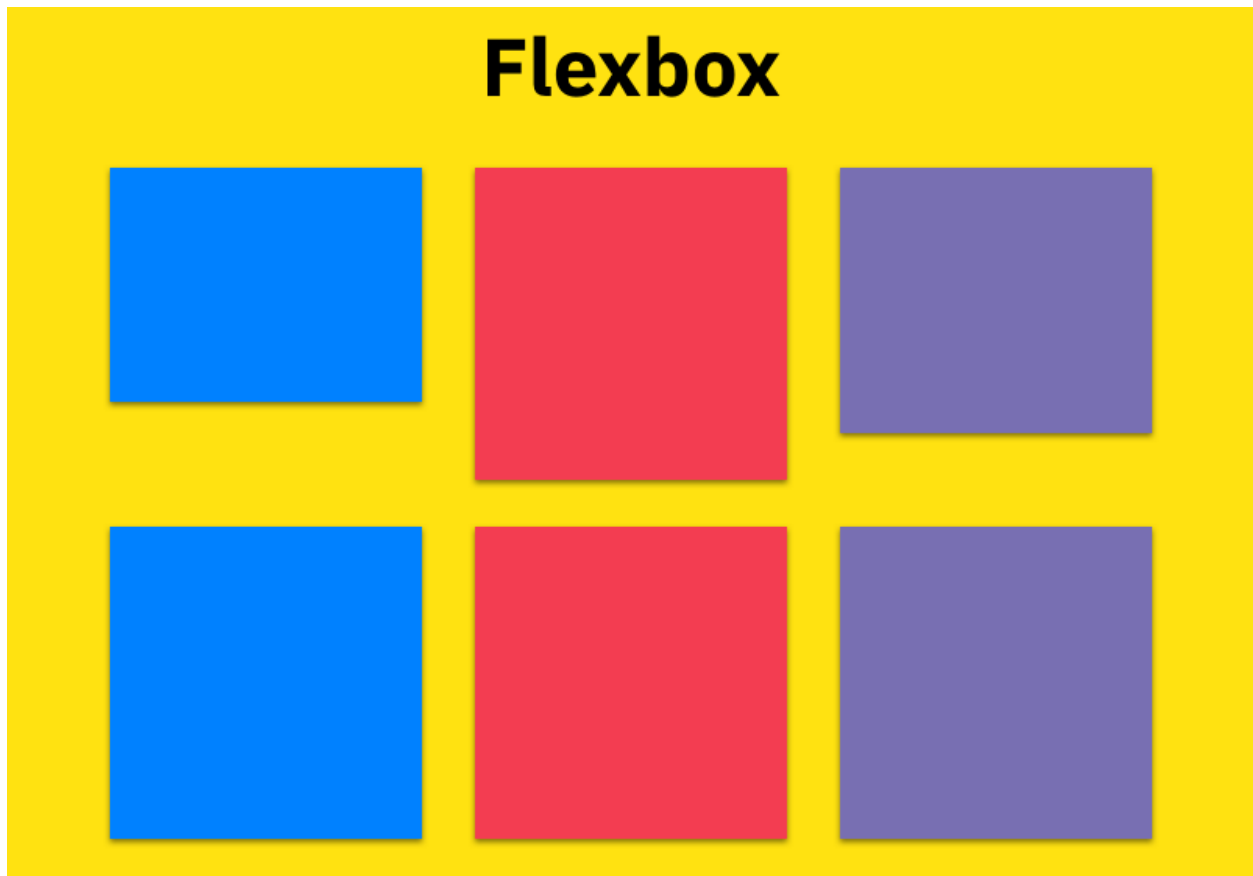
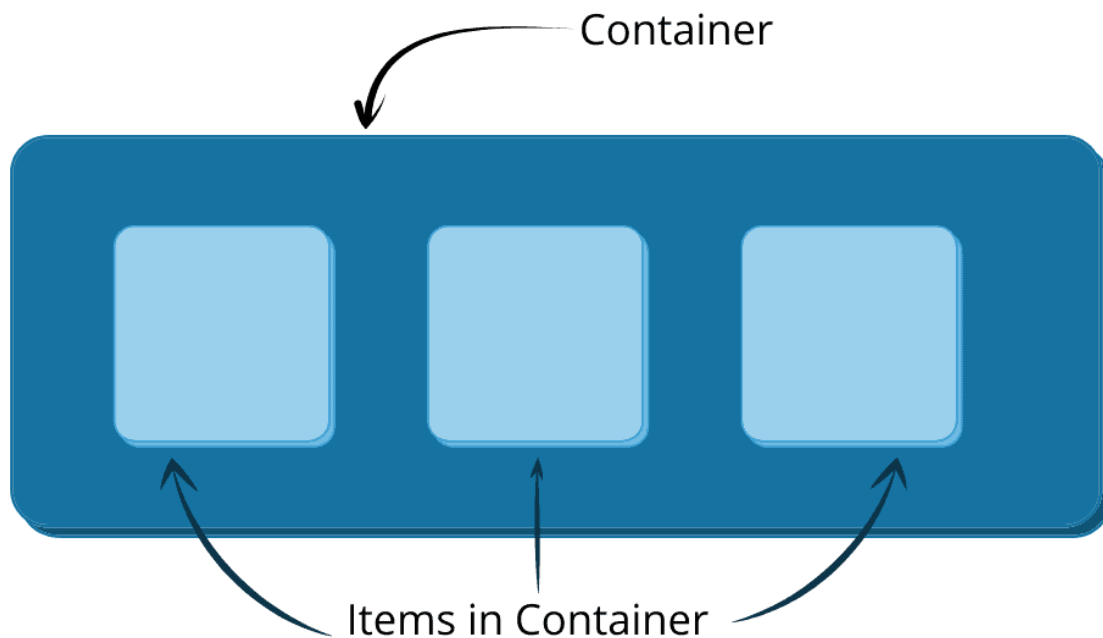


# CSS Flexbox



Flexbox คือ layout mode แบบใหม่ ที่เรียกว่า “Flex Layout” โดย flex layout มีลักษณะเป็นกล่อง ที่มีความยืดหยุ่นมากขึ้น (Flexible Box หรือเรียกสั้น ๆ ว่า Flexbox) ซึ่งเป็นเครื่องมือที่ CSS ใช้ช่วยในการจัดการตำแหน่งของ Element ต่าง ๆ ที่มีความซับซ้อนได้เป็นอย่างดี โดยจะมีความยืดหยุ่นในการใช้งานที่ค่อนข้างสูงมาก ๆ ซึ่งนั่นก็หมายความว่า สามารถจัด Layout ได้ในหลาย ๆ รูปแบบ



รูปแบบการทำงานนั้น จำเป็นที่จะต้องมีย่อแม่ คือ **Container** เพื่อคลุมกล่องลูก คือ **items**

จากรูปด้านบนจะสังเกตเห็นได้ว่า จะมี Container คลุม items ไว้อยู่ ซึ่งการใช้งาน Flex layout จะต้องถูกใช้ผ่าน Container เพื่อควบคุม items ที่อยู่ด้านในเท่านั้น

โดยรูปแบบการเขียนจะใช้ property เป็น display แล้วตามด้วย value เป็น flex บน Container ทดสอบดังนี้

#### HTML

```
<body>
  <div class="container">
    <div class="items">1</div>
    <div class="items">2</div>
    <div class="items">3</div>
    <div class="items">4</div>
    <div class="items">5</div>
  </div>
</body>
```

## รูปแบบของ Flexbox

- "flex-direction"
- "flex-wrap"
- "justify-content"
- "align-items"

### Flex-Direction

flex-direction คือ การจัดเรียง items ให้อยู่ในรูปแบบ row(แถว) หรือ column(คอลัมน์)  
โดย default จะเรียง items เป็นแนวนอน แต่สามารถกำหนดตามที่ต้องการได้



เพิ่ม flex-direction เป็นแนว Column



คำถาม flex-direction กำหนดได้กี่แบบ ..... อะไรบ้าง..... ????

## Flex-Wrap

Flex-wrap คือ การจัด items เมื่อ items นั้นถูกบังคับก็จะทำการเลื่อนตัวลงมาบรรทัดใหม่

```
.container{  
  display:flex;  
  flex-wrap: wrap;  
}
```

ก่อนใช้ wrap

หลังใช้ wrap

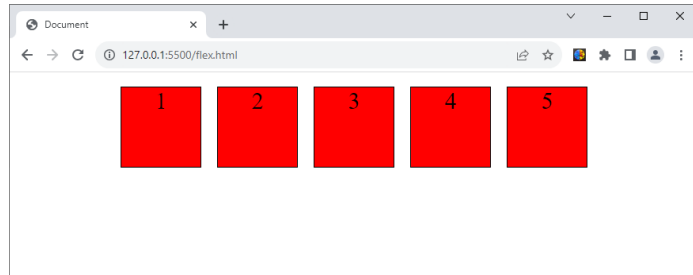
คำถาม flex-wrap กำหนดได้ที่แบบ ..... อะไรบ้าง..... ????

## Justify-Content

justify-content คือ การรูปแบบจัด items ที่อยู่ในแกน x หรือการจัดรูปแบบในแนวนอนทั้งหมด ซึ่งจะจัดให้ items ให้อยู่ในรูปแบบต่าง ๆ ไม่ว่าจะเป็นซ้าย กลาง หรือ ขวา โดยมี 3 รูปแบบ ดังนี้

### Center

```
.container{  
  display:flex;  
  justify-content: center;  
}
```



### flex-start , flex-end

```
.container{  
  display:flex;  
  justify-content:flex-start;  
}
```



flex-start

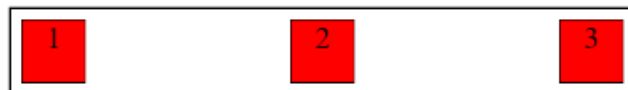
```
.container{  
  display:flex;  
  justify-content:flex-end;  
}
```



flex-end

### Space-between

```
.container{  
  display:flex;  
  justify-content:space-between;  
}
```



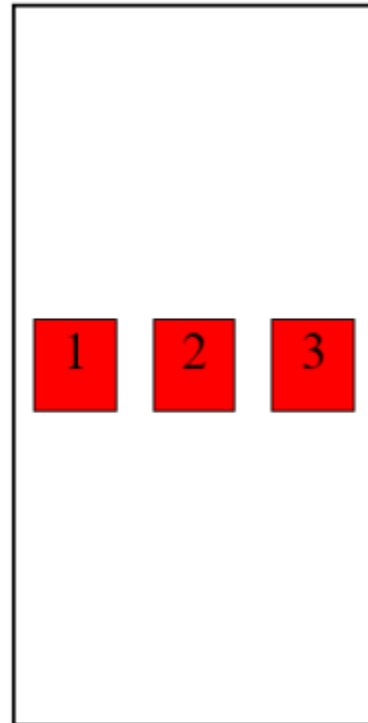
space-between

## align-items

align-items คือ การจัดรูปแบบของ items ที่อยู่ในแกน y หรือในแนวตั้ง มี 4 รูปแบบ

### Center

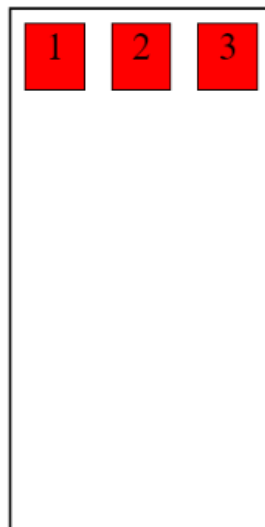
```
.container{  
  display:flex;  
  align-items: center;  
}
```



### flex-start, flex-end

```
.container{  
  display:flex;  
  align-items:flex-start;  
}
```

```
.container{  
  display:flex;  
  align-items:flex-end;  
}
```



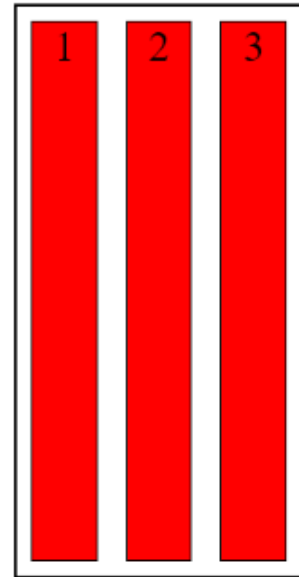
flex-start



flex-end

## Stretch

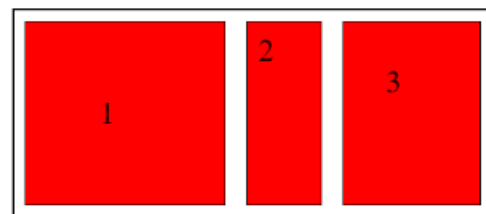
```
.container{  
  display:flex;  
  align-items:stretch;  
}
```



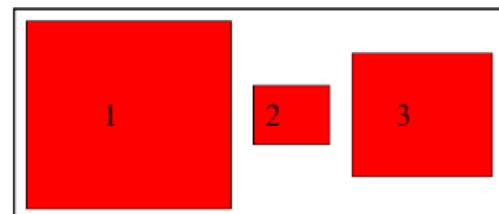
stretch

## Baseline

```
.container{  
  display:flex;  
  align-items:baseline;  
}
```



ก่อนใช้ baseline



หลังใช้ baseline

# Trick ที่ควรรู้ใน Flexbox

## Flex-Flow

flex-flow คือ shorthand property เพื่อปรับแต่ง Flex-direction และ Flex-wrap ให้เขียนน้อยลง โดยที่ถ้าเราต้องการที่จะเขียน Flex-direction และ Flex-wrap ใน Element เดียวกันเราก็ต้องใช้เวลาเขียนถึง 2 อันเลยใช่ไหมครับ แต่ถ้าใช้ Flex-flow เราจะเขียนแค่ Flex-flow อันเดียวแล้วใช้ value ของ Flex-direction และ Flex-wrap นั้นเอง ยังไง ? มาดูกัน

```
.container{
  display:flex;
  flex-direction: column;
  flex-wrap: wrap;
}
```

Flex-direction และ Flex-wrap

```
.container{
  display:flex;
  flex-flow: column wrap;
}
```

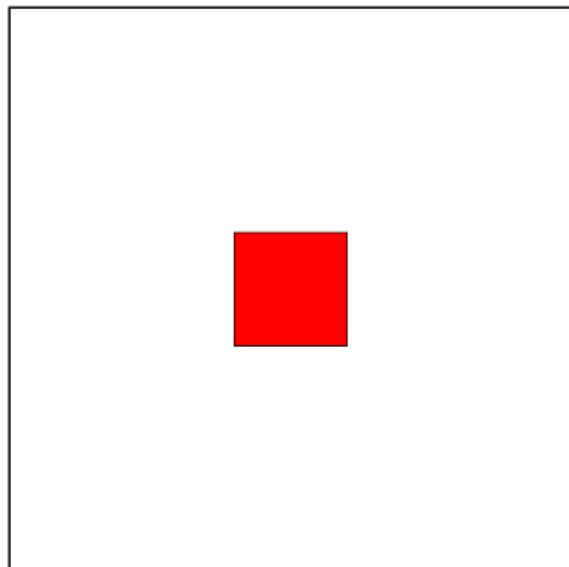
Flex-flow

โดยจะสังเกตได้ว่าถ้าใช้ Flex-flow นั้นจะเรียงโดยใช้ Flex-direction ขึ้นก่อนเป็น column ตามมาด้วย Flex-wrap ที่เป็น wrap นั้นเอง และยังทำให้ Code สั้นลงไปอีกด้วย

## Perfect Centering

สำหรับ perfect centering คือ การให้ items ใน container นั้นอยู่ตรงกลางแบบสมบูรณ์นั่นเอง

```
.container{
  display:flex;
  justify-content: center;
  align-items: center;
}
```



Perfect Centering



## The order property

CSS

```
.flex-container{
  display: flex;
  align-items: stretch;
  background-color: #f1f1f1;
}
.flex-container>div{
  background-color: rgb(209, 78, 8);
  color: aliceblue;
  width: 100px;
  margin: 10px;
  text-align: center;
  line-height: 75px;
  font-size: 30px;
}
```

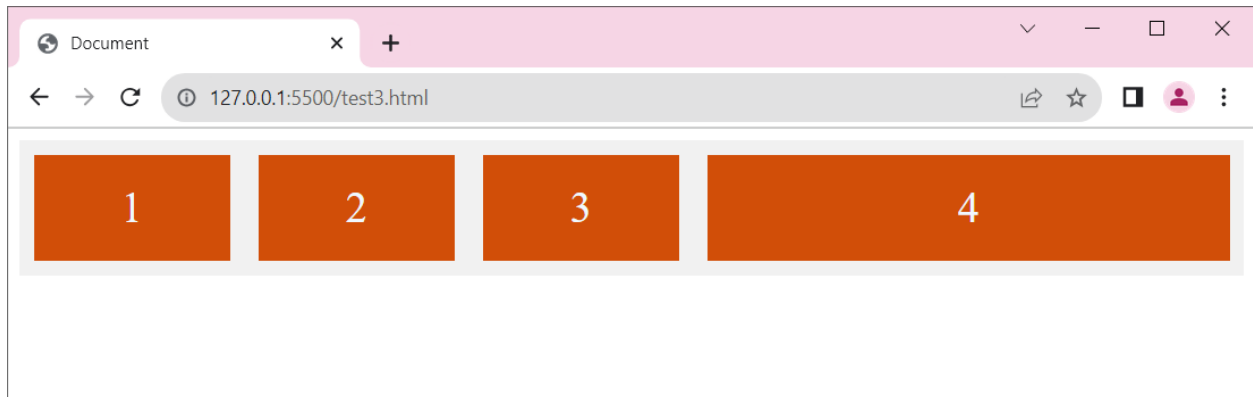
html

```
<div class="flex-container">
  <div style="order: 3">1</div>
  <div style="order: 2">2</div>
  <div style="order: 4">3</div>
  <div style="order: 1">4</div>
</div>
```



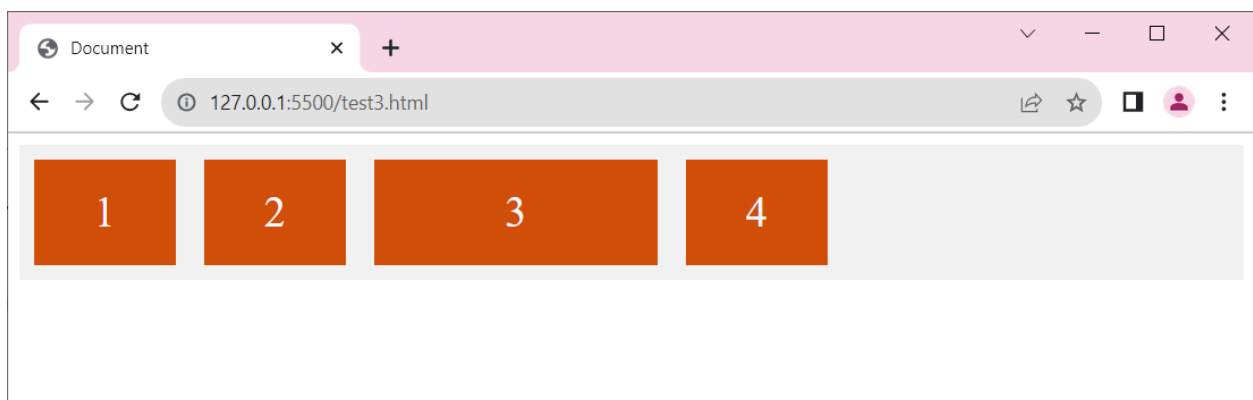
## The flex-grow Property

```
<div class="flex-container">
  <div style="flex-grow: 1">1</div>
  <div style="flex-grow: 1">2</div>
  <div style="flex-grow: 1">3</div>
  <div style="flex-grow: 7">4</div>
</div>
```



## The flex-basis Property

```
<div class="flex-container">
  <div>1</div>
  <div>2</div>
  <div style="flex-basis: 200px">3</div>
  <div>4</div>
</div>
```



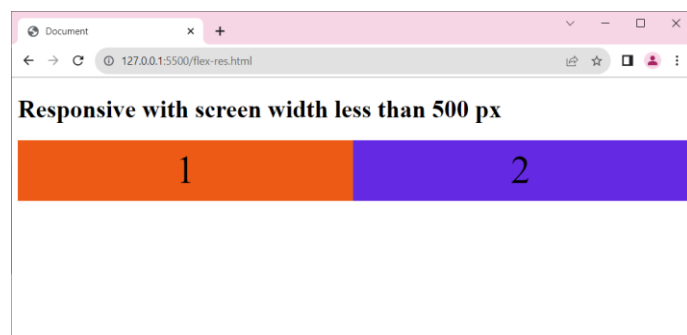
# CSS Flex Responsive

html

```
<h1>Responsive with screen width less than 500 px </h1>
<div class="flex-container">
  <div class="flex-item-left">1</div>
  <div class="flex-item-right">2</div>
</div>
```

CSS

```
/*
  box-sizing: border-box;
}
.flex-container{
  display:flex;
  flex-direction: row;
  font-size: 50px;
  text-align: center;
}
.flex-item-left{
  background-color: #ec5a16;
  padding: 10px;
  flex: 50%;
}
.flex-item-right{
  background-color: #6429e2;
  padding: 10px;
  flex: 50%;
}
/* Responsive layout*/
@media (max-width:500px) {
  .flex-container{
    flex-direction: column;
  }
}
```



# Responsive Website using Flexbox

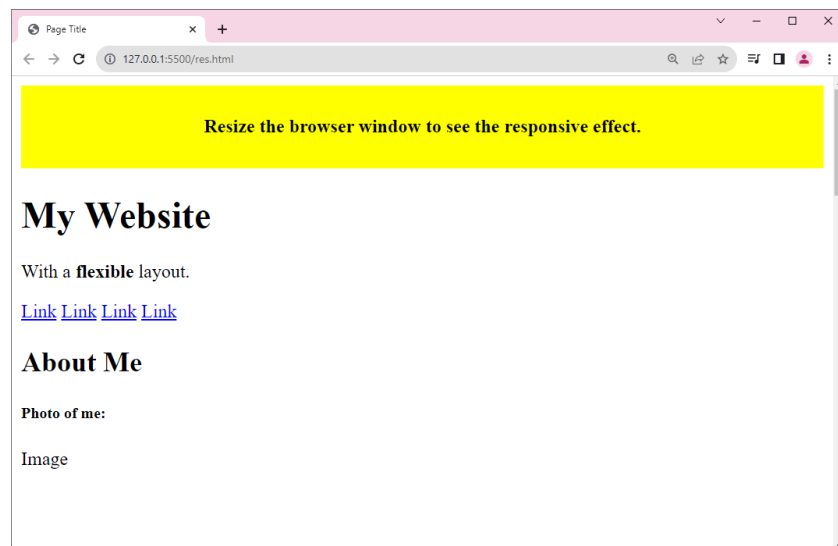
สร้างไฟล์ *test-res-web.html*

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <title>Page Title</title>
5  <meta charset="UTF-8">
6  <meta name="viewport" content="width=device-width, initial-scale=1">
7  <link rel="stylesheet" href="res.css">
8  </head>
9  <body>
10
11  <!-- Note -->
12  <div style="background: yellow;padding:5px">
13      <h4 style="text-align:center">
14          Resize the browser window to see the responsive effect.
15      </h4>
16  </div>
17
18  <!-- Header -->
19  <div class="header">
20      <h1>My Website</h1>
21      <p>With a <b>flexible</b> layout.</p>
22  </div>
23
24  <!-- Navigation Bar -->
25  <div class="navbar">
26      <a href="#">Link</a>
27      <a href="#">Link</a>
28      <a href="#">Link</a>
29      <a href="#">Link</a>
30  </div>
31
32  <!-- The flexible grid (content) -->
33  <div class="row">
34      <div class="side">
35          <h2>About Me</h2>
36          <h5>Photo of me:</h5>
37          <div class="fakeimg" style="height:200px;">Image</div>
38          <p>ทดสอบการแสดงผลบนหน้าเว็บ </p>
```

```

39     <h3>More Text</h3>
40     <p>แสดงเมนู</p>
41     <div class="fakeimg" style="height:60px;">Image</div><br>
42     <div class="fakeimg" style="height:60px;">Image</div><br>
43     <div class="fakeimg" style="height:60px;">Image</div>
44 </div>
45 <div class="main">
46     <h2>TITLE HEADING</h2>
47     <h5>Title description, Dec 7, 2017</h5>
48     <div class="fakeimg" style="height:200px;">Image</div>
49     <p>แสดงหัวข้อข่าว</p>
50     <p>มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือวิทยาเขตปทุมธานี</p>
51     <br>
52     <h2>TITLE HEADING</h2>
53     <h5>Title description, Sep 2, 2017</h5>
54     <div class="fakeimg" style="height:200px;">Image</div>
55     <p>Some text..</p>
56     <p>คณะเทคโนโลยีและการจัดการอุตสาหกรรม</p>
57 </div>
58 </div>
59
60 <!-- Footer -->
61 <div class="footer">
62     <h2>Footer</h2>
63 </div>
64
65 </body>
66 </html>
67

```



## สร้างไฟล์ res.css

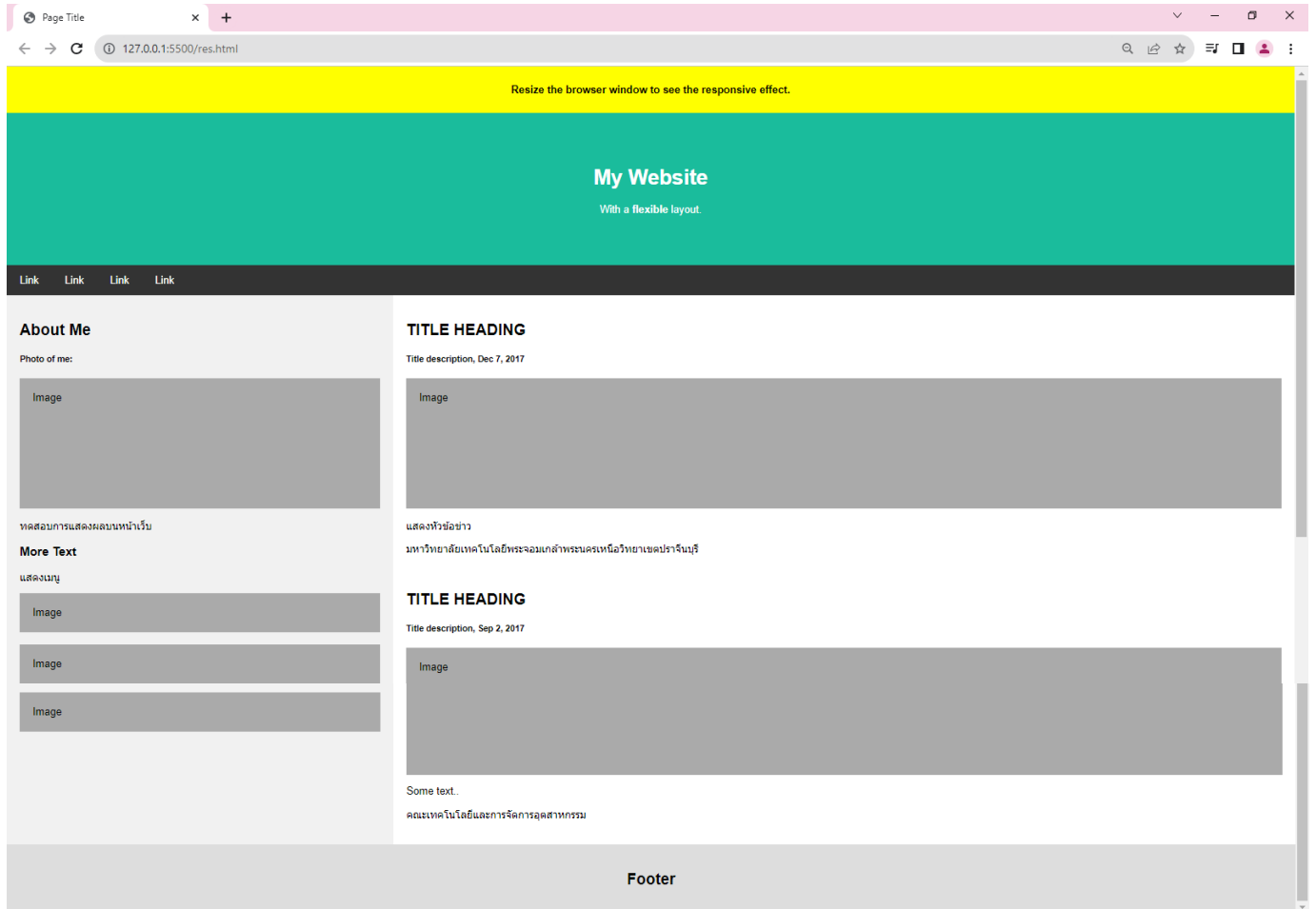
# res.css > ...

```
1  * {
2    box-sizing: border-box;
3  }
4
5  /* Style the body */
6  body {
7    font-family: Arial;
8    margin: 0;
9  }
10
11 /* Header/logo Title */
12 .header {
13   padding: 60px;
14   text-align: center;
15   background: #1abc9c;
16   color: white;
17 }
18
19 /* Style the top navigation bar */
20 .navbar {
21   display: flex;
22   background-color: #333;
23 }
24
25 /* Style the navigation bar links */
26 .navbar a {
27   color: white;
28   padding: 14px 20px;
29   text-decoration: none;
30   text-align: center;
31 }
32
33 /* Change color on hover */
34 .navbar a:hover {
35   background-color: #ddd;
36   color: black;
```

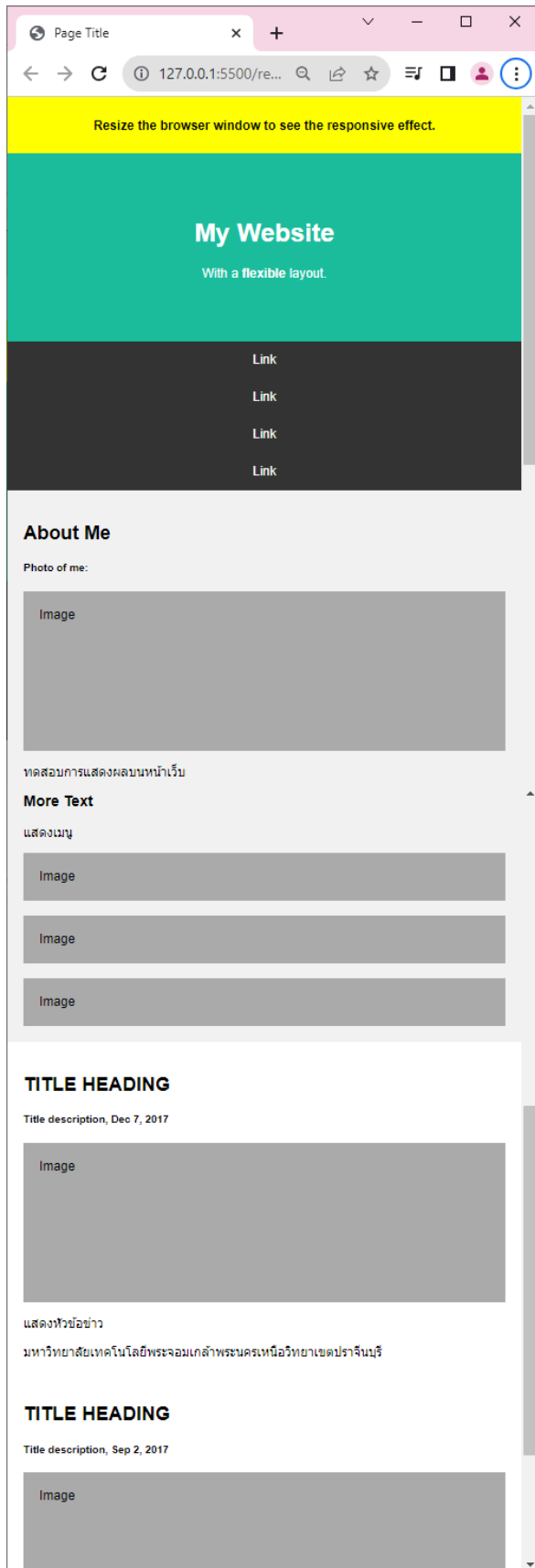
```
37 }
38
39 /* Column container */
40 .row {
41     display: flex;
42     flex-wrap: wrap;
43 }
44
45 /* Create two unequal columns that sits next to each other */
46 /* Sidebar/left column */
47 .side {
48     flex: 30%;
49     background-color: #f1f1f1;
50     padding: 20px;
51 }
52
53 /* Main column */
54 .main {
55     flex: 70%;
56     background-color: white;
57     padding: 20px;
58 }
59
60 /* Fake image, just for this example */
61 .fakeimg {
62     background-color: #aaa;
63     width: 100%;
64     padding: 20px;
65 }
66
67 /* Footer */
68 .footer {
69     padding: 20px;
70     text-align: center;
71     background: #ddd;
72 }
73
74 /* Responsive layout - when the screen is less than 700px wide,
75 make the two columns stack on top of each other instead of next to each other */
76 @media screen and (max-width: 700px) {
77     .row, .navbar {
78         flex-direction: column;
79     }
80 }
```

# ทดสอบการแสดงผล

## แสดงผลบนหน้าจอปกติ แนวนอน







แสดงผลบนหน้าจอแนวตั้ง