

🚀 김쌤과 함께 쉽고 빠르게

부트스트랩4를 활용한 반응형 웹 만들기



STEP 16

그밖의 유틸리티들4_ textHide부터 visibility속성까지

INDEX

이미지를 대체하는 클래스

인터렉션 관련 클래스

오버플로우 관련 클래스

위치 관련 클래스

그림자 스타일 클래스

범위 관련 클래스

텍스트 관련 클래스



```
<h1 class="text-hide" style="background-image: url('...');">Bootstrap</h1>
```

Warning

The `text-hide()` class and mixin has been deprecated as of v4.1. It will be removed entirely in v5.

Utilize the `.text-hide` class or mixin to help replace an element's text content with a background image.

```
<h1 class="text-hide">Custom heading</h1>
```

Copy

```
.text-hide {  
  font: 0/0 a;  
  color: transparent;  
  text-shadow: none;  
  background-color: transparent;  
  border: 0;  
}
```

```
<a class="sr-only sr-only-focusable" href="#content">Skip to main content</a>
```

```
.sr-only {  
  position: absolute;  
  width: 1px;  
  height: 1px;  
  padding: 0;  
  margin: -1px;  
  overflow: hidden;  
  clip: rect(0, 0, 0, 0);  
  white-space: nowrap;  
  border: 0;  
}
```

Text selection

Change how the content is selected when the user interacts with it. Note that Internet Explorer and Legacy Edge have no support for the `all` value for `user-select`, and as such, `.user-select-all` is not supported by either browser.

This paragraph will be entirely selected when clicked by the user.

This paragraph has the default select behavior.

This paragraph will not be selectable when clicked by the user.

```
<p class="user-select-all">This paragraph will be entirely selected when clicked by th
<p class="user-select-auto">This paragraph has the default select behavior.</p>
<p class="user-select-none">This paragraph will not be selectable when clicked by the
```

<

>

Barebones **overflow** functionality is provided for two values by default, and they are not responsive.

element with set width and height dimensions. By design, this content will vertically scroll.



This is an example of using **.overflow-hidden** on an element with set width and height dimensions.

[Copy](#)

```
<div class="overflow-auto">...</div>  
<div class="overflow-hidden">...</div>
```

.position-static / .position-relative / .position-absolute / .position-fixed / .position-sticky / .fixed-top / .fixed-bottom / .sticky-top

Common values

Quick positioning classes are available, though they are not responsive.

```
<div class="position-static">...</div>
<div class="position-relative">...</div>
<div class="position-absolute">...</div>
<div class="position-fixed">...</div>
<div class="position-sticky">...</div>
```

Fixed top

Position an element at the top of the viewport, from edge to edge. understand the ramifications of fixed position in your project; you i additional CSS.

```
<div class="fixed-top">...</div>
```

Sticky top

Position an element at the top of the viewport, from edge to edge, scroll past it. The `.sticky-top` utility uses CSS's `position: sticky`, v supported in all browsers.

IE11 and IE10 will render `position: sticky` as `position: relative` styles in a `@supports` query, limiting the stickiness to only browsers properly.

```
<div class="sticky-top">...</div>
```

Fixed bottom

Position an element at the bottom of the viewport, from edge to understand the ramifications of fixed position in your project; yo additional CSS.

```
<div class="fixed-bottom">...</div>
```


No shadow

Small shadow

Regular shadow

Larger shadow

```
<div class="shadow-none p-3 mb-5 bg-light rounded">No shadow</div>  
<div class="shadow-sm p-3 mb-5 bg-white rounded">Small shadow</div>  
<div class="shadow p-3 mb-5 bg-white rounded">Regular shadow</div>  
<div class="shadow-lg p-3 mb-5 bg-white rounded">Larger shadow</div>
```

Width 25%

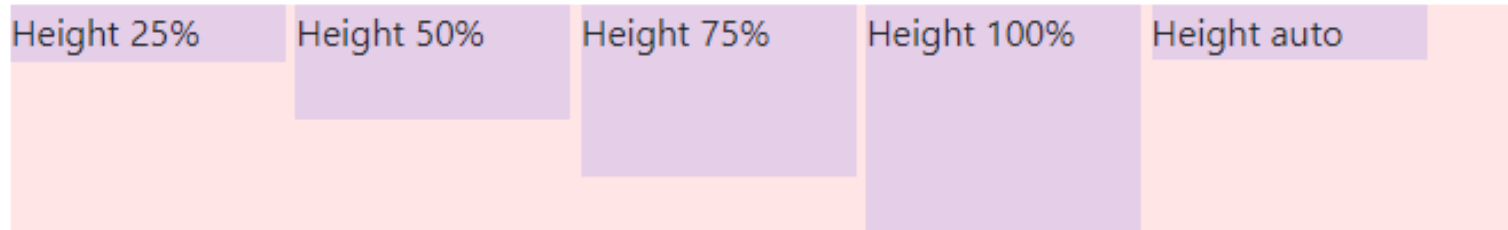
Width 50%

Width 75%

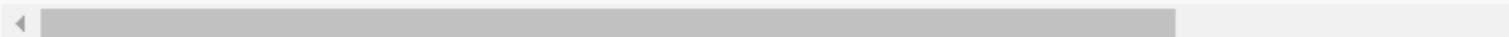
Width 100%

Width auto

```
<div class="w-25 p-3" style="background-color: #eee;">Width 25%</div>  
<div class="w-50 p-3" style="background-color: #eee;">Width 50%</div>  
<div class="w-75 p-3" style="background-color: #eee;">Width 75%</div>  
<div class="w-100 p-3" style="background-color: #eee;">Width 100%</div>  
<div class="w-auto p-3" style="background-color: #eee;">Width auto</div>
```



```
<div style="height: 100px; background-color: rgba(255,0,0,0.1);">  
  <div class="h-25 d-inline-block" style="width: 120px; background-color: rgba(0,0,255,  
  <div class="h-50 d-inline-block" style="width: 120px; background-color: rgba(0,0,255,  
  <div class="h-75 d-inline-block" style="width: 120px; background-color: rgba(0,0,255,  
  <div class="h-100 d-inline-block" style="width: 120px; background-color: rgba(0,0,255  
  <div class="h-auto d-inline-block" style="width: 120px; background-color: rgba(0,0,25  
</div>
```



You can also use `max-width: 100%;` and `max-height: 100%;` utilities as needed.

Max-width 100%

```

```

```
.mw-100 {  
  max-width: 100% !important;  
}
```

Max-height
100%

```
<div style="height: 100px; background-color: rgba(255,0,0,0.1);">  
  <div class="mh-100" style="width: 100px; height: 200px; background-color: rgba(0,0,255,0.1);">  
  </div>  
</div>
```

```
.mh-100 {  
  max-height: 100% !important;  
}
```



Card with stretched link

Some quick example text to build on the card title and make up the bulk of the card's content.

Go somewhere

```
<div class="card" style="width: 18rem;">
  
  <div class="card-body">
    <h5 class="card-title">Card with stretched link</h5>
    <p class="card-text">Some quick example text to build on the card title and
    <a href="#" class="btn btn-primary stretched-link">Go somewhere</a>
  </div>
</div>
```

```
.stretched-link::after {
  position: absolute;
  top: 0;
  right: 0;
  bottom: 0;
  left: 0;
  z-index: 1;
  pointer-events: auto;
  content: "";
  background-color: rgba(0, 0, 0, 0);
}
```

Identifying the containing block

If the stretched link doesn't seem to work, the **containing block** will probably be the cause. The following CSS properties will make an element the containing block:

- A **position** value other than **static**
- A **transform** or **perspective** value other than **none**
- A **will-change** value of **transform** or **perspective**
- A **filter** value other than **none** or a **will-change** value of **filter** (only works on Firefox)

Text alignment

Easily realign text to components with text alignment classes.

Some placeholder text to demonstrate justified text alignment. Will you do the same for me? It's time to face the music I'm no longer your muse. Heard it's beautiful, be the judge and my girls gonna take a vote. I can feel a phoenix inside of me. Heaven is jealous of our love, angels are crying from up above. Yeah, you take me to utopia.

```
<p class="text-justify">Some placeholder text to demonstrate justified text ali
```

.text-left / .text-center / .text-right / .text-미디어사이즈-정렬방향

Left aligned text on all viewport sizes.

Center aligned text on all viewport sizes.

Right aligned text on all viewport sizes.

Left aligned text on viewports sized SM (small) or wider.

Left aligned text on viewports sized MD (medium) or wider.

Left aligned text on viewports sized LG (large) or wider.

Left aligned text on viewports sized XL (extra-large) or wider.

```
<p class="text-left">Left aligned text on all viewport sizes.</p>
<p class="text-center">Center aligned text on all viewport sizes.</p>
<p class="text-right">Right aligned text on all viewport sizes.</p>

<p class="text-sm-left">Left aligned text on viewports sized SM (small) or wider
<p class="text-md-left">Left aligned text on viewports sized MD (medium) or wider
<p class="text-lg-left">Left aligned text on viewports sized LG (large) or wider
<p class="text-xl-left">Left aligned text on viewports sized XL (extra-large) or
```

```
@media (min-width: 576px) {
  .text-sm-left {
    text-align: left !important;
  }
  .text-sm-right {
    text-align: right !important;
  }
  .text-sm-center {
    text-align: center !important;
  }
}
```

`.align-baseline / .align-top / .align-middle / .align-bottom` `/ .align-text-top / .align-text-bottom`

baseline top middle bottom text-top text-bottom

```
<span class="align-baseline">baseline</span>
<span class="align-top">top</span>
<span class="align-middle">middle</span>
<span class="align-bottom">bottom</span>
<span class="align-text-top">text-top</span>
<span class="align-text-bottom">text-bottom</span>
```

With table cells:

baseline top text-top text-bottom
middle
bottom

```
<table style="height: 100px;">
  <tbody>
    <tr>
      <td class="align-baseline">baseline</td>
      <td class="align-top">top</td>
      <td class="align-middle">middle</td>
      <td class="align-bottom">bottom</td>
      <td class="align-text-top">text-top</td>
      <td class="align-text-bottom">text-bottom</td>
    </tr>
  </tbody>
</table>
```


Text wrapping and overflow

Wrap text with a `.text-wrap` class.

This text
should wrap.

```
<div class="badge badge-primary text-wrap" style="width: 6rem;">  
  This text should wrap.  
</div>
```

```
.text-wrap {  
  white-space: normal !important;  
}
```

Prevent text from wrapping with a `.text-nowrap` class.

This text should overflow the parent.

```
<div class="text-nowrap bd-highlight" style="width: 8rem;">  
  This text should overflow the parent.  
</div>
```

```
.text-nowrap {  
  white-space: nowrap !important;  
}
```

For longer content, you can add a `.text-truncate` class to truncate the text with an ellipsis.

Requires `display: inline-block` or `display: block`.

Praetere...

Praeterea iter est q...

```
<!-- Block level -->
<div class="row">
  <div class="col-2 text-truncate">
    Praeterea iter est quasdam res quas ex communi.
  </div>
</div>

<!-- Inline level -->
<span class="d-inline-block text-truncate" style="max-width: 150px;">
  Praeterea iter est quasdam res quas ex communi.
</span>
```

```
.text-truncate {
  overflow: hidden;
  text-overflow: ellipsis;
  white-space: nowrap;
}
```

Prevent long strings of text from breaking your components' layout by using `.text-break` to set `word-wrap: break-word` and `word-break: break-word`. We use `word-wrap` instead of the more common `overflow-wrap` for wider browser support, and add the deprecated `word-break: break-word` to avoid issues with flex containers.

[illegible]

19

Text transform

Transform text in components with text capitalization classes.

lowercased text.

UPPERCASED TEXT.

CapiTaliZed Text.

```
<p class="text-lowercase">Lowercased text.</p>  
<p class="text-uppercase">Uppercased text.</p>  
<p class="text-capitalize">CapiTaliZed text.</p>
```

Note how `.text-capitalize` only changes the first letter of each word, leaving the case of any other letters unaffected.

.font-weight-bold / .font-weight-normal / .font-weight-light / .font-weight-lighter / .font-italic

Font weight and italics

Quickly change the weight (boldness) of text or italicize text.

Bold text.

Bolder weight text (relative to the parent element).

Normal weight text.

Light weight text.

Lighter weight text (relative to the parent element).

Italic text.

```
<p class="font-weight-bold">Bold text.</p>
<p class="font-weight-bolder">Bolder weight text (relative to the parent
<p class="font-weight-normal">Normal weight text.</p>
<p class="font-weight-light">Light weight text.</p>
<p class="font-weight-lighter">Lighter weight text (relative to the paren
<p class="font-italic">Italic text.</p>
```

Reset color

Reset a text or link's color with `.text-reset`, so that it inherits the color from its parent.

Muted text with a reset link.

```
<p class="text-muted">  
  Muted text with a <a href="#" class="text-reset">reset link</a>.  
</p>
```








Text decoration

Remove a text decoration with a `.text-decoration-none` class.

Non-underlined link

```
<a href="#" class="text-decoration-none">Non-underlined link</a>
```

꼭 기억해야 할 포인트 정리

1. 이미지를 대체하는 클래스로 이미지는 보이고 텍스트는 숨겨서 스크린 리더기가 읽어주려면  클래스명을, 이미지와 텍스트 모두 숨기고 스크린 리더기가 읽어주려면  클래스명을 사용한다.
2. 콘텐츠가 부모요소보다 흘러넘쳐서 부모 영역 안에서 자동으로 스크롤이 생기도록 하려면  클래스를, 넘치는 콘텐츠를 숨기려면  클래스를 사용한다.
3. 인터렉션 관련 클래스로 텍스트를 클릭시 텍스트가 모두 선택되려면  클래스를 사용한다.
4. 텍스트가 보이는 화면의 너비에 따라 자동으로 줄넘김되려면  클래스명을, 생략표시인 ...으로 표시하려면  클래스명을 사용한다.

꼭 기억해야 할 포인트 정리

1. 이미지를 대체하는 클래스로 이미지는 보이고 텍스트는 숨겨서 스크린 리더기가 읽어주려면 `.text-hide` 클래스명을, 이미지와 텍스트 모두 숨기고 스크린 리더기가 읽어주려면 `.sr-only` 클래스명을 사용한다.
2. 콘텐츠가 부모요소보다 흘러넘쳐서 부모 영역 안에서 자동으로 스크롤이 생기도록 하려면 `.overflow-hidden` 클래스를, 넘치는 콘텐츠를 숨기려면 `.text-overflow` 클래스를 사용한다.
3. 인터렉션 관련 클래스로 텍스트를 클릭시 텍스트가 모두 선택되려면 `.selectable` 클래스를 사용한다.
4. 텍스트가 보이는 화면의 너비에 따라 자동으로 줄넘김되려면 `.text-wrap` 클래스명을, 생략표시인 ...으로 표시하려면 `.text-truncate` 클래스명을 사용한다.

꼭 기억해야 할 포인트 정리

1. 이미지를 대체하는 클래스로 이미지는 보이고 텍스트는 숨겨서 스크린 리더기가 읽어주려면 `.text-hide` 클래스명을, 이미지와 텍스트 모두 숨기고 스크린 리더기가 읽어주려면 `.sr-only` 클래스명을 사용한다.
2. 콘텐츠가 부모요소보다 흘러넘쳐서 부모 영역 안에서 자동으로 스크롤이 생기도록 하려면 `.overflow-auto` 클래스를, 넘치는 콘텐츠를 숨기려면 `.overflow-hidden` 클래스를 사용한다.
3. 인터렉션 관련 클래스로 텍스트를 클릭시 텍스트가 모두 선택되려면 `.select-all` 클래스를 사용한다.
4. 텍스트가 보이는 화면의 너비에 따라 자동으로 줄넘김되려면 `.text-wrap` 클래스명을, 생략표시인 ...으로 표시하려면 `.text-overflow` 클래스명을 사용한다.

꼭 기억해야 할 포인트 정리

1. 이미지를 대체하는 클래스로 이미지는 보이고 텍스트는 숨겨서 스크린 리더기가 읽어주려면 `.text-hide` 클래스명을, 이미지와 텍스트 모두 숨기고 스크린 리더기가 읽어주려면 `.sr-only` 클래스명을 사용한다.
2. 콘텐츠가 부모요소보다 흘러넘쳐서 부모 영역 안에서 자동으로 스크롤이 생기도록 하려면 `.overflow-auto` 클래스를, 넘치는 콘텐츠를 숨기려면 `.overflow-hidden` 클래스를 사용한다.
3. 인터렉션 관련 클래스로 텍스트를 클릭시 텍스트가 모두 선택되려면 `.user-select-all` 클래스를 사용한다.
4. 텍스트가 보이는 화면의 너비에 따라 자동으로 줄넘김되려면 `.text-wrap` 클래스명을, 생략표시인 ...으로 표시하려면 `.text-overflow` 클래스명을 사용한다.

꼭 기억해야 할 포인트 정리

1. 이미지를 대체하는 클래스로 이미지는 보이고 텍스트는 숨겨서 스크린 리더기가 읽어주려면 **.text-hide** 클래스명을, 이미지와 텍스트 모두 숨기고 스크린 리더기가 읽어주려면 **.sr-only** 클래스명을 사용한다.
2. 콘텐츠가 부모요소보다 흘러넘쳐서 부모 영역 안에서 자동으로 스크롤이 생기도록 하려면 **.overflow-auto** 클래스를, 넘치는 콘텐츠를 숨기려면 **.overflow-hidden** 클래스를 사용한다.
3. 인터렉션 관련 클래스로 텍스트를 클릭시 텍스트가 모두 선택되려면 **.user-select-all** 클래스를 사용한다.
4. 텍스트가 보이는 화면의 너비에 따라 자동으로 줄넘김되려면 **.text-break** 클래스명을, 생략표시인 ...으로 표시하려면 **.text-truncate** 클래스명을 사용한다.

영상 코딩 디자인은 영코디 김쌤



영코디 김쌤

