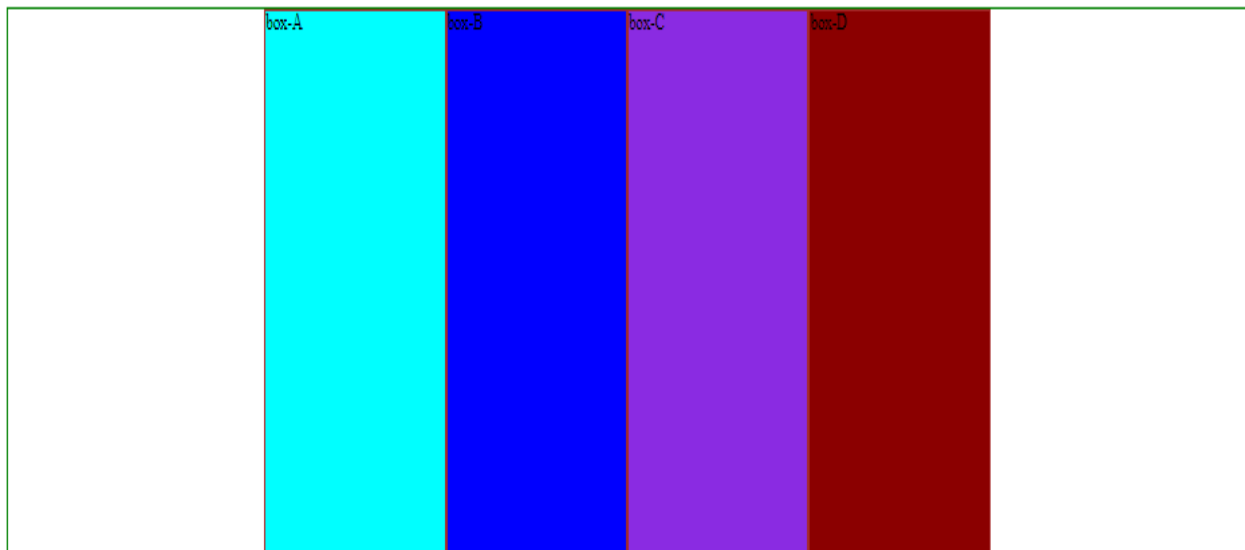
Output



Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```

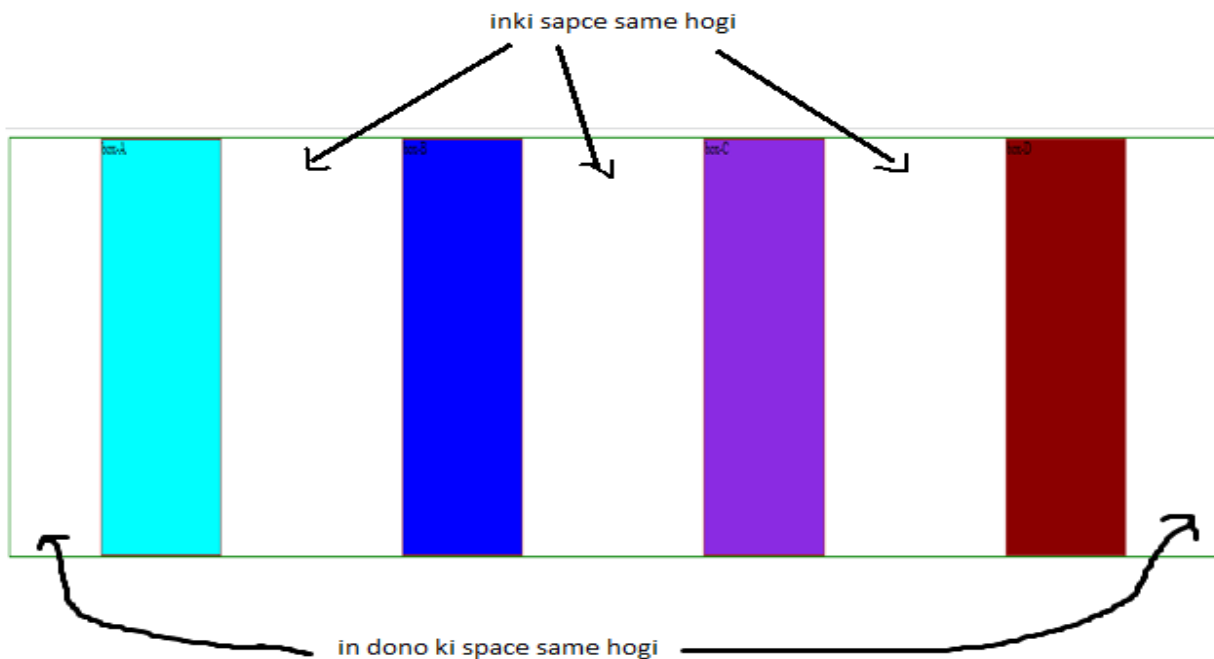
Output



Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: space-around;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```

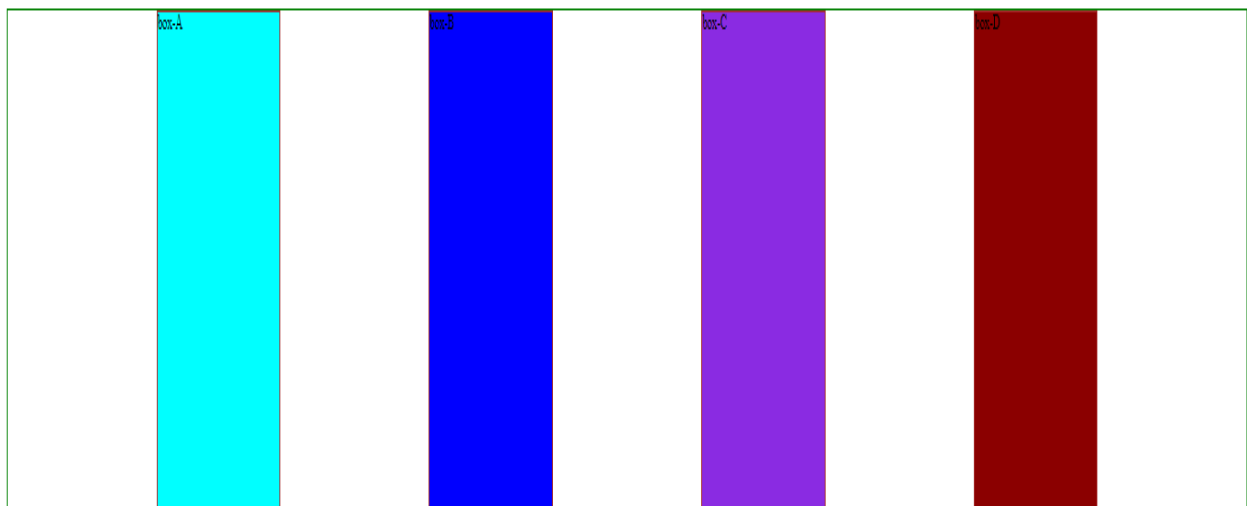
Output



Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: space-evenly;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```

Output

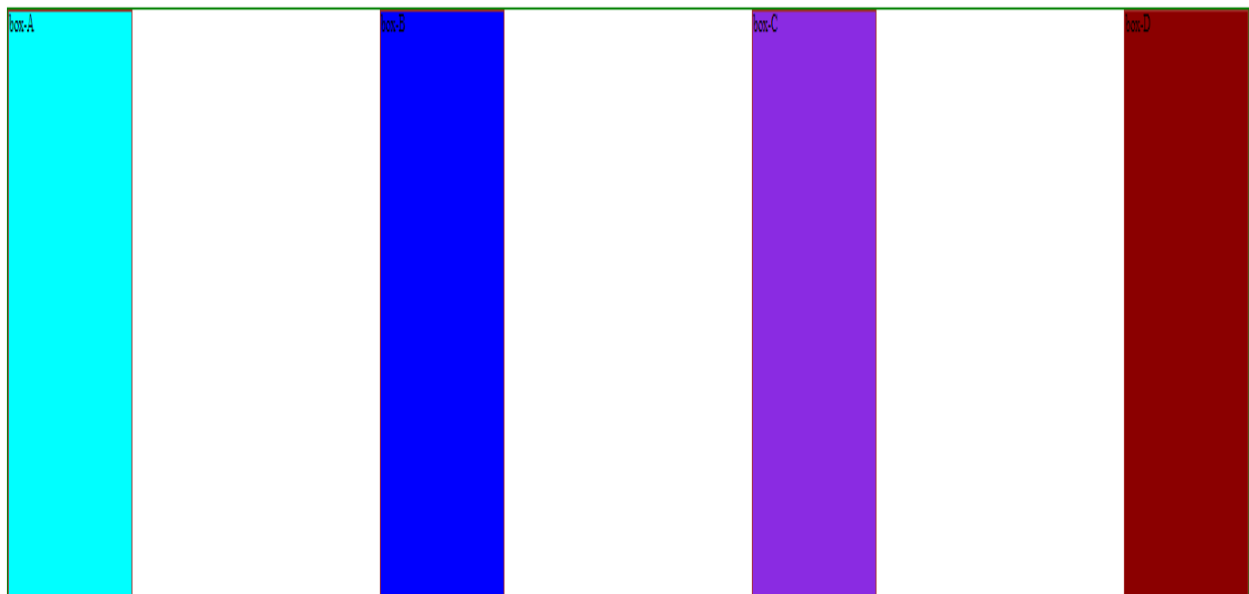


Note : evenly mai sabki space same hogi

Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: space-between;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```

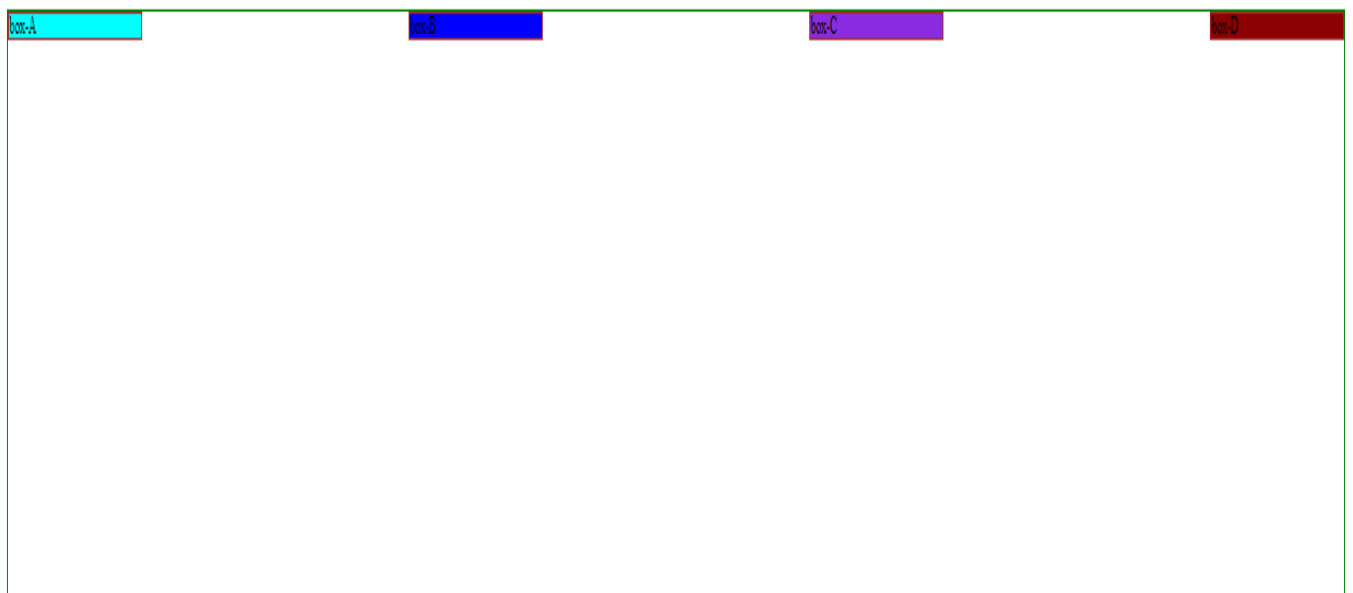
Output



Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: space-between;
  align-items: flex-start;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```

Output



Note : align-items mai cross axis ke uppar properties alignment hoti hai...

vertical axis

Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: space-between;
  align-items: flex-end;
}
.box{
  border:1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color:blue;
}
#boxC{
  background-color:blueviolet;
}
#boxD{
  background-color:darkred;
}
```

Output



Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: space-between;
  align-items: center;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```

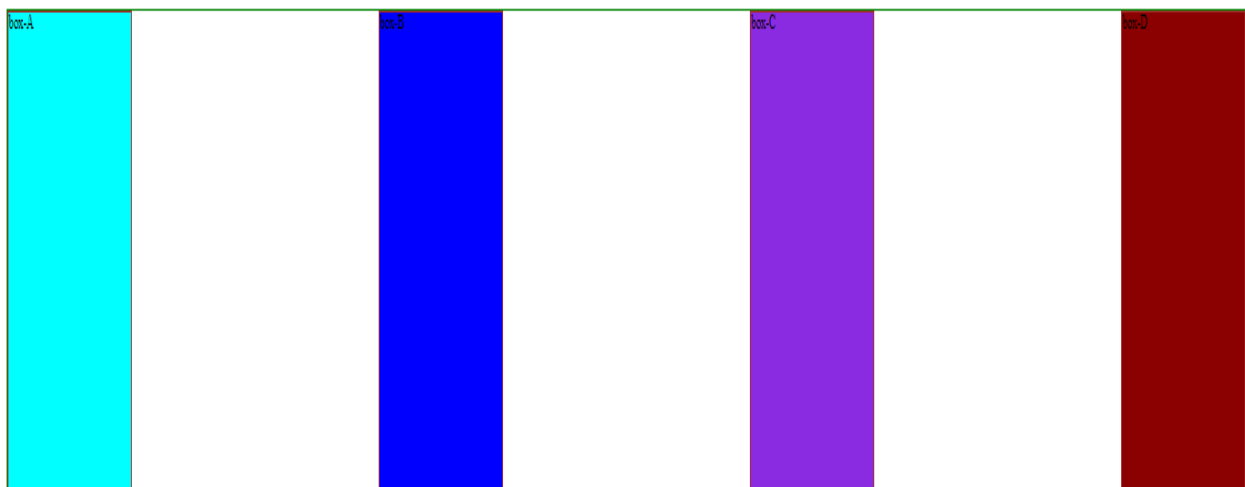
Output



Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: space-between;
  align-items: stretch;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```

Output



Note :align-items ki by default value stretch hoti hai..

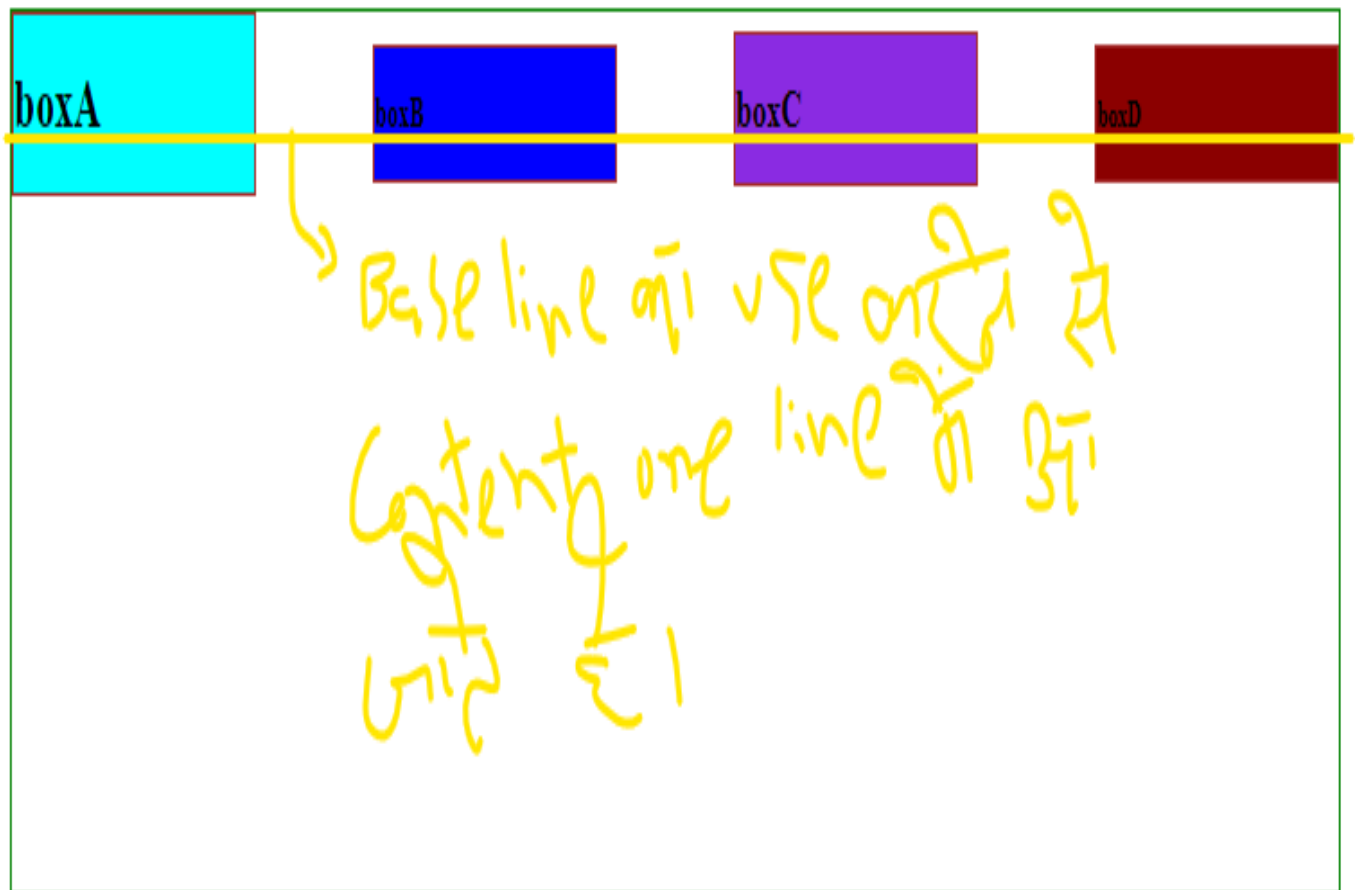
Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/html3_2.css">
</head>
<body>
  <div class="container">
    <div class="box" id="boxA"><h1>boxA</h1></div>
    <div class="box" id="boxB"><h3>boxB</h3></div>
    <div class="box" id="boxC"><h2>boxC</h2></div>
    <div class="box" id="boxD"><h4>boxD</h4></div>
  </div>
</body>
</html>
```

Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: space-between;
  align-items: baseline;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```

Output

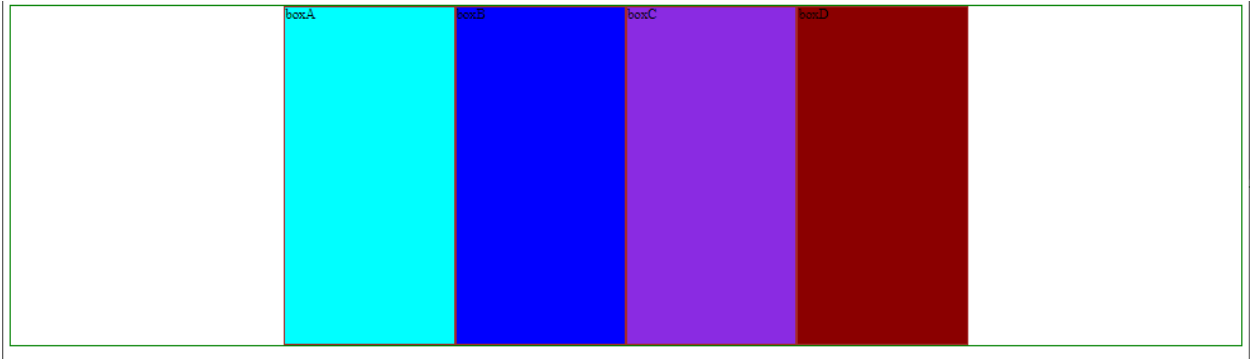


Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/html3_2.css">
</head>
<body>
  <div class="container">
    <div class="box" id="boxA">boxA</div>
    <div class="box" id="boxB">boxB</div>
    <div class="box" id="boxC">boxC</div>
    <div class="box" id="boxD">boxD</div>
  </div>
</body>
</html>
```

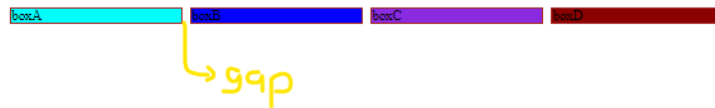
Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```

Css part

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: center;
  gap: 10px;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```



Css

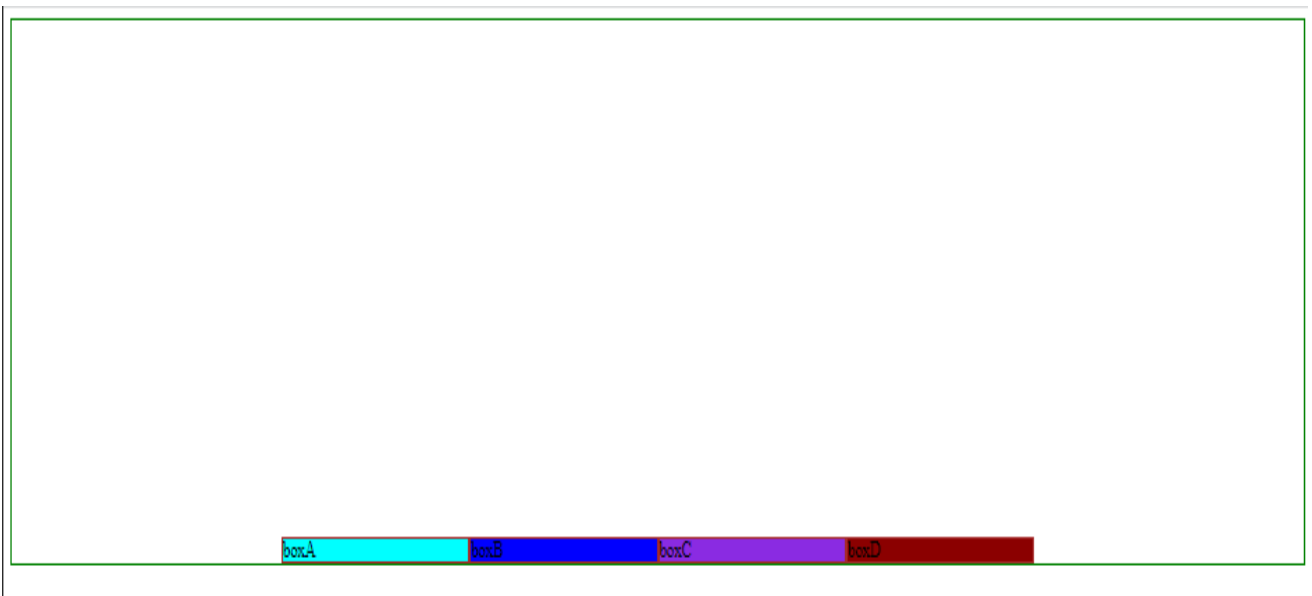
```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: center;
  align-content: flex-start;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```



Note : align-content multiple row ke gap mai space ko handle karta hai...

Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: center;
  align-content: flex-end;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```



Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: center;
  align-content: flex-start;
}
.box{
  border:1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color:blue;
}
#boxC{
  background-color:blueviolet;
}
#boxD{
  background-color:darkred;
}
```

boxA	boxB
boxC	boxD

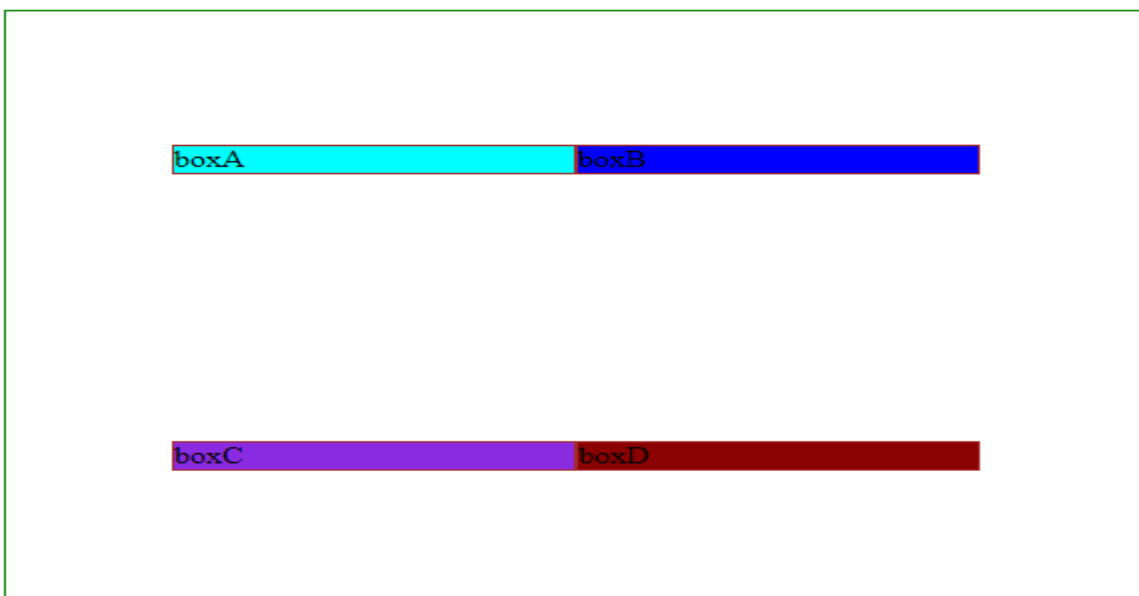
Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: center;
  align-content: flex-end;
}
.box{
  border:1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color:blue;
}
#boxC{
  background-color:blueviolet;
}
#boxD{
  background-color:darkred;
}
```



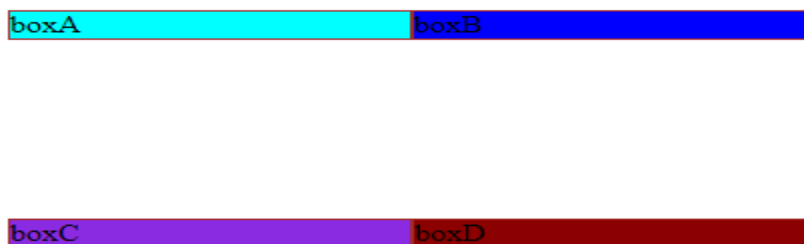
Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: center;
  align-content: space-around;
}
.box{
  border:1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color:blue;
}
#boxC{
  background-color:blueviolet;
}
#boxD{
  background-color:darkred;
}
```



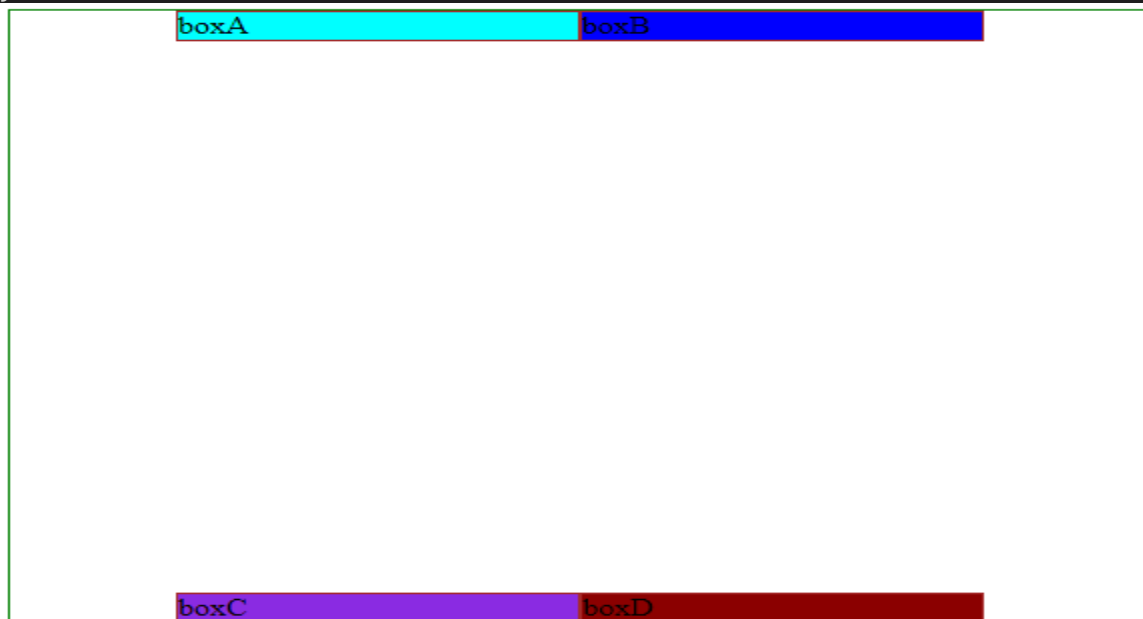
Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: center;
  align-content: space-evenly;
}
.box{
  border:1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color:blue;
}
#boxC{
  background-color:blueviolet;
}
#boxD{
  background-color:darkred;
}
```



Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: center;
  align-content: space-between;
}
.box{
  border: 1px solid brown;
  width: 200px;
}
#boxA{
  background-color: cyan;
}
#boxB{
  background-color: blue;
}
#boxC{
  background-color: blueviolet;
}
#boxD{
  background-color: darkred;
}
```



Flex items properties

Order

Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/html3_3.css">
</head>
<body>
  <div class="container">
    <div class="box" id="boxA">boxA</div>
    <div class="box" id="boxB">boxB</div>
    <div class="box" id="boxC">boxC</div>
    <div class="box" id="boxD">boxD</div>
  </div>
</body>
</html>
```

Css :

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: start;
  gap: 10px;
}
.box{
  height: 50px;
  width: 50px;
}
#boxA{
  order: 1;
  background-color: chartreuse;
}
```

```
#boxC{
  order: 8;
  background-color: blue;
}
#boxB{
  background-color:cyan;
}
#boxD{
  background-color:tomato;
}
```

Output



Flex-grow

Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/html3_3.css">
</head>
<body>
  <div class="container">
    <div class="box" id="boxA">boxA</div>
    <div class="box" id="boxB">boxB</div>
    <div class="box" id="boxC">boxC</div>
    <div class="box" id="boxD">boxD</div>
  </div>
</body>
</html>
```

Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: start;
  gap: 10px;
}
.box{
  height: 50px;
  width: 50px;
  flex-grow: 1;
}
#boxA{
  order: 1;
  background-color: chartreuse;
}
```

```
#boxC{
  order: 8;
  background-color: blue;
}
#boxB{
  background-color:cyan;
}
#boxD{
  background-color:tomato;
}
```

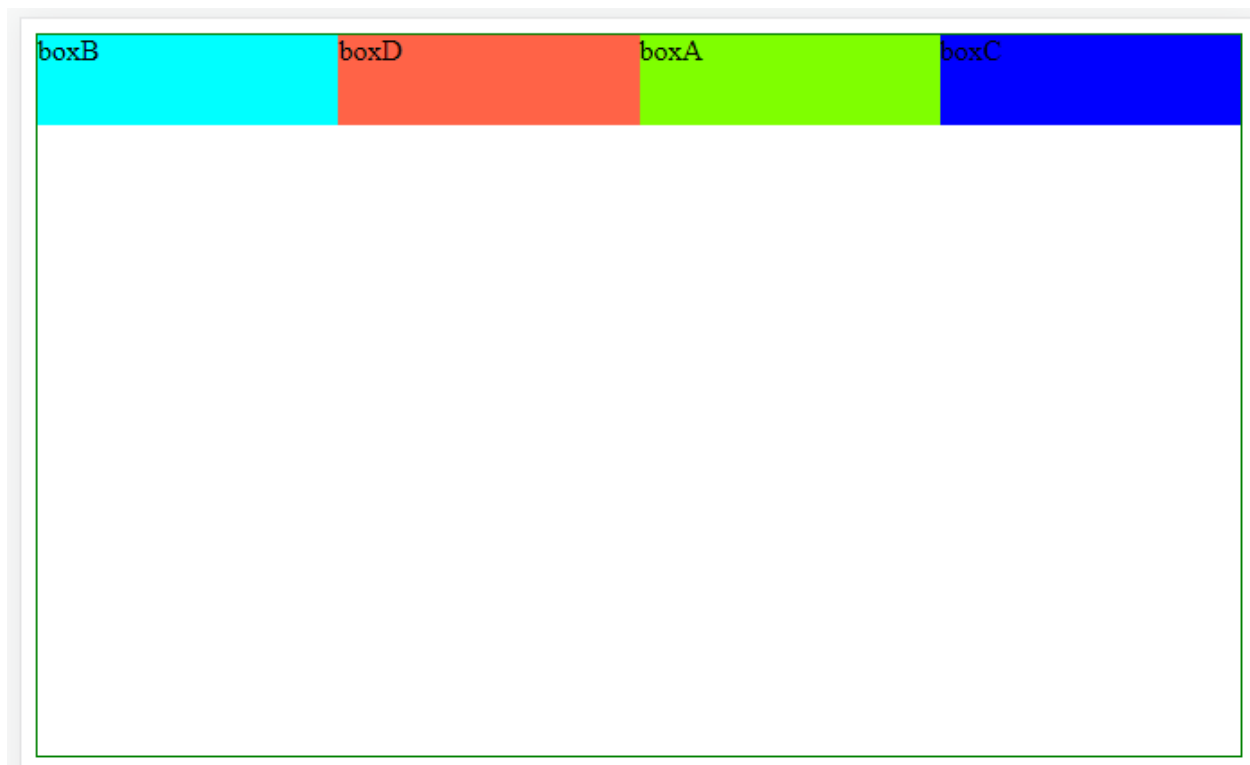
Output



Flex-grow :

Css :

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: start;
  /* gap: 10px; */
}
.box{
  height: 50px;
  width: 50px;
  flex-grow: 1;
}
#boxA{
  order: 1;
  background-color: chartreuse;
}
#boxC{
  order: 8;
  background-color: blue;
}
#boxB{
  background-color: cyan;
}
#boxD{
  background-color: tomato;
}
```

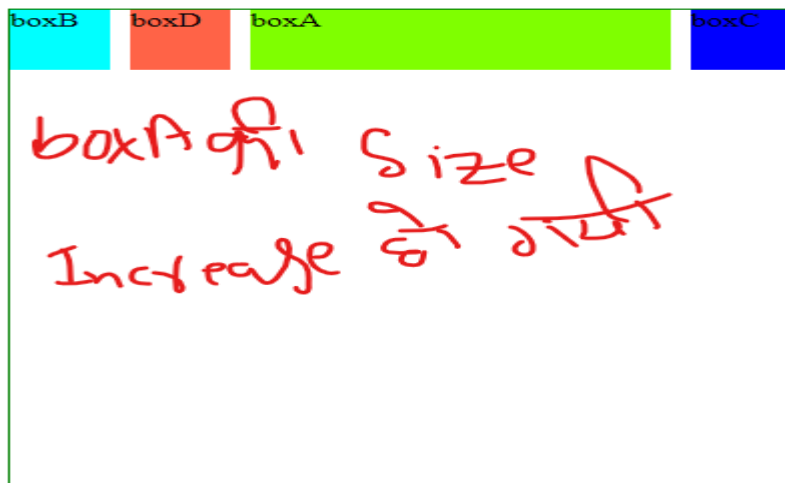


Note : flex-grow ka use karne se div equal width lega..

Html code part same

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: start;
  gap: 10px;
}
.box{
  height: 50px;
  width: 50px;
}
#boxA{
  order: 1;
  background-color: chartreuse;
  flex-grow: 1;
}
#boxC{
  order: 8;
  background-color: blue;
}
#boxB{
  background-color: cyan;
}
#boxD{
  background-color: tomato;
}
```

Output



Flex-shrink

Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  /* flex-wrap: wrap; */
  justify-content: center;
  align-items: start;
  gap: 10px;
}
.box{
  height: 50px;
  width: 100px;
}
#boxA{
  flex-shrink: 4;
  order: 1;
  background-color: chartreuse;
}
#boxC{
  order: 8;
  background-color: blue;
}
#boxB{
  flex-shrink: 1;
  background-color: cyan;
}
#boxD{
  background-color: tomato;
}
```

Output

Note :by default flex – shrink ki value 1 hoti hai

flex-shrink 2
shrink 2

boxB	boxD	boxA	boxC
------	------	------	------

||

flex-basis

iska use karne se content ke hisaab se increase hoga

Html

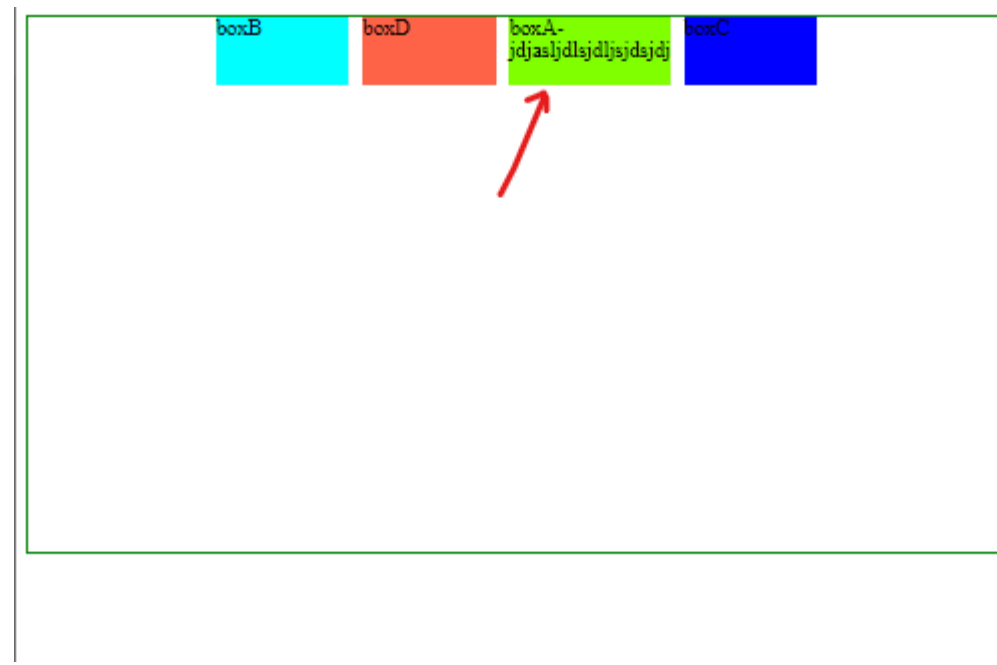
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/html3_3.css">
</head>
<body>
  <div class="container">
    <div class="box" id="boxA">boxA-jdjasljdlsjdljsjdsjdj</div>
    <div class="box" id="boxB">boxB</div>
    <div class="box" id="boxC">boxC</div>
    <div class="box" id="boxD">boxD</div>
  </div>
</body>
</html>
```

Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: start;
  gap: 10px;
}
.box{
  height: 50px;
  /* width: 100px; */
  flex-basis: 100px;
}
```

```
#boxA{
  /* flex-shrink: 4; */
  order: 1;
  background-color: chartreuse;
}
#boxC{
  order: 8;
  background-color: blue;
}
#boxB{
  /* flex-shrink: 1; */
  background-color: cyan;
}
#boxD{
  background-color: tomato;
}
```

Output :



align-self

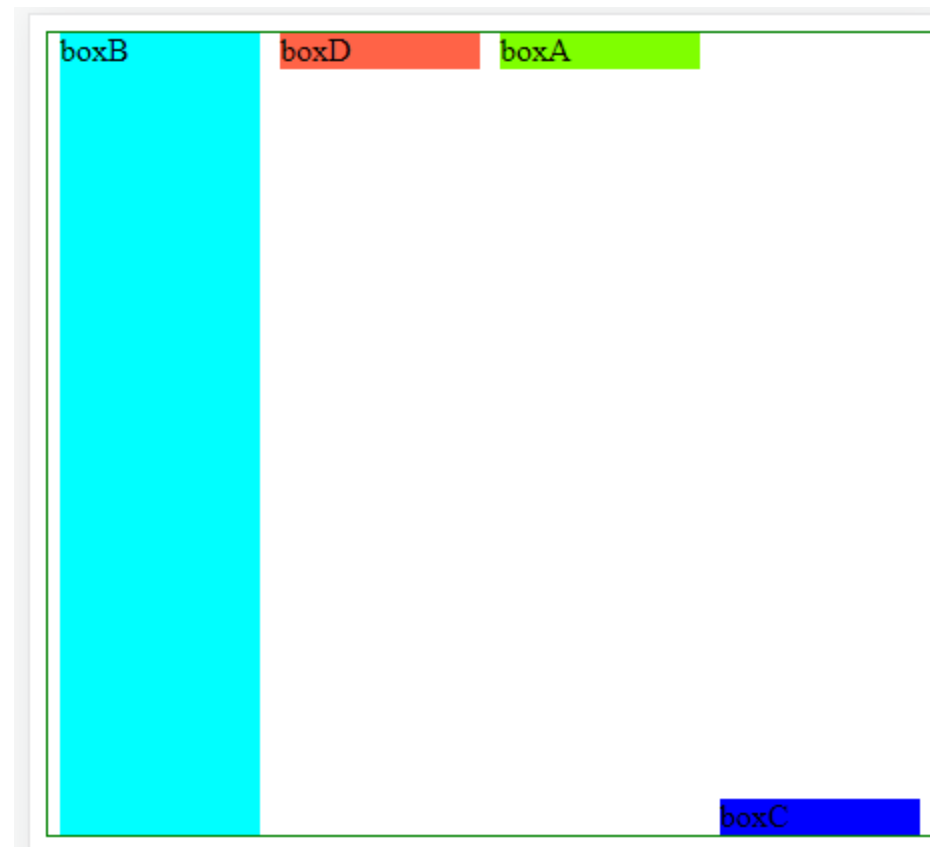
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/html3_3.css">
</head>
<body>
  <div class="container">
    <div class="box" id="boxA">boxA</div>
    <div class="box" id="boxB">boxB</div>
    <div class="box" id="boxC">boxC</div>
    <div class="box" id="boxD">boxD</div>
  </div>
</body>
</html>
```

Css

```
.container{
  border: 1px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: start;
  gap: 10px;
}
.box{
  /* height: 50px; */
  width: 100px;
}
#boxA{
  /* flex-shrink: 4; */
  order: 1;
  background-color: chartreuse;
}
```

```
#boxC{
  align-self: self-end;
  order: 8;
  background-color: blue;
}
#boxB{
  align-self: stretch;
  background-color:cyan;
}
#boxD{
  background-color:tomato;
}
```

Output



Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle/html3_3.css">
</head>
<body>
  <div class="container">
    <div class="box" id="boxA">boxA</div>
    <div class="box" id="boxB">boxB</div>
    <div class="box" id="boxC">boxC</div>
    <div class="box" id="boxD">boxD</div>
  </div>
</body>
</html>
```

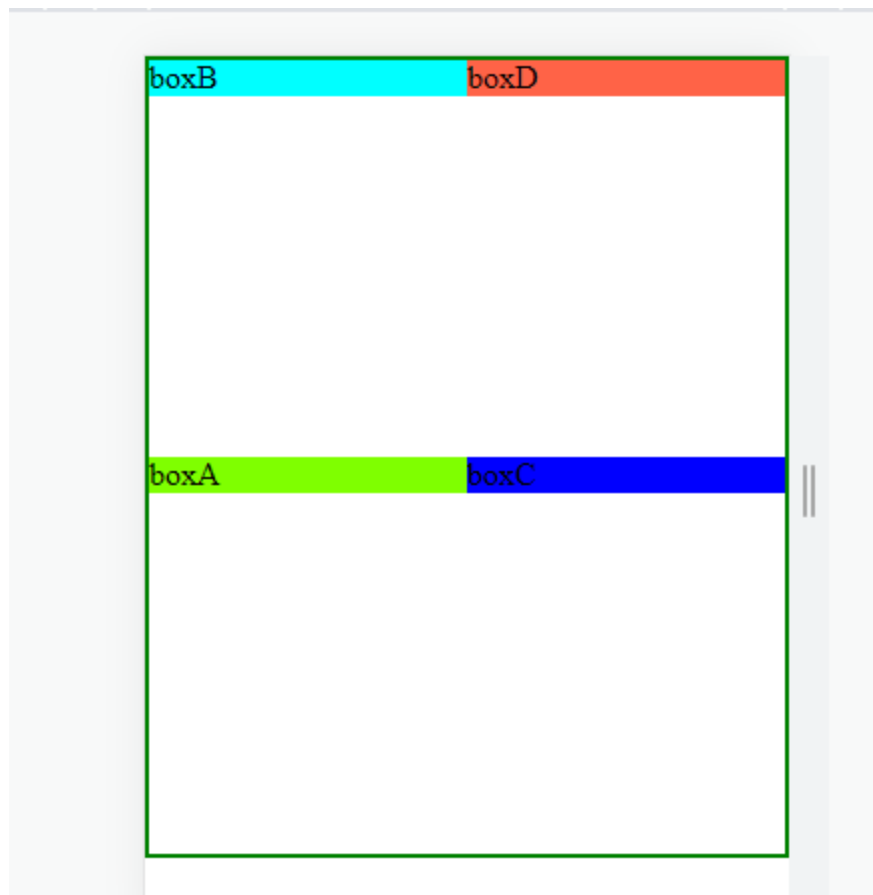
Css

```
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
.container{
  border: 2px solid green;
  height: 400px;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: start;
  /* gap: 10px; */
}
.box{
  /* height: 50px; */
  width: 150px;
  flex-basis: 50%;
}
```



```
#boxA{
  /* flex-shrink: 4; */
  order: 1;
  background-color: chartreuse;
}
#boxC{
  /* align-self: self-end; */
  order: 8;
  background-color: blue;
}
#boxB{
  /* align-self: stretch; */
  background-color: cyan;
}
#boxD{
  background-color: tomato;
}
```

Output



CSS 3D Transform

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle//html3_4.css">
</head>
<body>
  <div class="div"></div>

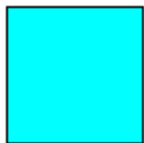
  <div class="div" id="translated"></div>

  <div class="div" id="scaled"></div>

</body>
</html>
```

Css

```
.div{
  border: 1px solid black;;
  background-color: aqua;
  width: 100px;
  height: 100px;
}
#translated{
  transform: perspective(15px) translateZ(-15px);
}
```



Note : **perspective** working z-axis aur hamare eye se leker screen ke beech ka jo distance hai usmai 10px de diya hai .

TranslatedZ(-15px) andar hogi ya bahar hogi

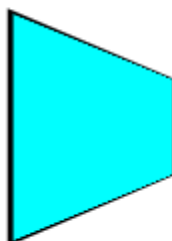
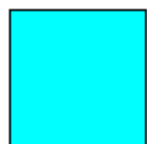


```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle//html3_4.css">
</head>
<body>
  <div class="div"></div>
  <div class="div" id="translated"></div>
  <div class="div" id="scaled"></div>

</body>
</html>
```

Css

```
.div{
  border: 1px solid black;;
  background-color: aqua;
  width: 100px;
  height: 100px;
}
#translated{
  transform: perspective(15px) translateZ(-15px);
}
#scaled{
  margin-left: 200px;
  transform: perspective(15px) scaleZ(1) rotateX(0deg) rotateY(7deg);
}
```



CSS -2D transforms

1.Translate

Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/addStyle//html4.css">
</head>
<body>
  <div class="box1">box1</div>
</body>
</html>
```

Css

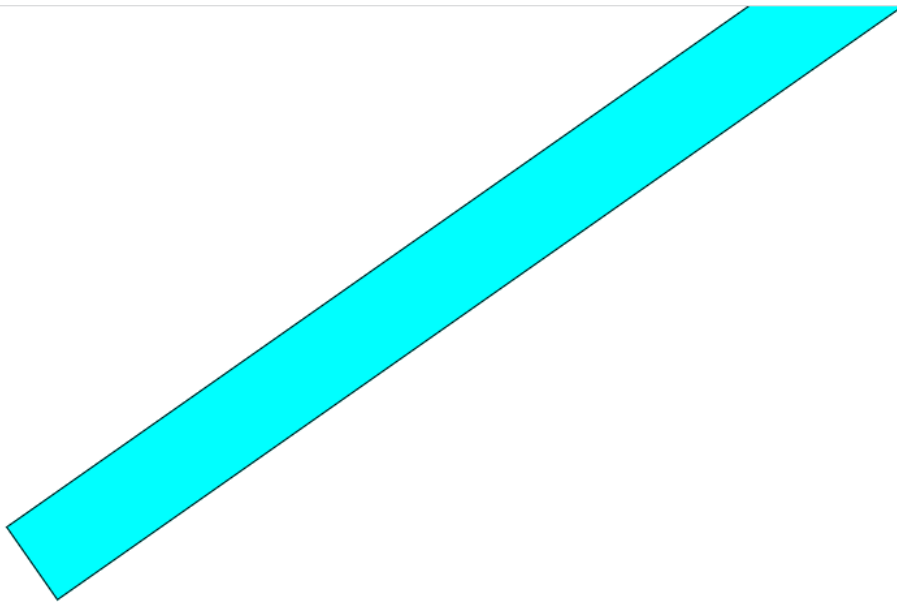
```
.box1{
  background-color: aqua;
  height: 200px;
  width: 200px;
  transform: translate(150px,300px);
}
```

box1



2.Rotate

```
.box1{  
  background-color: aqua;  
  border: 1px solid black;  
  height: 100px;  
  transform: translate(0,300px);  
  transform: rotate(145deg);  
}
```



3.scale

```
.box1{  
  background-color: aqua;  
  border: 1px solid black;  
  height: 100px;  
  margin: 200px;  
  transform: scale(1,4);  
}
```



4.skew

```
.box1{  
  background-color: aqua;  
  border: 1px solid black;  
  height: 100px;  
  margin: 200px;  
  transform: skew(45deg)  
}
```



CSS BASICS -IV

Grid

Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>grid</title>
  <link rel="stylesheet" href="/Html/html2.css">
</head>
<body>
  <div class="container">
    <div class="box">box1</div>
    <div class="box">box2</div>
    <div class="box">box3</div>
    <div class="box">box4</div>
    <div class="box">box5</div>
  </div>
</body>
</html>
```

Css

```
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
.container{
  background-color: brown;
  height: 500px;
  border: 3px solid black;
  display: grid;
}
.box{
  background-color: aqua;
  border: 3px solid black;
  width: 80px;
  height: 80px;
}
```

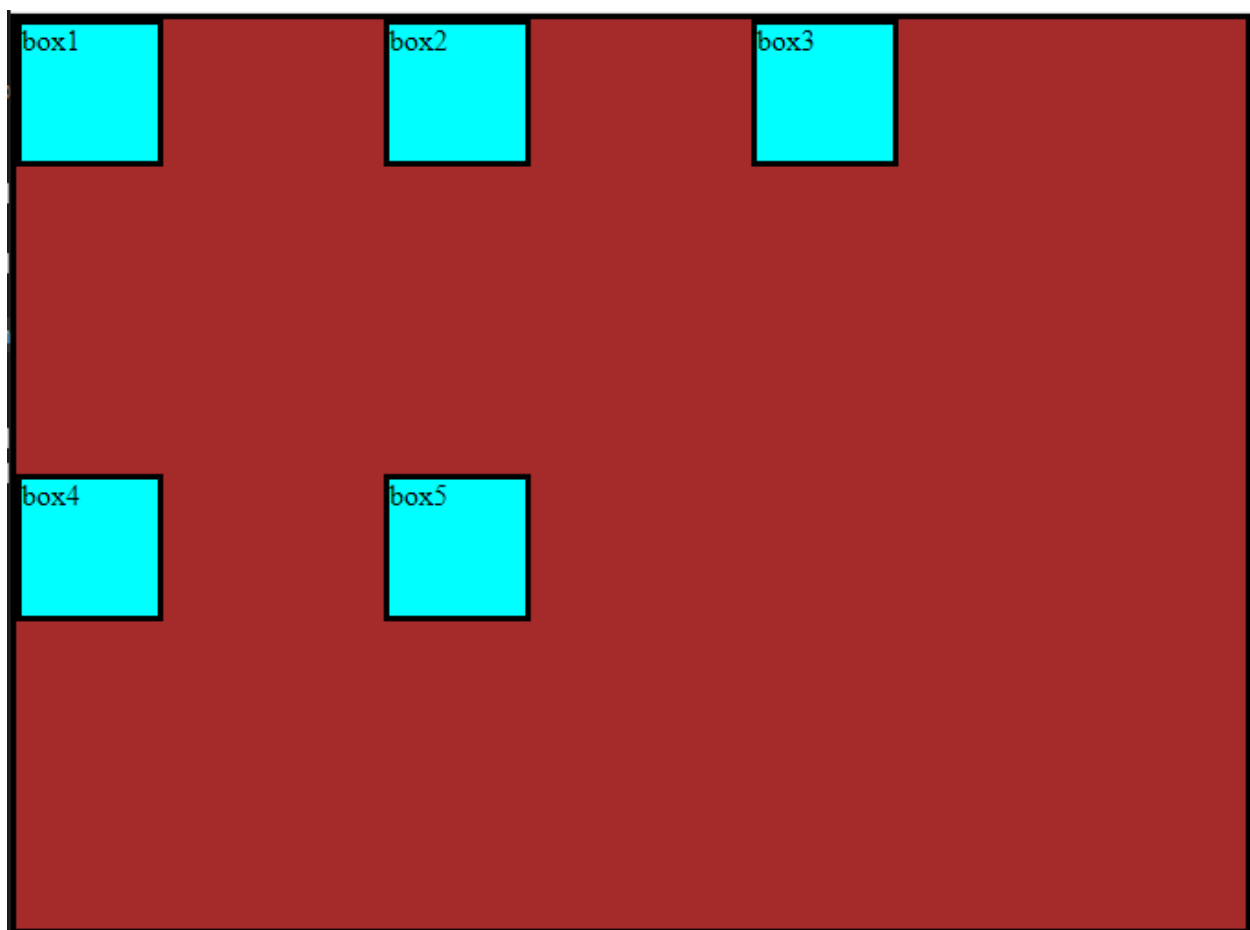


grid-template-columns: 200px 200px 210px;

html part same

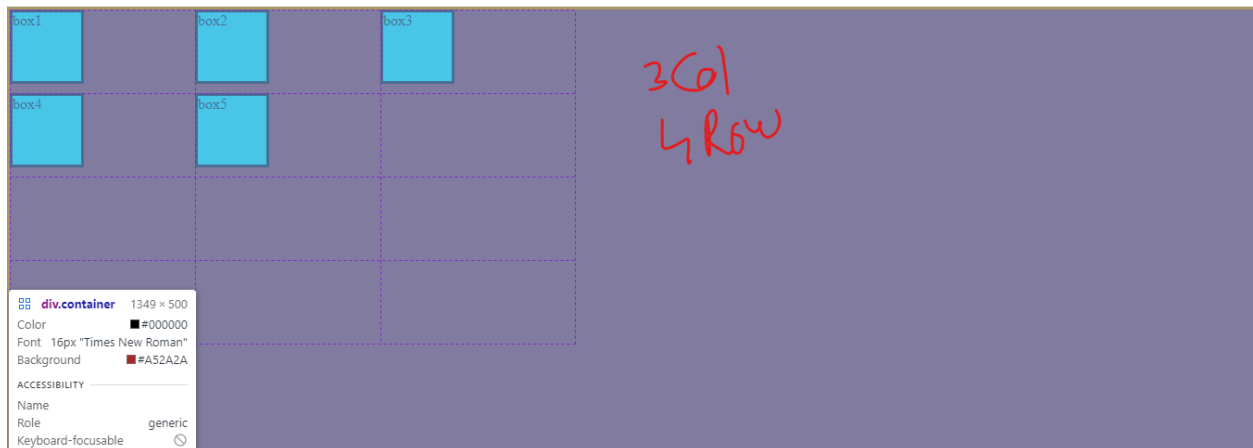
css part

```
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
.container{
  background-color: brown;
  height: 500px;
  border: 3px solid black;
  display: grid;
  grid-template-columns: 200px 200px 210px;
}
.box{
  background-color: aqua;
  border: 3px solid black;
  width: 80px;
  height: 80px;
}
```



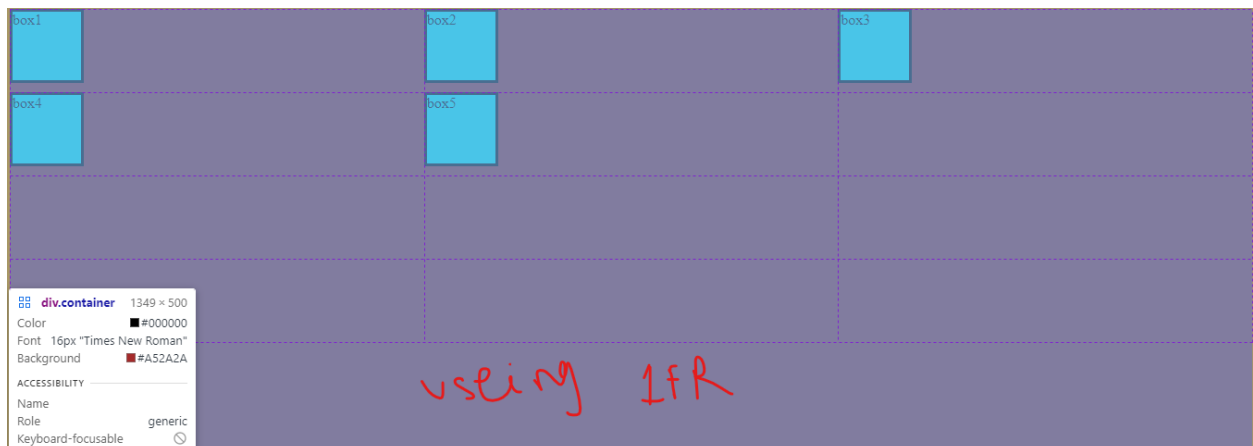
Css

```
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
.container{
  background-color: brown;
  height: 500px;
  border: 3px solid black;
  display: grid;
  grid-template-columns: 200px 200px 210px;
  grid-template-rows: 90px 90px 90px 90px;
}
.box{
  background-color: aqua;
  border: 3px solid black;
  width: 80px;
  height: 80px;
}
```



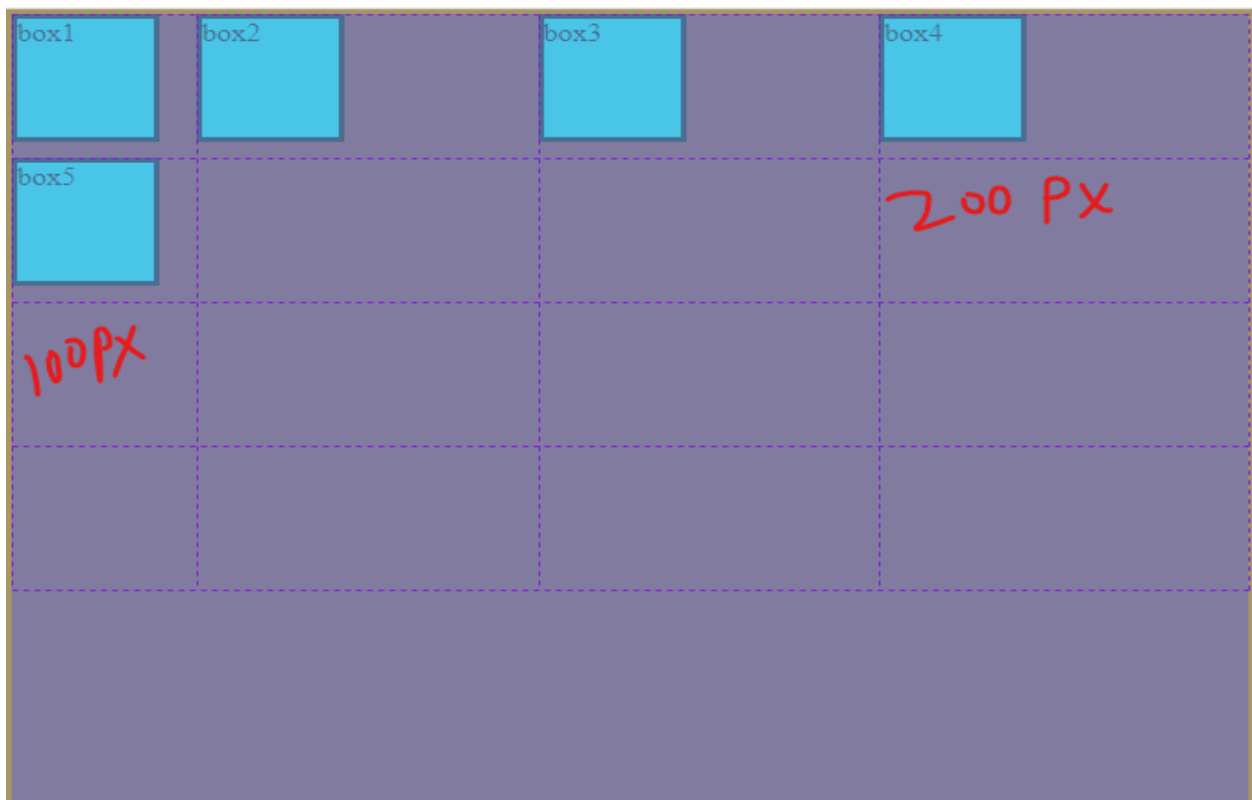
Css

```
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
.container{
  background-color: brown;
  height: 500px;
  border: 3px solid black;
  display: grid;
  grid-template-columns: 1fr 1fr 1fr;
  /* Is ko aisa bhi likh sakhte hai same output aayega grid-template-columns:
  repeat(3,1fr); */
  grid-template-rows: 90px 90px 90px 90px;
}
.box{
  background-color: aqua;
  border: 3px solid black;
  width: 80px;
  height: 80px;
}
```



Css

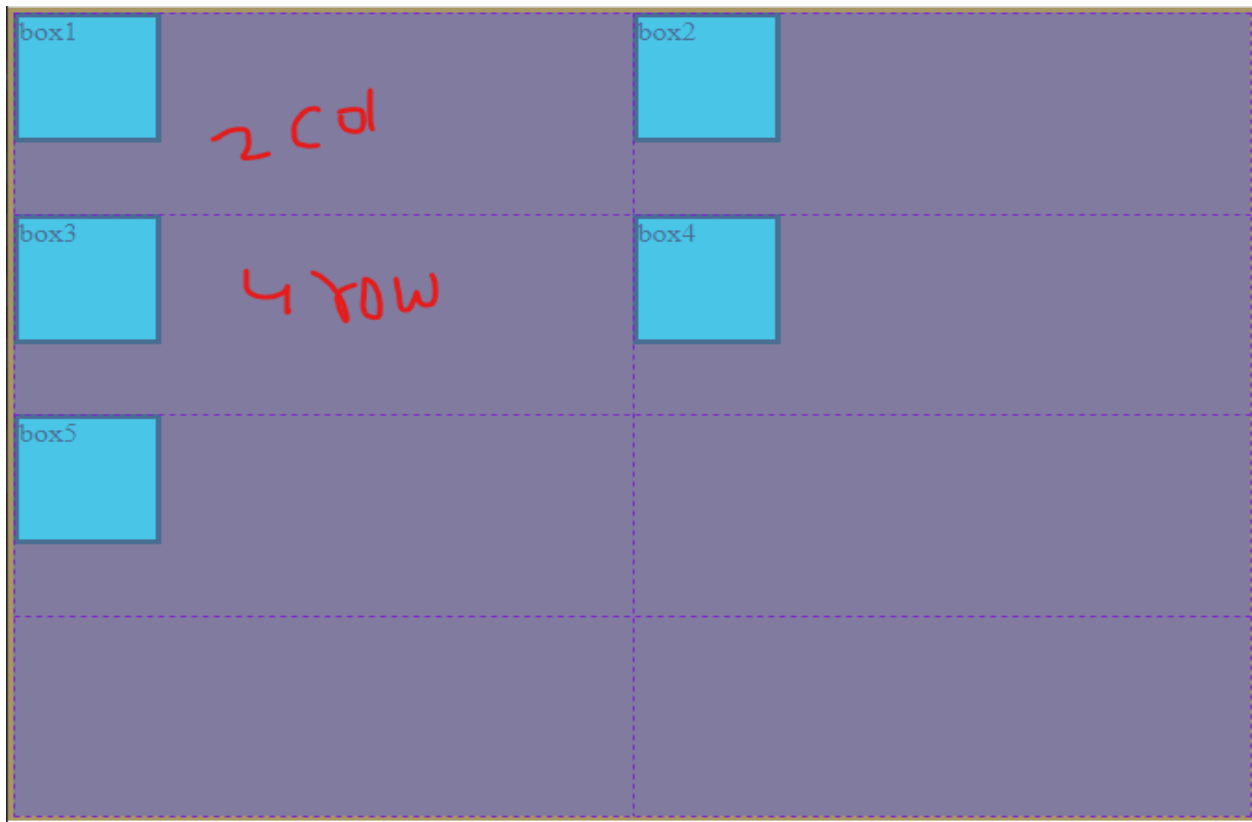
```
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
.container{
  background-color: brown;
  height: 500px;
  border: 3px solid black;
  display: grid;
  grid-template-columns: 100px repeat(2,1fr) 200px;
  grid-template-rows: 90px 90px 90px 90px;
}
.box{
  background-color: aqua;
  border: 3px solid black;
  width: 80px;
  height: 80px;
}
```



Css

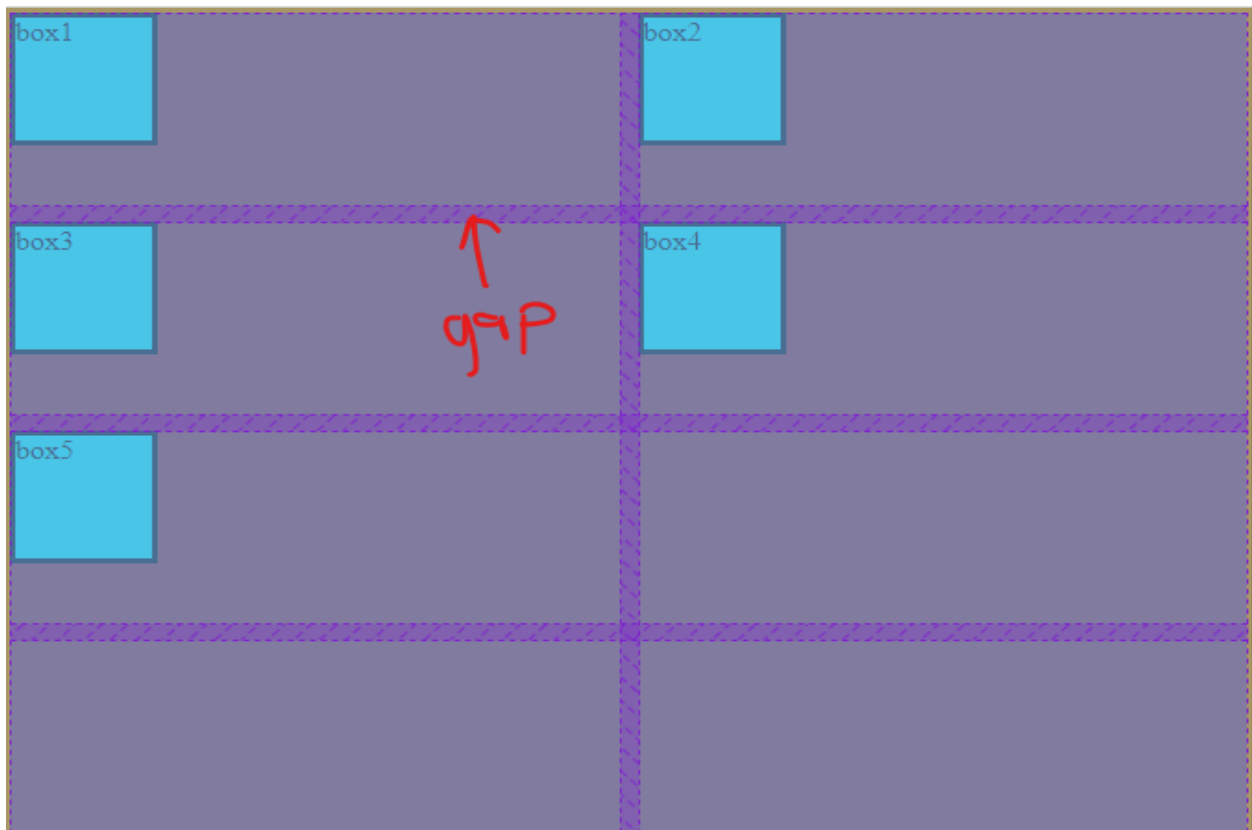
```
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
.container{
  background-color: brown;
  height: 500px;
  border: 3px solid black;
  display: grid;
  grid-template-columns: repeat(2,1fr);

  grid-template-rows: repeat(4,1fr);
}
.box{
  background-color: aqua;
  border: 3px solid black;
  width: 80px;
  height: 80px;
}
```



Css

```
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
.container{
  background-color: brown;
  height: 500px;
  border: 3px solid black;
  display: grid;
  grid-template-columns: repeat(2,1fr);
  grid-template-rows: repeat(4,1fr);
  gap: 10px;
}
.box{
  background-color: aqua;
  border: 3px solid black;
  width: 80px;
  height: 80px;
}
```



Animations & Responsive

Advanced Grids, Media Queries & Brakpoints

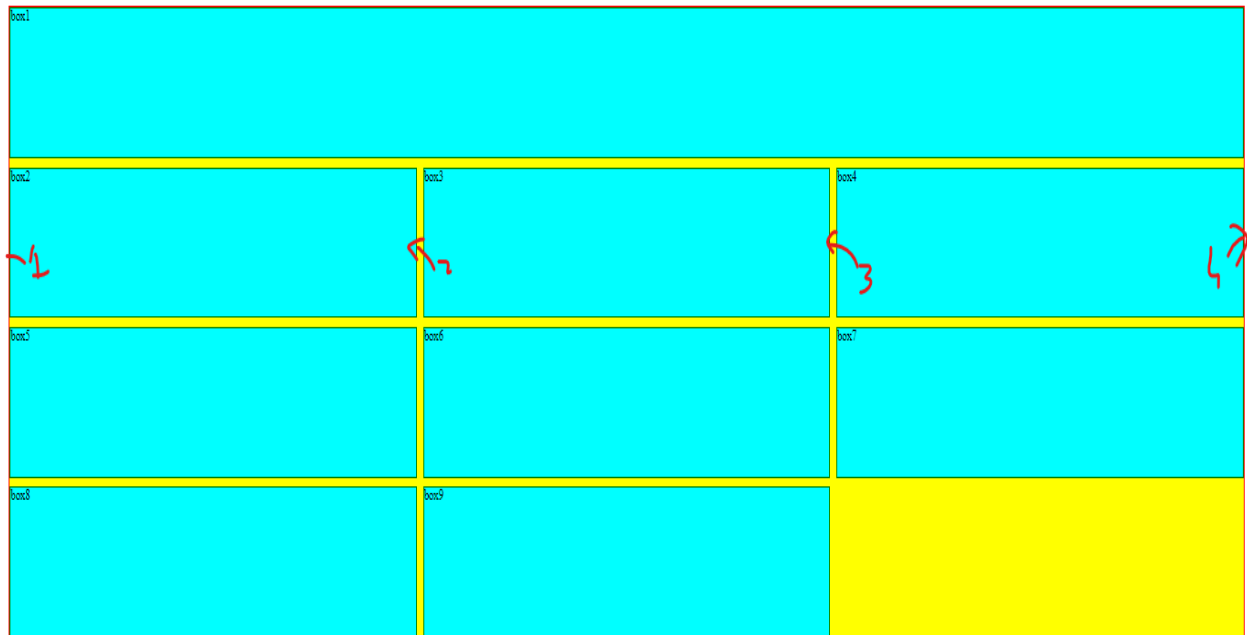
Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Html/html3.css">
</head>
<body>
  <div class="container">
    <div class="box" id="box1">box1</div>
    <div class="box">box2</div>
    <div class="box">box3</div>
    <div class="box">box4</div>
    <div class="box">box5</div>
    <div class="box">box6</div>
    <div class="box">box7</div>
    <div class="box">box8</div>
    <div class="box">box9</div>
  </div>
</body>
</html>
```

Css

```
.container{
  height: 700px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-template-columns: repeat(3,1fr);
  grid-template-rows: repeat(4,1fr);
  grid-gap: 10px;
}
```

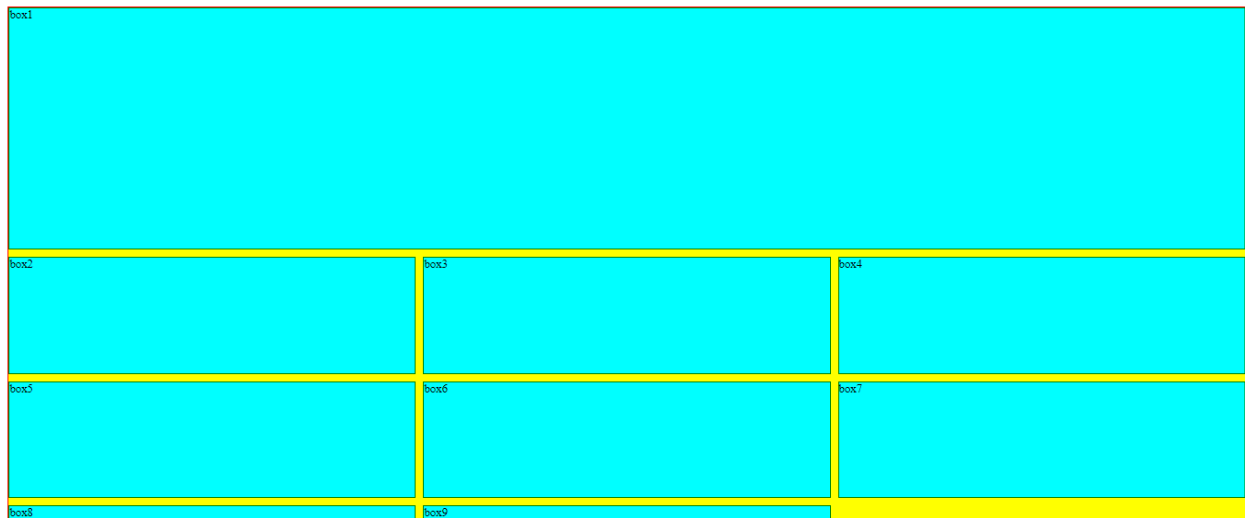
```
.box{
  border: 1px solid green;
  background-color: aqua;
  /* height: 100px;
  width: 100px;; */
}
#box1{
  grid-column-start: 1;
  grid-column-end: 4;
}
```



Html part same only change in css part

```
.container{
  height: 700px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-template-columns: repeat(3,1fr);
  grid-template-rows: repeat(4,1fr);
  grid-gap: 10px;
}
.box{
  border: 1px solid green;
  background-color: aqua;
  /* height: 100px;
  width: 100px;; */
}
#box1{
  grid-column-start: 1;
  grid-column-end: 4;

  grid-row-start: 1;
  grid-row-end: 3;
}
```



Css

Grid-area

```
.container{
  height: 700px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-template-columns: repeat(3,1fr);
  grid-template-rows: repeat(4,1fr);
  grid-gap: 10px;
}
.box{
  border: 1px solid green;
  background-color: aqua;
  /* height: 100px;
  width: 100px;; */
}
#box1{
  /* grid-column-start: 1;
  grid-column-end: 4;

  grid-row-start: 1;
  grid-row-end: 3; */

  grid-area: 1/1/3/4 ;
}
```



Grid-template-areas

Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Html/html4.css">
</head>
<body>
  <div class="container">
    <div class="box" id="header">header</div>
    <div class="box" id="sidebar">sidebar</div>
    <div class="box" id="contents">contents</div>
    <div class="box" id="footer">footer</div>
  </div>
</body>
</html>
```

Css

```
.container{
  height: 700px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-template-columns: 400px;
  grid-template-rows: 30px 150px 400px 30px;

  grid-template-areas:
    "hd"
    "side"
    "main"
    "ft";
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
```



```
#header{
  grid-area: hd;
}
#sidebar{
  grid-area: side;
}
#contents{
  grid-area: main;
}
#footer{
  grid-area: ft;
}
```



Css

```
.container{
  height: 700px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;

  grid-template-columns: 200px 400px;
  grid-template-rows: 30px 500px 30px;

  /* grid-template-areas:
    "hd"
    "side"
    "main"
    "ft"; */

  grid-template-areas:
    "hd hd"
    "side main"
    "ft ft";
}

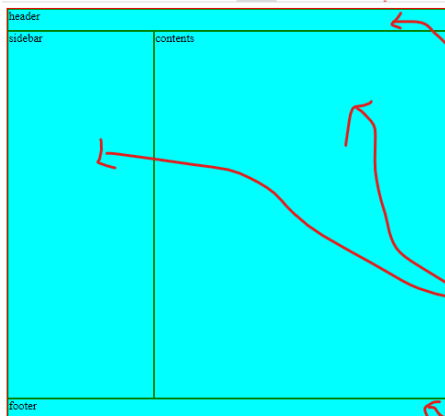
.box{
  border: 1px solid green;
  background-color: aqua;
}

#header{
  grid-area: hd;
}

#sidebar{
  grid-area: side;
}

#contents{
  grid-area: main;
}

#footer{
  grid-area: ft;
}
```



using grid-template-areas!

"hd hd"

"side main"

"ft ft";

Advanced Grid Concepts

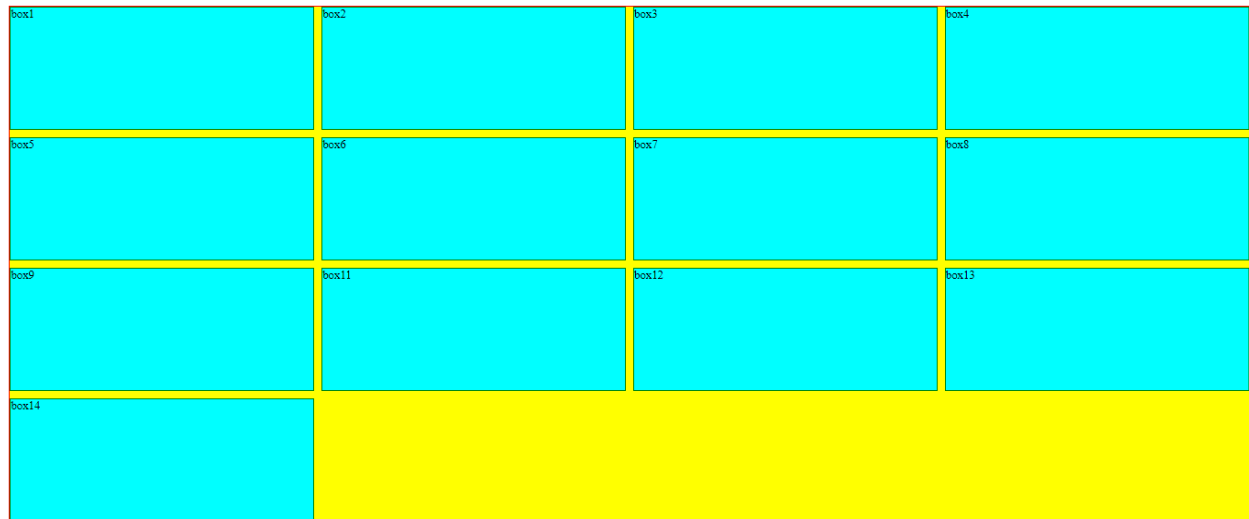
Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Html/h5.css">
</head>
<body>
  <div class="container">
    <div class="box">box1</div>
    <div class="box">box2</div>
    <div class="box">box3</div>
    <div class="box">box4</div>
    <div class="box">box5</div>
    <div class="box">box6</div>
    <div class="box">box7</div>
    <div class="box">box8</div>
    <div class="box">box9</div>
    <div class="box">box11</div>
    <div class="box">box12</div>
    <div class="box">box13</div>
    <div class="box">box14</div>
  </div>
</body>
</html>
```

Css

```
.container{
  height: 700px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: repeat(4,1fr);
  grid-auto-rows: minmax(100px,auto);
}
```

```
.box{  
  border: 1px solid green;  
  background-color: aqua;  
}
```



Grid auto rows mai automatic row add hota ,agar content add hota rahega.

Grid-properties

Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Html/h5.css">
</head>
<body>
  <div class="container">
    <div class="box">box1</div>
    <div class="box">box2</div>
    <div class="box">box3</div>
    <div class="box">box4</div>
    <div class="box">box5</div>
    <div class="box">box6</div>
    <div class="box">box7</div>
    <div class="box">box8</div>
    <div class="box">box9</div>
    <div class="box">box11</div>
    <div class="box">box12</div>
    <div class="box">box13</div>
    <div class="box">box14</div>
  </div>
</body>
</html>
```

Css

```
.container{
  height: 700px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  justify-content: start;
  /* justify-content: end; */
}
```

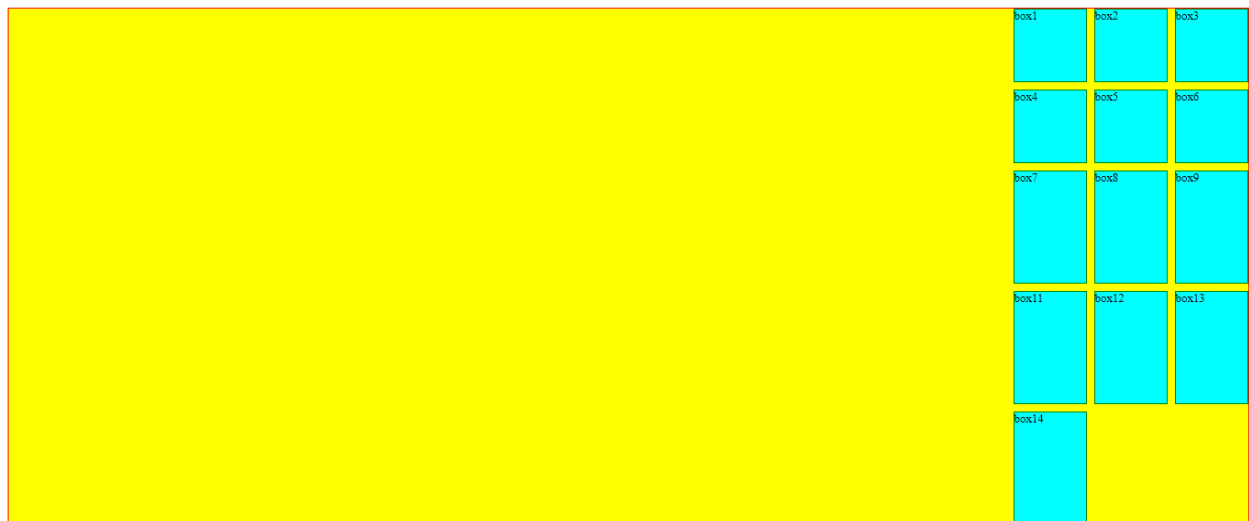
```
.box{  
  border: 1px solid green;  
  background-color: aqua;  
}
```

box1	box2	box3
box4	box5	box6
box7	box8	box9
box11	box12	box13
box14		

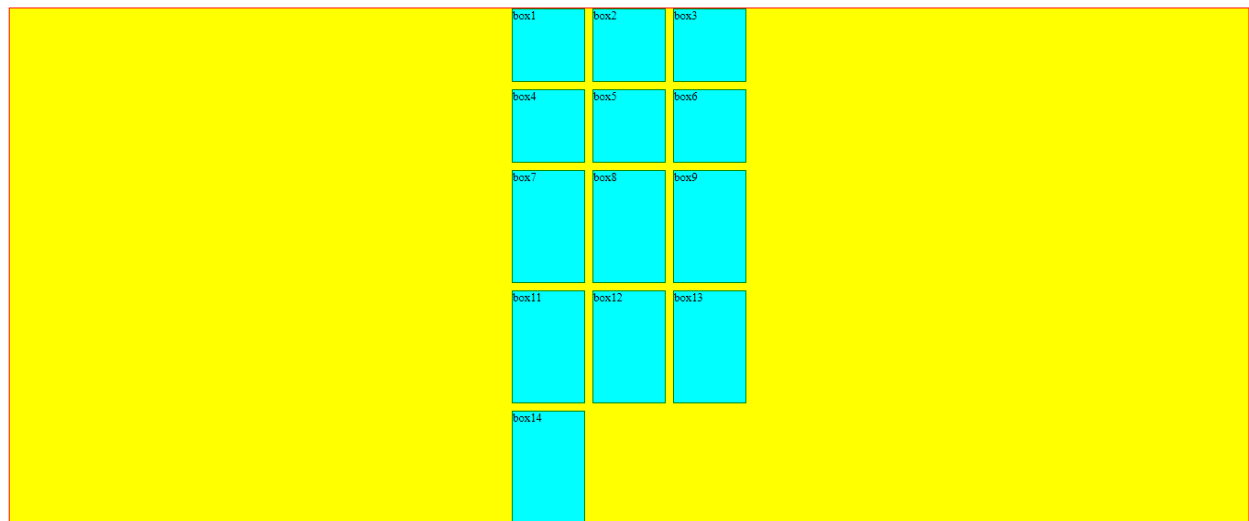
Justify-Content: start

Css (justify-content :end)

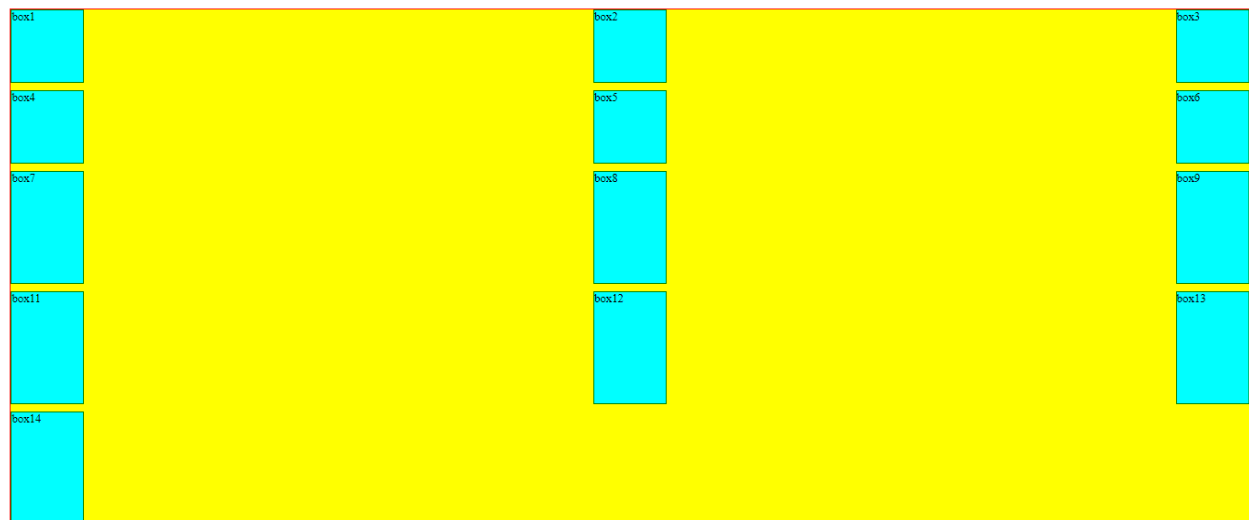
```
.container{
  height: 700px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  /* justify-content: start; */
  justify-content: end;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
```



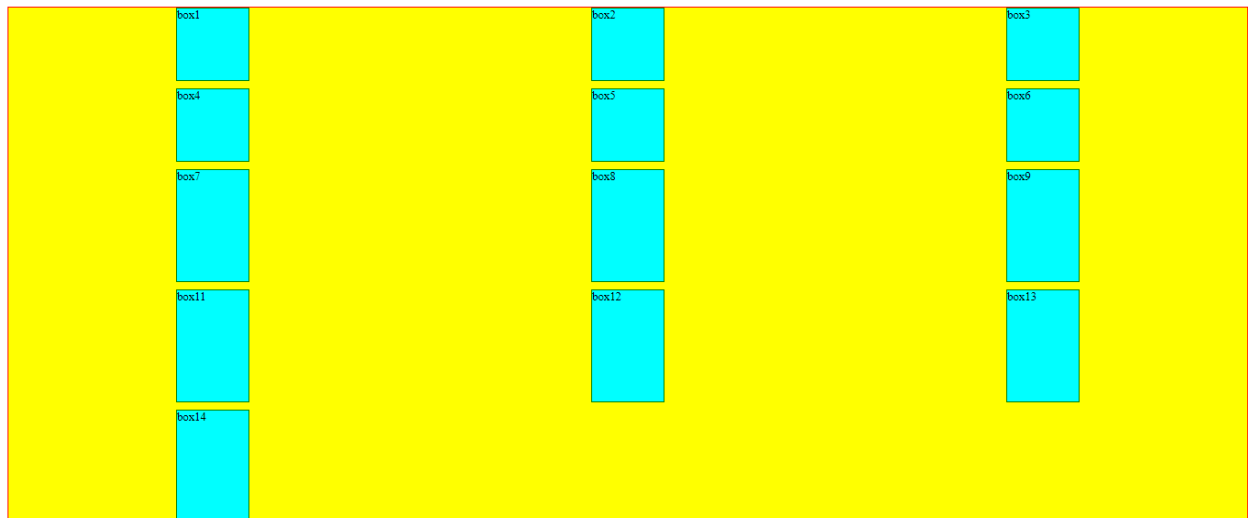

```
.container{
  height: 700px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  /* justify-content: start; */
  /* justify-content: end; */
  justify-content: center;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
```



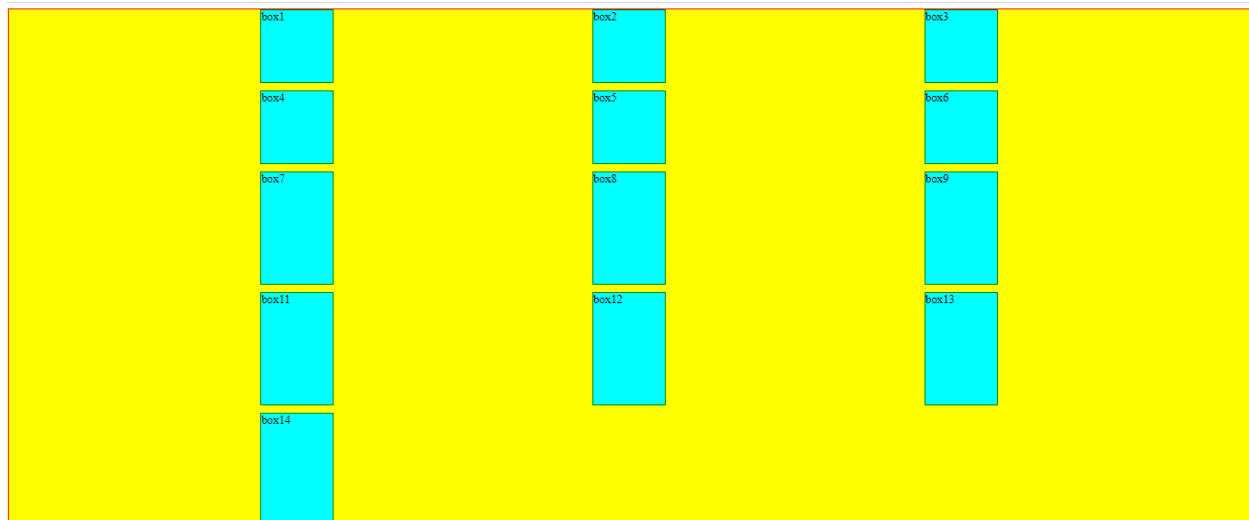
```
.container{
  height: 700px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  /* justify-content: start; */
  /* justify-content: end; */
  /* justify-content: center; */
  justify-content: space-between;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
```



```
.container{
  height: 700px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  /* justify-content: start; */
  /* justify-content: end; */
  /* justify-content: center; */
  justify-content: space-around;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
```



```
.container{
  height: 700px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  /* justify-content: start; */
  /* justify-content: end; */
  /* justify-content: center; */
  justify-content: space-evenly;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
```

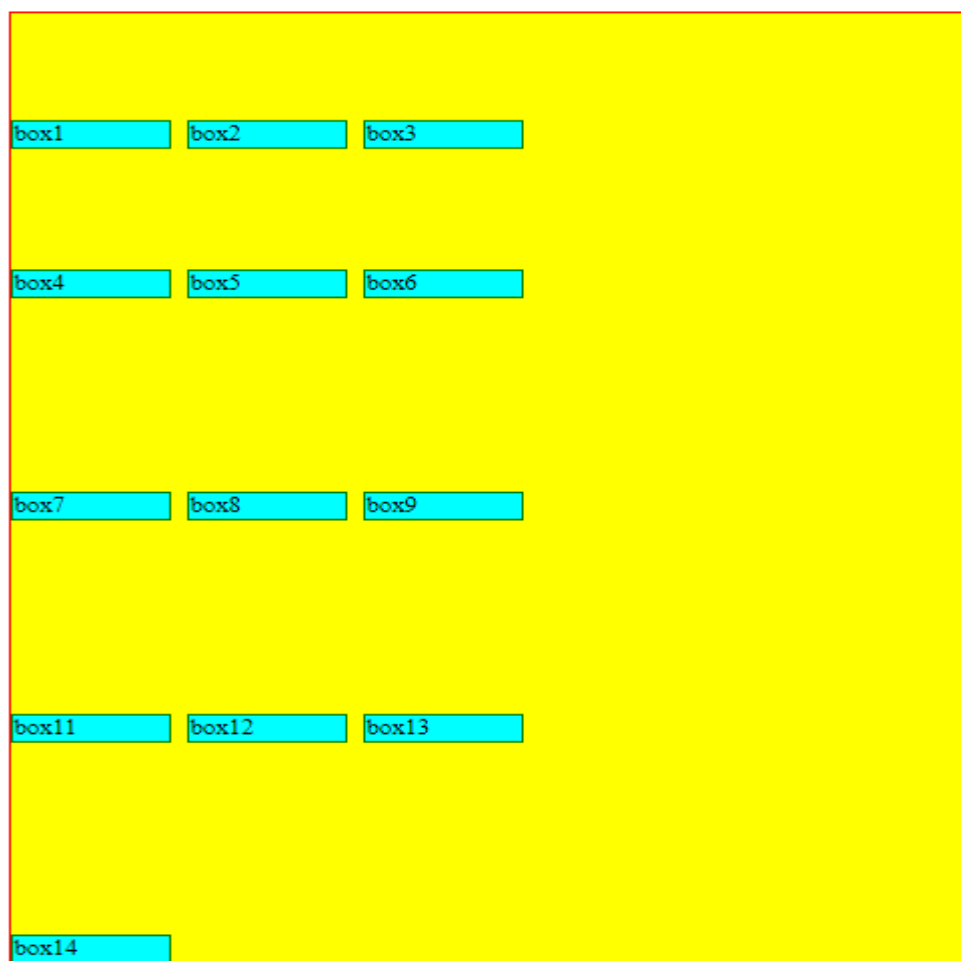


Align-items:start;

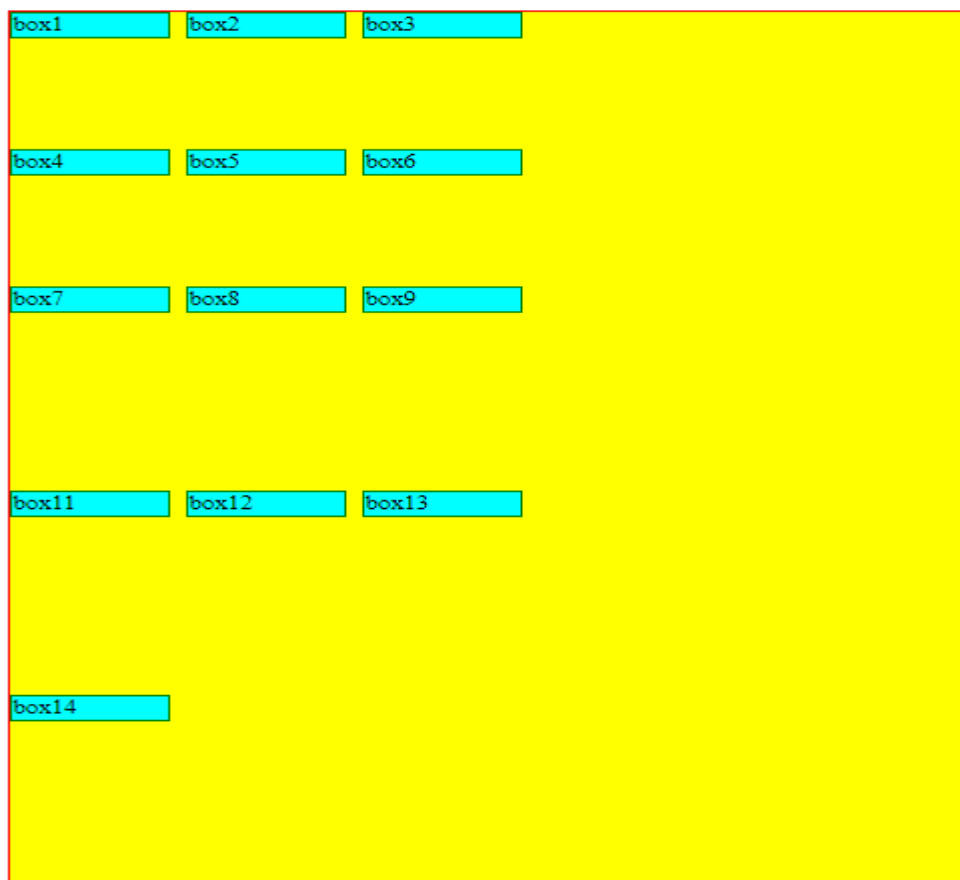
```
.container{
  height: 700px;
  width: 600px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  align-items: start;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
```



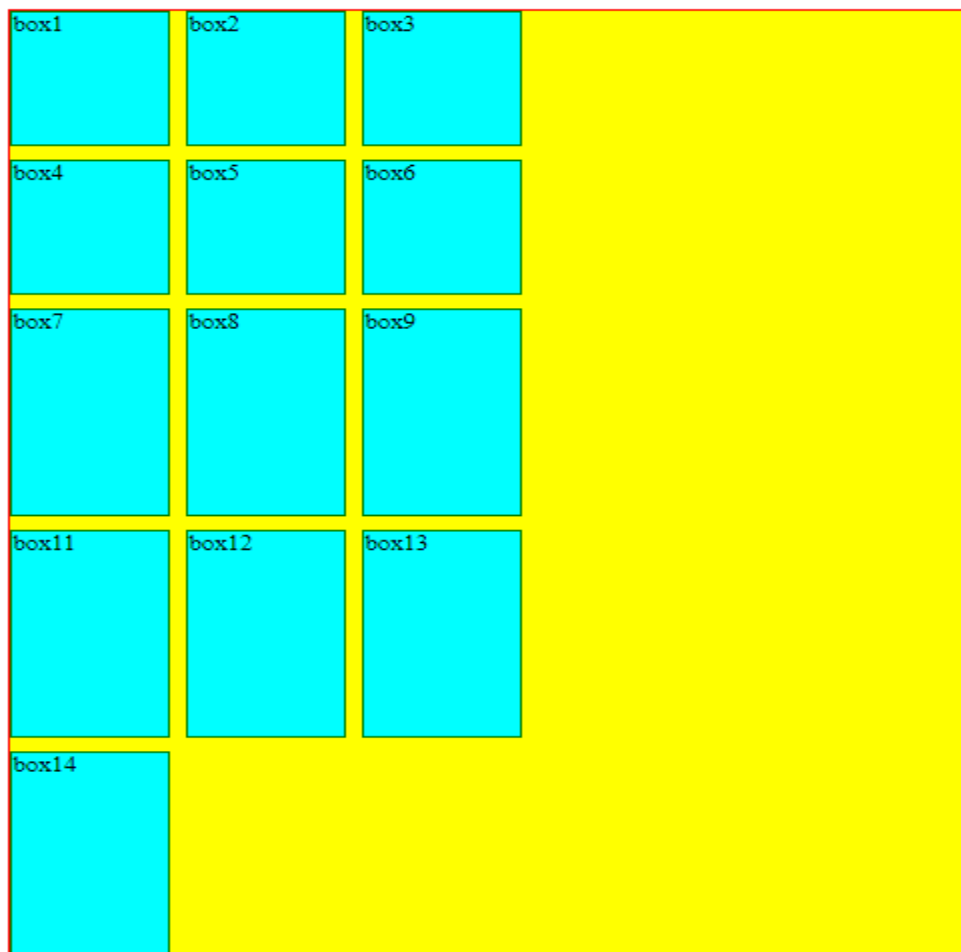
```
.container{
  height: 700px;
  width: 600px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  align-items: end;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
```



```
.container{
  height: 700px;
  width: 600px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  align-items: baseline;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
```



```
.container{
  height: 700px;
  width: 600px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  align-items: stretch;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
```



Justify-self:start (jitna content hoga uthna horizontal size lega)

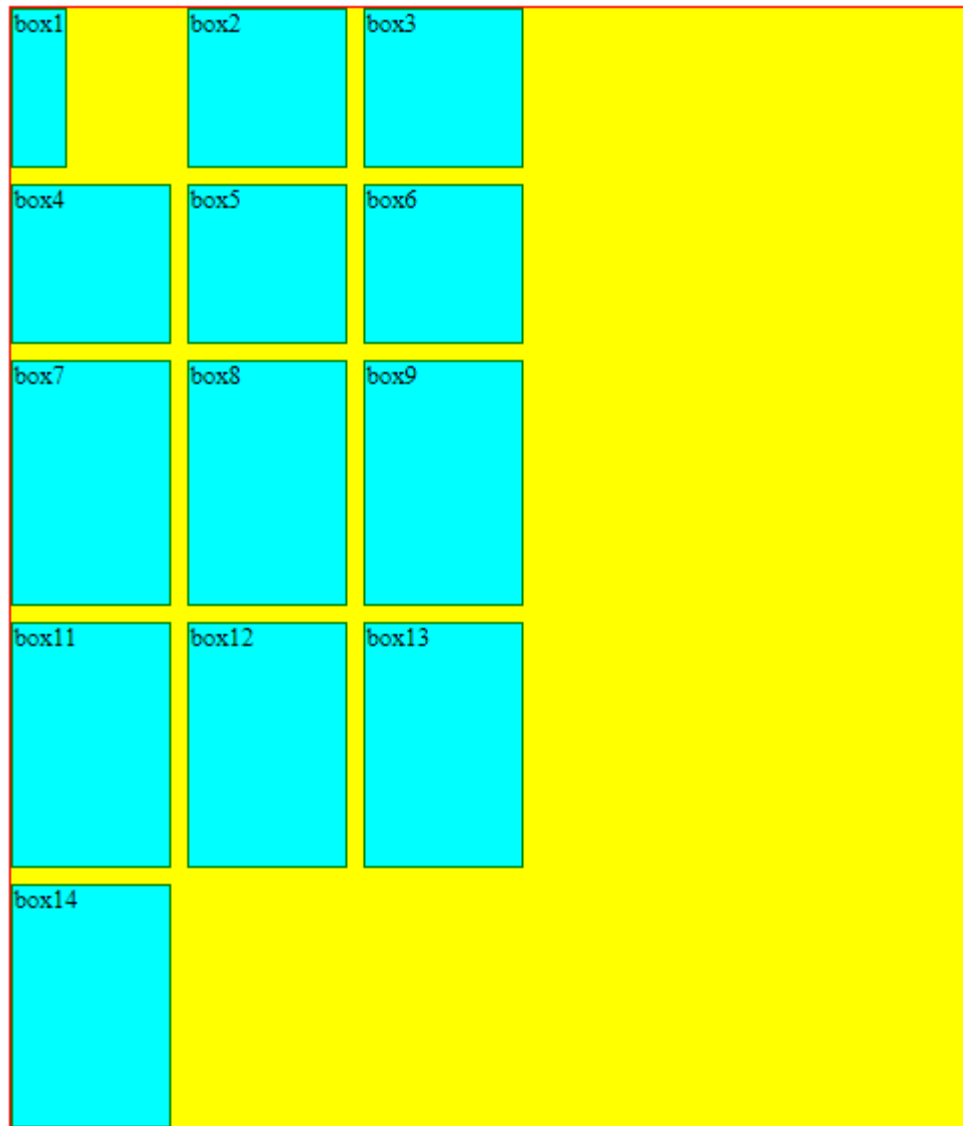
Html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Html/h5.css">
</head>
<body>
  <div class="container">
    <div class="box" id="box1">box1</div>
    <div class="box">box2</div>
    <div class="box">box3</div>
    <div class="box">box4</div>
    <div class="box">box5</div>
    <div class="box">box6</div>
    <div class="box">box7</div>
    <div class="box">box8</div>
    <div class="box">box9</div>
    <div class="box">box11</div>
    <div class="box">box12</div>
    <div class="box">box13</div>
    <div class="box">box14</div>
  </div>
</body>
</html>
```

Css

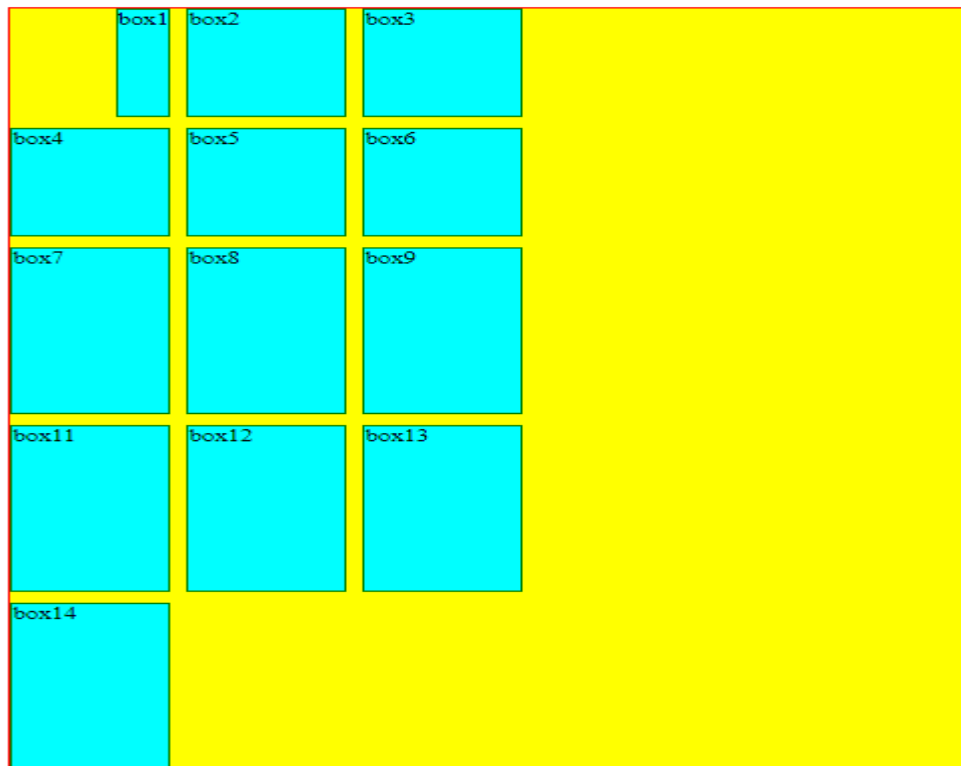
```
.container{
  height: 700px;
  width: 600px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  align-items: stretch;
}
```

```
.box{  
  border: 1px solid green;  
  background-color: aqua;  
}  
#box1{  
  justify-self: start;  
}
```



Css

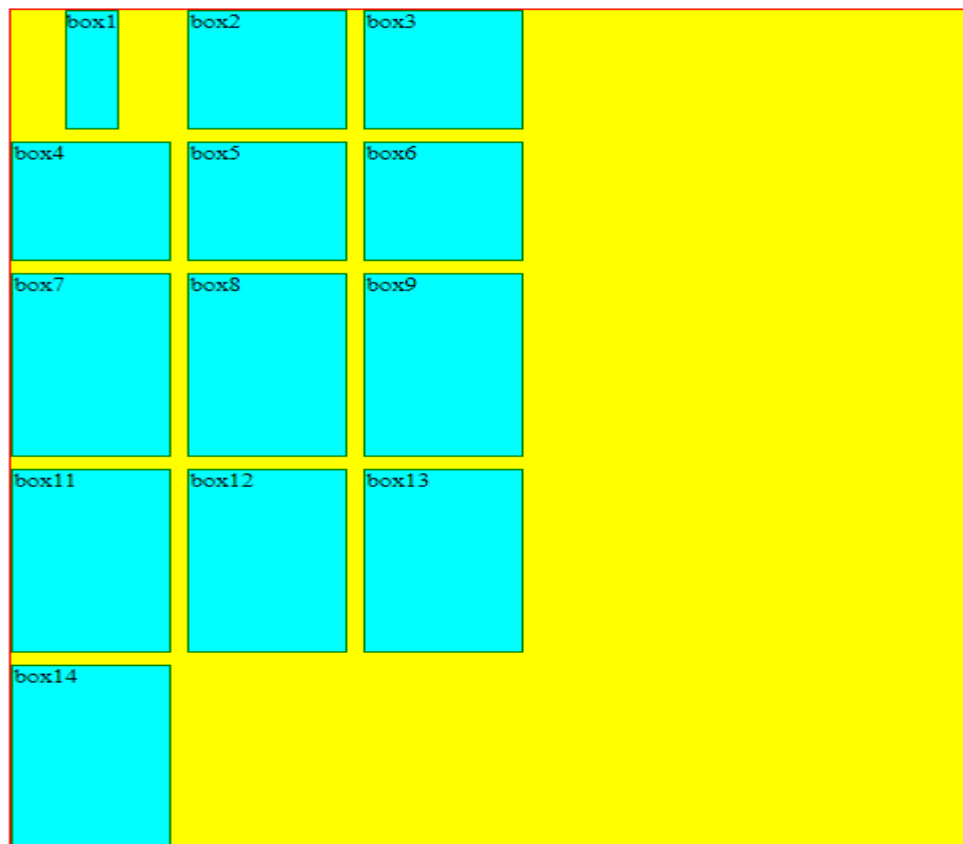
```
.container{
  height: 700px;
  width: 600px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  align-items: stretch;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
#box1{
  justify-self: end;
}
```



```

.container{
  height: 700px;
  width: 600px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  align-items: stretch;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
#box1{
  justify-self: center;
}

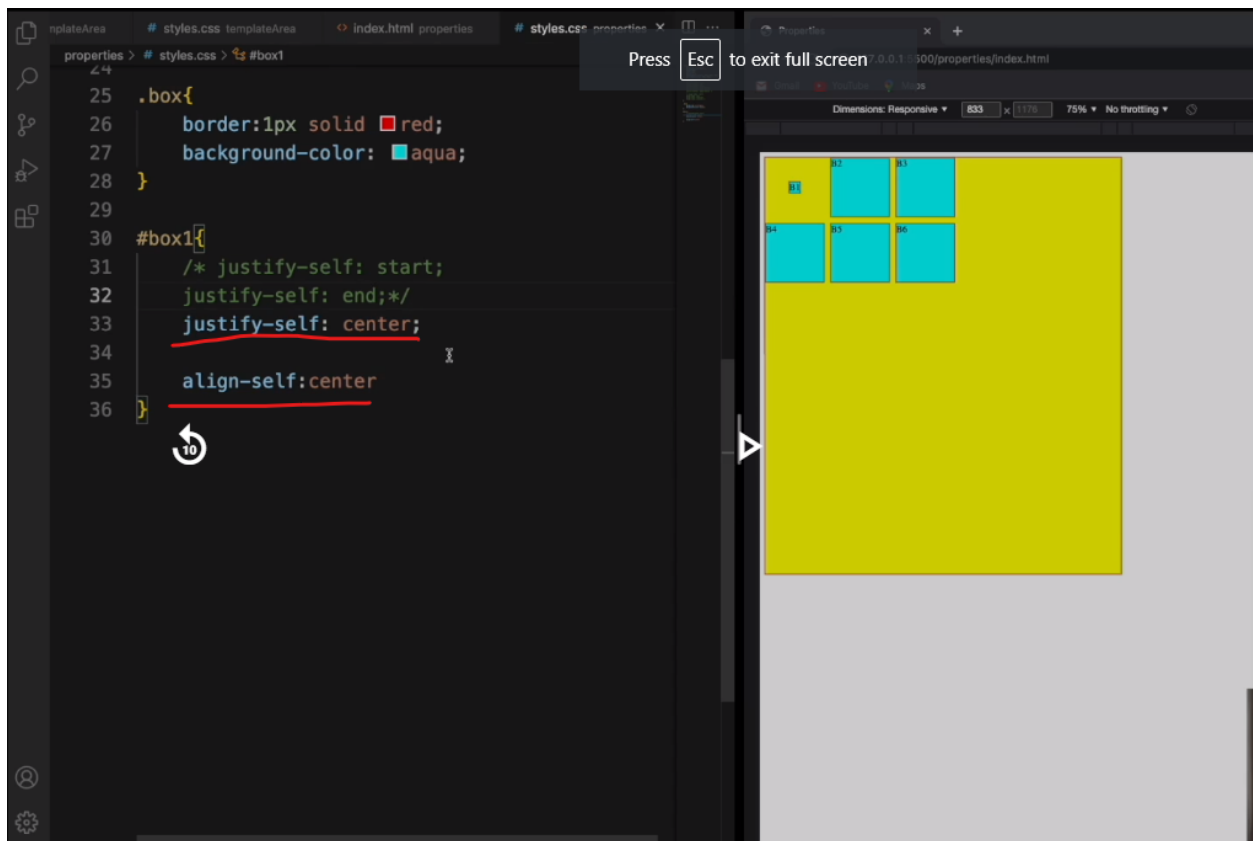
```

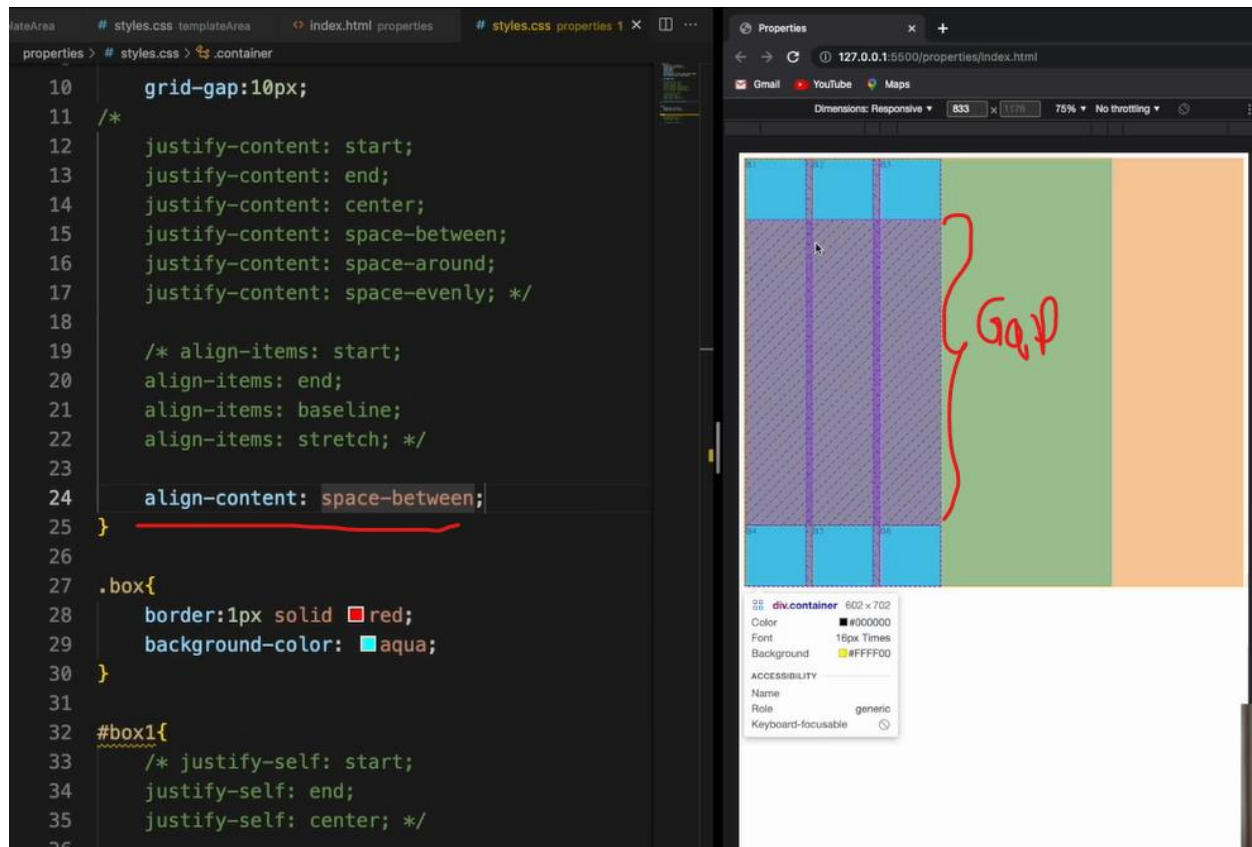


Align-self

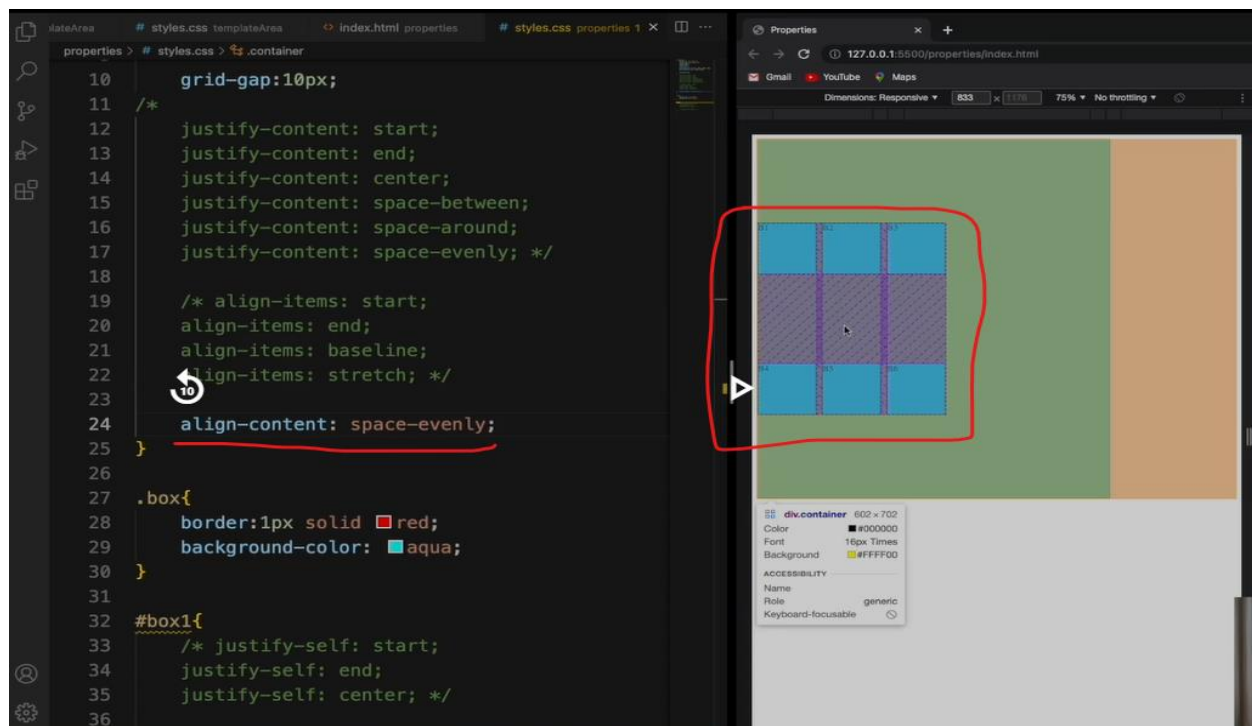
```
.container{
  height: 700px;
  width: 600px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-gap: 10px;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;
  align-items: stretch;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
#box1{
  align-self: center;
}
```

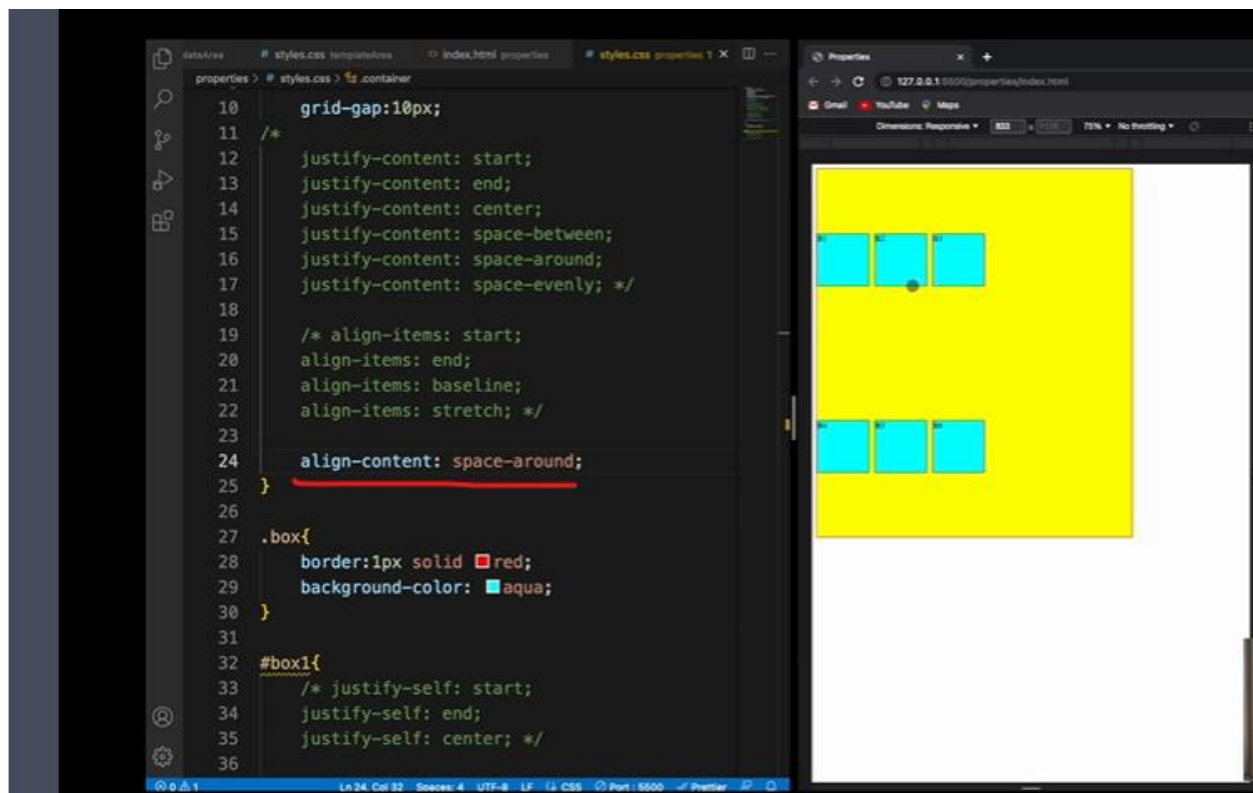






Align-content karne se gap aa gaya





Place-content:center

Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Html/h5.css">
</head>
<body>
  <div class="container">
    <div class="box" id="box1">box1</div>
    <div class="box">box2</div>
    <div class="box">box3</div>
    <div class="box">box4</div>
    <div class="box">box5</div>
    <div class="box">box6</div>
  </div>
</body>
</html>
```

```
.container{
  height: 700px;
  width: 600px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;

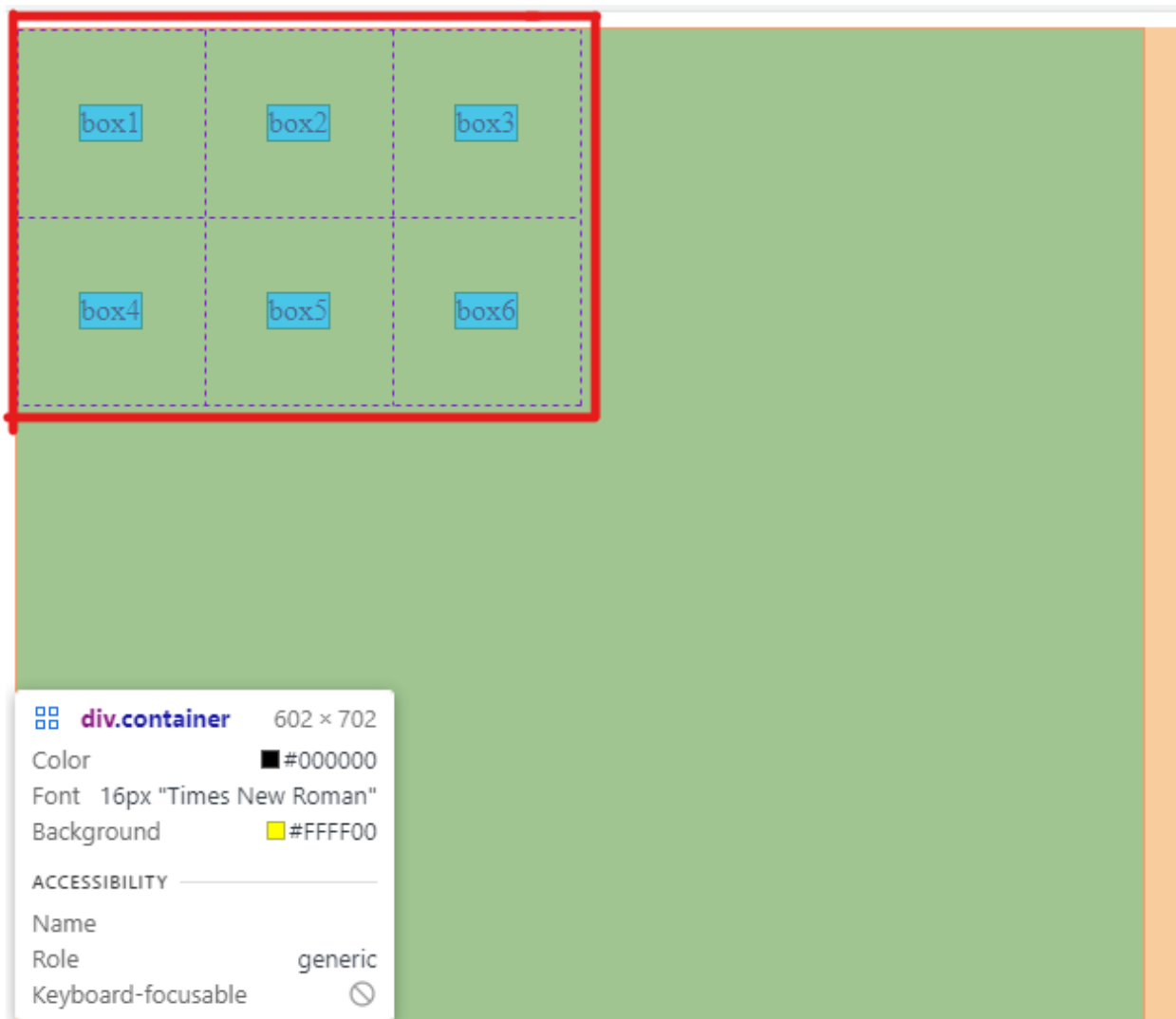
  place-content: center;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
```

box1	box2	box3
box4	box5	box6

Place –items:center

```
.container{
  height: 700px;
  width: 600px;
  border: 1px solid red;
  background-color: yellow;
  display: grid;
  grid-template-columns: 100px 100px 100px;
  grid-template-rows: 100px 100px;

  place-items: center;
}
.box{
  border: 1px solid green;
  background-color: aqua;
}
```



Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Html/post.css">
</head>
<body>
  <div class="container">
    <div class="box" id="header">
      <a href="/Html/h6.html">Click here to move forward</a>
    </div>
    <div class="box" id="sidebar">sidebar</div>
    <div class="box" id="content">content</div>
    <div class="box" id="footer">footer</div>
  </div>
</body>
</html>
```

Css

```
.container{
  border: 1px solid black;
  background-color: yellow;
  height: 700px;
  display: grid;
  grid-template-rows: 30px 1fr 1fr 30px;
  grid-template-columns: 1fr;

  grid-gap: 4px;
  grid-template-areas:
    "hd "
    "side"
    "main"
    "ft";
}
.box{
  border: 1px solid red;
  background-color: aqua;
}
```

```
#header{
  grid-area: hd;
}
#sidebar{
  grid-area: side;
}
#content{
  grid-area: main;
}
#footer{
  grid-area: ft;
}
```

[Click here to move forward](#)

sidebar

content

Footer

Html

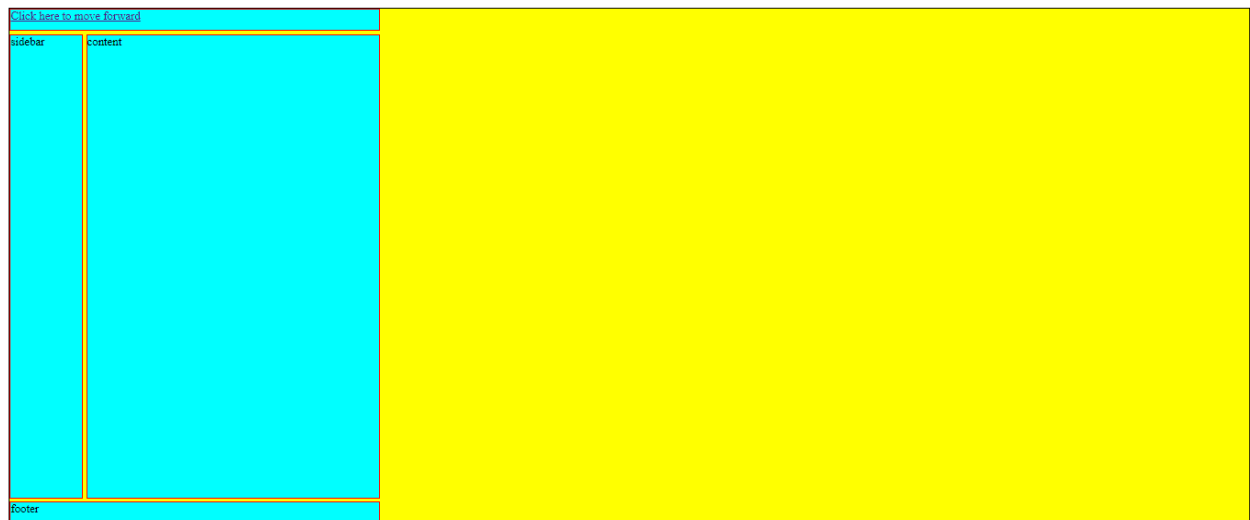
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Html/h6.css">
</head>
<body>
  <div class="container">
    <div class="box" id="header">
      <a href="/Html/post.html">Click here to move forward</a>
    </div>
    <div class="box" id="sidebar">sidebar</div>
    <div class="box" id="content">content</div>
    <div class="box" id="footer">footer</div>
  </div>
</body>
</html>
```

Css

```
.container{
  border: 1px solid black;
  background-color: yellow;
  height: 700px;
  display: grid;
  grid-template-rows: 30px 1fr 30px;
  grid-template-columns: 100px 400px;

  grid-gap: 4px;
  grid-template-areas:
    "hd hd"
    "side main"
    "ft ft";
}
.box{
  border: 1px solid red;
  background-color: aqua;
}
```

```
#header{
  grid-area: hd;
}
#sidebar{
  grid-area: side;
}
#content{
  grid-area: main;
}
#footer{
  grid-area: ft;
}
```



Media-queries

Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Html/h7.css">
</head>
<body>
  <div class="container">
    <div class="box" id="header">header</div>
    <div class="box" id="sidebar">sidebar</div>
    <div class="box" id="content">content</div>
    <div class="box" id="footer">footer</div>
  </div>
</body>
</html>
```

Css

```
/* *{
  margin: 0px;
  padding: 0px;
  box-sizing: border-box;
} */
.container{
  border: 1px solid black;
  background-color: yellow;
  height: 700px;
  display: grid;
  grid-template-rows: 30px 1fr 30px;
  grid-template-columns: 100px 400px;

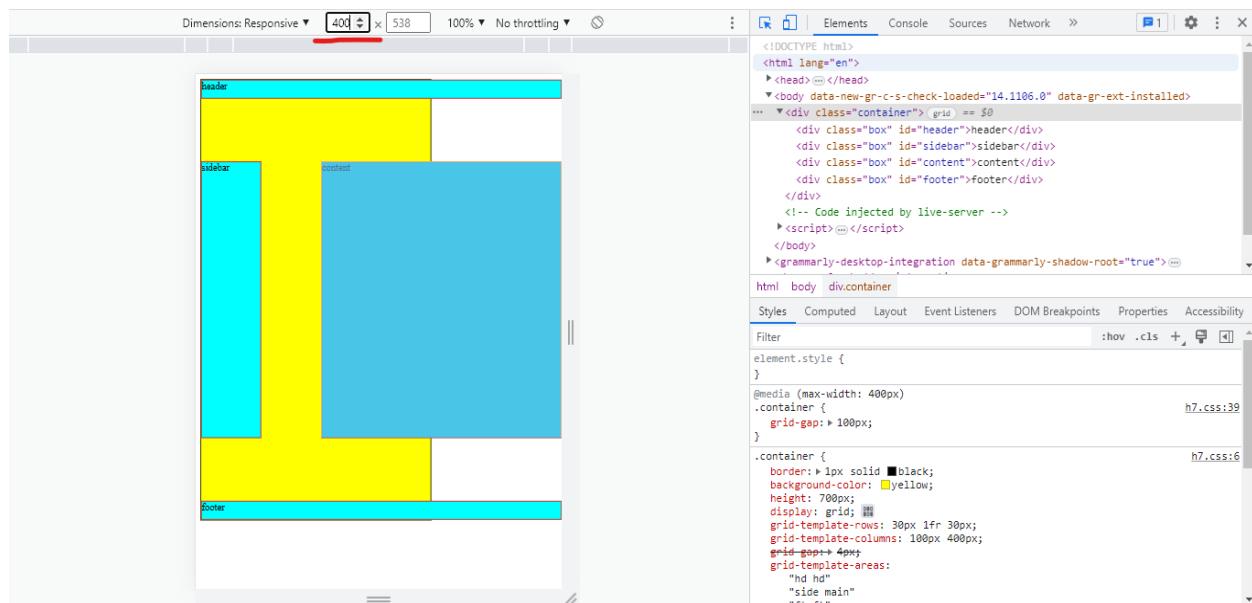
  grid-gap: 4px;
  grid-template-areas:
    "hd hd"
    "side main"
    "ft ft";
}
```

```

.box{
  border: 1px solid red;
  background-color: aqua;
}
#header{
  grid-area: hd;
}
#sidebar{
  grid-area: side;
}
#content{
  grid-area: main;
}
#footer{
  grid-area: ft;
}

@media(max-width:400px){
  .container{
    grid-gap: 100px;
  }
}

```



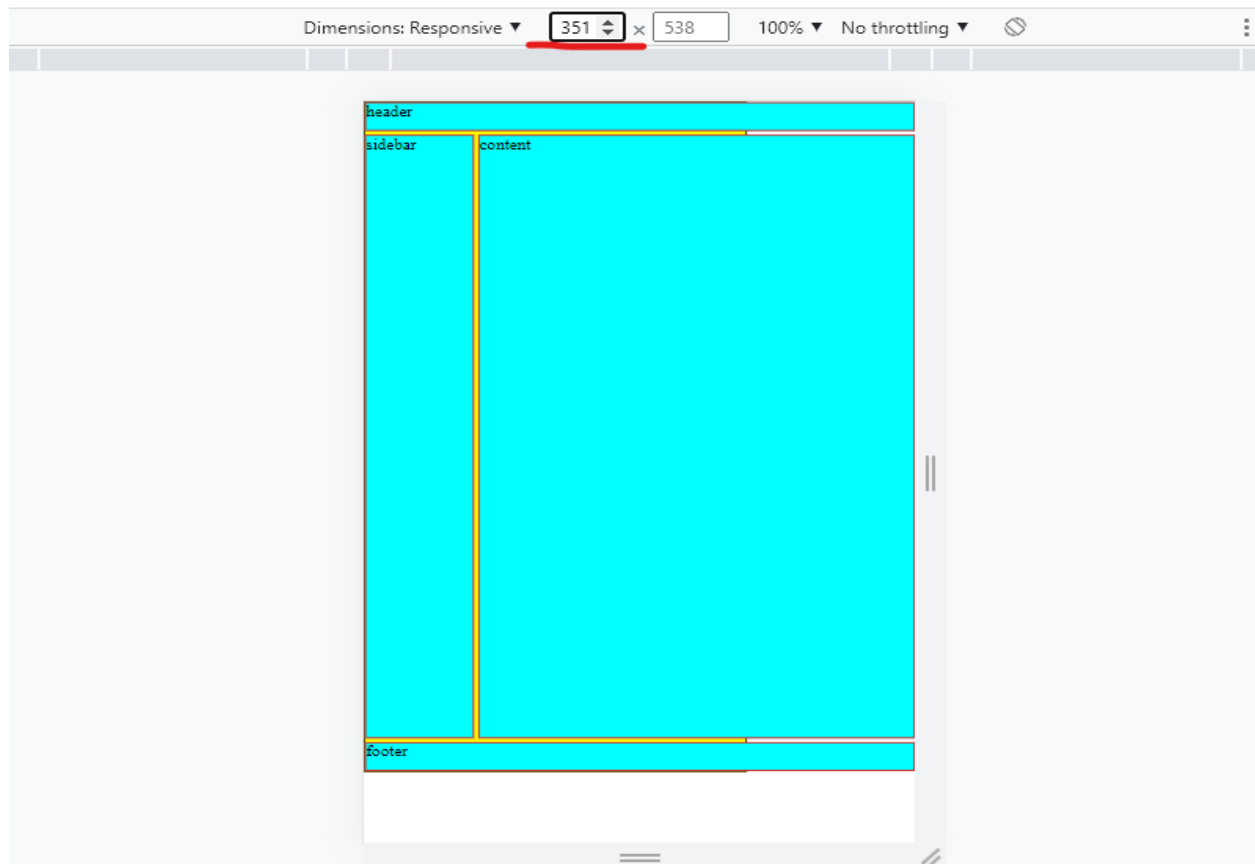
Css

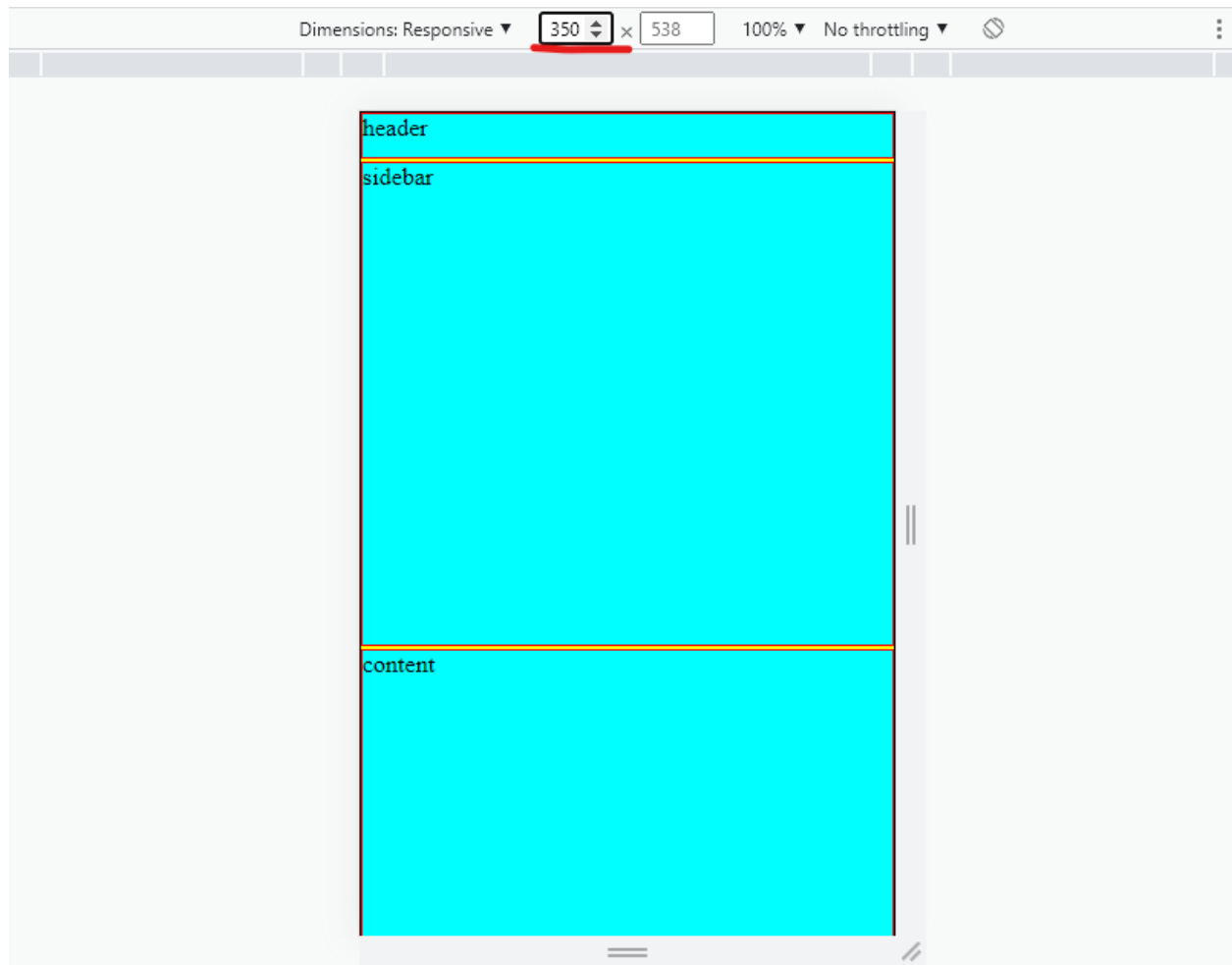
```
*{
    margin: 0px;
    padding: 0px;
    box-sizing: border-box;
}
.container{
    border: 1px solid black;
    background-color: yellow;
    height: 700px;
    width: 510;
    display: grid;
    grid-template-rows: 30px 1fr 30px;
    grid-template-columns: 100px 400px;

    grid-gap: 4px;
    grid-template-areas:
        "hd hd"
        "side main"
        "ft ft";
}
.box{
    border: 1px solid red;
    background-color: aqua;
}
#header{
    grid-area: hd;
}
#sidebar{
    grid-area: side;
}
#content{
    grid-area: main;
}
#footer{
    grid-area: ft;
}

/* @media(max-width:400px){
    .container{
        grid-gap: 100px;
    }
} */
```

```
@media(max-width:350px){  
  .container{  
    grid-template-rows: 30px 1fr 1fr 30px;  
    grid-template-columns: 1fr;  
  
    grid-gap: 2px;  
    grid-template-areas:  
      "hd "  
      "side "  
      "main"  
      "ft";  
  }  
}
```





Nested Grid

Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Html/h8.css">
</head>
<body>
  <div class="container">
    <div class="box" id="header">header</div>
    <div class="box" id="sidebar">sidebar</div>
    <div class="box" id="content">
      <div class="childGrid">
        <div class="element">img1</div>
        <div class="element">img2</div>
        <div class="element">img3</div>
        <div class="element">img4</div>
      </div>
    </div>
    <div class="box" id="footer">footer</div>
  </div>
</body>
</html>
```

Css

```
*{
  margin: 0px;
  padding: 0px;
  box-sizing: border-box;
}
```

```
.container{
  border: 1px solid black;
  background-color: yellow;
  height: 700px;
  width: 510;
  display: grid;
  grid-template-rows: 30px 1fr 30px;
  grid-template-columns: 100px 400px;

  grid-gap: 4px;
  grid-template-areas:
    "hd hd"
    "side main"
    "ft ft";
}

.box{
  border: 1px solid red;
  background-color: aqua;
}

#header{
  grid-area: hd;
}

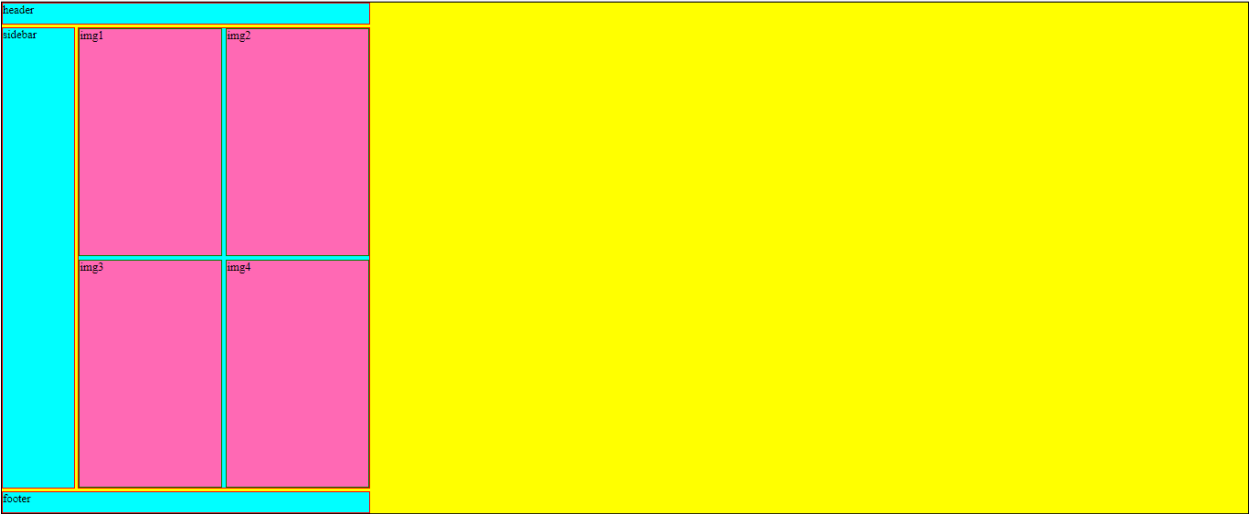
#sidebar{
  grid-area: side;
}

#content{
  grid-area: main;
}

#footer{
  grid-area: ft;
}

.childGrid{
  height: 100%;
  width:100%;
  display: grid;
  grid-template-columns: 1fr 1fr;
  grid-template-rows: 1fr 1fr;
  grid-gap: 5px;
}

.element{
  border: 1px solid green;
  background-color: hotpink;
}
```

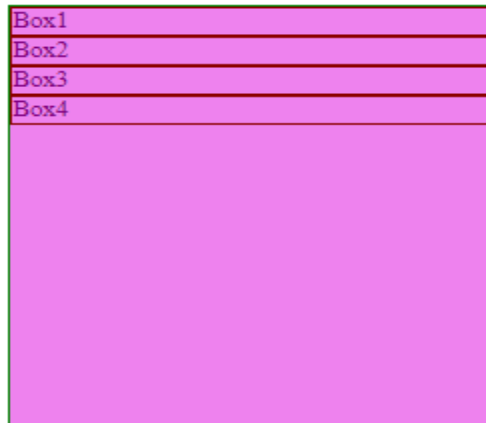
CSS Functions, Variables, Transitions & Animations

Local var()

Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Html/h9.css">
</head>
<body>
  <div class="container">
    <div class="box">Box1</div>
    <div class="box">Box2</div>
    <div class="box">Box3</div>
    <div class="box">Box4</div>
  </div>
</body>
</html>
```

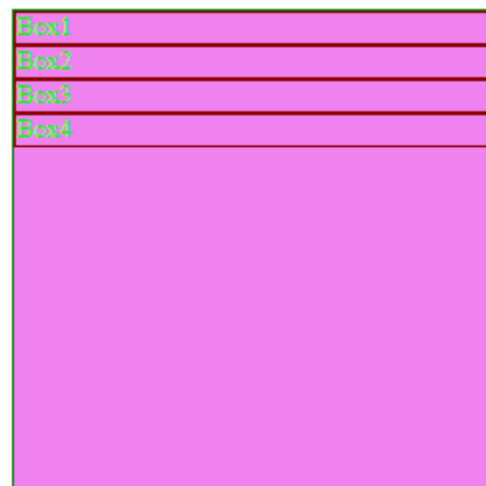
```
.container{
  border: 1px solid green;
  background-color:violet;
  height: 300px;
  width: 300px;
}
.box{
  border: 1px solid darkred;
  color: purple;
}
```



Html part same ,only change in css

```
.container{
  border: 1px solid green;
  background-color:violet;
  height: 300px;
  width: 300px;
}
.box{
  border: 1px solid darkred;
  color: purple;

  --lime:#00FF00;
  color:var(--lime)
}
```

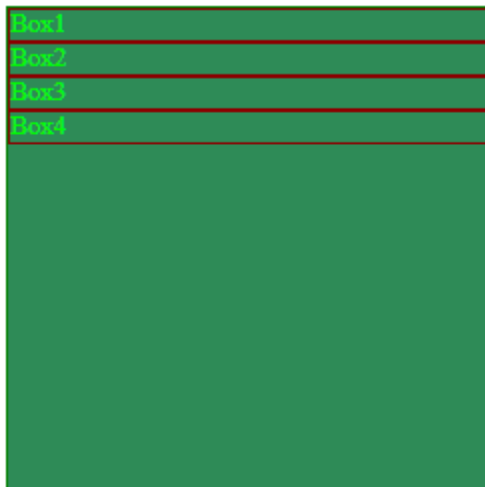


Global variable

Css

```
:root{
  --seagreen:#2E8B57;
}
.container{
  border: 1px solid green;
  background-color:violet;
  height: 300px;
  width: 300px;
  background-color: var(--seagreen);
}
.box{
  border: 1px solid darkred;
  color: purple;

  --lime:#00FF00;
  color:var(--lime)
}
```



Transitions in CSS

Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Html/h10.css">
</head>
<body>
  <div class="container">
    <div class="box">Box1</div>
  </div>
</body>
</html>
```

Css

```
.container{
  height: 500px;
  width: 500px;
  background-color: green;
  border: 3px solid red ;
  display: flex;
}
.box{
  height: 150px;
  width: 150px;
  background-color:aqua;
  border: 2px solid orange ;
  margin: 20px;
  transition: 2s;
}
.box:hover{
  background-color: magenta;
}
```

Box1



↑ transition: $1s_j$

-
- A diagram illustrating a function f mapping from a set Q_n to a set B_n . A curved arrow labeled f points from a box labeled Q_n to a box labeled B_n .

Animation

Animation Keyframe

No Sound

↓ 0°
Reload

transition:- $\xleftarrow[\text{property}]{\text{by-color}}$ $\xleftrightarrow[2s]{\text{duration}}$ $\xleftrightarrow[\text{transition ring func}]{\text{each call and call-intr}}$ $\xrightarrow[4s]{\text{delay}}$

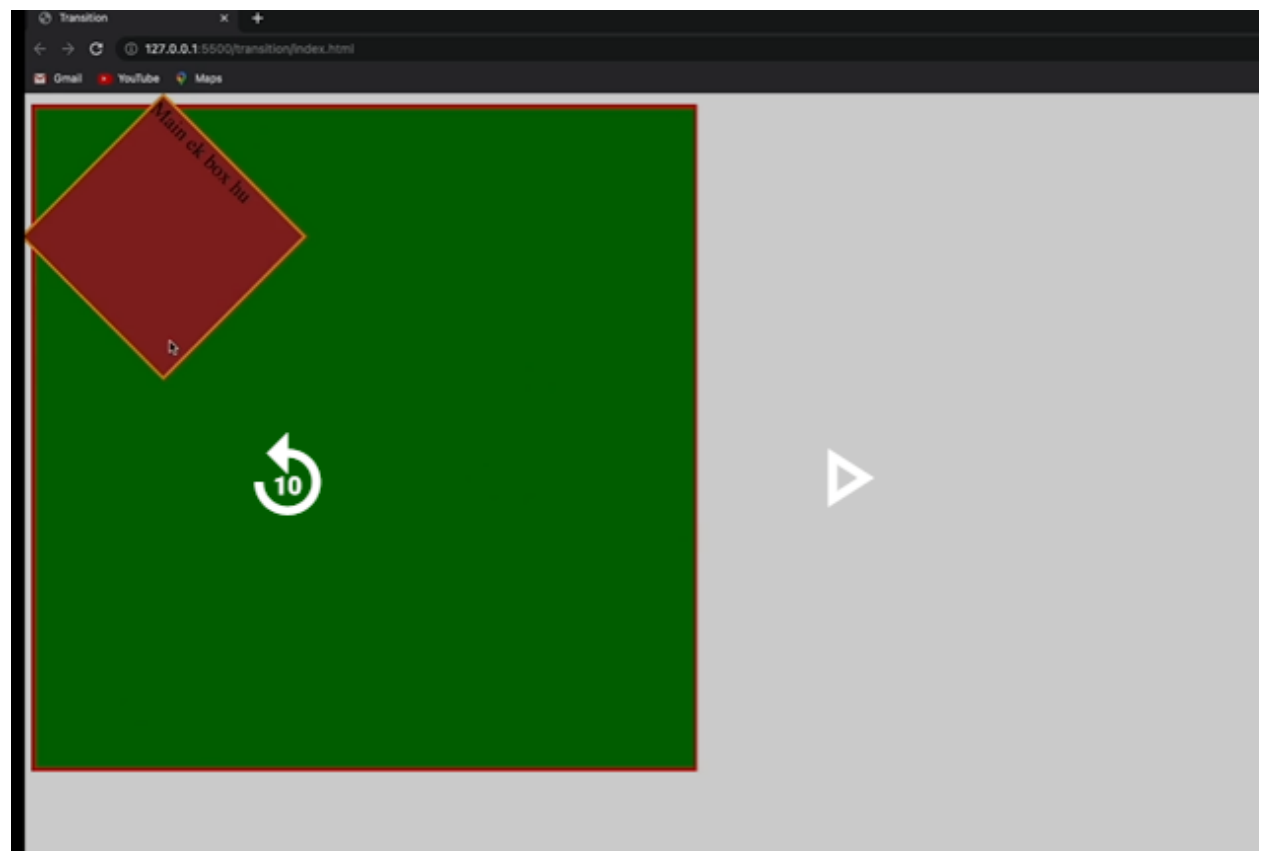
styles.css — animations

Get Started index.html # styles.css x

transition > # styles.css > .box

Press Esc to exit full screen

```
10     height:150px;
11     width:150px;
12     margin:20px;
13     border:2px solid orange;
14     background-color: aqua;
15
16     /* transition: 2s; */
17
18     /* transition-property: background-color; */
19     /* transition-delay: 5s; */
20
21     10 transition: background-color 0.3s ease ; 10
22     transition: all 200ms ease 1s;
23
24 }
25
26 .box:hover{
27     background-color: brown;
28     transform: rotate(45deg);
29 }
```



Key frame

Html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Html/h11.css">
</head>
<body>
  <div class="container">
    <div class="box">Box1</div>
  </div>
</body>
</html>
```

Css

```
.container{
  width: 100%;
  height: 214px;
  background-color: yellow;
  border: 2px solid red;
}
.box{
  width: 212px;
  height: 212px;
  background-color: aqua;
  border: 2px solid green;

  animation-name: anandkj;
  animation-duration: 3s;
  animation-iteration-count: 4;
  animation-timing-function: ease-in;
  animation-delay: 1s;
}
@keyframes anandkj{
  from{
    width: 212px;
  }
  to{
    width: 800px;
  }
}
```

}



CSS Functions, Variables, transitions & Animations [HD]

Transitions in Css :

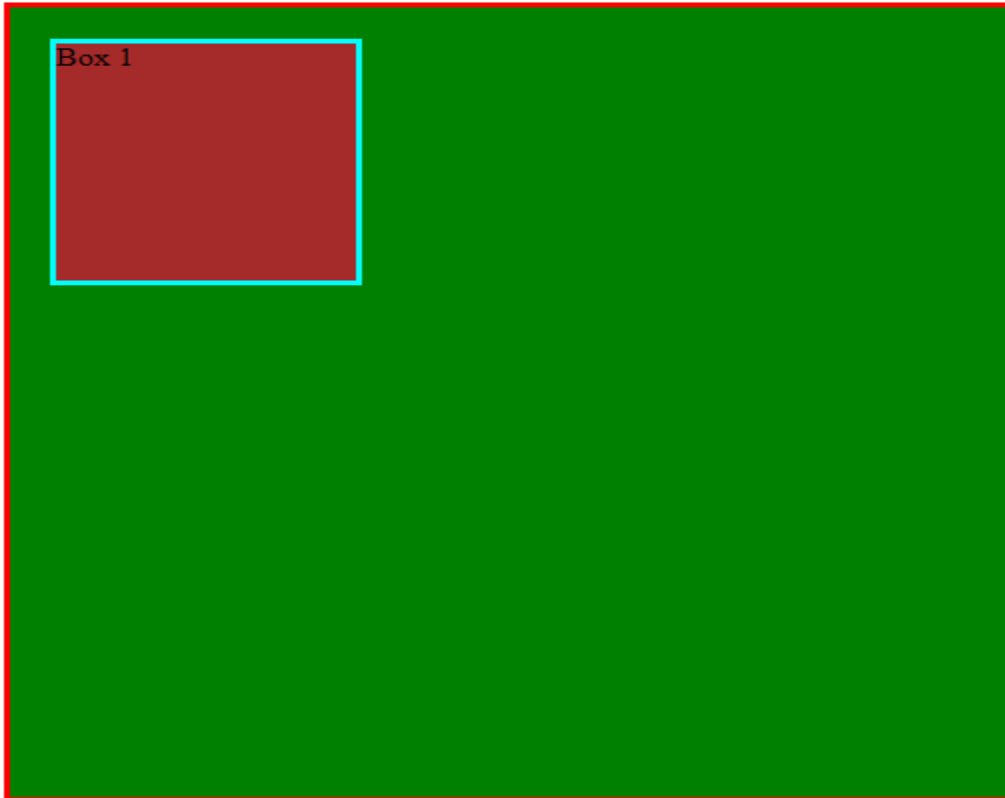
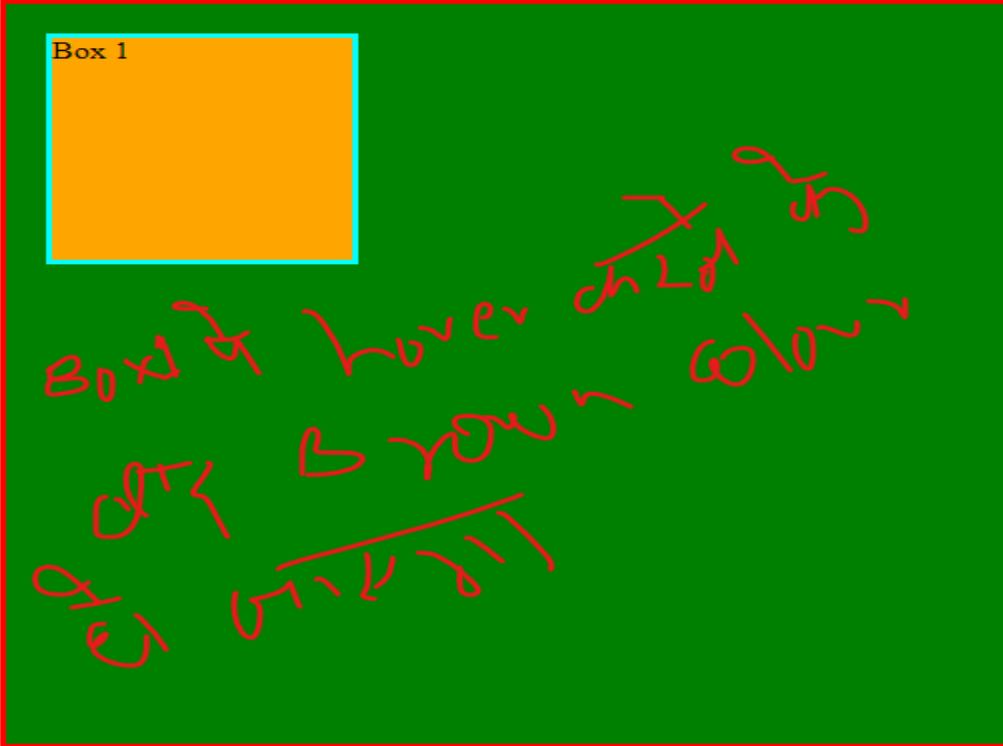
Html code

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Transitions/style.css">
</head>
<body>
  <div class="container">
    <div class="box1">Box 1</div>
  </div>
</body>
</html>
```

Css code

```
.container{
  height:500px;
  width:500px;
  background-color: green;
  border:3px solid red;
  display: flex;
}
.box1{
  height:150px;
  width:150px;
  margin: 20px;
  background-color: orange;
  border:3px solid aqua;
}
.box1:hover{
  background-color: brown;
}
```

Output:



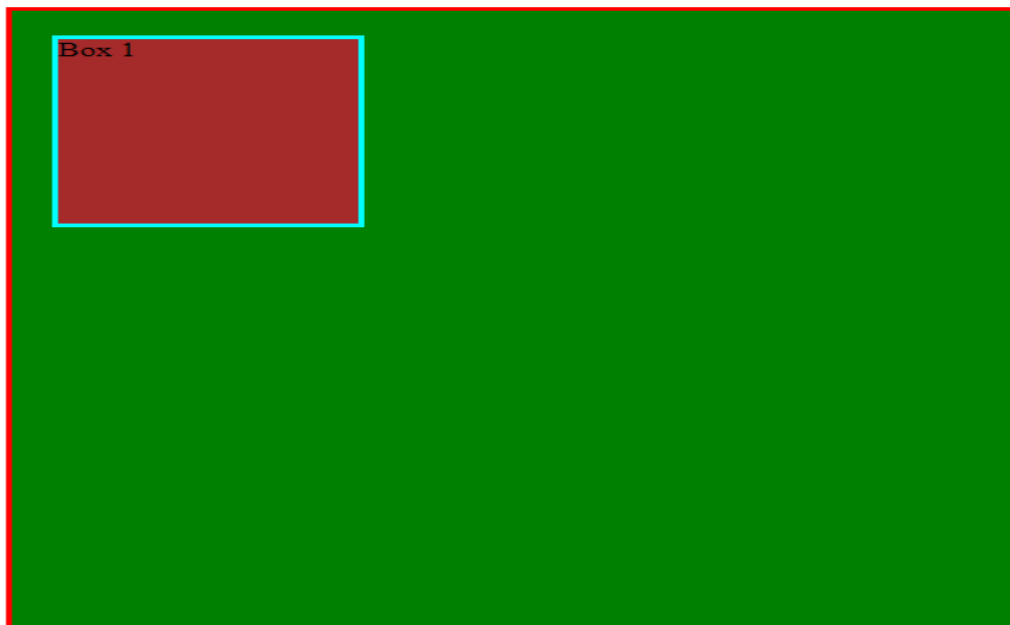
1) transitions

Html code same

Css code :

```
.container{
  height:500px;
  width:500px;
  background-color: green;
  border:3px solid red;
  display: flex;
}
.box1{
  height:150px;
  width:150px;
  margin: 20px;
  background-color: orange;
  border:3px solid aqua;
  transition: 2s;
}
.box1:hover{
  background-color: brown;
}
```

Output:

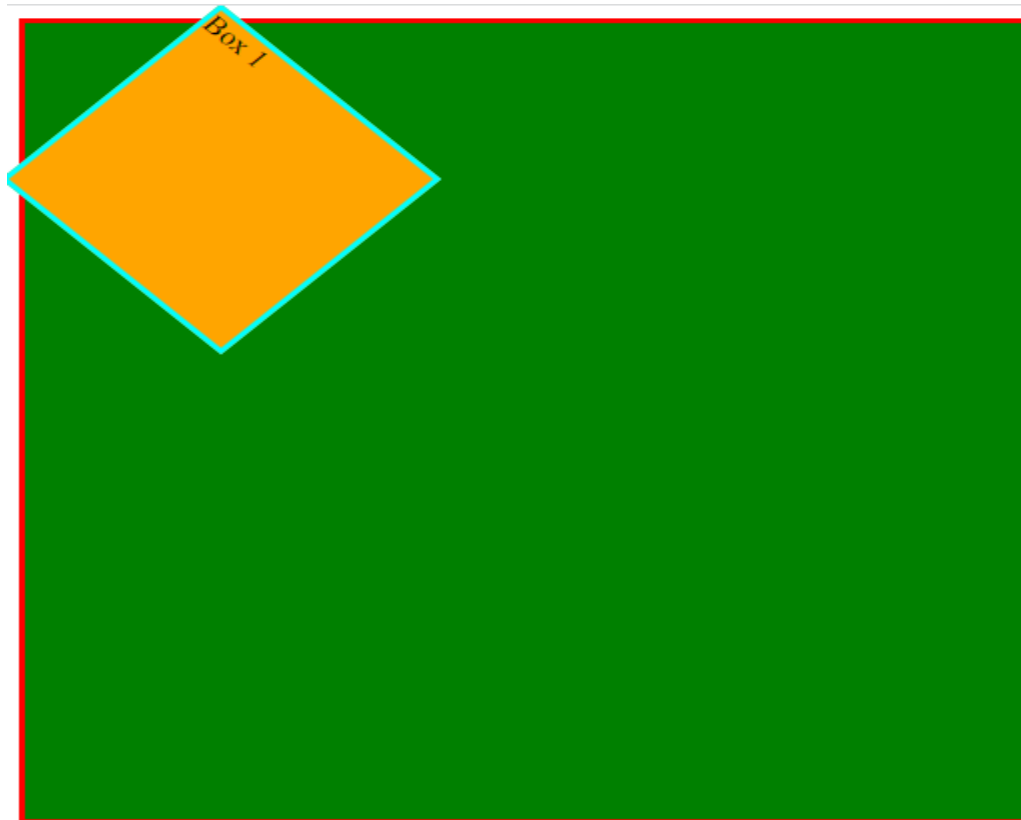


Mouse hover karne ke baad background color change ho jayega jo ki 2 seconde lega

Html part same

Css code :

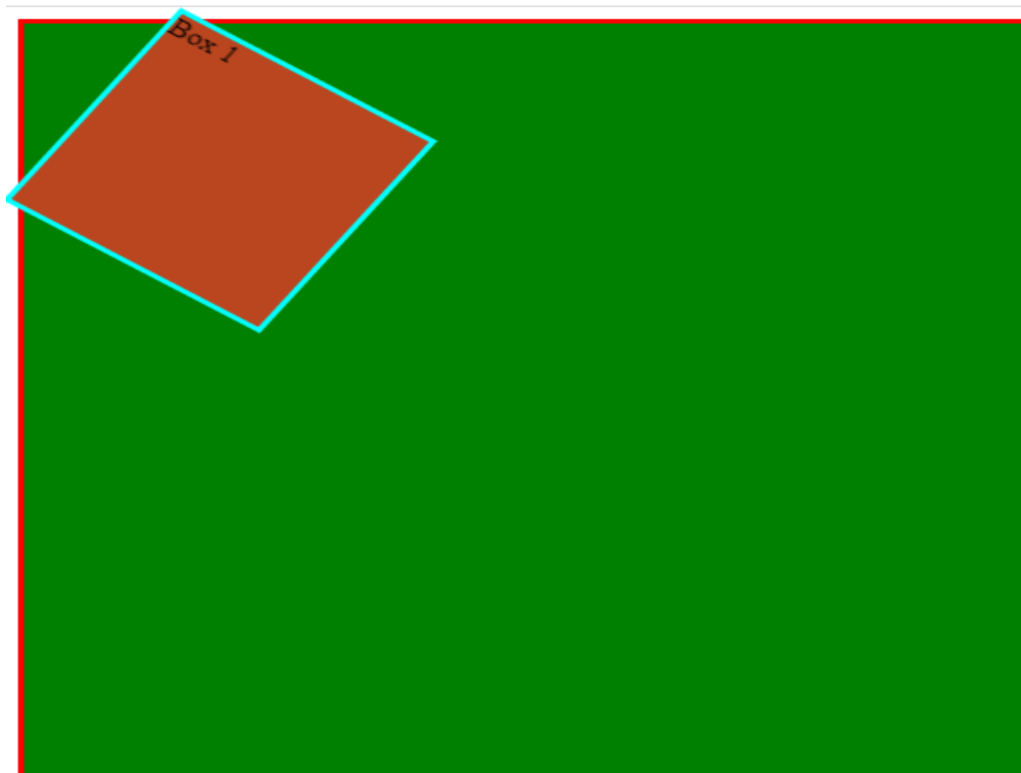
```
.container{
  height:500px;
  width:500px;
  background-color: green;
  border:3px solid red;
  display: flex;
}
.box1{
  height:150px;
  width:150px;
  margin: 20px;
  background-color: orange;
  border:3px solid aqua;
  transition: transform 1s ease 3s;
}
.box1:hover{
  transform: rotate(45deg);
}
```



Html part same

Css code

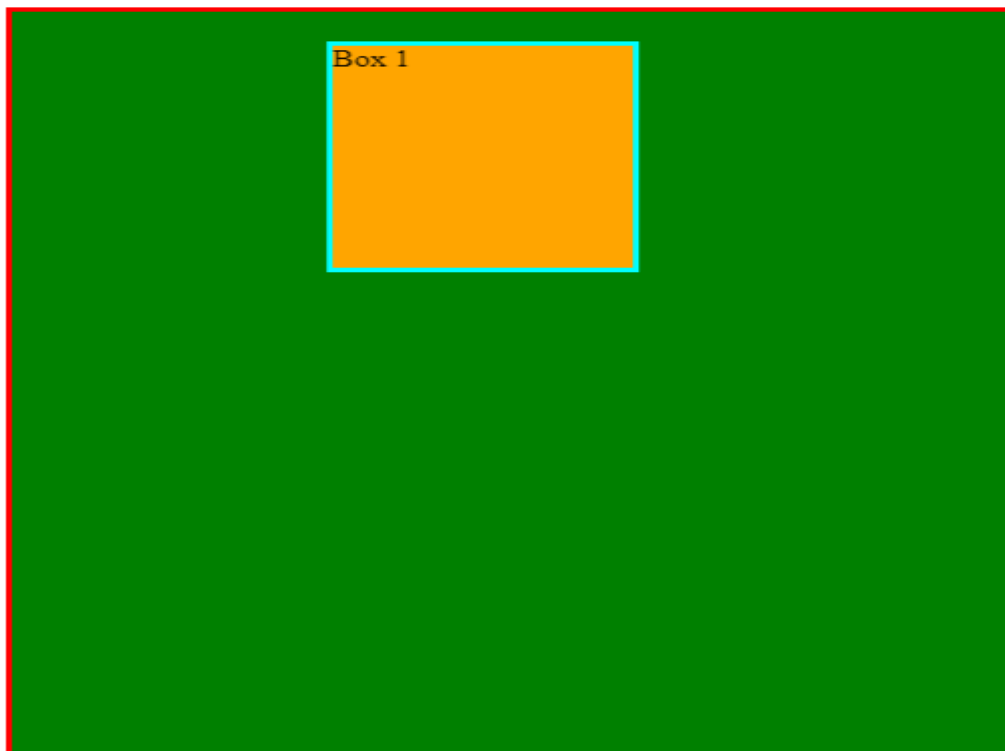
```
.container{
  height:500px;
  width:500px;
  background-color: green;
  border:3px solid red;
  display: flex;
}
.box1{
  height:150px;
  width:150px;
  margin: 20px;
  background-color: orange;
  border:3px solid aqua;
  transition: all 5s ease ;
}
.box1:hover{
  transform: rotate(45deg);
  background-color: brown;
}
```



Html part same

Css code

```
.container{
  height:500px;
  width:500px;
  background-color: green;
  border:3px solid red;
  display: flex;
}
.box1{
  height:150px;
  width:150px;
  margin: 20px;
  background-color: orange;
  border:3px solid aqua;
  transition: transform 1s ease-in ;
}
.box1:hover{
  transform: translateX(500px);
}
```



Animation properties

Html code

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="/Transitions/style1.css">
</head>
<body>
  <div class="container">
    <div class="box1">box1</div>
  </div>
</body>
</html>
```

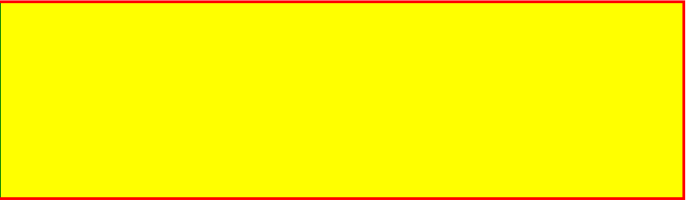
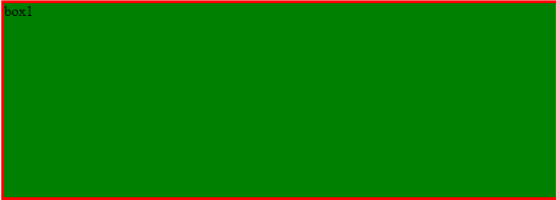
Css code

```
.container{
  height: 100%;
  width: 100%;
  background-color: yellow;
  border: 3px solid red;
}
.box1{
  height: 212px;
  width: 212px;
  background-color: green;
  border-color: brown;

  animation-name: anandkumar;
  animation-duration: 3s;
  animation-iteration-count: 4;
  animation-timing-function: ease-in;
  animation-delay: 1s;
}
@keyframes anandkumar {
  from {
    width: 212px;
  }
  to{
    width: 900px;
  }
}
```

```
}  
}
```

box1



Html part same

Css code

```
.container{
  height: 100%;
  width: 100%;
  background-color: yellow;
  border: 3px solid red;
}
.box1{
  height: 212px;
  width: 212px;
  background-color: green;
  border-color: brown;

  animation-name: anandkumar;
  animation-duration: 3s;
  animation-iteration-count: 4;
  animation-timing-function: ease-in;
  animation-delay: 1s;
  animation-direction: alternate;
}

@keyframes anandkumar {
  0% {
    width: 212px;
    background-color: orange;
  }
  45%{
    width: 35%;
    background-color: white;
  }
  100%{
    width: 100%;
    background-color: green;
  }
}
```

box1

