

เริ่มต้นใช้งาน

# Bootstrap 4

สำหรับผู้เริ่มต้น



By Monkeywebstudio.com

Bootstrap ในฐานะคนทำเว็บหรือ คิดจะทำเว็บ คงจะเคยได้ยิน หรือ ใช้งานอยู่เป็นประจำอยู่แล้ว สำหรับ CSS Framework ตัวนี้ แต่สำหรับ ใครที่เพิ่งเริ่มจะศึกษาการทำเว็บ หรือ ยังไม่เคยใช้งานมาก่อน ก็ อาจจะสงสัยว่า จะใช้งานมันยังไง ขอบอกไว้ก่อนเลยครับว่ามันง่ายมาก ๆ (เด็กอุบങก์ได้แล้ว) ในเอกสารชุดนี้ เราจะมาแนะนำการใช้งานเริ่มต้นให้ดูกันครับ

## พร้อมแล้วมาเริ่มกันเลยครับ

ก่อนจะเริ่มใช้งานเรามาทำความเข้าใจกันก่อนครับ ว่า Bootstrap CSS Framework มันคืออะไรกันแน่ สรุปสั้น ๆ เป็นภาษาhtmlcssjs คือ คำสั่ง HTML CSS JS ที่ถูกเขียนมาแล้วแบบสำเร็จรูปสามารถเรียกใช้งานได้ทันที ก็ได้หน้าเว็บที่มีปุ่ม หรือ รูปแบบการแสดงผลแบบ สวยงาม ๆ เมื่อ онตามตัวอย่างที่ทางผู้พัฒนาได้ออกแบบไว้ และ ที่สำคัญมันรองรับการแสดงผลแบบ responsive ด้วย หมายความว่า โครงสร้างเว็บเราไปเปิดที่ไหนก็ ตาม ไม่ว่าจะเป็น PC , Mobile หรือ Notebook ฯลฯ เราแสดงผลได้หมด จะว่า

ข่าวร้ายสำหรับคนใช้ IE (Internet Explorer) ตั้งแต่ version 10 ลงมา ต้องขอแสดงความเสียใจด้วย ครับ (คุณไม่ได้ไปต่อครับ) ตรงนี้จะมีปัญหามากครับ สำหรับบางองค์กร หรือ หน่วยงานราชการ ที่ใช้ Software รุ่นเก่า ที่รองรับเฉพาะ IE เท่านั้น แต่ถ้าใครใช้ Internet Explorer 10 -11 หรือ Microsoft Edge สามารถใช้งานได้ครับ

แล้วเจ้า Bootstrap จะไนยมมีมาแล้ว กี่ version กันแน่

สำหรับ Bootstrap ในตอนที่เขียน E-book อยู่นี้ได้เดินทางมาถึง Version 4 แล้วครับ

เริ่มต้นใช้งานจริง ๆ

ก่อนอื่น ไปที่เว็บไซต์ของผู้พัฒนา กันก่อนที่ <https://getbootstrap.com/> เมื่อเข้าไปก็จะพบหน้าจอ ต้องรับประมาณนี้



# Bootstrap

Build responsive, mobile-first projects on the web with the world's most popular front-end component library.



# การติดตั้งและเรียกใช้งาน

วิธีการติดตั้ง สามารถ Download เป็นไฟล์มาเก็บไว้ในเครื่องเรา หรือ หากไม่อยาก Download จะใช้งานแบบ Online โดยใช้บริการจาก CDN ที่ Bootstrap เตรียมไว้ให้ ก็ได้ครับ <https://getbootstrap.com/> ไฟล์ที่ต้องเรียกใช้งานมีดังนี้

- bootstrap.min.css
- jquery-3.2.1.slim.min.js
- popper.min.js
- bootstrap.min.js

สำหรับตัวอย่างทั้งหมดใน E-book เล่มนี้ จะเรียกใช้งานจาก CDN ครับ

## ตัวอย่างการเรียกใช้งาน

---

```
<!doctype html>
<html lang="en">
  <head>
    <title>Hello, world!</title>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta.2/css/bootstrap.min.css" >
  </head>
  <body>

    <h1>Hello, world!</h1>

    <!-- Optional JavaScript -->
    <!-- jQuery first, then Popper.js, then Bootstrap JS -->
    <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" ></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.3/umd/popper.min.js"></script>
    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta.2/js/bootstrap.min.js"></script>
  </body>
</html>
```

# Grid System

ระบบ Grid System คือ หลักการแบ่งหน้าจอ ออกเป็น 12 columns เท่า ๆ กัน โดยในแต่ละ 行 (row) จะมีกี่ columns ก็ได้ แต่ว่ามักจะต้องไม่เกิน 12 ช่อง มาลองดูภาพตัวอย่างกันเลย

span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1				
span 4				span 4				span 4							
span 4				span 8				span 6							
span 6						span 6									
span 12															

นอกเหนือจากนั้นเรายังสามารถกำหนดได้ว่าต้องให้ columns ไหนแสดงในหน้าจอแต่ละขนาดได้ด้วย อธิบายตามความเข้าใจของครั้งก็คือ อย่างให้บน Desktop แสดงเป็น 6 columns และถ้าไปเปิดหน้าเดิมบน Mobile ให้แสดงเป็น 12 columns ก็สามารถทำได้

โดย Bootstrap 4 ได้มีการแบ่งหน้าจอขนาดต่าง ๆ ดังนี้

	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	Extra large ≥1200px
<b>Max container width</b>	None (auto)	540px	720px	960px	1140px
<b>Class prefix</b>	.col-	.col-sm-	.col-md-	.col-lg-	.col-xl-
<b># of columns</b>	12				
<b>Gutter width</b>	30px (15px on each side of a column)				

- Extra small ( เรียกใช้งาน class .col )
- Small ( เรียกใช้งาน class col-sm-12 )
- Medium ( เรียกใช้งาน class col-md-12 )
- Large ( เรียกใช้งาน class col-lg-12 )
- Extra large ( เรียกใช้งาน class col-xl-12 )

## ตัวอย่างการใช้งาน

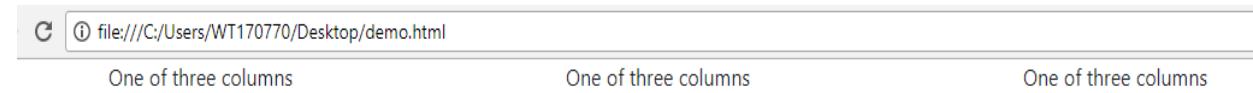
ในหน้าจอขนาด Medium แบ่งหน้าจอออกเป็น 3 ส่วนเท่า ๆ กัน ก็จะแบ่ง ได้ column ละ 4 มีทั้งหมด 3 column ก็จะได้เท่ากับ 12 พอดี

การเรียกใช้งาน Grid เราจะสร้าง class row ขึ้นมาก่อน และ เรียก class="col-ขนาดหน้าจอ-จำนวน Grid"

## ตัวอย่างการเรียกใช้งาน

```
<div class="container">
<div class="row">
<div class="col-md-4">
  One of three columns
</div>
<div class="col-md-4">
  One of three columns
</div>
<div class="col-md-4">
  One of three columns
</div>
</div>
</div>
```

ตัวอย่างการแสดงผล จะเห็นว่า ได้มีการแบ่งหน้าจอออกเป็น 3 ส่วน ตามที่เรากำหนดไว้



## Auto-layout columns

เราสามารถให้ Bootstrap กำหนดขนาดของ columns ให้เราอัตโนมัติ โดยใช้ class .col

```
<div class="row">
<div class="col">
  1 of 2
</div>
<div class="col">
  2 of 2
</div>
</div>
```

## ตัวอย่างการแสดงผล

1 of 2

2 of 2

# Tables

สำหรับ Table ปกติ เวลาที่เราเขียน HTML เวลาต้องแสดงข้อมูลที่เป็น ตาราง เราอาจต้องมาเขียนในส่วนของการจัดรูปแบบตาราง เช่น ใส่เส้นขอบ(border), ใส่สี (background-color), หรือ จัดระยะของตาราง (margin , padding) ฯลฯ

ซึ่งบางครั้งเราต้องการความรวดเร็วในการพัฒนา การใช้ Bootstrap Tables สามารถช่วยในส่วนนี้ได้มากทำให้สามารถสร้าง Table ที่มีความสวยงามในเวลาอันรวดเร็ว

สำหรับในตัวอย่างนี้เราจะแนะนำการใช้งาน Table ในแบบต่าง ๆ ดังนี้ Basic, Striped, bordered สามารถศึกษาการใช้งานเพิ่มเติมได้ที่ <http://getbootstrap.com/docs/4.0/content/tables/>

## Basic Table

การใช้งานเพียงแค่ เพิ่ม class ที่ชื่อว่า table ลงไปก็สามารถใช้งานได้ทันที

```
<table class="table">
  <thead>
    <tr>
      <th scope="col"># </th>
      <th scope="col">First Name </th>
      <th scope="col">Last Name </th>
      <th scope="col">Username </th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th scope="row">1 </th>
      <td>Mark </td>
      <td>Otto </td>
      <td>@mdo </td>
    </tr>
    <tr>
      <th scope="row">2 </th>
      <td>Jacob </td>
      <td>Thornton </td>
      <td>@fat </td>
    </tr>
    <tr>
      <th scope="row">3 </th>
      <td>Larry </td>
      <td>the Bird </td>
      <td>@twitter </td>
    </tr>
  </tbody>
</table>
```

## ตัวอย่างการแสดงผล เมื่อใช้ bootstrap table

#	First Name	Last Name	Username
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

เปรียบเทียบการแสดงผล กราณีที่ไม่ได้ใช้ bootstrap

### # First Name Last Name Username

1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

จะเห็นว่า หากไม่ได้ใช้ bootstrap เราต้องเขียนคำสั่ง CSS เพื่อจัดการกับตารางเองทั้งหมดซึ่งอาจใช้เวลามาก

## Striped rows

คือการสลับสี ขาว – เทา (zebra-striping) ใน ตารางให้มีความสวยงามมากยิ่งขึ้น โดยการเพิ่ม class table-striped ใน table

ตัวอย่าง การเพิ่ม class table-striped ใน table

```
<table class="table table-striped">
```

## ตัวอย่างการแสดงผล

#	First Name	Last Name	Username
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

## Bordered table

คือการเติมเส้นขอบให้กับตาราง โดยการเพิ่ม class table-bordered ลงใน table

ตัวอย่าง การเพิ่ม class table-bordered ลงใน table

```
<table class="table table-bordered">
```

ตัวอย่างการแสดงผล จะมีการเพิ่มเส้นขอบให้กับตารางให้มีความเรียบง่ายมากยิ่งขึ้น

#	First Name	Last Name	Username
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

## Hover Table

เมื่อนำมาสู่ปุ่มที่ตาราง จะมี hover สีเทาที่ตารางเพื่อให้เห็นว่าเรากำลังซึ่งที่ข้อมูลในແຕ່ໄດ້ຍຸ້ງ โดยเพิ่ม class .table-hover

```
<table class="table table-hover">
```

ตัวอย่างการแสดงผล

#	First Name	Last Name	Username
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry the Bird		@twitter

สำหรับการใช้งาน table นั้น ทาง bootstrap ได้เตรียมรูปแบบให้เราอีกหลายแบบทั้งนี้ สามารถดูการใช้งานเพิ่มเติมได้ที่ <http://getbootstrap.com/docs/4.0/content/tables/>

## Image

สำหรับในส่วนของรูปภาพ ทาง bootstrap ได้เตรียมรูปแบบ ต่าง ๆ ให้เราใช้งาน ดังนี้

- .rounded คือการเพิ่มขอบมลให้กับรูปภาพ
- .rounded-circle กำหนดให้รูปภาพเป็นวงกลม
- .img-thumbnail คือการแสดงภาพตัวอย่าง (Thumbnail) แบบมีเส้นขอบรอบรูปภาพ
- .img-fluid คือ การกำหนดให้รูปภาพแสดงผลแบบ responsive

ตัวอย่างการใช้งาน แต่ละแบบ เพิ่ม class ที่เราต้องการให้กับ tag img ดังนี้

### Rounded เพิ่มขอบมลให้กับภาพ

```

```



### Rounded-circle กำหนดให้รูปภาพเป็นวงกลม

```

```



Img-thumbnail แสดงภาพตัวอย่าง (Thumbnail) แบบมีเส้นขอบ

```

```

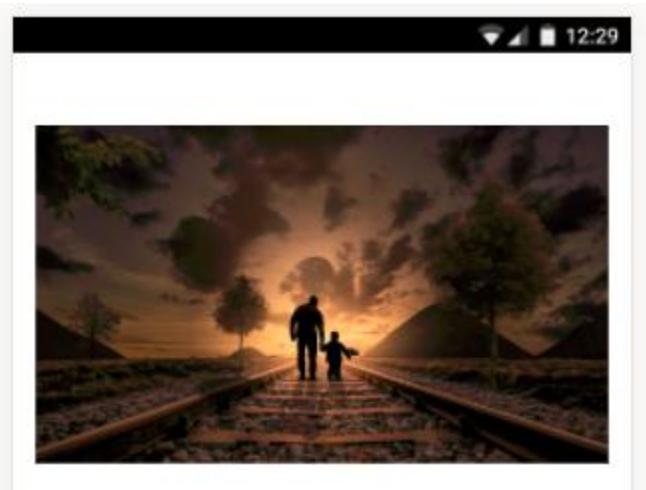


Img-fluid คือ การกำหนดให้รูปภาพแสดงผลแบบ responsive

```

```

ตัวอย่างการแสดงผล



สามารถดูตัวอย่างการแสดงผลรูปภาพเพิ่มเติมได้ที่ <https://getbootstrap.com/docs/4.0/content/images/>

# Alert

คือกล่องข้อความแจ้งเตือนต่าง ๆ ในระบบ เรียกใช้งานได้โดยใช้ class .alert alert-สีของกล่องข้อความ

```
<div class="alert alert-primary" role="alert">  
  This is a primary alert—check it out!  
</div>  
<div class="alert alert-secondary" role="alert">  
  This is a secondary alert—check it out!  
</div>  
<div class="alert alert-success" role="alert">  
  This is a success alert—check it out!  
</div>  
<div class="alert alert-danger" role="alert">  
  This is a danger alert—check it out!  
</div>  
<div class="alert alert-warning" role="alert">  
  This is a warning alert—check it out!  
</div>  
<div class="alert alert-info" role="alert">  
  This is a info alert—check it out!  
</div>  
<div class="alert alert-light" role="alert">  
  This is a light alert—check it out!  
</div>  
<div class="alert alert-dark" role="alert">  
  This is a dark alert—check it out!  
</div>
```

ตัวอย่างการแสดงผล



## Button

ในส่วนของปุ่ม นั้นทาง Bootstrap ได้เตรียมรูปแบบไว้ให้เราใช้ หลายแบบ ตัวอย่างการแสดงผล ดังนี้



ตัวอย่าง code การใช้งาน button เพิ่ม class btn btn-ชื่อปุ่มที่ต้องการ เช่น btn btn-primary เป็นต้น

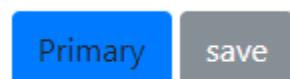
```
<button type="button" class="btn btn-primary">Primary</button>
<button type="button" class="btn btn-secondary">Secondary</button>
<button type="button" class="btn btn-success">Success</button>
<button type="button" class="btn btn-danger">Danger</button>
<button type="button" class="btn btn-warning">Warning</button>
<button type="button" class="btn btn-info">Info</button>
<button type="button" class="btn btn-light">Light</button>
<button type="button" class="btn btn-dark">Dark</button>

<button type="button" class="btn btn-link">Link</button>
```

ทั้งนี้ เราสามารถ เพิ่ม class btn btn-ชื่อปุ่มที่ต้องการ ให้กับ Tag อื่น ๆ ได้ เช่น a , input , button

```
<a class="btn btn-primary">Primary</a>
<input type="submit" value="save" class="btn btn-secondary">
```

ตัวอย่างการแสดงผล



## Button Outline

คือ การแสดงผล button แบบมีเฉพาะเส้นขอบ การใช้งานเพียงแค่ เพิ่ม outline ลงไปใน class เช่น

```
<button type="button" class="btn btn-outline-primary">Primary</button>
<button type="button" class="btn btn-outline-secondary">Secondary</button>
<button type="button" class="btn btn-outline-success">Success</button>
<button type="button" class="btn btn-outline-danger">Danger</button>
<button type="button" class="btn btn-outline-warning">Warning</button>
<button type="button" class="btn btn-outline-info">Info</button>
<button type="button" class="btn btn-outline-light">Light</button>
<button type="button" class="btn btn-outline-dark">Dark</button>
```

ตัวอย่างการแสดงผล



## Button Size

เราสามารถ ปรับ ขนาดของ button ได้ โดย เพิ่ม .btn-lg สำหรับขนาดใหญ่ หรือ .btn-sm สำหรับ ขนาดเล็ก

```
<button type="button" class="btn btn-primary btn-lg">Large button</button>
<button type="button" class="btn btn-primary btn-sm">Small button</button>
```

ตัวอย่างการแสดงผล



เราสามารถกำหนดให้ button ขยายความกว้างเต็มขนาด ได้โดยการเพิ่ม .btn-block

```
<button type="button" class="btn btn-primary btn-lg btn-block">Block level button</button>
```

Block level button

# Card

ในส่วนของ Card นั้น ได้ถูกออกแบบมา สำหรับจัดรูปแบบข้อมูลให้อยู่ในกรอบ ในลักษณะ คล้าย ๆ กับ Card หรือ บัตร ซึ่งสามารถช่วยให้เนื้อหาของเรามีความสวยงามมากยิ่งขึ้น

## Basic Card

```
<div class="card">
  <div class="card-body">
    This is some text within a card body.
  </div>
</div>
```

การแสดงผล จะเห็นว่ามีกรอบ包围ข้อความทำให้มีความสวยงามมากยิ่งขึ้น

This is some text within a card body.

## Header and footer

```
<div class="card">
  <div class="card-header">
    Featured
  </div>
  <div class="card-body">
    <h4 class="card-title">Special title treatment</h4>
    <p class="card-text">With supporting text below as a natural lead-in to additional content.</p>
    <a href="#" class="btn btn-primary">Go somewhere</a>
  </div>
</div>
```

ตัวอย่างการแสดงผล

Featured

Special title treatment

With supporting text below as a natural lead-in to additional content.

Go somewhere

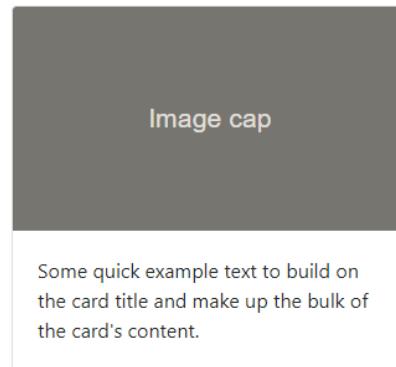
## Card Images

ใน card เราสามารถใส่รูปภาพแล้วมีข้อความประกอบด้านล่างได้ สามารถนำไปประยุกต์ใช้งานได้หลายแบบ เช่น แสดงข้อมูลสินค้า เป็นต้น

```
<div class="card" style="width: 20rem;">
  
  <div class="card-body">
    <p class="card-text">Some quick example text to build on the card title and make up the bulk of
    </div>
</div>
```

Copy

ตัวอย่างการแสดงผล



## List groups

```
<div class="card" style="width: 20rem;">
  <ul class="list-group list-group-flush">
    <li class="list-group-item">Cras justo odio</li>
    <li class="list-group-item">Dapibus ac facilisis in</li>
    <li class="list-group-item">Vestibulum at eros</li>
  </ul>
</div>
```

Copy

ตัวอย่างการแสดงผล สามารถแสดงข้อมูลในรูปแบบ List เรียงลงมาใน Card ได้



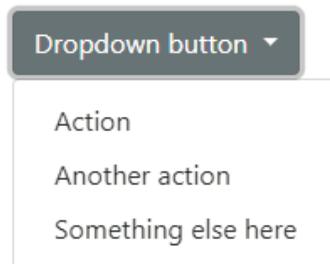
## Dropdown

คือ ปุ่มที่สามารถย่อ หรือ แสดงเมนูที่อยู่ด้านในได้ เนื่องจากในกรณีที่เรามีเมนูจำนวนมาก เมื่อใช้งาน Dropdown จะช่วยลดพื้นที่ของหน้าเว็บในการแสดงเมนูได้

ตัวอย่างการแสดงผล สามารถคลิกเพื่อแสดงหรือย่อเมนูได้

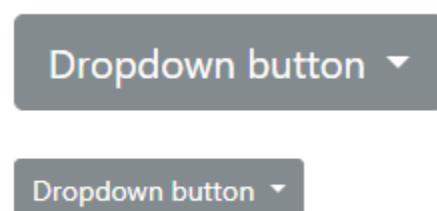
```
<div class="dropdown">
<button class="btn btn-secondary dropdown-toggle" type="button" id="dropdownMenuButton" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
  Dropdown button
</button>
<div class="dropdown-menu" aria-labelledby="dropdownMenuButton">
  <a class="dropdown-item" href="#">Action</a>
  <a class="dropdown-item" href="#">Another action</a>
  <a class="dropdown-item" href="#">Something else here</a>
</div>
</div>
```

ตัวอย่างการแสดงผล



## Dropdowns Sizing

เราสามารถขยายหรือเพิ่มขนาดของ Dropdowns ได้ โดยการเพิ่ม class `btn-lg` (ขนาดใหญ่) หรือ `btn-sm` (ขนาดเล็ก)



# Forms

สำหรับ Forms คือส่วนที่ใช้สำหรับการรับข้อมูลจากผู้ใช้ เช่น ช่องกรอกข้อความ ต่างๆ เป็นต้น ใน Bootstrap ได้เตรียมรูปแบบ Forms ให้เจ้าใช้งานดังนี้

## Form-Group และ Form controls

เริ่มต้นการใช้งาน form เพิ่ม class form-group และ class form-control สำหรับการ Code ดังนี้

```
<form>
  <div class="form-group">
    <label for="exampleFormControlInput1">Email address</label>
    <input type="email" class="form-control" id="exampleFormControlInput1" placeholder="name@example.com">
  </div>
  <div class="form-group">
    <label for="exampleFormControlSelect1">Example select</label>
    <select class="form-control" id="exampleFormControlSelect1">
      <option>1</option>
      <option>2</option>
      <option>3</option>
      <option>4</option>
      <option>5</option>
    </select>
  </div>
</form>
```

ตัวอย่างการแสดงผล

Email address

name@example.com

Example select

1

## Form Sizing

เราสามารถปรับขนาดของ input โดยใช้ class .form-control-lg (ขนาดใหญ่) และ class .form-control-sm (ขนาดเล็ก)

```
<form>
  <div class="form-group">
    <input class="form-control form-control-lg" type="text" placeholder=".form-control-lg">
  </div>
  <div class="form-group">
    <input class="form-control" type="text" placeholder="Default input">
  </div>
  <div class="form-group">
    <input class="form-control form-control-sm" type="text" placeholder=".form-control-sm">
  </div>
</form>
```

ตัวอย่างการแสดงผล

.form-control-lg

Default input

.form-control-sm

## Read only

ในบางครั้งเราต้องการที่จะแสดงข้อมูลในช่อง input แต่ไม่ต้องการให้ผู้ใช้แก้ไขข้อมูลในช่องได้ เราสามารถกำหนดให้ช่อง input อ่านได้อย่างเดียวเท่านั้น โดยใส่ attribute readonly

```
<input class="form-control" type="text" placeholder=" Readonly input here..." readonly>
```

ตัวอย่างการแสดงผล จะเห็นว่าในช่อง input จะเป็นสีเทา และจะไม่สามารถแก้ไขข้อมูลได้

Readonly input here...

## Read only plain text

ในบางครั้งหากต้องการแสดงข้อมูล text ใน Form เราสามารถกำหนด class .form-control-plaintext เพื่อแสดงข้อความ

```
<form>
  <div class="form-group row">
    <label for="staticEmail" class="col-sm-2 col-form-label">Email</label>
    <div class="col-sm-10">
      <input type="text" readonly class="form-control-plaintext" id="staticEmail" value="email@example.com">
    </div>
  </div>
  <div class="form-group row">
    <label for="inputPassword" class="col-sm-2 col-form-label">Password</label>
    <div class="col-sm-10">
      <input type="password" class="form-control" id="inputPassword" placeholder="Password">
    </div>
  </div>
</form>
```

ตัวอย่างการแสดงผล

Email email@example.com

Password

Password

## Form Grid

สำหรับ Form ใน Bootstrap เราสามารถแบ่งการแสดงผล Form โดยใช้ grid ได้ดังนี้

```
<form>
  <div class="row">
    <div class="col">
      <input type="text" class="form-control" placeholder="First name">
    </div>
    <div class="col">
      <input type="text" class="form-control" placeholder="Last name">
    </div>
  </div>
</form>
```

ตัวอย่างการแสดงผล

First name

Last name

## Horizontal form

ในส่วนของ horizontal form ในส่วนนี้จะมีการเพิ่ม class col-form-label สำหรับแสดงข้อความ และ มีการใช้ Grid ในการแบ่ง Columns ของ Form

```
<form>
  <div class="form-group row">
    <label for="inputEmail3" class="col-sm-2 col-form-label">Email</label>
    <div class="col-sm-10">
      <input type="email" class="form-control" id="inputEmail3" placeholder="Email">
    </div>
  </div>
  <div class="form-group row">
    <label for="inputPassword3" class="col-sm-2 col-form-label">Password</label>
    <div class="col-sm-10">
      <input type="password" class="form-control" id="inputPassword3" placeholder="Password">
    </div>
  </div>
  <div class="form-group row">
    <div class="col-sm-10">
      <button type="submit" class="btn btn-primary">Sign in</button>
    </div>
  </div>
</form>
```

ตัวอย่างการแสดงผล

Email	<input type="text" value="Email"/>
Password	<input type="password" value="Password"/>
<button style="background-color: blue; color: white;" type="button">Sign in</button>	

## Inline forms

คือ ฟอร์ม ที่มี input เรียงต่อกันไปบนเดียวกัน โดยเราจะเพิ่ม class .form-inline ใน form

```
<form class="form-inline">
  <label class="sr-only" for="inlineFormInputName2">Name</label>
  <input type="text" class="form-control mb-2 mr-sm-2 mb-sm-0" id="inlineFormInputName2" placeholder="Jane Doe">

  <label class="sr-only" for="inlineFormInputGroupUsername2">Username</label>
  <div class="input-group mb-2 mr-sm-2 mb-sm-0">
    <div class="input-group-addon">@</div>
    <input type="text" class="form-control" id="inlineFormInputGroupUsername2" placeholder="Username">
  </div>

  <div class="form-check mb-2 mr-sm-2 mb-sm-0">
    <label class="form-check-label">
      <input class="form-check-input" type="checkbox"> Remember me
    </label>
  </div>

  <button type="submit" class="btn btn-primary">Submit</button>
</form>
```

ตัวอย่างการแสดงผล

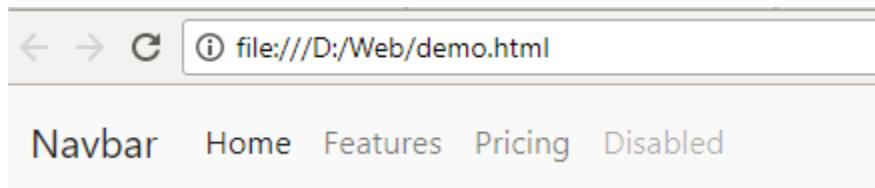
The screenshot shows a horizontal form layout. On the left, there is a text input field containing "Jane Doe". Next to it is a label "@". To the right of the label is another text input field containing "Username". Below the first two fields is a checkbox labeled "Remember me" which is checked. To the right of the checkbox is a blue "Submit" button.

# NavBar

Navbar โดยปกติจะอยู่ที่ด้านบนของเว็บ มีไว้สำหรับแสดง Logo / ชื่อ เว็บ, เมนู หรือ icon ต่าง ๆ เพื่อเป็น Link เข้าสู่ส่วนต่าง ๆ ในเว็บของเรา

```
<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <a class="navbar-brand" href="#">Navbar</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="navbarNav">
    <ul class="navbar-nav">
      <li class="nav-item active">
        <a class="nav-link" href="#">Home <span class="sr-only">(current)</span></a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Features</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Pricing</a>
      </li>
      <li class="nav-item">
        <a class="nav-link disabled" href="#">Disabled</a>
      </li>
    </ul>
  </div>
</nav>
```

ตัวอย่างการแสดงผล



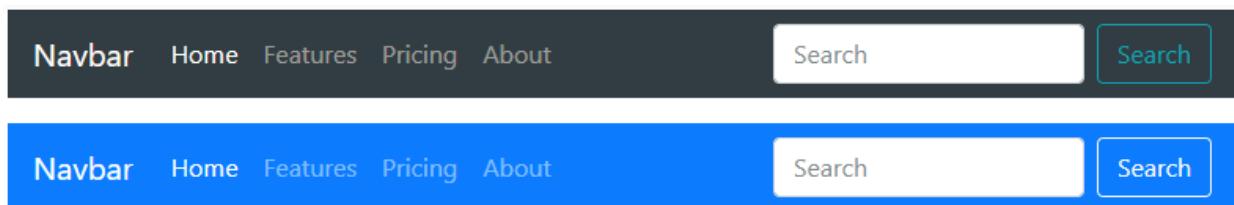
## Navbar Color

เราสามารถเปลี่ยนสีของ Navbar ได้โดยเพิ่ม class .bg-dark หรือ .bg-primary

```
<nav class="navbar navbar-dark bg-dark">
  <!-- Navbar content -->
</nav>

<nav class="navbar navbar-dark bg-primary">
  <!-- Navbar content -->
</nav>
```

ตัวอย่างการแสดงผล



## Navbar Placement

เราสามารถกำหนดตำแหน่งการวาง Navbar ให้อยู่ด้านบน (top) หรือ ด้านล่าง (bottom) ของจอ ดังนี้

Fixed top กำหนดให้ Navbar ยึดติดอยู่กับด้านบนสุดของหน้าจอตลอดเวลา

ตัวอย่างการเรียกใช้งาน

```
<nav class="navbar fixed-top navbar-light bg-light">
```

Fixed bottom กำหนดให้ Navbar ติดอยู่ด้านล่างสุดของจอ

ตัวอย่างการเรียกใช้งาน

```
<nav class="navbar fixed-bottom navbar-light bg-light">
```

## Popovers

สำหรับ Popovers เจ้าสามารถแสดงข้อความแบบ show/hide ให้กับ Button และสามารถกำหนดได้ว่าจะให้ข้อความอยู่ด้านบน ล่าง ซ้าย หรือ ขวา ของ Button ได้ โดยจะทำงานก็ต่อเมื่อคลิกที่ Button ก่อนการใช้งานเราต้องมีการเรียกใช้งาน Javascript Jquery กันก่อน เพื่อให้ popover ทำงาน

```
<script>
  $(function () {
    $('[data-toggle="popover"]').popover()
  });
</script>
```

### ตัวอย่างการเรียกใช้งาน Popovers

```
<button type="button" class="btn btn-secondary" data-container="body" data-toggle="popover"
data-placement="top" data-content="Vivamus sagittis lacus vel augue laoreet rutrum faucibus.">
  Popover on top
</button>

<button type="button" class="btn btn-secondary" data-container="body" data-toggle="popover"
data-placement="right" data-content="Vivamus sagittis lacus vel augue laoreet rutrum faucibus.">
  Popover on right
</button>

<button type="button" class="btn btn-secondary" data-container="body" data-toggle="popover"
data-placement="bottom" data-content="Vivamus sagittis lacus vel augue laoreet rutrum faucibus.">
  Popover on bottom
</button>

<button type="button" class="btn btn-secondary" data-container="body" data-toggle="popover"
data-placement="left" data-content="Vivamus sagittis lacus vel augue laoreet rutrum faucibus.">
  Popover on left
</button>
```

### ตัวอย่างการแสดงผล



# Tooltips

เราสามารถแสดงข้อความแบบ Tooltips ให้กับ button ได้ โดยจะทำงานเมื่อนำมาส์ไปวางที่ button หรือ Tag Html ที่เรากำหนดค่า tooltip ได้ หมายความว่าการแสดงข้อความแนะนำการใช้งานปุ่มต่าง ๆ เป็นต้น

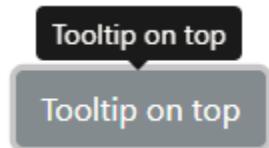
ก่อนการใช้งานเราต้องมีการเรียกใช้งาน JQuery เพื่อให้ tooltip ทำงานดังนี้

```
<script>
$(function () {
    $('[data-toggle="tooltip"]').tooltip()
});
</script>
```

ตัวอย่างการเรียกใช้งาน

```
<button type="button" class="btn btn-secondary" data-toggle="tooltip" data-placement="top" title="Tooltip on top">
    Tooltip on top
</button>
<button type="button" class="btn btn-secondary" data-toggle="tooltip" data-placement="right" title="Tooltip on right">
    Tooltip on right
</button>
<button type="button" class="btn btn-secondary" data-toggle="tooltip" data-placement="bottom" title="Tooltip on bottom">
    Tooltip on bottom
</button>
<button type="button" class="btn btn-secondary" data-toggle="tooltip" data-placement="left" title="Tooltip on left">
    Tooltip on left
</button>
```

ตัวอย่างการแสดงผล เมื่อนำมาส์ไปวางบน Button จะมี Tooltip แสดงข้อความขึ้นมาทันที



เราสามารถกำหนดให้ Tooltip แสดงข้อมูลตรงตำแหน่ง ด้านบน ล่าง ซ้าย หรือ ขวา โดยกำหนดให้ที่ data-placement

ตัวอย่าง กำหนดให้ Tooltip แสดงด้านซ้ายของ Button

```
<button type="button" class="btn btn-secondary" data-toggle="tooltip" data-placement="left" title="Tooltip on left">
    Tooltip on left
</button>
```

ตัวอย่างการแสดงผล



# Modal

สำหรับ Modal เราสามารถแสดงข้อความ เป็นกล่องข้อความ เดิมชื่นมากอยู่หน้าเว็บของเรา ซึ่งเป็นที่นิยมในการแสดงข้อความหรือ ประยุกต์ใช้งานได้อีกหลากหลายรูปแบบ

ตัวอย่างการเรียกใช้งาน

สร้าง Button สำหรับเรียกใช้งาน modal

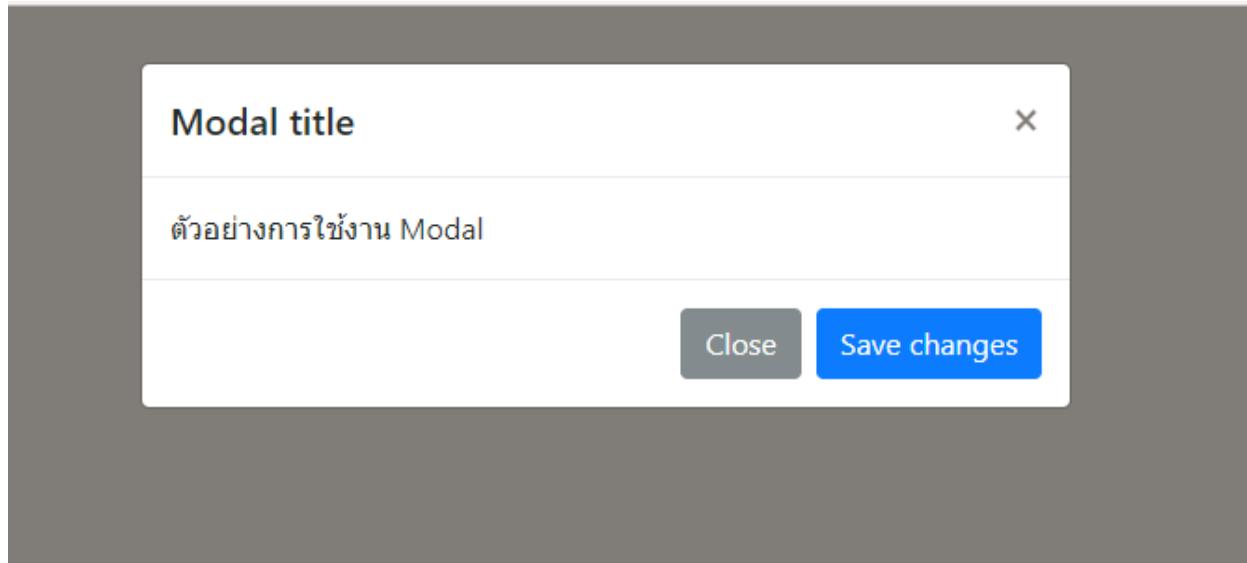
```
<!-- Button trigger modal -->
<button type="button" class="btn btn-primary" data-toggle="modal" data-target="#exampleModal">
  Launch demo modal
</button>
```

สร้าง class .modal สำหรับแสดงข้อความใน modal

โดยเราสามารถ ใส่ข้อความ ได้ภายในส่วนของ class .modal-body

```
<!-- Modal -->
<div class="modal fade" id="exampleModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title" id="exampleModalLabel">Modal title</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <div class="modal-body">
        ...
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-secondary" data-dismiss="modal">Close</button>
        <button type="button" class="btn btn-primary">Save changes</button>
      </div>
    </div>
  </div>
</div>
```

## ตัวอย่างการแสดงผล



## ส่งท้าย

สำหรับใน เนื้อหาที่แนะนำทั้งหมดใน เอกสารชุดนี้เป็นเพียงแค่การแนะนำบางส่วนของทั้งหมดซึ่งสามารถดูตัวอย่างและวิธีการใช้งานเพิ่มเติมจาก <https://getbootstrap.com/> ซึ่งผู้เขียนได้เขียนเอกสารนี้ขึ้น เพื่อเป็นแนวทางในการเริ่มต้นสำหรับผู้เริ่มต้น หรือ ยังไม่เคยใช้งานเท่านั้น ผู้เขียนหวังเป็นอย่างยิ่งว่าข้อมูลใน เอกสารชุดนี้จะมีประโยชน์สำหรับผู้อ่าน หากมีข้อผิดพลาดประการใดทางผู้เขียนต้องขออภัยมา ณ ที่นี่ ด้วยครับ

By Monkeywebstudio.com