

셋째 주

- SCSS
- CSS overflow
- CSS position
- HTML ul, ol
- CSS list-style
- UI 예서의 ul, li
- CSS font

SCSS 란

SCSS

SCSS를 알기전에 SASS를 먼저 알아보자면, SASS는 CSS의 단점을 보완하기 위한 CSS의 확장으로 나온 스크립트 언어 입니다. 기존 CSS의 개념에서 variable, nesting, mixin 등의 개념을 추가시켜서 CSS를 더욱 간결하고 사용하기 쉽게 만든것이 SASS 입니다.

SASS는 들여쓰기를 통해 구문의 구조를 파악하지만 ‘Sassy CSS’ 를 의미하는 SCSS의 경우 {} 를 기준으로 구조를 만들기 때문에 기존 CSS에 익숙한 사용자들이 사용하기 쉽다.

SASS, SCSS 예

SASS

```
nav
  ul
    margin: 0
    padding: 0

    li
      display: inline-block
```

SCSS

```
nav {
  ul {
    margin: 0;
    padding: 0;
  }

  li {
    display: inline-block;
  }
}
```

SCSS Variable

Variable

Variable 은 특정값을 저장하는데 사용됩니다.
디자인시스템을 구축할 때 유용하게 사용되는 기능입니다.

```
$desktopContentWidth: 1180px;  
$tabletContentWidth: calc(100% - 48px);  
$mobileContentWidth: calc(100% - 32px);  
$contentWidth: $desktopContentWidth;  
  
@media screen and (min-width: 1228px) {  
    $contentWidth: $desktopContentWidth;  
}  
  
@media screen and (min-width: 769px) and (max-width: 1227px) {  
    $contentWidth: $tabletContentWidth;  
}  
  
main {  
    width: $contentWidth;  
    margin: 0 auto;  
}
```

SCSS Nesting

Nesting

Nesting 은 CSS를 중첩해서 사용하는걸 말합니다.
Nesting 종류에는 셀렉터 네스팅과 속성 네스팅이 있습니다.

셀렉터 네스팅

```
.main {  
  .body {  
    .article {  
      background-color: black;  
    }  
  }  
}
```

속성 네스팅

```
.main {  
  .body {  
    .article {  
      font: {  
        family: "Noto Serif CJK KR", serif;  
        size: 20px;  
        weight: 400;  
      }  
    }  
  }  
}
```

SCSS Mixin

Mixin

mixin은 variable 처럼 단일 값을 담는것이 아닌, 여러 속성을 담아서 코드 재사용성을 높여주는 기능이다.

```
@mixin title1 {  
  font-weight: 500;  
  font-size: 20px;  
  line-height: 28px;  
}  
  
header {  
  h1 {  
    @include title1;  
  }  
}
```

SCSS Ampersand

Ampersand

Ampersand 상위(부모) 선택자를 참조하는걸 말합니다.
중첩안에서 부모 선택자를 사용해야할 때 사용됩니다.

```
button {  
  color: black;  
  border-radius: 4px;  
  
  &.active {  
    color: red;  
  }  
}
```

```
// css 결과  
button{ color: black; border-radius: 4px;}  
button.active { color: red }
```

CSS overflow

overflow

overflow는 자식 요소가 부모 요소의 크기를 벗어날 경우 어떻게 보여질지를 설정해주는 속성입니다.
visible, hidden, scroll, auto 값을 설정해줄 수 있습니다.

예)

overflow: hidden;

overflow: auto;

overflow: scroll;

CSS overflow

overflow: visible

visible은 overflow의 기본값입니다.
자식 요소가 부모 요소보다 더 클 경우 넘치는 부분을 그대로 보여줍니다.

overflow: hidden

hidden은 부모 요소보다 더 큰 자식요소의 부분을 가립니다.

overflow: scroll

scroll은 부모 요소가 스크롤 가능하도록 만들어서 스크롤을 통해 넘치는 부분을 볼 수 있게 합니다.

overflow: auto

auto는 자식 요소가 부모 요소보다 작을경우엔 스크롤을 생성 안하지만
자식 요소가 부모 요소보다 커지면 스크롤을 생성합니다.

CSS overflow

overflow

overflow는 자식 요소가 부모 요소의 크기를 벗어날 경우 어떻게 보여질지를 설정해주는 속성입니다.
visible, hidden, scroll, auto 값을 설정해줄 수 있습니다.

예)

overflow: hidden;

overflow: auto;

overflow: scroll;

EX1



visible



auto



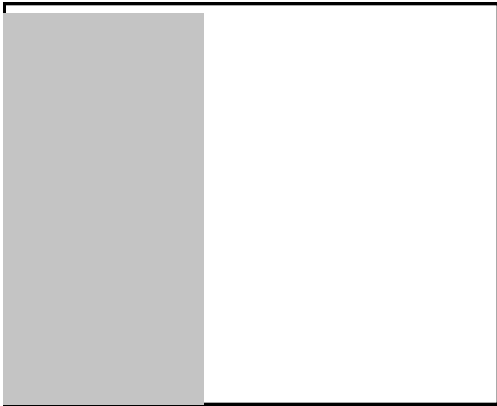
hidden



auto



scroll-x



scroll-y

CSS Position

position

position은 HTML에서 element를 배치하는 방법을 지정하는 속성입니다.

position 속성에 설정 가능한 값은 static, relative, absolute, fixed, sticky 가 있습니다.

예)

position: relative;

position: absolute;

position: fixed;

CSS Position

position: static

static은 기본값으로 position에 대해 아무런 설정하지 않은 상태입니다.
static 상태일땐 top, right, bottom, left 속성을 사용할 수 없습니다.

position: relative

relative는 static과 마찬가지로 position을 설정하지 않은 기본 위치에 element를 배치합니다.
하지만 top, right, bottom, left 속성을 사용 가능하며
해당 속성들은 element의 기본위치를 기준으로 요소의 위치를 계산해줍니다.

예)

```
position: relative;
```

```
top: 20px;
```

```
left: 40px;
```

CSS Position

position: absolute

absolute는 기준이 되는 요소에서 위치값을 적용시키는 속성입니다.
이 때 기준이 되는 요소는 부모 요소중에 relative나 absolute 속성을 가진 요소입니다.

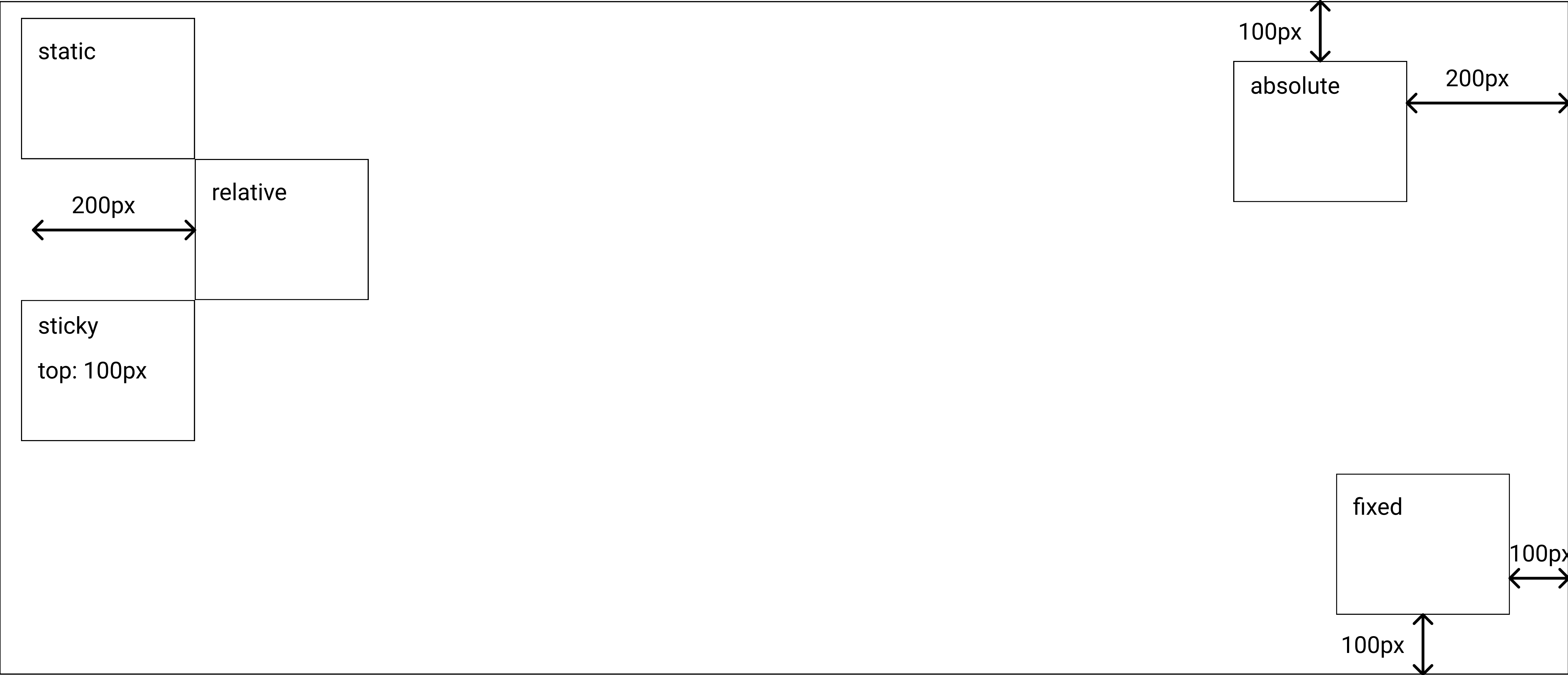
position: fixed

fixed는 화면 전체를 기준으로 좌표값을 적용시킵니다.
스크롤에 영향을 받지 않고 항상 같은 위치에 있기 때문에 고정된 UI를 만들 때 사용됩니다.

position: sticky

sticky는 평소에는 static요소의 위치에 있지만 스크롤을 내릴 경우 설정한 값에 맞춰서 위치를 유지합니다.
기준이 되는 스크롤은 element의 부모 요소에 영향을 받습니다.

EX2



CSS z-index

z-index

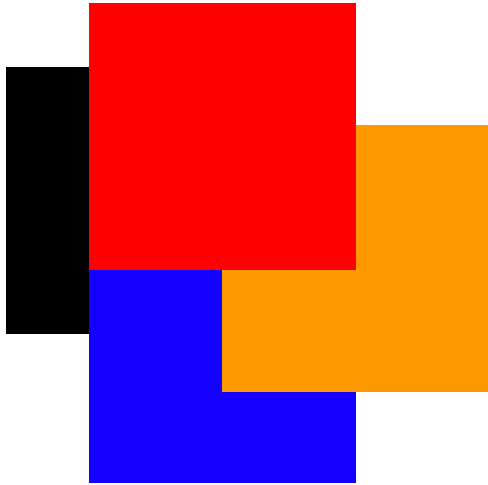
position 속성을 이용해서 요소들의 위치가 바꼈을 때, 요소들끼리 겹치는 경우가 발생합니다. 보통 요소들이 겹치면 코드 상 나중에 입력한 것일 수록 위로 올라오는데 z-index를 통해서 그 순서를 변경할 수 있습니다.

z-index는 숫자가 클 수록 위로 올라가고, 작을 수록 아래로 내려갑니다. 음수또한 사용 가능합니다.

// .box2 .box1 .box3 .box4 의 순서를 가집니다.

```
.box1 { position: absolute; z-index: 10}  
.box2 { position: absolute; z-index: 100}  
.box3 { position: absolute; z-index: 1}  
.box4 { position: absolute; z-index: -1}
```


EX3



EX4

————— This is Text —————

HTML ul, ol

목록 태그 ul, ol

html 에서 목록을 나타내는 태그로는 ul, ol 이 있습니다.

ul은 unordered list 의 약자로 주로 점을 이용한 목록을 만들 때 사용됩니다.

ol은 ordered list의 약자로 순서가 필요한 목록을 만들 때 사용됩니다.

ul, ol은 리스트를 감싸는 태그로 해당 요소의 목록을 작성할 땐 li 요소가 사용됩니다.

예)

- ul입니다1
- ul입니다2
 - ul입니다2-1

1. ol입니다1
2. ol입니다2
 - a. ol입니다2-1

CSS list-style

list-style-type

list-style-type은 목록 마커의 타입을 설정하는 속성입니다.

disc: 기본값, 검은 원

circle: 원

square: 사각형

decimal: 사각형

none: 표시안함

예)

```
list-style-type: circle;
```

list-style-position

목록 마커의 position을 설정합니다.

inside: 문단 안쪽에 놓습니다.

outside: 문단 바깥쪽에 놓습니다.

예)

```
list-style-position: inside;
```

CSS list-style

list-style-image

항목의 종류를 이미지로 넣는 속성 입니다.

none: 기본값입니다. 이미지를 사용하지 않습니다.

url: 사용할 이미지의 경로를 입력합니다.

예)

```
list-style-image: url('assets/image/check.png');
```

list-style

list-style은 ul, ol, li 요소에 주는 스타일로

리스트 앞에 붙어 나오는 점이나 숫자(목록 마커)를 설정하는 속성 입니다.

list-style-type, list-style-position, list-style-image 의 단축 속성으로 각각 따로 지정 가능합니다.

예)

```
list-style-image: square inside url('assets/image/check.png');
```

```
list-style-image: square;
```

```
list-style-image: inside;
```

```
list-style-image: url('assets/image/check.png');
```

UI에서의 ul, li

UI에서의 ul, li

ul, li는 단순한 목록을 보여줄때 사용하는경우보단 리스트형 ui를 표현할 때 많이 사용됩니다.

예)

Some fruits
Lemons
Raspberries
Strawberries
Blackberries
Kiwis
Grapefruit
Avocado
Watermelon

CE	Carla Espinosa Nurse
BK	Bob Kelso Doctor of Medicine
J	Janitor Janitor
PC	Perry Cox Doctor of Medicine
BS	Ben Sullivan Comedian and talk show host

An item
A second item
A third item
A fourth item
And a fifth one

EX5

1370px



HOME

FEATURES

ARCHIVE

PAGES

ABOUT

CONTACT

CSS font

font

CSS를 통해 글자의 모양을 제어할 수 있습니다.

글자의 크기나 색상등을 설정을 통해 변경할 수 있으며, 우선적으로 많이 다루는 속성은 다음과 같습니다.

font-size: 글자의 크기를 설정할 수 있습니다.

font-weight: 글자의 두께를 설정합니다.

line-height: 줄 간격을 설정합니다.

letter-spacing: 자간을 설정합니다.

font-family: 글꼴을 설정합니다.

예)

font-size: 20px;

font-weight: 500;

line-height: 20px;

letter-spacing: -0.3px;

font-family: "Noto Sans Kr";

CSS font

font

CSS를 통해 글자의 모양을 제어할 수 있습니다.

글자의 크기나 색상등을 설정을 통해 변경할 수 있으며, 우선적으로 많이 다루는 속성은 다음과 같습니다.

font-size: 글자의 크기를 설정할 수 있습니다.

font-weight: 글자의 두께를 설정합니다.

line-height: 줄 간격을 설정합니다.

letter-spacing: 자간을 설정합니다.

font-family: 글꼴을 설정합니다.

예)

font-size: 20px;

font-weight: 500;

line-height: 20px;

letter-spacing: -0.3px;

font-family: "Noto Sans Kr";

CSS font

font-size

font-size는 글자의 크기를 설정합니다.

px, em, rem 단위를 주로 많이 사용하고 in, pt, cm, vw, vh, % 단위도 있지만 잘 사용되지 않습니다.

예)

```
font-size: 20px;
```

font-weight

font-weight는 글자의 두께를 설정합니다.

100, 200, ~ 900 까지의 값을 가질 수 있습니다.

수치 말고 normal, bold 등의 값들도 사용가능합니다.

light: 300

normal: 400 (기본굵기)

medium: 500

semi bold: 600

bold: 700

예)

```
font-weight: 400;
```

```
font-weight: normal;
```

CSS font

line-height

line-height는 줄간격을 설정합니다.

숫자값, px, % 등으로 설정 가능합니다.

숫자값으로 설정할 경우 글자크기에 해당 숫자만큼의 배수가 적용된 값이 설정됩니다.

예)

line-height: 20px;

line-height: 1.2;

letter-spacing

letter-spacing 자간을 설정합니다.

숫자값, px, % 등으로 설정 가능합니다.

예)

letter-spacing: 2px;

CSS font

font-family

font-family 는 글꼴을 정할 때 사용합니다.

설정하고자 하는 글꼴을 설치 후 설치할 때 등록된 이름으로 불러서 사용합니다.

글꼴을 여러개 설정할 경우 순서대로 적용 가능한지 찾아서 가능한 글꼴을 적용합니다.

예)

font-family: Georgia, "Times New Roman", serif;

EX6

[HOME](#)

[FEATURES](#)

[ARCHIVE](#)

[PAGES](#)

[ABOUT](#)

[CONTACT](#)

