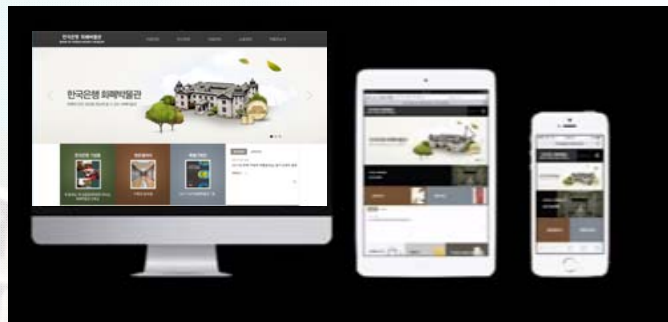


# 반응형 웹(Responsive Web)



# 반응형 웹

## 반응형 웹(Responsive Web)이란?

미디어쿼리 기술 등을 사용하여 다양한 기기에 최적화된 구조로 변경하여 보여주는 웹

즉, 화면이 작은 기기에서 반응형 웹으로 제작된 웹사이트를 접속했을 때는  
웹사이트의 구조를 작은 화면에 최적화된 구조로 변경하여 보여주고,  
큰 화면을 가진 기기에서는 웹사이트의 구조를 큰 화면에 최적화된 구조로 변경하여 보여줌.  
이처럼 기기의 화면이나 환경에 맞게 자유자재로 변하는 것을 반응형 웹이라 함



<http://www.naju.go.kr/tour/>

# 반응형 웹

- 반응형 웹은 사이트에 접속하는 다양한 기기에 반응하는 웹으로서, 반응형 웹을 만들려면 기본적으로 미디어 쿼리를 사용하며, 미디어 쿼리를 직접 작성하기 어려우면 미디어 쿼리와 가변 그리드 레이아웃을 쉽게 사용할 수 있도록 지원하는 CSS 프레임워크를 사용

## 1) 미디어 쿼리(Media Query)

- CSS3 모듈 중 하나로, 사이트에 접속하는 기기 별로 적용할 CSS 파일을 다르게 지정해 주는 기술
  - . 기기의 크기 또는 종류를 감지하여 그 기기에 맞게 웹 사이트의 레이아웃을 지정(변경)해 주는 기술
  - . 기기마다 서로 다른 레이아웃을 표시할 수 있음
  - . 미디어 쿼리는 CSS3의 표준 규약을 사용

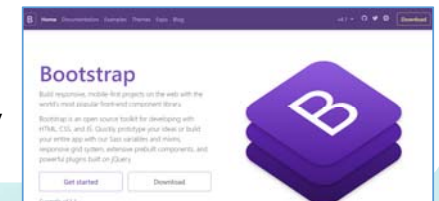


<https://mediaqueri.es/>

## 2) CSS 프레임워크

- 미디어 쿼리와 가변 <sup>flexbox?</sup> 그리드 레이아웃을 쉽게 사용할 수 있도록 지원하는 프레임워크
  - . <sup>트위터에서 만들</sup> 부트스트랩(Bootstrap) : 가장 많이 사용하는 프레임워크로 반응형 그리드 시스템과 다양한 컴포넌트, 자바스크립트 기능을 제공하며, 많은 플러그인들을 사용할 수 있어 기능 확장이 용이
  - . 파운데이션(Foundation), 스켈레톤(Skeleton)

<https://getbootstrap.com/>



# **1. 미디어쿼리**

## 미디어쿼리(Media Query)

- 미디어쿼리란? 장치 특성에 따라 브라우저가 해석해야 할 CSS 코드를 분기 처리하는 규칙으로, 사이트에 접속하는 장치에 따라 사이트의 레이아웃을 다양하게 적용할 수 있기 때문에 반응형 웹 구현의 가장 기본적인 방법
- 컴퓨터나 기기에게 “너는 어떤 종류의 미디어니?” 또는 “미디어의 화면 크기는 어느 정도나 되니?”라고 미디어에게 질의(Query)하고 감지하여 웹사이트를 변경하는 기술
- 즉, 미디어의 화면의 크기와 환경을 감지하여 웹사이트를 변경하는 기술을 말함
- 미디어 쿼리를 이용해 웹 사이트를 제작하면, PC나 태블릿, 스마트폰의 웹 브라우저 화면 크기에 따라 사이트 레이아웃이 바뀜



# 미디어쿼리 기본 문법

@media [ONLY | NOT] 미디어 유형 [AND 또는 콤마] ( 조건문) {실행문}

- 미디어 쿼리를 사용하기 위해서는 CSS <style> 태그 안에 위와 같은 구문을 작성(대·소문자 구별 안함)
- @media : 미디어쿼리 문법의 시작을 알리는 문법
- ONLY | NOT : 미디어 쿼리를 적용할 유형을 지정할 때 사용하는 접두어(Prefix)
  - ONLY : ONLY 다음의 미디어 유형에만 적용
  - NOT : NOT 다음에 지정하는 미디어 유형을 제외한 나머지 유형에만 적용
- 미디어 유형 : 미디어별로 적용할 CSS를 따로 작성해야 함으로 @media 속성 다음에 미디어 유형을 지정함  
예) all(모든장치), print(인쇄장치), screen(화면), tv(영상/음성출력장치), ...
- AND : AND 조건이 만족할 때 뒤에 따라오는 것을 해석
- 콤마 : OR 조건이 만족할 때 뒤에 따라오는 것을 해석
- 조건문 : 조건문이 사실일 때 실행문을 실행
- 실행문 : 실행할 CSS 코드

# 미디어쿼리 구문

@media [ONLY | NOT] 미디어 유형 [AND 또는 콤마] ( 조건문) {실행문}

## • 사용 예

```
/* Tablet & Desktop Device */  
@media all and (min-width:768px) {  
사용자 해상도가 768px 이상일 때  
}
```

```
/* Tablet Device */  
@media all and (min-width:768px) and (max-width:1024px) {  
사용자 해상도가 768px 이상이고 1024px 이하일 때  
}
```

```
/* Desktop Device */  
@media all and (min-width:1025px) {  
사용자 해상도가 1025px 이상일 때  
}
```

```
/* Mobile Device */  
@media all and (max-width:767px) {  
사용자 해상도가 767px 이하일 때  
}
```

Break point

min-width: 이상  
max-width: 이하

# 미디어쿼리에서 사용하는 조건

실제 웹 문서 내용이 화면에 보여지는 영역

가로 너비와 세로 높이 : 뷰포트의 너비와 높이

width, height	웹 페이지의 가로 너비, 세로 높이
min-width, min-height	최소 너비, 최소 높이
max-width, max-height	최대 너비, 최대 높이

예)

```
@media all, print, screen, tv 등screen and (max-width:960px) {  
  body { background-color:green; }  
}
```

단말기의 가로 너비와 세로 높이 : 단말기의 너비와 높이

device-width, device-height	단말기의 가로 너비, 세로 높이
min-device-width, min-device-height	단말기의 최소 너비, 최소 높이
max-device-width, max-device-height	단말기의 최대 너비, 최대 높이

예)

```
@media all and (device-width:320px) and  
(device-height:480px) { ... }
```



## 미디어쿼리에서 사용하는 조건

**화면 회전** : 디바이스를 세로로 또는 가로로 보기

orientation: portrait	단말기의 세로 방향
orientation: landscape	단말기의 가로 방향

예)

```
@media all and (orientation:portrait) {  
    ...  
}
```

**화면 비율** : 브라우저 화면 너비 값(width)을 높이 값(height)으로 나눈 것

aspect-ratio	화면 비율(width 값/ height 값)
min-aspect-ratio	최소 화면 비율
max-aspect-ratio	최대 화면 비율

예)

```
@media all and (aspect-ratio:16/9) {  
    ...  
}
```

## 미디어쿼리에서 사용하는 조건

**단말기의 물리적 화면 비율** : 단말기 너비 값(device-width)을 높이로 나눈 것

device-aspect-ratio	단말기 화면 비율(단말기 너비/단말기 높이)
min-device-aspect-ratio	단말기 최소 화면 비율
max-device-aspect-ratio	단말기 최대 화면 비율

예)

```
@media all and (device-aspect-ratio:16/9) {  
    ...  
}
```

**색상당 비트 수** : 단말기에서 사용하는 최대 색상 비트 수

color	비트 수
min-color	최소 비트 수
max-color	최대 비트 수

예)

```
@media all and (color) { ... }  
@media all and (color:0) { ... }
```

# 미디어쿼리 적용하기

## 기본형

```
<style>
  @media 미디어 쿼리 조건 {
    스타일 규칙들
  }
</style>
```

예)

```
<style>
  @media screen and (max-width:320px) {
    body {background-color:orange}
  }
</style>
```

너비가 320px 이하일 때  
배경색을 orange로

# 미디어쿼리 적용 예

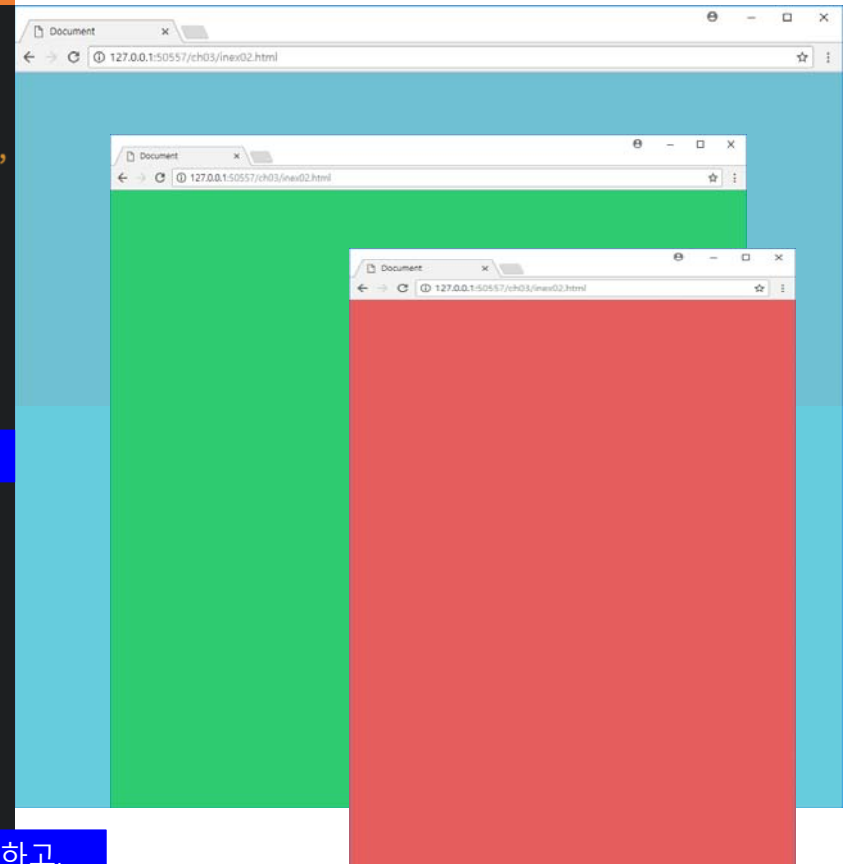
Index01.html

```
1 <!DOCTYPE HTML>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1,
6     minimum-scale=1, maximum-scale=1, user-scalable=no">
7   <title>Document</title>
8 </head>
9 <style>
10   /* 웹 브라우저 크기가 320px 이상일 때 */
11   @media all and (min-width:320px){
12     body{
13       background:#e65d5d;
14     }
15   }
16   /* 웹 브라우저 크기가 768px 이상일 때 */
17   @media all and (min-width:768px){
18     body{
19       background:#2dcc70;
20     }
21   }
22   /* 웹 브라우저 크기가 960px 이상일 때 */
23   @media all and (min-width:960px){
24     body{
25       background:#6fc0d1;
26     }
27   }
28 </style>
29 <body>
30
31 </body>
32 </html>
```

✓ and 구문 뒤에는 반드시 공백을 한 칸 띄어야 함

✓ 접두사 min을 사용할 때는 크기가 작은 순서대로 작성하고, max를 사용할 때는 크기가 큰 순서대로 작성한다.

✓ min : 최소 또는 그 이상, max : 최대 또는 그 이하를 의미



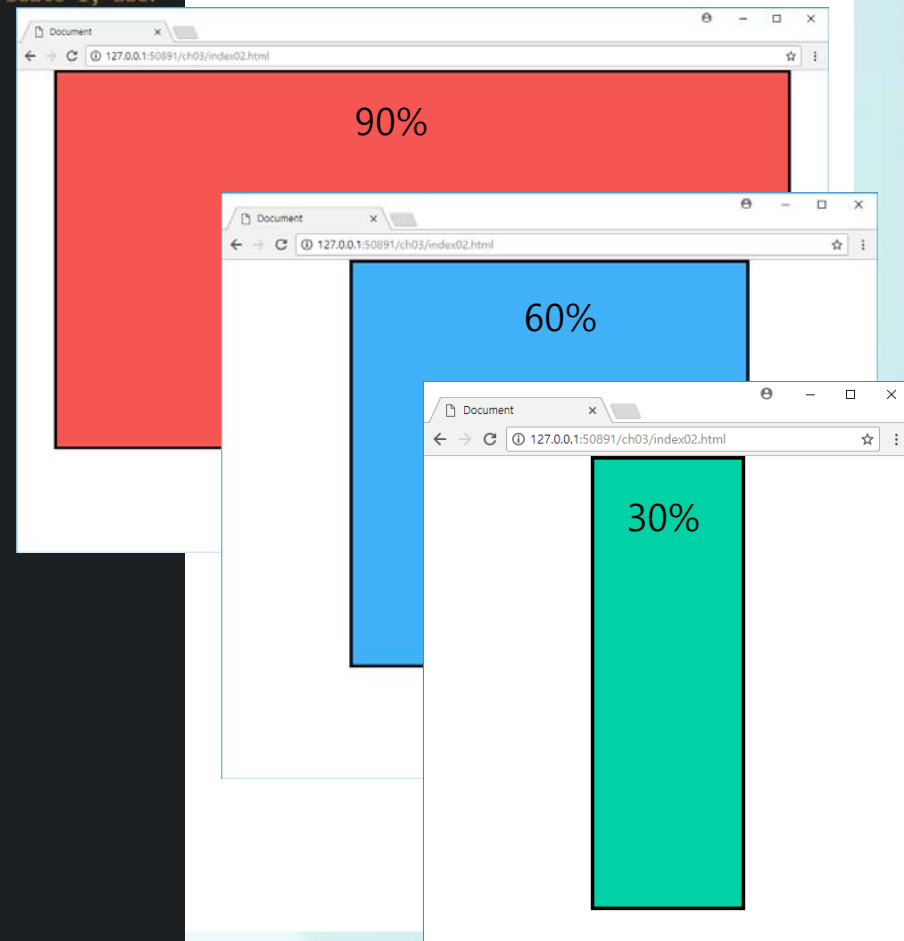
크롬 -> F12 -> Toggle Device Toolbar 클릭하여 Size 편리하게 확인 가능

# 미디어쿼리 적용 예

Index02.html

```
1 <!DOCTYPE HTML>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1, minimum-scale=1, maximum-scale=1, user-
6     scalable=no">
7   <title>Document</title>
8   <style>
9     /* 공통으로 적용할 스타일을 먼저 작성하고 난 다음에 미디어 쿼리를 작성 */
10    * {
11      margin: 0;
12      padding: 0;
13    }
14    #wrap {
15      height: 500px;
16      margin: 0 auto;
17      border: 4px solid #000;
18    }
19    /* 미디어쿼리 */
20    @media all and (min-width:320px) {
21      #wrap {
22        width: 30%;
23        background: #00d2a5;
24      }
25    }
26    @media all and (min-width:768px) {
27      #wrap {
28        width: 60%;
29        background: #40b0f9;
30      }
31    }
32    @media all and (min-width:1024px) {
33      #wrap {
34        width: 90%;
35        background: #f45750;
36      }
37    }
38  </style>
39 </head>
40 <body>
41   <div id="wrap"></div>
42 </body>
43 </html>
```

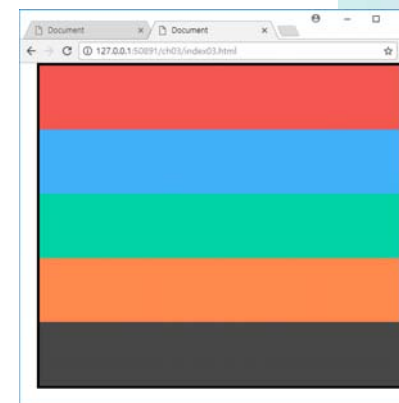
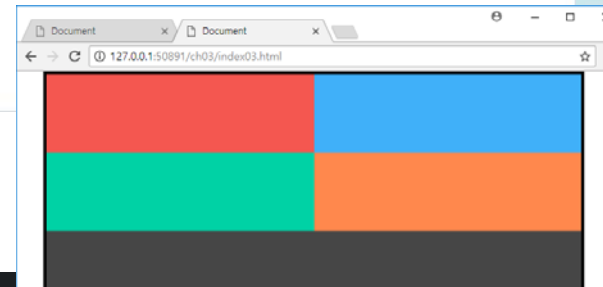
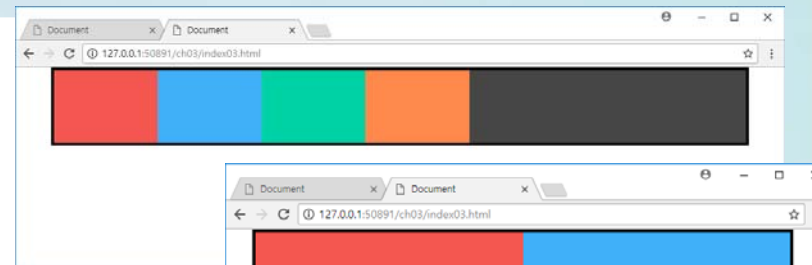
auto : 가운데 정렬



# 미디어쿼리 적용 예

Index03.html

```
1 <!DOCTYPE HTML>
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1, minimum-scale=1,
6     scalable=no">
7   <title>Document</title>
8   <style>
9     /* 공통으로 적용할 스타일을 먼저 작성하고 난 다음에 미디어 쿼리를 작성 */
10    * {
11      margin: 0;
12      padding: 0;
13    }
14    #wrap {
15      width: 90%;
16      margin: 0 auto;
17      border: 4px solid #000;
18      overflow: hidden;
19    }
20    #wrap div {
21      height: 100px;
22    }
23    #wrap div:first-child {
24      background: #f45750;
25      float: left;
26    }
27    #wrap div:nth-child(2) {
28      background: #40b0f9;
29      float: left;
30    }
31    #wrap div:nth-child(3) {
32      background: #00d2a5;
33      float: left;
34    }
35    #wrap div:nth-child(4) {
36      background: #ff884d;
37      float: left;
38    }
39    #wrap div:last-child {
40      background: #464646;
41      float: left;
42    }
43    /* 브라우저 크기가 320px 이상일 때 모든 박스의 너비값을 100%로 설정 */
44    @media all and (min-width:320px) {
45      #wrap div {
46        width: 100%;
47      }
48    }
49    /* 브라우저 크기가 768px 이상일 때 너비를 50%로 설정하고, 마지막 박스 너비는 100%로 설정 */
50    @media all and (min-width:768px) {
51      #wrap div {
52        width: 50%;
53      }
54      #wrap div:last-child {
55        width: 100%;
56      }
57    }
58    /* 브라우저 크기가 1024px 이상일 때 너비를 15%로 설정하고, 마지막 박스 너비는 40%로 설정 */
59    @media all and (min-width:1024px) {
60      #wrap div {
61        width: 15%;
62      }
63      #wrap div:last-child {
64        width: 40%;
65      }
66    }
67  </style>
68 </head>
69 <body>
70   <div id="wrap">
71     <div></div><div></div><div></div><div></div><div></div>
72   </div>
73 </body>
74 </html>
```





## **2. 뷰 포트**

# 뷰포트

- 뷰포트(Viewport)는 화면에서 실제 내용이 표시되는 영역으로,  
데스크톱은 사용자가 설정한 해상도가 뷰포트 영역이 되고,  
모바일 기기는 기본으로 설정되어 있는 값이 뷰포트 영역이 됨
- 따라서 모바일 기기는 기본으로 설정되어 있는 뷰포트 영역으로  
인해 미디어쿼리가 정상적으로 작동하지 않는 문제가 발생함
- 이러한 문제를 방지하기 위해 `<head>`에 뷰포트 메타태그를 선언하여  
화면의 크기나 배율을 조절해야 반응형 웹을 만들 수 있음

# 뷰포트

## ❖ 뷰포트 <meta> 태그 형식

이 부분까지는 필수사항

```
<meta name="viewport" content="width=device-width, initial-scale=1, minimum-scale=1, maximum-scale=1, user-scalable=no">
```

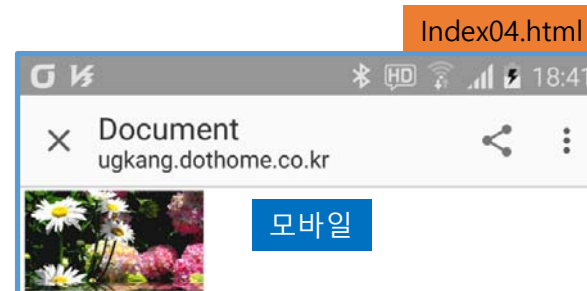
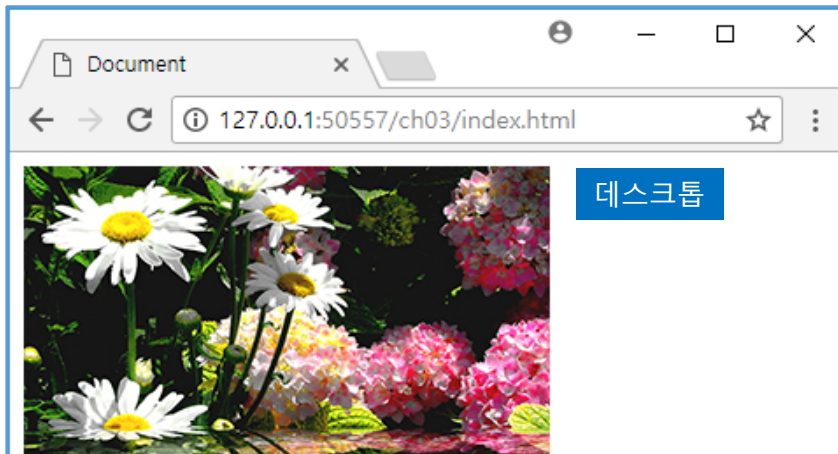
- width : 뷰포트의 너비 설정
- initial-scale : 초기 배율 설정
- minimum-scale : 최소 축소 비율 설정
- maximum-scale : 최대 확대 비율 설정
- user-scalable : 확대/축소 여부 설정

# 뷰포트

## ❖ 뷰포트 속성

width	뷰포트의 너비를 지정합니다. 기본 값은 device-width입니다.
height	뷰포트의 높이를 지정합니다. 기본 값은 device-height입니다.
initial-scale	초기 배율을 나타내며, 기본 값은 1입니다. 1보다 작은 값을 사용하면 축소된 페이지를 표시하고, 1보다 큰 값을 사용하면 확대된 페이지를 표시합니다.
user-scalable	사용자가 페이지를 확대/축소할 수 있는지의 여부를 지정합니다. 기본 값은 'yes'인데, 'no'로 지정하면 사용자가 화면을 확대하거나 축소할 수 없습니다.
minimum-scale	사용자가 축소할 수 있는 최솟값을 지정합니다. 기본 값은 0.25이며, 가로 값을 기준으로 합니다.
maximum-scale	사용자가 확대할 수 있는 최댓값을 지정합니다. 기본 값은 5.0입니다.

## 뷰포트가 적용되지 않은 경우

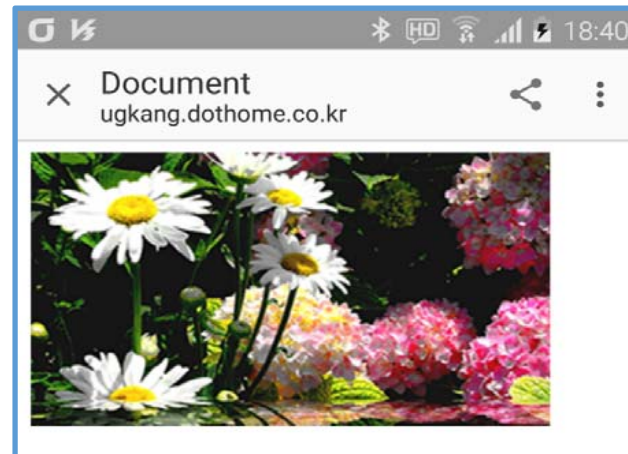
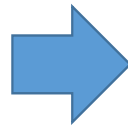
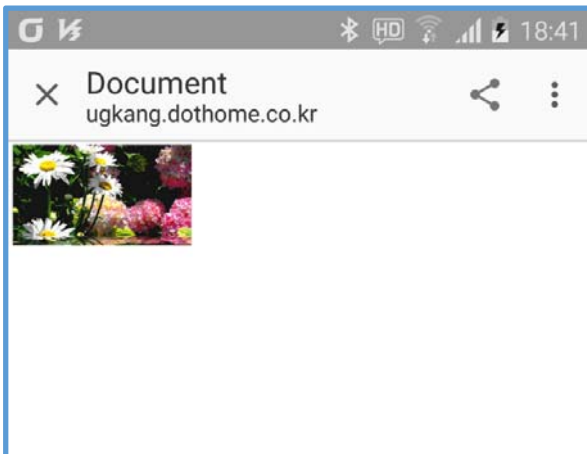


✓ 데스크톱에서는 정상적으로 보이지만  
모바일에서는 작게 보임

```
1 <!DOCTYPE HTML>
2 <html lang="ko">
3   <head>
4     <meta charset="UTF-8">
5
6     <title>Document</title>
7   </head>
8   <body>
9     
10
11  </body>
12 </html>
```

# 뷰포트가 적용된 경우

Index05.html



```
1 <!DOCTYPE HTML>
2 <html lang="ko">
3   <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1,
6       minimum-scale=1, maximum-scale=1, user-scalable=no">
7     <title>Document</title>
8   </head>
9   <body>
10    
11  </body>
12 </html>
```



### **3. 그리드 레이아웃**

## 그리드 레이아웃

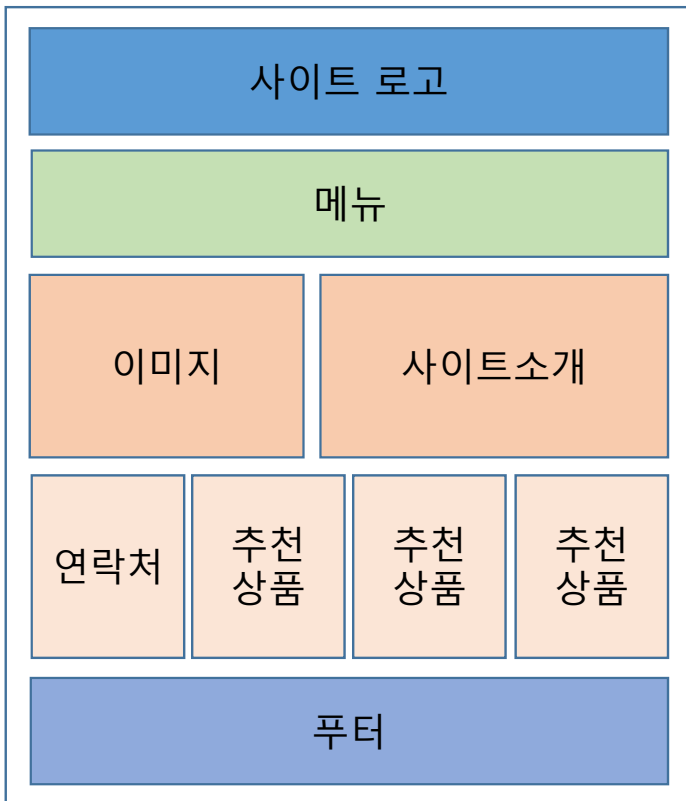
- 반응형 웹 사이트를 제작할 때 사이트 전체의 디자인이나 일관성을 유지하기 쉬운 구조가 그리드 레이아웃이다.
- 그리드 레이아웃 기법은 그리드(Grid)란 용어가 “격자”라는 뜻을 가지고 있는 것처럼, 브라우저 화면을 바둑판처럼 여러 개의 가로와 세로로 나누어 필요에 따라 두세 개의 컬럼을 합치기도 하고, 위치를 옮기기도 하면서 레이아웃을 다양하게 꾸미는 방법이다.
- 직접 CSS 소스를 작성해 그리드 레이아웃을 만들 수 있지만 플렉스 박스 레이아웃(Flex Box Layout)을 활용하면 좀 더 편리하고 안정된 사이트를 만들 수 있다.

## ■ 그리드 레이아웃 구상하기

- 반응형 웹 디자인을 염두에 두고 사이트 레이아웃을 구상할 때는 사이트의 각 부분이 화면 크기에 따라 위치가 옮겨질 수 있다는 점을 생각해야 한다.
- 가장 먼저 해야 할 일은 개발하고자 하는 사이트의 레이아웃을 구상하는 것으로, PC뿐만 아니라, 태블릿이나 스마트폰에 맞는 레이아웃을 함께 구상한다.
- 구상한 레이아웃이 화면의 크기에 맞게 움직일 수 있게 하려면 사이트에 들어가는 내용들을 웹 표준을 준수해야 한다.

# 그리드 레이아웃

## ❖ 그리드 레이아웃 구상



PC용 레이아웃



태블릿용 레이아웃

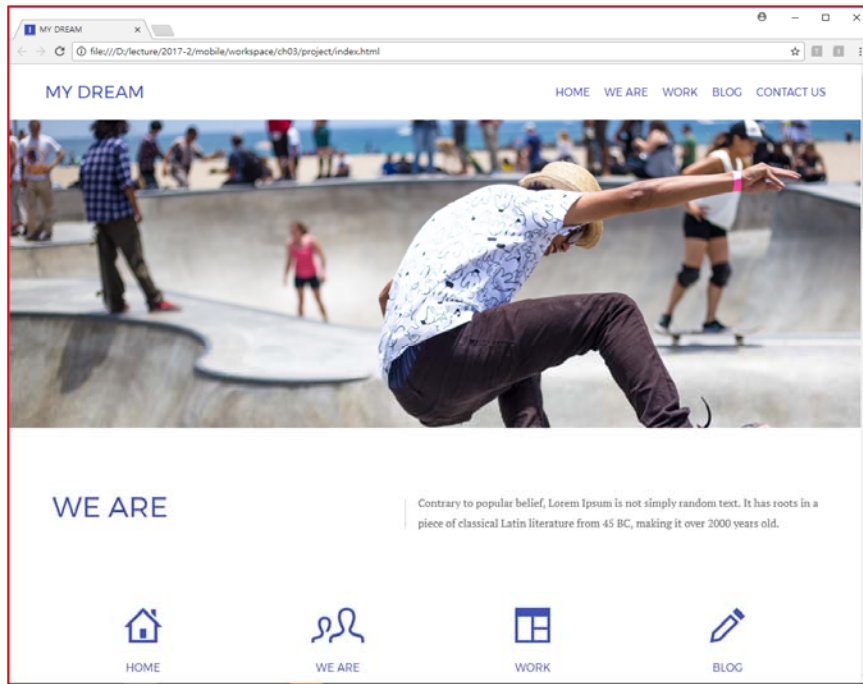
작은 사이즈인 스마트폰부터



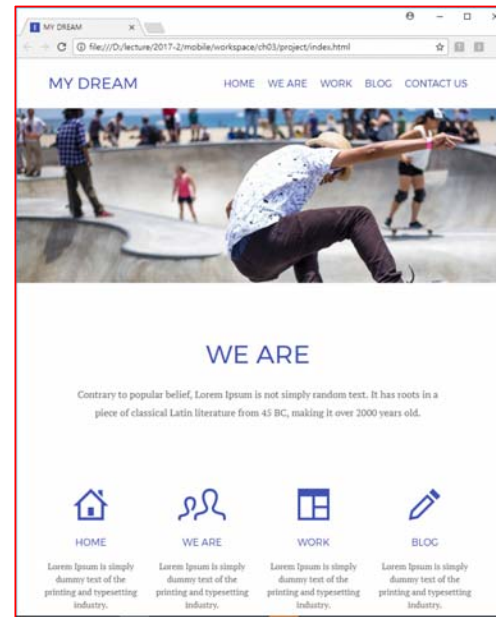
스마트폰용 레이아웃

# **반응형 웹 페이지 만들기**

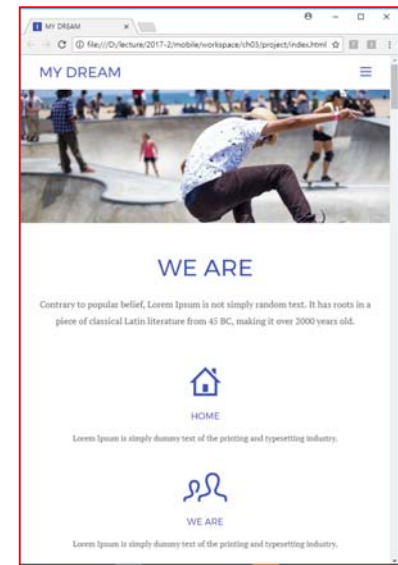
# 반응형 웹 페이지 만들기(프로젝트1)



데스크톱



태블릿



스마트폰



# 반응형 웹 페이지 만들기(프로젝트1)

header

logo

nav(gnb)

MY DREAM

HOME WE ARE WORK BLOG CONTACT US

article

slider

헤더 영역(header)

슬라이드 영역(article)

컨텐츠 영역(section)

푸터 영역(footer)

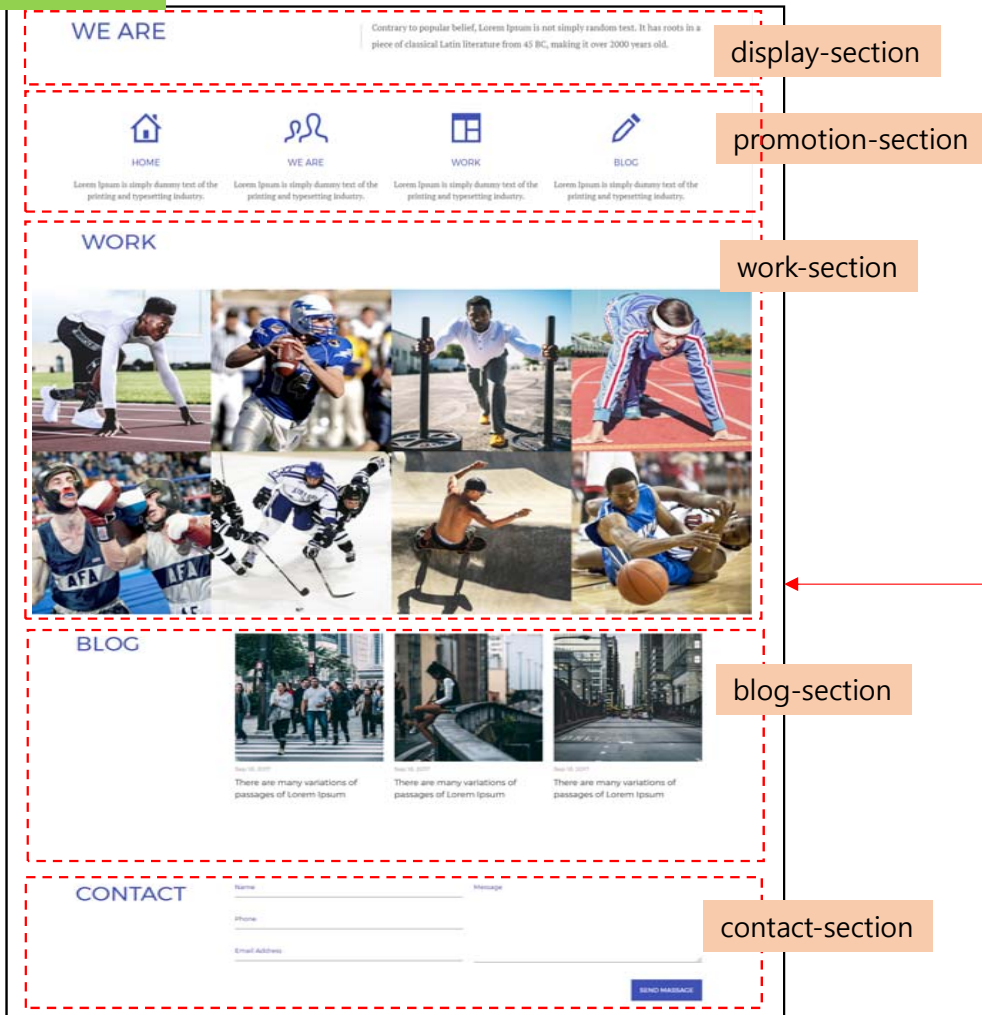
MY DREAM



menu-toggle-btn

# 반응형 웹 페이지 만들기(프로젝트1)

section



헤더 영역(header)

슬라이드 영역(article)

컨텐츠 영역(section)

푸터 영역(footer)

# 반응형 웹 페이지 만들기(프로젝트1)

헤더 영역(header)

슬라이드 영역(article)

컨텐츠 영역(section)

footer

iframe

푸터 영역(footer)

copyright



# 반응형 웹 페이지 만들기(프로젝트1)

헤더 영역(header)

슬라이드 영역(article)

컨텐츠 영역(section)

푸터 영역(footer)

footer

구글 맵스에서 공유 -> 지도 퍼가기로 장소 변경



iframe

copyright



# 참고 문헌

## [ References]

- 기본 서적
  - 2017, 김은아, "Do it! 반응형 웹 만들기", 이지스퍼블리싱(주)
  - 2018, 고경희' "Do it! 프론트엔드 웹 디자인 입문, 이지스퍼블리싱(주)
- Website
  - <http://www.android.com/>