

10. 반응형 웹과 미디어 쿼리



10-1 반응형 웹 알아보기

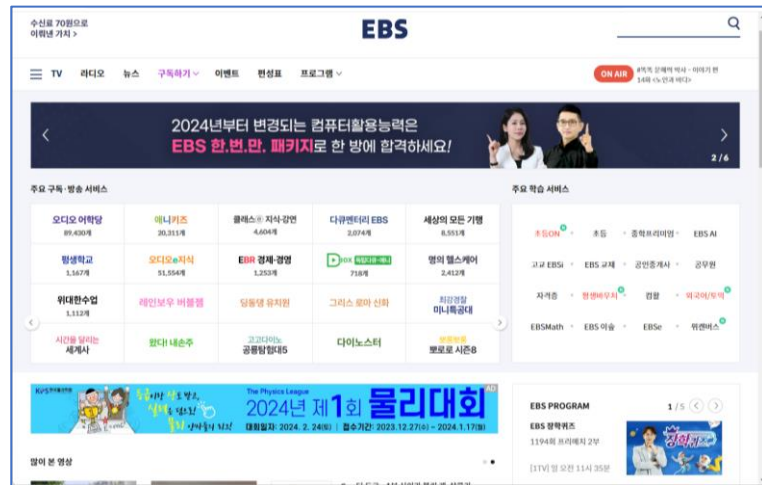
10-2 반응형 요소 알아보기

10-3 미디어 쿼리 알아보기

반응형 웹 알아보기

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



모바일 기기를 위한 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
- VS Code에서 자동으로 작성된 코드에는 뷰포트 코드가 지정되어 있음

일반적인 사용법

뷰포트의 너비를 스마트폰 화면 너비에 맞추고 초기 화면 배율을 1로 지정

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

종류	설명	사용할 수 있는 값	기본값
width	뷰포트 너비	device-width 또는 크기	브라우저 기본값
height	뷰포트 높이	device-height 또는 크기	브라우저 기본값
user-scalable	확대·축소 가능 여부	yes 또는 no (yes는 1로, device-width와 device-height의 값은 10으로 간주)	yes
initial-scale	초기 확대·축소한 값	1~10	1

반응형 요소 만들기 - em 단위와 rem 단위

em 단위

- 부모 요소에서 지정한 글꼴의 대문자 M 너비를 1em으로 놓고 상대적인 크기를 계산함
- 부모 요소에서 글자 크기를 지정하지 않으면 body 요소의 기본 크기 사용

```
<style>
  p { font-size: 1em; }
  .content { font-size: 1.5em; }
</style>
```

.....

```
<h1>레드향</h1>
```

```
<p>껍질에 붉은 빛이 돌아 레드향이라 불린다.</p>
```

16px

```
<div class="content">
```

```
  <p>레드향은 한라봉과 귤을 교배한 것으로</p>
```

24px

```
  <p>일반 귤보다 2~3배 크고, 과육이 붉고 통통하다.</p>
```

```
</div>
```

부모 요소가 다르기 때문에 p 요소의 글자 크기가 달라짐

반응형 요소 만들기

rem 단위

- root에서 지정한 크기를 기준으로 상대적인 크기를 계산함

```
<style>
  html { font-size: 16px; }
  p { font-size: 1rem; }
  .content { font-size: 1.5rem; }
</style>
```

.....

```
<h1>레드향</h1>
```

```
<p>껍질에 붉은 빛이 돌아 레드향이라 불린다.</p>
```

```
<div class="content">
```

```
  <p>레드향은 한라봉과 귤을 교배한 것으로</p>
```

```
  <p>일반 귤보다 2~3배 크고, 과육이 붉고 통통하다.</p>
```

```
</div>
```

em 단위가 부모 요소를 기준으로 한다는 점만 기억한다면 em 단위나 rem 단위는 상황에 따라 어떤 것을 사용해도 무방함

16px

반응형 요소 만들기 - 반응형 이미지

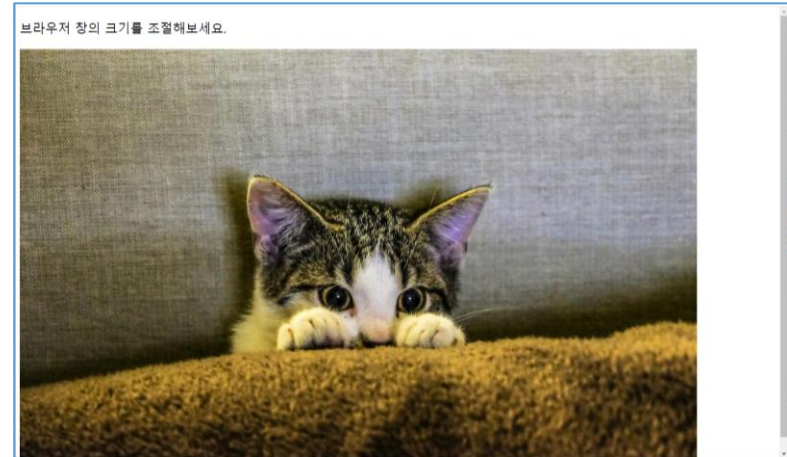
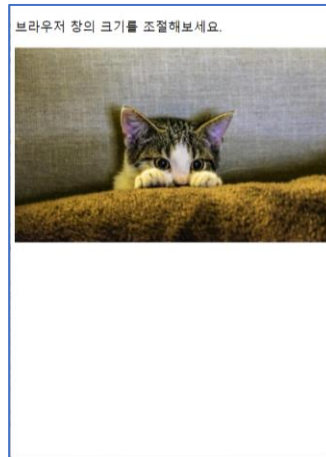
width 속성과 max-width 속성

- width: 100%로 지정하면 부모 요소의 너비가 커질 때 화질이 떨어질 수 있음.
- max-width: 100%로 지정하면 부모 요소 너비가 커져서 이미지가 커지더라도 이미지 원본 크기만 큼만 커짐

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

<p>브라우저 창의 크기를 조절해보세요.</p>

```



반응형 요소 만들기 - 반응형 이미지

object-fit 속성

- 이미지나 비디오 등의 가로, 세로 비율을 유지하면서 크기를 조절할 수 있다

속성값	설명
fill	콘텐츠의 원래 비율을 무시하고 요소 전체 영역을 채움. 기본 값
contain	비율을 유지하면서 요소의 전체 영역에 맞출 수 있을만큼 확대. 영역을 모두 채우지 못할 수도 있다.
cover	비율을 유지하면서 요소의 전체 영역을 채울 수 있을만큼 확대. 영역을 채우기 위해 콘텐츠의 일부가 잘릴 수 있다.
none	콘텐츠의 원래 크기 유지
scale-down	none과 contain 중 콘텐츠 크기가 더 작아지는 값 선택

<style>

.....

.fill { object-fit: fill; } /* 요소에 꽉 차게 채움 */

.contain { object-fit: contain; } /* 비율 유지. 요소 크기에 맞춤 */

.cover { object-fit: cover; } /* 비율 유지. 요소에 꽉 차게 채움 */

.none { object-fit: none; } /* 이미지 원래 크기 유지 */

.scale-down { object-fit: scale-down; } /* none과 contain 중 작은 것 */

</style>



object-fit: fill



object-fit: contain



object-fit: cover



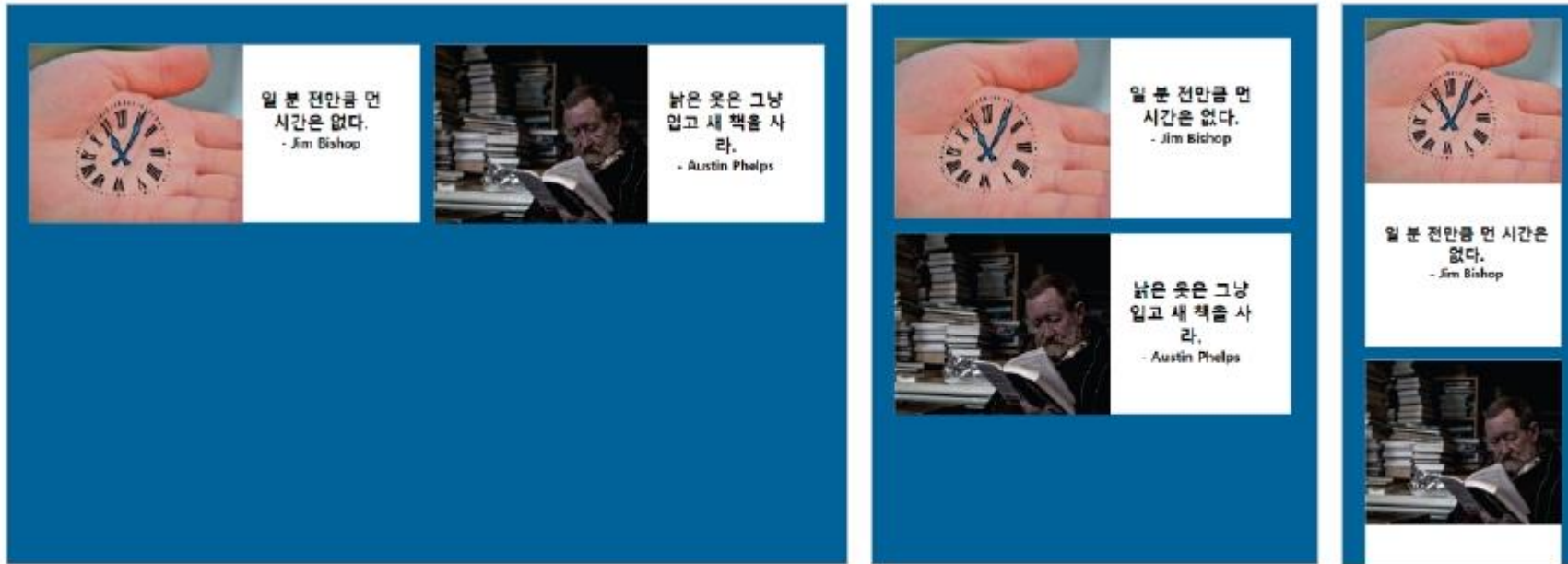
object-fit: none



object-fit: scale-down

미디어 쿼리 알아보기

미디어 쿼리(media queries) : 접속하는 장치(미디어)에 따라 특정한 CSS 스타일을 사용하는 방법



웹 브라우저 창의 크기에 따라 바뀌는 사이트 레이아웃

미디어 쿼리 알아보기

미디어 쿼리 구문

- @media 속성을 사용해 특정 미디어에서 어떤 CSS를 적용할 것인지 지정함
- <style> 태그와 </style> 태그 사이에 사용

기본형 @media [only | not] 미디어 유형 [and 조건] * [and 조건]

- ① only: 미디어 쿼리를 지원하지 않는 웹 브라우저에서는 미디어 쿼리를 무시하고 실행하지 않습니다.
- ② not: not 다음에 지정하는 미디어 유형을 제외합니다. 예를 들어 not tv라고 지정하면 TV를 제외한 미디어 유형에만 적용합니다.
- ③ and: 조건을 여러 개 연결해서 추가할 수 있습니다.

예) 미디어 유형이 screen이면서 최소 너비가 768px이고
최대 너비는 1439px일 경우에 적용할 CSS

```
@media screen and (min-width: 768px) and (max-width: 1439px) {  
  (... 생략 ...)  
}
```

미디어 유형의 종류

종류	설명
all	모든 미디어 유형에서 사용할 CSS를 정의합니다.
print	인쇄 장치에서 사용할 CSS를 정의합니다.
screen	컴퓨터 스크린에서 사용할 CSS를 정의합니다. 스마트폰의 스크린도 포함됩니다.
tv	음성과 영상이 동시에 출력되는 TV에서 사용할 CSS를 정의합니다.
aural	음성 합성 장치(주로 화면을 읽어 소리로 출력해 주는 장치)에서 사용할 CSS를 정의합니다.
braille	점자 표시 장치에서 사용할 CSS를 정의합니다.
handheld	패드(pad)처럼 손에 들고 다니는 장치를 위한 CSS를 정의합니다.
projection	프로젝터를 위한 CSS를 정의합니다.
tty	디스플레이 기능이 제한된 장치에 맞는 CSS를 정의합니다. 이런 장치에서는 픽셀(px) 단위를 사용할 수 없습니다.
embossed	점자 프린터에서 사용할 CSS를 정의합니다.

미디어 쿼리 알아보기

웹 문서의 가로 너비와 세로 높이(뷰포트)

실제 웹 문서 내용이 나타나는 영역의 너비와 높이를 조건으로 사용

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

예) screen에서 너비가 1440px 이상일 때

```
@media screen and (min-width: 1440px)
{
  ... 생략 ...
}
```

화면 회전

스마트폰이나 태블릿에서 기기를 가로나 세로로 돌려보는지 확인

종류	설명
orientation: portrait	단말기의 세로 모드
orientation: landscape	단말기의 가로 모드

이 외에도 화면 비율이나 단말기 화면 비율, 색상당 비트 수 같은
여러 가지 미디어 쿼리 조건이 있음.

참고 : [미디어 쿼리 레벨 5](#)

미디어 쿼리 중단점

- **중단점(breakpoint)** : 서로 다른 CSS를 적용할 화면 크기
- 대부분 기기의 화면 크기 기준.
- 모든 기기를 반영할 수 없기 때문에 스마트폰과 태블릿, 데스크톱 정도로 구분
- 모바일 퍼스트(mobile first) : 모바일 기기 레이아웃을 기본으로 작성 → 태블릿 & PC 레이아웃 작성

일반적인 사용 예

(미디어 쿼리 중단점은 개발자나 작업 조건에 따라 달라질 수 있다.)

- **스마트폰**: 모바일 페이지는 미디어 쿼리를 사용하지 않고 기본 CSS로 작성.
만일 스마트폰의 방향까지 고려해서 제작한다면 min-width의 세로와 가로로 각각 portrait 320px, landscape 480px로 지정.
- **태블릿**: 세로 크기가 768px 이상이면 태블릿으로 지정. 가로 크기는 데스크톱과 똑같이 1024px 이상으로 지정.
- **데스크톱**: 화면 크기가 1024px 이상이면 데스크톱으로 설정.

<참고> 미디어 쿼리를 작성하는데 필요한 뷰포트 값이 정리된 사이트 yesviz.com/devices.php

미디어 쿼리 적용하기

1. 외부 CSS 파일 연결

방법1) <link> 태그 사용하기

<head>와 </head> 사이에 삽입

기본형 <link rel="stylesheet" media="미디어 쿼리 조건" href="css 파일 경로">

예) <link rel="stylesheet" media="print" href="css/print.css">

방법2) @import 구문 사용하기

<style>와 </style> 사이에 삽입

기본형 @import url(css 파일 경로) 미디어 쿼리 조건

예) @import url("css/tablet.css") only screen and (min-width: 321px) and (max-width: 768px);

2. 웹 문서에서 직접 정의하기

방법1) <style> 태그 안에서 media 속성 사용하기

기본형 <style media="<조건>"> {
}
</style>

예)

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


방법2) @media문 사용하기

기본형 <style>
 @media <조건> {
 }
</style>

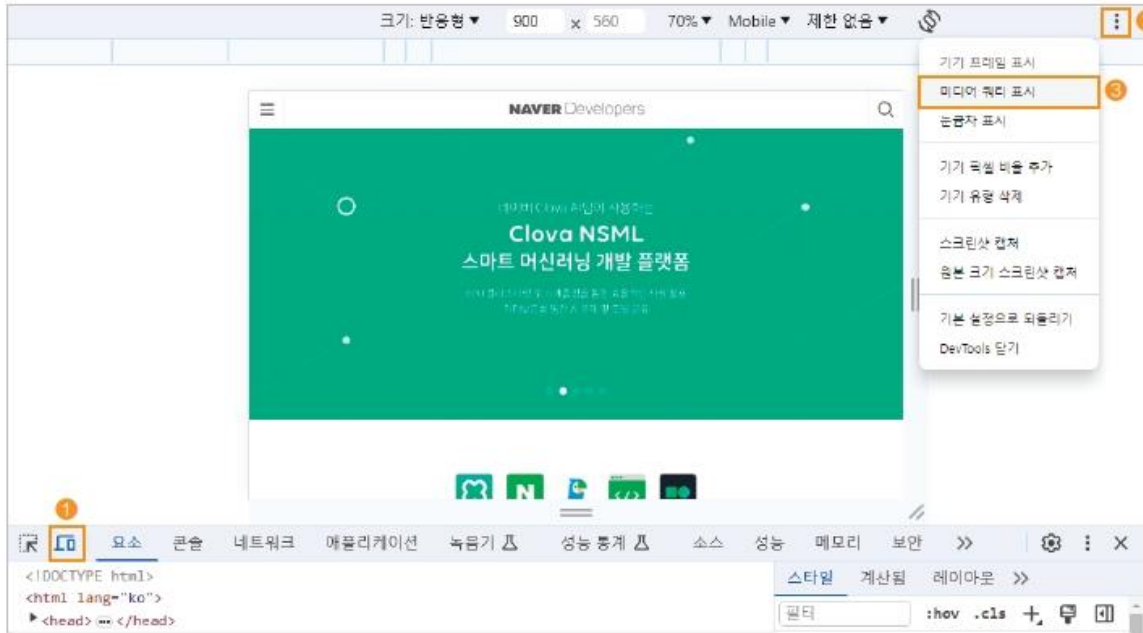
예)


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

웹 개발자 도구 창에서 미디어 쿼리 표시하기

- 1) 웹 개발자 도구 창에서 기기 툴바를 표시한 상태에서 툴바 오른쪽 끝에 있는  클릭
- 2) [미디어 쿼리 표시] 선택

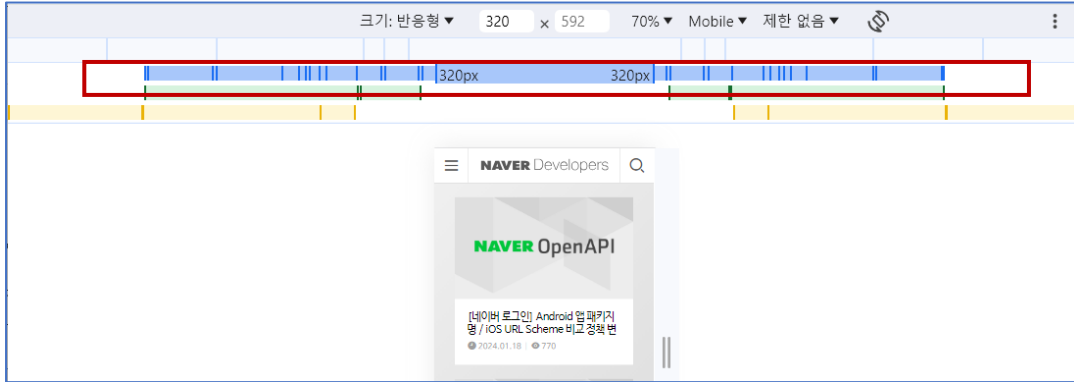
- 3) 미디어 쿼리가 색상별로 구분되어 나타남



 1160px	최대 너비 <code>max-width</code> 를 기준으로 중단점을 나눈 미디어 쿼리
 540px 360px	최소 너비와 최대 너비를 기준으로 중단점을 나눈 미디어 쿼리
 650px	최소 너비 <code>min-width</code> 를 기준으로 중단점을 나눈 미디어 쿼리

웹 개발자 도구 창에서 미디어 쿼리 표시하기

4) 중단점과 중단점 사이의 공간을 클릭하면 조건에 따라 사이트가 어떻게 달라지는지 확인



5) 미디어 쿼리 막대를 우클릭한 후 [소스 코드에서 보기] 위로
마우스 커서를 올리면 CSS 파일 경로 알려줌

6) CSS 파일 선택하면 사용한 미디어 쿼리 코드 확인 가능

