

14. 반응형 웹이란

14-1 모바일 기기와 웹 디자인

14-2 가변 그리드 레이아웃

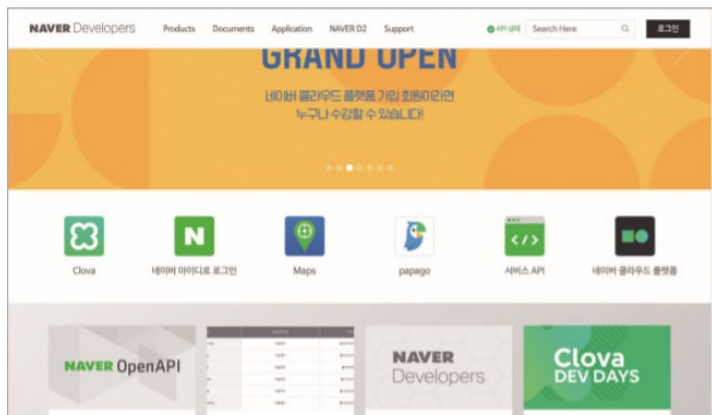
14-3 가변 요소



모바일 기기와 웹 디자인

반응형 웹 디자인

- 웹 사이트의 내용을 그대로 유지하면서 다양한 화면 크기에 맞게 웹 사이트를 표시하는 방법
- 다양한 화면 크기의 모바일 기기들이 계속 쏟아져 나오는데 그 때마다 그 크기에 맞춘 사이트를 별도로 제작하는 것은 비효율적
→ 화면 크기에 '반응'해 화면 요소들을 자동으로 바꾸어 사이트를 구현하는 것이 바로 반응형 웹 디자인



반응형 웹 디자인의 장단점

장점

- 모든 스마트 기기에서 접속 가능
- 가로 모드에 맞춘 레이아웃 변경 가능
- 사이트 유지,관리 용이

단점

반응형 웹 기술이 최신 웹 표준인 CSS3의 일부
→ 최신 모던 웹 브라우저에서만 지원됨

모바일 기기와 웹 디자인

뷰포트(viewport)

- 뷰포트 : 실제 내용이 표시되는 영역
- PC 화면과 모바일 화면의 픽셀 표시 방법이 다르기 때문에 모바일 화면에서 의도한대로 표시되지 않음
→ 뷰포트를 지정하면 기기 화면에 맞춰 확대/축소해서 내용 표시

뷰포트 지정하기

- <head> 태그 안에서 <meta> 태그를 이용해 뷰포트 지정

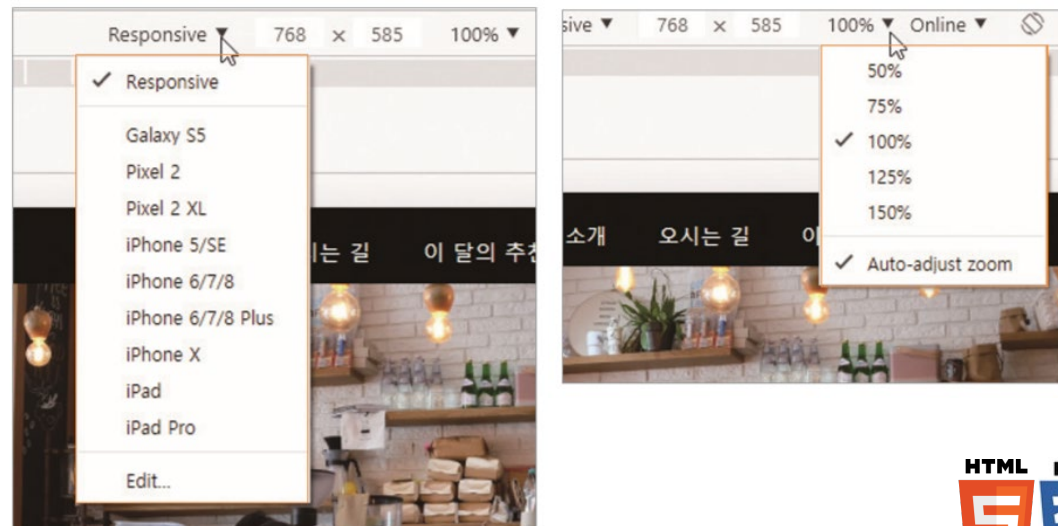
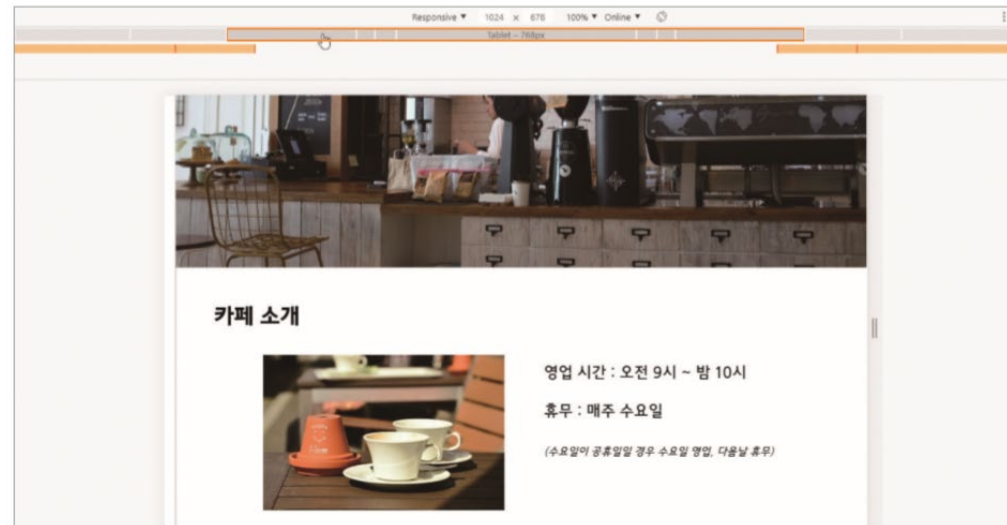
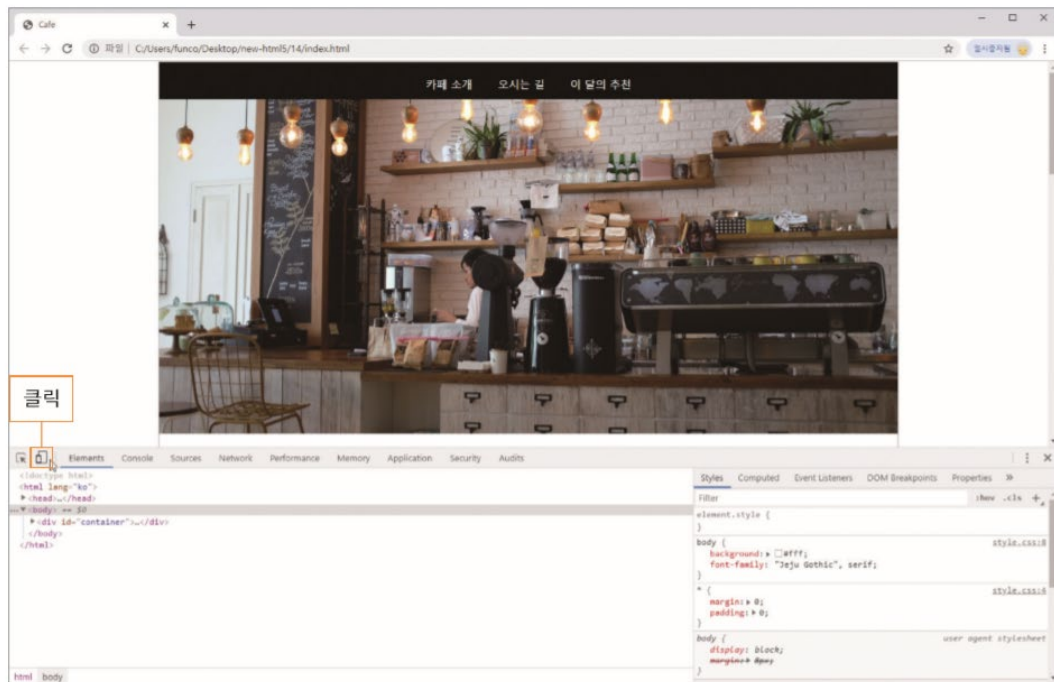
기본형 `<meta name="viewport" content="속성1=값, 속성2=값2, ... ">`

속성	설명	사용 가능한 값	기본 값
width	뷰포트 너비	device-width 또는 크기	브라우저 기본 값
height	뷰포트 높이	device-height 또는 크기	브라우저 기본 값
user-scalable	확대/축소 가능 여부	yes 또는 no	yes
initial-scale	초기 확대/축소 값	1~10	1
minimum-scale	최소 확대/축소 값	0~10	0.25
maximum-scale	최대 확대/축소 값	0~10	1.6

- 일반적인 사용법 : 뷰포트의 너비를 스마트폰 화면 너비에 맞추고 초기 화면 배율을 1로 지정

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

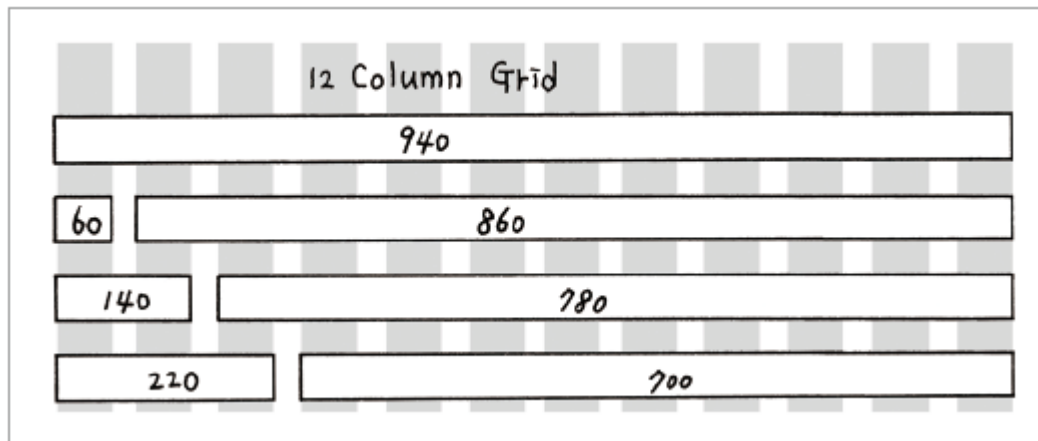
[실습] 크롬의 디바이스 모드 활용하기



가변 그리드 레이아웃

그리드 시스템(grid system)

- 화면을 여러 개의 칼럼(column)으로 나누어, 필요할 때마다 칼럼들을 묶어 배치하는 방법
- 화면 너비 값에 따라 '960 그리드 시스템', '1200 그리드 시스템' 등으로 나뉨
- 칼럼 개수에 따라 '12 칼럼 그리드 시스템', '16 칼럼 그리드 시스템', '24 칼럼 그리드 시스템' 등으로 나뉨
- 주로 960 픽셀 12 칼럼의 그리드 시스템 사용
- 고정 그리드 : 화면 너비를 일정하게 고정하고 레이아웃 만들.
- 가변 그리드 : 화면 너비를 % 같은 가변 값으로 지정.
- 가변 그리드 레이아웃을 사용할 경우, 너비 값이 줄어들면 실제 콘텐츠를 확인하기 불편하므로 가능하면 간결한 디자인을 사용하는 것이 좋음.

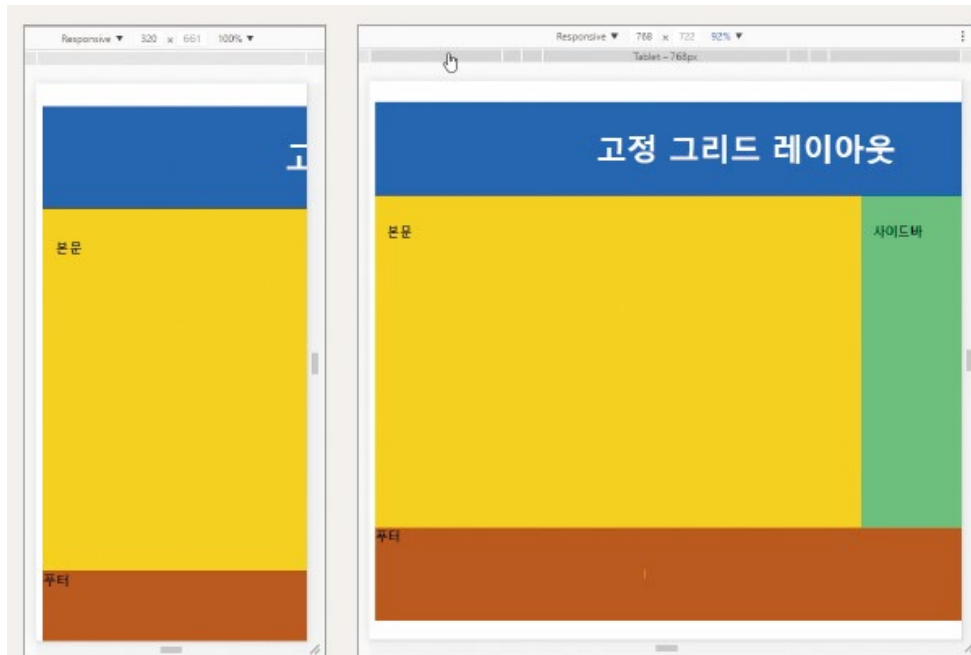


문서 좌우에 10px씩의 패딩이 있다고 가정한 레이아웃 →

가변 그리드 레이아웃

고정 그리드 레이아웃일 경우

- 문서의 맨 바깥 부분을 #wrapper 요소로 묶고 너비를 960px로 지정
- 헤더와 본문, 사이드 바, 푸터를 배치. 이 때 너비는 px 값.
- 화면 너비가 좁아질 경우 내용의 일부가 가려질 수 있음.



```
<style>
#wrapper {
  width:960px;
  margin:0 auto;
}
header {
  width:960px;
  height:120px;
}
.content {
  float:left;
  width:600px;
}
.right-side {
  float:right;
  width:300px;
}
footer {
  clear:both;
  width:960px;
}
</style>
```

가변 그리드 레이아웃

고정 그리드 레이아웃일 경우

```
1  <!doctype html>
2  <html lang="ko">
3  <head>
4  <meta charset="utf-8">
5  <title>Fluid Grid Layout</title>
6  <style>
7      #wrapper {
8          width:960px;
9          margin:0 auto;
10     }
11     header { /* 헤더 */
12         width:960px;
13         height:120px;
14         background-color: #066cfa;
15         border-bottom:1px solid black;
16     }
17     .header-text{
18         font-size:40px;
19         color:white;
20         text-align:center;
21         line-height:120px;
22     }
23     .content { /* 본문 */
24         float:left;
25         width:600px;
26         height:400px;
27         padding:15px;
28         background-color: #ffd800;
29     }
30     .right-side { /* 사이드 바 */
31         float:right;
32         width:300px;
33         height:400px;
34         padding:15px;
35         background-color: #00ff90;
36     }
37     footer { /* 푸터 */
38         clear:both;
39         height:120px;
40         background-color: #c3590a;
41     }
42 </style>
43 </head>
```

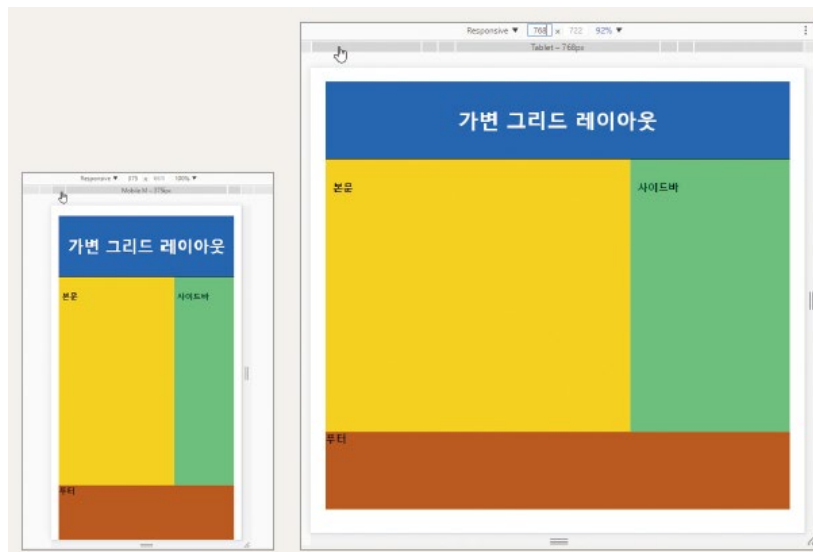
```
44
45 <body>
46     <div id="wrapper">
47         <header>
48             <h1 class="header-text">고정 그리드 레이아웃</h1>
49         </header>
50         <section class="content">
51             <h4>본문</h4>
52         </section>
53         <aside class="right-side">
54             <h4>사이드바</h4>
55         </aside>
56         <footer>
57             <h4>푸터</h4>
58         </footer>
59     </div>
60 </body>
61 </html>
```

가변 그리드 레이아웃

가변 그리드 레이아웃 만들기

- ① 전체를 감싸는 요소의 너비를 %로 변환 (화면에 딱 차게 하고 싶다면 100%, 여유를 두려면 적당히)
- ② 전체를 감싸는 요소의 너비를 기준으로 각 요소의 너비를 계산

$(\text{요소의 너비} / \text{콘텐츠 전체를 감싸는 요소의 너비}) * 100$



```
<style>
#wrapper {
  width:96%;
  margin:0 auto;
}
header {
  width:100%;
  height:120px;
}
.content {
  float:left;
  width:62.5%;
  height:400px;
  padding:1.5625%;
  background-color:#ffd800;
}
.right-side {
  float:right;
  width:31.25%;
  height:400px;
  padding:1.5625%;
  background-color:#00ff90;
}
footer {
  clear:both;
  width:100%;
  height:120px;
  background-color:#c3590a;
}
```


가변 그리드 레이아웃

가변 그리드 레이아웃 만들기

(요소의 너비 / 콘텐츠 전체를 감싸는 요소의 너비) * 100

```
1  <!doctype html>
2  <html lang="ko">
3  <head>
4  <meta charset="utf-8">
5  <title>Fluid Grid Layout</title>
6  <meta name="viewport" content="width=device-width, initial-scale=1">
7  <style>
8      #wrapper {
9          width:96%;
10         margin:0 auto;
11     }
12     header { /* 헤더 */
13         width: 100%;
14         height: 120px;
15         background-color: #066cfa;
16         border-bottom: 1px solid black;
17     }
18     .header-text{
19         font-size:32px;
20         color:white;
21         text-align:center;
22         line-height:120px;
23     }
24     .content { /* 본문 */
25         float:left;
26         width:62.5%;
27         height:400px;
28         padding:1.5625%;
29         background-color: #ffd800;
30     }
31     .right-side { /* 사이드 바 */
32         float:right;
33         width:31.25%;
34         height:400px;
35         padding:1.5625%;
36         background-color: #00ff90;
37     }
38     footer { /* 푸터 */
39         clear:both;
40         width:100%;
41         height:120px;
42         background-color: #c3590a;
43     }
44 </style>
45 </head>
```

```
47 <body>
48     <div id="wrapper">
49         <header>
50             <h1 class="header-text">가변 그리드 레이아웃</h1>
51         </header>
52         <section class="content">
53             <h4>본문</h4>
54         </section>
55         <aside class="right-side">
56             <h4>사이드바</h4>
57         </aside>
58         <footer>
59             <h4>푸터</h4>
60         </footer>
61     </div>
62 </body>
63 </html>
```

가변 요소

가변 글꼴

em 단위

부모 요소 폰트의 대문자 M 너비를 1em으로 지정.

1em=16px

$$\text{글자 크기(em)} = \frac{\text{글자 크기(px)}}{16\text{px}}$$

```
1 <!doctype html>
2 <html lang="ko">
3 <head>
4   <meta charset="utf-8">
5   <title>Fluid Grid Layout</title>
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <style>
8     #wrapper {
9       width:96%;
10      margin:0 auto;
11    }
12    header {
13      width: 100%;
14      height: 120px;
15      background-color: #066cfa;
16      border-bottom: 1px solid black;
17    }
18    .header-text{
19      font-size:2em;
20      color:white;
21      text-align:center;
22      line-height:120px;
23    }
24    .content {
25      float: left;
26      width: 62.5%;
27      height: 400px;
28      padding: 1.5625%;
29      background-color: #fffd800;
30      text-align: center;
31      line-height: 380px;
32      font-size: 1.5em;
33    }
34    .right-side {
35      float: right;
36      width: 31.25%;
37      height: 400px;
38      padding: 1.5625%;
39      background-color: #00ff90;
40      text-align: center;
41      line-height: 380px;
42      font-size: 1.5em;
43    }
44    footer {
45      clear: both;
46      width: 100%;
47      height: 120px;
48      background-color: #c3590a;
49      text-align:center;
50      line-height:120px;
51      font-size: 1.5em;
52    }
53  </style>
54 </head>
55
56 <body>
57   <div id="wrapper">
58     <header>
59       <h1 class="header-text">가변 그리드 레이아웃</h1>
60     </header>
61     <section class="content">
62       <h4>본문</h4>
63     </section>
64     <aside class="right-side">
65       <h4>사이드바</h4>
66     </aside>
67     <footer>
68       <h4>푸터</h4>
69     </footer>
70   </div>
71 </body>
72 </html>
```

가변 요소

rem 단위

- em 단위는 부모 요소가 중첩될 경우 글자 크기가 계속 달라짐
- rem은 처음부터 기본 크기를 지정하고 그것을 기준으로 글자 크기 지정

```
1 <!doctype html>
2 <html lang="ko">
3 <head>
4   <meta charset="utf-8">
5   <title>Fluid Grid Layout</title>
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <style>
8     #wrapper {
9       width:96%;
10      margin:0 auto;
11      font-size:12px;
12    }
13    .header-text{
14      font-size:2em;
15      text-align:center;
16    }
17    .fluid-text {
18      font-size:1.5em;
19    }
20  </style>
21 </head>
22
23 <body>
24   <div id="wrapper">
25     <header class="header-text">
26       <p>가변 그리드 레이아웃 </p>
27       <p class="fluid-text">가변 폰트 </p>
28     </header>
29   </div>
30 </body>
31 </html>
```

```
<style>
  body { font-size:16px; }
  .header-text{ font-size:2rem; }
  .fluid-text { font-size:1.5rem; }
</style>
```

가변 요소

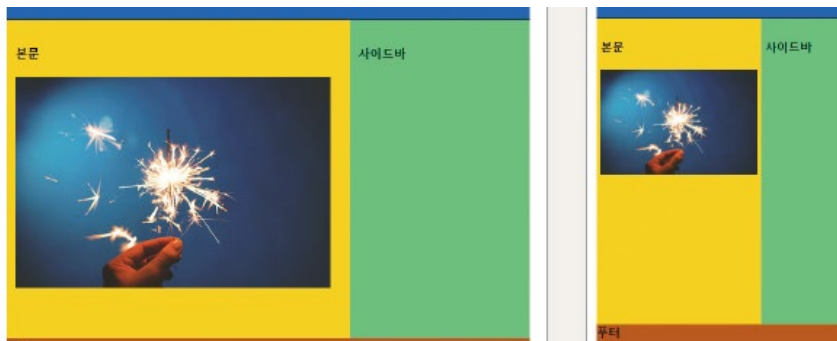
가변 이미지

- 브라우저 창의 너비가 변하더라도 이미지 너비 값은 변하지 않음 → 브라우저 화면 너비를 줄일 경우 이미지 일부가 가려짐
- 가변 이미지(fluid image)로 만들면 창의 너비에 따라 이미지 너비도 조절됨

1) CSS를 이용한 방법

이미지를 감싸고 있는 부모 요소만큼만 커지거나 작아지도록 max-width 속성 값을 100%로 지정

```
<style>
  .content img {
    max-width: 100%;
    height: auto;
  }
</style>
```



가변 요소

가변 이미지

```
1 <!doctype html>
2 <html lang="ko">
3 <head>
4 <meta charset="utf-8">
5 <title>Fluid Grid Layout</title>
6 <meta name="viewport" content="width=device-width, initial-scale=1">
7 <style>
8   #wrapper {
9     width:96%;
10    margin:0 auto;
11  }
12  header {
13    width: 100%;
14    height: 120px;
15    background-color: #066cfa;
16    border-bottom: 1px solid black;
17  }
18  .header-text{
19    font-size:32px;
20    color:white;
21    text-align:center;
22    line-height:120px;
23  }
24  .content {
25    float:left;
26    width:62.5%;
27    height:400px;
28    padding:1.5625%;
29    background-color: #ffd800;
30  }
31  .content img {
32    max-width:100%;
33    height:auto;
34  }
35  .right-side {
36    float:right;
37    width:31.25%;
38    height:400px;
39    padding:1.5625%;
40    background-color: #00ff90;
41  }
```

```
42  footer {
43    clear:both;
44    width:100%;
45    height:120px;
46    background-color: #c3590a;
47  }
48 </style>
49 </head>
50
51 <body>
52   <div id="wrapper">
53     <header>
54       <h1 class="header-text">가변 그리드 레이아웃</h1>
55     </header>
56     <section class="content">
57       <h4>본문</h4>
58       
59     </section>
60     <aside class="right-side">
61       <h4>사이드바</h4>
62     </aside>
63     <footer>
64       <h4>푸터</h4>
65     </footer>
66   </div>
67 </body>
68 </html>
```

가변 요소

가변 이미지

2) 태그와 srcset 속성

화면 너비 값이나 픽셀 밀도에 따라 고해상도의 이미지 파일 지정 가능

기본형

```
[, <이미지2>, <이미지3>, ...]">
```

예)

```

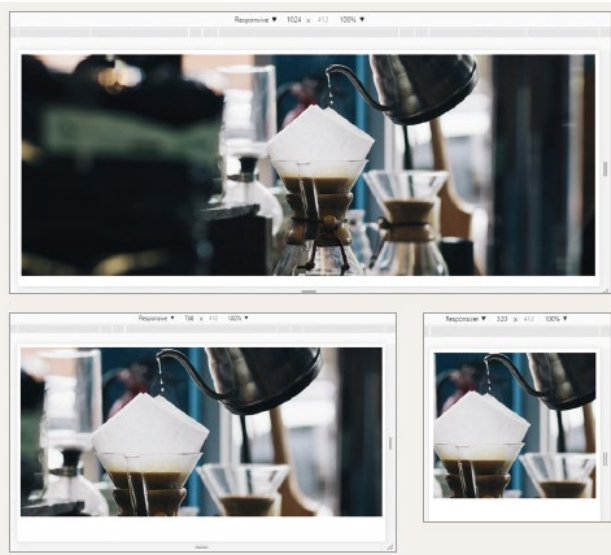
```

```
<picture>
  <source srcset="images/shop-large.jpg" media="(min-width:1024px)">
  <source srcset="images/shop-medium.jpg" media="(min-width:768px)">
  <source srcset="images/shop-small.jpg" media="(min-width:320px)">
  
```

3) <picture> 태그와 <source> 태그

화면 해상도뿐만 아니라 화면 너비에 따라 다른 이미지 파일 표시

속성	설명
srcset	이미지 파일의 경로
media	srcset에 지정한 이미지를 표시하기 위한 조건(속성 값은 14-4 미디어 쿼리 참고).
type	파일 유형
sizes	파일의 크기



가변 요소

가변 이미지

```
1 <!doctype html>
2 <html lang="ko">
3 <head>
4 <meta charset="utf-8">
5 <title>Fluid Image</title>
6 <meta name="viewport" content="width=device-width, initial-scale=1">
7 </head>
8
9 <body>
10   <picture>
11     <source srcset="images/shop-large.jpg" media="(min-width:1024px)">
12     <source srcset="images/shop-medium.jpg" media="(min-width:768px)">
13     <source srcset="images/shop-small.jpg" media="(min-width:320px)">
14     
15   </picture>
16 </body>
17 </html>
```

가변 요소

가변 비디오

CSS를 사용해 max-width 속성을 100%로 지정

```
<style>  
  video { max-width: 100%; }  
</style>
```

```
<video autoplay loop src="assets/cars.mp4"> </video>
```



가변 요소

가변 비디오

```
1 <!doctype html>
2 <html lang="ko">
3 <head>
4 <meta charset="utf-8">
5 <title>Fluid Image</title>
6 <meta name="viewport" content="width=device-width, initial-scale=1">
7 <style>
8   video {
9     max-width:100%;
10  }
11 </style>
12 </head>
13
14 <body>
15   <video src="cars.mp4" controls></video>
16 </body>
17 </html>
```