



텍스트 스타일

🕒 작성일시	@2023년 12월 12일 오전 10:05
📁 유형	CSS
☑ 진행	<input type="checkbox"/>

텍스트 스타일

- 🔔 텍스트 크기 : [font-size](#)
- 🔔 글자 스타일 : [font-style](#)
- 🔔 글자 색 : [color](#)
- 🔔 텍스트에 줄 표시/없애기 : [text-decoration](#)
- 🔔 텍스트 간격 : [letter-spacing](#), [word-spacing](#)
- 🔔 텍스트 그림자 : [text-shadow](#)
- 🔔 텍스트 대/소문자 변환 : [text-transform](#)
- 🔔 텍스트 정렬 : [text-align](#)
- 🔔 텍스트 정렬 시 공백 조정 : [text-justify](#)
- 🔔 글자 줄 간격 : [line-height](#)
- 🔔 텍스트 공백 처리 : [white-space](#)
- 🔔 텍스트 줄 바꿈 : [word-break](#), [overflow-wrap](#)
- 🔔 글자 세로로 작성 : [writing-mode](#) / [text-orientation](#) : [upright](#)
- 🔔 텍스트 말 줄임표

텍스트 스타일

🔔 텍스트 크기 : font-size

`font-size` 속성은 텍스트 크기를 조절할 때 사용한다.

텍스트 크기는 모든 브라우저에서 16px을 기본값으로 설정되어 있다.

단위는 `px` 같은 절대 단위 고정 값을 주로 사용하고 반응형 제작 시 상대에 따라 변화 시켜줄 수 있는 `rem`, `vw`, `vh` 를 사용할 수 있다.

```
<style>
  body {
    /*font-size: 10px;*/
    font-size: 62.5%; /* 10px */
  }
  p:first-child {
    font-size: 20px;
  }
  p:nth-child(2) {
    font-size: 2rem; /* body의 초기 폰트크기에 따라 변화됨, 20px */
  }
  p:last-child {
    font-size: 3vw; /*브라우저 너비값에 따라 지속적으로 달라짐*/
  }
</style>

<body>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit</p>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit</p>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit</p>
</body>
```

🔔 글자 스타일 : font-style

`font-style` 속성은 글자 스타일을 조절할 때 사용하며, 기본스타일, 이탤릭체, 기울어진 스타일로 설정할 수 있다.

속성값으로는 `normal`, `italic`, `oblique` 가 있다.

`italic`, `oblique` 의 스타일은 차이가 없으며, 웹에서는 주로 `italic` 을 사용한다.

```
em {
  /*글자 스타일 기본값, 기울어진 모습을 기본 형태로 만들어준다.*/
  font-style:normal;
}

div {
  /*글자 모양을 이탤릭체로 표시해준다.*/
  font-style : italic
}

p {
  /*본래 글꼴에서 기울어진 모양으로 표시해준다.*/
  font-style : oblique;
}
```

🔔 글자 색 : color

`color` 속성은 텍스트 색상을 지정할 때 사용한다.

`color` 색상 이름, `HEX(#ff0000 16진수)`, `rgb(0, 0, 0)`, `rgba(0, 0, 0, 0.5)`, hsl등의 속성 값으로 사용할 수 있다.

`rgb()` 와 `rgba()` 의 차이는 투명도를 적용하는 것으로, a는 투명도 즉, opacity값을 추가하면 된다.

디자인 프로그램에서 작업 시 opacity값을 50% 되어 있다면 `rgba` 값에서 0.5 또는 .5로 표기할 수 있다.

```
<style>
  p:first-child {
    color:blue;
  }
  p:nth-child(2) {
    color:#ff0000;
  }
  p:last-child {
    color:rgba(0, 0, 0, 0.5);
  }
</style>

...
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit</p>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit</p>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit</p>
```

🔔 텍스트에 줄 표시/없애기 : text-decoration

`text-decoration` 속성은 텍스트에 밑줄, 취소선, 윗줄등을 표시해줄 수 있다.

속성값 `none` `underline` `overline` `line-through`

```
a {
  /*기본값, 밑줄 없애기, a태그한테는 기본적으로 밑줄이 표시되어 있어서 없애버릴 수 있다.*/
  text-decoration: none;
```

```

}
p {
  /*텍스트 밑줄*/
  text-decoration: underline;
}
div {
  /*텍스트 윗줄*/
  text-decoration: overline;
}
span {
  /*텍스트 취소선*/
  text-decoration: line-through;
}

```

🔔 텍스트 간격 : letter-spacing, word-spacing

`letter-spacing` 은 글자 사이 간격을 조절할 수 있다.

`word-spacing` 은 단어 사이 간격을 조절할 수 있다.

주로 `letter-spacing` 을 많이 사용하며, 음수와 양수를 표현할 수 있습니다. 단위는 `px` 나 `%` 로 설정할 수 있다.

```

p {
  /*글자사이 간격 양수, 글자사이 넓어짐*/
  letter-spacing: 2px;
}
a {
  /*글자사이 간격 음수, 글자사이 좁아짐*/
  letter-spacing: -2px;
}
div {
  /*문자사이 간격 양수, 문자사이 넓어짐.*/
  letter-spacing: 2px;
}
strong {
  /*문자사이 간격 음수, 문자사이 좁아짐.*/
  letter-spacing: -2px;
}

```

🔔 텍스트 그림자 : text-shadow

`text-shadow` 속성은 글자 모양대로 그림자를 적용해줄 수 있다.

box-shadow와 동일하게 `가로거리 세로거리 번짐정도 색상` 을 적용하여 값을 설정할 수 있다.

거리, 번짐은 음수, 양수 값을 사용할 수 있으며, 단위는 `px` 이다.

- **가로 거리** : 텍스트로부터 그림자 까지의 가로 거리(오른쪽)
- **세로 거리** : 텍스트로부터 그림자 까지의 세로 거리(아래쪽)
- **번짐 정도** : 양수 값을 사용하면 모든 방향으로 퍼져나가 그림자가 크게 표시되며, 음수 값일 땐 모든 방향으로 축소되어 보인다. 기본 값은 0이다.
- **색상** : 한 가지 색상, 혹은 공백으로 구분하여 여러 색상을 지정할 수 있다. 기본 값은 현재 글자 색이다.

```

p {
  /*기본값*/

```

```

    text-shadow : none;
}
div {
    /*양수적용 오른쪽 아래쪽 번짐정도 색상*/
    text-shadow : 4px 4px 10px rgba(0, 0, 0, 0.5);
}
span {
    /*음수적용 왼쪽 위쪽 번짐정도 색상*/
    text-shadow : -4px -4px 10px rgba(0, 0, 0, 0.5);
}
strong {
    /*양쪽으로 그림자 적용, 상하좌우로 그림자를 생성할 수 있음.*/
    text-shadow : 0px 0px 10px rgba(0, 0, 0, 0.5);
}
}

```

🔔 텍스트 대/소문자 변환 : text-transform

`text-transform` 속성은 텍스트 대소문자를 변환 시켜 줄 수 있으며, 영문에 대부분 사용된다.

속성 값 `none` `capitalize` `uppercase` `lowercase` `full-width`

- **none** : 기본 값
- **capitalize** : 첫 글자 대문자
- **uppercase** : 모두 대문자
- **lowercase** : 모두 소문자
- **full-width** : 전각 문자 변환

```

p {
    /*기본값*/
    text-transform: none;
}
div {
    /*첫글자 대문자*/
    text-transform: capitalize;
}
span {
    /*모두 대문자*/
    text-transform: uppercase;
}
strong {
    /*모두 소문자*/
    text-transform: lowercase;
}
b {
    /*전각문자 변환,
한중일 인쇄 문화에서 사용되는 용어,
문자의 폭이 일반적인 영문자의 고정폭의 두 배 정도의 폭을 가지는 문자를 전각문자라고 함.*/
    text-transform: full-width;
}
}

```

🔔 텍스트 정렬 : text-align

`text-align` 속성은 글자의 정렬을 시켜줄 수 있지만, 글자 뿐만 아니라 inline요소를 포함하고 있는 구성 요소를 정렬 시켜줄 수 있다.

속성 값 `start` `end` `left` `right` `center` `justify` `match-parent`

- **start** : 현재 텍스트 방향의 시작 위치에 문단 정렬
- **end** : 현재 텍스트 방향의 끝 위치에 문단 정렬
- **left** : 왼쪽에 맞추어 문단 정렬 (기본 값)
- **right** : 오른쪽에 맞추어 문단 정렬
- **center** : 가운데 맞추어 문단 정렬
- **justify** : 양쪽에 맞추어 문단 정렬
- **match-parent** : 부모 요소에 따라 문단 정렬

```
p {
  text-align: center;
}
div {
  text-align: right;
}
```



img태그 를 이용하여 배치할 때 img가 부모 영역의 좌측, 가운데, 우측으로 배치하고자 한다면 부모에게 **text-align** 을 사용하여 img를 정렬 시켜 줄 수 있다.

🔔 텍스트 정렬 시 공백 조정 : text-justify

text-align: justify 를 적용할 때 글자 사이 간격이 어색하게 벌어질 수 있다. 이를 조정하는 속성이 **text-justify** 이다.

속성 값 **auto** **none** **inter-word** **distribute**

- **none** : 정렬하지 않음
- **auto** : 웹 브라우저에서 자동으로 지정
- **inter-word** : 단어 사이의 공백을 조절해 정렬
- **distribute** : 인접한 글자 사이의 공백을 똑같이 맞추어 정렬

```
p {
  /*단어 사이의 공백을 조절해 정렬*/
  text-justify : inter-word;
}
div {
  /*인접한 글자 사이의 공백을 똑같이 맞추어 정렬*/
  text-justify : distribute;
}
```

🔔 글자 줄 간격 : line-height

line-height 속성은 글자 세로 줄 간격을 설정해줄 수 있다.

글자 크기가 20px일 때 line-height값을 40px로 설정하면 위 아래 각 각 10px의 여백이 생긴다.

줄 간격이 글자 크기보다 작으면 글자가 겹쳐지게 된다.

속성 값으로 **normal** **px** **숫자** **%** **inherit** 으로 적용할 수 있다.

- **normal** : 브라우저에서 정한 기본 값. 보통 1.2이다.
- **px** : px 값으로 간격 지정
- **숫자** : 글자 크기의 배수(글자 크기의 1.3 ~ 1.6 정도가 가독성이 좋음)

- **백분율** : 글자의 몇 %인지 지정
- **inherit** : 부모 요소의 속성 값을 상속 받아서 설정

```
body {
  font-size: 16px;
}
p {
  /* line-height값은 16 * 1.5 => 24px*/
  line-height: 1.5;
}

.button {
  width : 120px;
  height : 40px;
  background-color: #333;
  color: #fff;
  text-align:center;
  line-height: 40px;
  border-radius: 5px;
}
```



박스 스타일에서 텍스트를 세로로 가운데 정렬(수직 가운데 정렬) 할 때 `line-height` 값을 `height` 값과 동일하게 지정해주면 된다.



텍스트 공백 처리 : white-space

html에서 아무리 많은 공백을 가지고 있더라도 하나의 띄워 쓰기로 처리가 된다. 그리고 텍스트의 길이가 길 경우 자동으로 줄 바꿈 처리가 발생되는데 이러한 현상들에 대한 스타일을 잡아줄 수 있는 것이 `white-space` 에 따른 속성 값들이 존재한다.

속성 값 `normal` `nowrap` `pre` `pre-line` `pre-wrap`

- **normal** : 여러 개의 공백을 하나로 표시 (기본 값)
- **nowrap** : 공백을 하나로 표시하고, 영역 너비를 넘어가는 내용은 줄을 바꾸지 않고 계속 한 줄로 표시
- **pre** : 여러 공백을 그대로 표시, 너비를 넘어가는 내용은 한 줄로 표시
- **pre-line** : 여러 공백을 하나로 표시, 너비를 넘어가면 줄 바꿈
- **pre-wrap** : 여러 공백을 그대로 표시, 너비를 넘어가면 줄 바꿈

```
p {
  white-space : nowrap;
}
div {
  white-space : pre-line;
}
```



텍스트 줄 바꿈 : word-break, overflow-wrap

`word-break` 속성을 이용하여 텍스트가 부모 영역을 벗어날 것인지, 글자 하나로 줄 바꿈 할 것인지, 단어로 할 것인지 결정할 수 있다.

기본적으로 국문은 글자로 줄 바꿈이 이루어지며, 영문은 단어로 줄 바꿈이 이루어진다.

속성 값 `normal` `break-all` `keep-all`

- **normal** : 기본 값으로 국문은 단어로 줄 바꿈이 이루어지고, 영문과 숫자는 띄워 쓰기가 있다면 그 부분이 줄 바꿈이 이루어진다. 만약 띄워 쓰기가 없다면 줄 바꿈이 적용이 안되고 부모 영역을 벗어나게 된다.

- **break-all** : 국문, 영문, 숫자 모두 글자로 줄 바꿈이 적용된다.
- **keep-all** : 단어를 기준으로 줄 바꿈이 적용된다. 하지만 영문, 숫자에서 띄워 쓰기 없다면 부모 영역을 벗어나게 된다.

💡 국문 단어로 줄 바꿈 해주고자 할 때 `word-break:keep-all` 을 사용할 수 있다.

`overflow-wrap` 속성은 한 줄로 배치할 수 없을 때 사용할 수 있다. 영문인 경우 띄워 쓰기가 없다면 줄 바꿈 처리가 이루어지지 못하는데 그런 상황일 때 사용할 수 있다.

속성 값 `normal` `anywhere` `break-word`

- **normal** : 기본적인 분리 지점에서 줄 바꿈이 발생된다.
- **anywhere** : 긴 단어나 URL과 같이 분리할 수 없는 문자열인 경우 임의의 지점에서 줄 바꿈이 발생한다.
- **break-word** : `anywhere` 과 동일하게 중단할 수 없는 문자열도 임의의 지점에서 줄 바꿈을 허용한다. **min-content** 고유 크기를 계산할 때 소프트 랩 기회를 고려하지 않는다.

💡 `word-break: break-all` 와 `overflow-wrap : break-work` 를 함께 사용하는 것을 추천한다.

word-break: break-all
overflow: break-word

능히 청춘 할지라도 할지
니, 그들의 것은 있으랴?
밝은 청춘은 그와 운다.
우리 천자만홍이 열매를
아니더면, 같은 가슴이 남
는 봄바람이다.

https://www.google.com/search?q=overflow-wrap&ei=PneFZOrrHsfQ-QaS8oSACw&ved=0ahUKEwjgpa22Lr_AhVHaN4KHRI5AbAQ4dUDCBA&uact=5&oq=overflow-wrap&gs_lcp=Cgxn d3Mtd2l6LXNlcnAQAzlFCAAQgAQyBQgAEIAEMgUIABCABDIFCAAQgAQyBQgAEIAEMgUIABCABDIFCAAQgAQyBQgAEIAEMgUIABCABDIFCAAQgAQ6CggAEecQ1gQQsANKBAhBGABQ7AFY4A1gqA9oBHABeACAAbQBIAHzB5IBAzAuOJgBAKABAcABAcgBCg&scient=gws-wiz-serp

12312412412312412412
31241241231241241231
24124123124124123124
12
4!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!

<https://velog.io/@ongddree/css-줄바꿈-스타일-word-break-overflow-wrap>

🔔 글자 세로로 작성 : `writing-mode / text-orientation : upright`

`writing-mode` 속성을 이용하여 텍스트를 세로로 작성할 수 있다. 세로에서 오른쪽에서 왼쪽으로, 왼쪽에서 오른쪽으로 설정할 수 있다.

속성 값 `vertical-lr` `vertical-rl`

- `vertical-lr` : 세로로 왼쪽에서 오른쪽으로 배치
- `vertical-rl` : 세로로 오른쪽에서 왼쪽으로 배치

`text-orientation : upright` 은 영어에서 추가적으로 사용할 수 있다. 영문으로 세로로 작성 하다 보면 90도로 돌아간 모습으로 표시가 되는데 이를 해결하기 위한 방법으로 사용할 수 있다.

```
p {
  writing-mode : vertical-rl;
  text-orientation : upright;
}
```



텍스트 말 줄임표



[참고사이트]

<https://gurtn.tistory.com/181>

<https://hyejin.tistory.com/101>

<https://velog.io/@ongddree/css-줄바꿈-스타일-word-break-overflow-wrap>