



Web Application Development

บรรยายโดย นายทวิวุฒิ นากอหม๊ะ (ท็อป)

นักวิชาการคอมพิวเตอร์ วิทยาลัยอิสลามศึกษา มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตปัตตานี /

Managing Director iTopCyberSoft / Full-Stack Developer & Lecturer

Site: <https://nakomah.com> หรือ E-mail: nakomah.web@gmail.com

กำหนดการอบรม Web Application Development
ระหว่างวันที่ 12-13 พฤศจิกายน 2561
ณ ห้องปฏิบัติการคอมพิวเตอร์ 202 อาคารศูนย์ภาษาและคอมพิวเตอร์ มหาวิทยาลัยราชภัฏสงขลา

วันที่ 12 พฤศจิกายน 2561

- รู้จักภาษา PHP และ MySQL เบื้องต้น
- จำลองเซิร์ฟเวอร์ด้วย Xampp และสร้างฐานข้อมูล
- การเขียนโปรแกรมเชื่อมต่อฐานข้อมูลโดยใช้ PHP DataObject (PDO)
 - <https://nakomah.com/article/view/php-dataobject-syntax>
- การใช้งาน PHP Function และการสร้างฟังก์ชันขึ้นมาใช้เอง
 - Datetime(), Array(), Explode(), Implode(), NumberFormat() เป็นต้น
- เรียนรู้การปรับแต่งหน้าเว็บไซต์ด้วย CSS Framework เช่น
 - Bootstrap Framework : <https://getbootstrap.com/>
 - Bulma framework : <https://bulma.io/>
 - Foundation framework : <https://foundation.zurb.com/>
- ติดตั้ง Node.js, npm
 - <https://nodejs.org/dist/v8.12.0/node-v8.12.0-x64.msi>
- ใช้ PDO ในการบันทึกข้อมูล แสดงข้อมูลและลบข้อมูล
- การอัปโหลดรูปภาพ/ไฟล์พร้อมบันทึกค่าลงฐานข้อมูล

วันที่ 13 พฤศจิกายน 2561

- แนะนำการใช้งาน Backend Template
 - <https://github.com/pratikborsadiya/vali-admin>
- การจัดโครงสร้างไฟล์โปรเจค
- สร้างระบบ Login, Logout, และการกำหนดสิทธิ์การใช้งานหน้าเว็บ
<https://nakomah.com/article/view/login-amp-logout>

รู้จักภาษา PHPและMySQL เบื้องต้น

พีเอชพี (PHP) คือ ภาษาคอมพิวเตอร์ในลักษณะเซิร์ฟเวอร์-ไซด์ สคริปต์ โดยลิขสิทธิ์อยู่ในลักษณะโอเพนซอร์ส ภาษาพีเอชพีใช้สำหรับจัดทำเว็บไซต์ และแสดงผลออกมาในรูปแบบ HTML โดยมีรากฐานโครงสร้างคำสั่งมาจากภาษา ภาษาซี ภาษาจาวา และ ภาษาเพิร์ล ซึ่ง ภาษาพีเอชพี นั้นง่ายต่อการเรียนรู้ ซึ่งเป้าหมายหลักของภาษานี้ คือให้นักพัฒนาเว็บไซต์สามารถเขียน เว็บเพจ ที่มีความตอบโต้ได้อย่างรวดเร็ว

รายการระบบฐานข้อมูลที่ PHP สามารถเชื่อมต่อได้คือ Oracle, Sybase, mSQL, **MySQL**, SOLID, ODBC, PostgreSQL, Adabas D, FilePro, Velocis, Informix, dbase, Unix dbm

MySQL คือ โปรแกรมระบบจัดการฐานข้อมูล ที่พัฒนาโดยบริษัท MySQL AB มีหน้าที่เก็บข้อมูลอย่างเป็นระบบ รองรับคำสั่ง SQL เป็นเครื่องมือสำหรับเก็บข้อมูล ที่ต้องใช้ร่วมกับเครื่องมือหรือโปรแกรมอื่นอย่างบูรณาการ เพื่อให้ได้ระบบงานที่รองรับ ความต้องการของผู้ใช้ เช่นทำงานร่วมกับเครื่องบริการเว็บ (Web Server) เพื่อให้บริการแก่ภาษาสคริปต์ที่ทำงานฝั่งเครื่องบริการ (Server-Side Script) เช่น ภาษา php ภาษา asp.net หรือภาษาเจเอสพี (PHP)

โดยสรุปความสามารถได้ดังนี้

- MySQL เป็นฟรีแวร์ทางด้านฐานข้อมูลที่มีประสิทธิภาพสูง
- นักพัฒนาฐานข้อมูลที่เคยใช้ MySQL ต่างยอมรับในความรวดเร็ว การรองรับจำนวนผู้ใช้ และขนาดของข้อมูลจำนวนมาก
- สนับสนุนการใช้งานบนระบบปฏิบัติการมากมาย เช่น UNIX OS/2 MAC OS Windows
- สามารถใช้งานร่วมกับ Web Development platform เช่น C, C++ , Java, Perl, PHP, Python, TCL, หรือ ASP
- ได้รับความนิยมอย่างมากในปัจจุบัน และมีแนวโน้มสูงขึ้นเรื่อย ๆ ในอนาคต

คำสั่ง PHP + Mysql ที่ใช้บ่อย

+ Options			id	name ชื่อสมาชิก	surname	nickname	tel	created วันที่สร้าง
<input type="checkbox"/>	Edit	Copy	Delete	1	taveevut	nakomah	top	0862887987 2016-10-09 06:44:23
<input type="checkbox"/>	Edit	Copy	Delete	2	sakarin	dalee	sak	0869512067 2016-10-09 06:44:23
<input type="checkbox"/>	Edit	Copy	Delete	3	somchai	soujai	chai	0901658478 2016-10-09 06:44:23

1.การเรียกข้อมูลจากฐานข้อมูลมาแสดงผล

รูปแบบคำสั่ง : SELECT * FROM ชื่อตาราง

ตัวอย่าง :

```
1. <table width="500" border="1">
2.   <tr>
3.     <th>ชื่อ</th>
4.     <th>นามสกุล</th>
5.     <th>ชื่อเล่น</th>
6.     <th>เบอร์โทรศัพท์</th>
7.   </tr>
8.   <?php
9.     $q = mysql_query("SELECT * FROM member ");
10.    while ($row = mysql_fetch_assoc($q)) {
11.      ?>
12.      <tr>
13.        <td>
14.          <?php echo $row["name"]; ?>
15.        </td>
16.        <td>
17.          <?php echo $row["surname"]; ?>
18.        </td>
19.        <td>
20.          <?php echo $row["nickname"]; ?>
21.        </td>
22.        <td>
23.          <?php echo $row["tel"]; ?>
24.        </td>
25.      </tr>
26.    <?php } ?>
27. </table>
```

ผลลัพธ์

ชื่อ	นามสกุล	ชื่อเล่น	เบอร์โทรศัพท์
taveevut	nakomah	top	0862887987
sakarin	dalee	sak	0869512067
somchai	soujai	chai	0901658478

2.การเรียกข้อมูลจากฐานข้อมูลมาแสดงผลแบบมีเงื่อนไข

รูปแบบคำสั่ง : SELECT * FROM ชื่อตาราง WHERE เงื่อนไขอย่างไร

ตัวอย่าง :

```
1. <table width="500" border="1">
2.   <tr>
3.     <th>ชื่อ</th>
4.     <th>นามสกุล</th>
5.     <th>ชื่อเล่น</th>
6.     <th>เบอร์โทรศัพท์</th>
7.   </tr>
8.   <?php
9.     $q = mysql_query("SELECT * FROM member WHERE id = 1 ");
10.    while ($row = mysql_fetch_assoc($q)) {
11.      ?>
12.      <tr>
13.        <td>
14.          <?php echo $row["name"]; ?>
15.        </td>
16.        <td>
17.          <?php echo $row["surname"]; ?>
18.        </td>
19.        <td>
20.          <?php echo $row["nickname"]; ?>
21.        </td>
22.        <td>
23.          <?php echo $row["tel"]; ?>
24.        </td>
25.      </tr>
26.    <?php }?>
27. </table>
```

ผลลัพธ์

ชื่อ	นามสกุล	ชื่อเล่น	เบอร์โทรศัพท์
taveevut	nakomah	top	0862887987

3.การเปลี่ยนแปลงค่าในฐานข้อมูล หรือ UPDATE

รูปแบบคำสั่ง : UPDATE ชื่อตาราง SET ชื่อฟิลด์1 = 'ข้อความ1',ชื่อฟิลด์2 = 'ข้อความ2',...
WHERE เงื่อนไขอย่างไร

ตัวอย่าง :

```
1. <?php
2. mysql_query("UPDATE member SET name = '".$_POST['name']."' ,surname = '".$_POST['surname']."' ,tel = '".$_POST['tel']."' ");
3. ?>
```

4.การลบค่าในฐานข้อมูล หรือ DELETE

รูปแบบคำสั่ง : DELET FROM ชื่อตาราง WHERE เงื่อนไขอย่างไร

ตัวอย่าง

```
1. <?php
2. mysql_query("DELETE FROM member WHERE id = '".$_GET['id']."' ");
3. ?>
```

ชนิดข้อมูลใน MySQL (Datatype) ที่ใช้บ่อย

VARCHAR : สำหรับเก็บข้อมูลประเภทตัวอักษร ทุกครั้งที่เลือกชนิดของฟิลด์เป็นประเภทนี้ จะต้องมีการกำหนดความยาวของข้อมูลลงไปด้วย ซึ่งสามารถกำหนดค่าได้ตั้งแต่ 1 - 255 ฟิลด์ชนิดนี้ เหมาะสำหรับการเก็บข้อมูลสั้น ๆ เช่น ชื่อ นามสกุล หรือหัวข้อต่าง ๆ เป็นต้น

CHAR : สำหรับเก็บข้อมูลประเภทตัวอักษร แบบที่ถูกจำกัดความกว้างเอาไว้คือ 255 ตัวอักษร ไม่สามารถปรับเปลี่ยนได้เหมือนกับ VARCHAR หากทำการสืบค้นโดยเรียงตามลำดับ ก็จะเรียงข้อมูลแบบ case-sensitive เว้นแต่จะกำหนดแอตทริบิวต์เป็น BINARY ที่จะทำให้การเรียงข้อมูลเป็นแบบ non case-sensitive เช่นเดียวกับ VARCHAR

TINYTEXT : ในกรณีที่ข้อความยาว ๆ หรือต้องการที่จะค้นหาข้อความ โดยอาศัยฟิเจอร์ FULL TEXT SEARCH ของ MySQL เราอาจจะเลือกที่จะไม่เก็บข้อมูลลงในฟิลด์ประเภท VARCHAR ที่มีข้อจำกัดแค่ 255 ตัวอักษร แต่เราจะเก็บลงฟิลด์ประเภท TEXT แทน โดย TINYTEXT นี้ จะสามารถเก็บข้อมูลได้ 255 ตัวอักษร ซึ่งมองเผิน ๆ ก็ไม่ต่างกับเก็บลงฟิลด์ประเภท CHAR หรือ VARCHAR(255) เลย แต่จริงๆ มันต่างกันตรงที่ มันทำ FULL TEXT SEARCH ได้

TEXT : สำหรับเก็บข้อมูลประเภทตัวอักษร เช่นเดียวกับ TINYTEXT แต่สามารถเก็บได้มากขึ้น โดยสูงสุดคือ 65,535 ตัวอักษร หรือ 64KB เหมาะสำหรับเก็บข้อมูลพวกเนื้อหาต่างๆ ที่ยาว ๆ

MEDIUMTEXT : เก็บข้อมูลประเภทตัวอักษร เช่นเดียวกับ TINYTEXT แต่เก็บข้อมูลได้ 16,777,215 ตัวอักษร

LONGTEXT : เก็บข้อมูลประเภทตัวอักษร เช่นเดียวกับ TINYTEXT แต่เก็บข้อมูลได้ 4,294,967,295 ตัวอักษร

TINYINT : สำหรับเก็บข้อมูลชนิดตัวเลขที่มีขนาด 8 บิต ข้อมูลประเภทนี้เราสามารถกำหนดเพิ่มเติมในส่วน of "แอตทริบิวต์" ได้ว่าจะเลือกเป็น UNSIGNED หรือ UNSIGNED ZEROFILL โดยจะมี ความแตกต่างดังนี้

- UNSIGNED : จะหมายถึงเก็บค่าตัวเลขแบบไม่มีเครื่องหมาย แบบนี้จะทำให้สามารถเก็บค่าได้ตั้งแต่ 0 - 255

- UNSIGNED ZEROFILL : เหมือนข้างต้น แต่ว่าหากข้อมูลที่กรอกเข้ามาไม่ครบตามจำนวน หลักที่เรากำหนด ตัว MySQL จะทำการเติม 0 ให้ครบหลักเอง เช่น ถ้ากำหนดให้ใส่ได้ 3 หลัก แล้วทำการเก็บข้อมูล 25 เข้าไป เวลาที่สืบค้นดู เราจะได้ค่าออกมาเป็น 025 หากไม่เลือก "แอ ตทริบิวต์" สิ่งที่เราจะได้ก็คือ SIGNED นั่นก็คือต้องเสียบิตหนึ่งไปเก็บเครื่องหมาย บวก/ลบ ทำให้สามารถเก็บข้อมูลได้อยู่ในช่วง -128 ถึง 127 เท่านั้น

SMALLINT : สำหรับเก็บข้อมูลประเภทตัวเลขที่มีขนาด 16 บิต จึงสามารถเก็บค่าได้ตั้งแต่ -32768 ถึง 32767 (ในกรณีแบบคิดเครื่องหมาย) หรือ 0 ถึง 65535 (ในกรณี UNSIGNED หรือไม่คิดเครื่องหมาย) ซึ่งสามารถเลือก Attribute เป็น UNSIGNED และ UNSIGNED ZEROFILL ได้เช่นเดียวกับ TINYINT

MEDIUMINT : สำหรับเก็บข้อมูลประเภทตัวเลขที่มีขนาด 24 บิต นั่นก็หมายความว่าสามารถเก็บข้อมูลตัวเลขได้ตั้งแต่ -8388608 ไปจนถึง 8388607 (ในกรณีแบบคิดเครื่องหมาย) หรือ 0 ถึง 16777215

(ในกรณีที่เป็น UNSIGNED หรือไม่คิดเครื่องหมาย) ซึ่งสามารถเลือก Attribute เป็น UNSIGNED และ UNSIGNED ZEROFILL ได้เช่นเดียวกับ TINYINT

INT : สำหรับเก็บข้อมูลประเภทตัวเลขที่มีขนาด 32 บิต หรือสามารถเก็บข้อมูลได้ตั้งแต่ -2147483648 ไปจนถึง 2147483647 ครับ (ในกรณีแบบคิดเครื่องหมาย) หรือ 0 ถึง 4294967295 (ในกรณีที่เป็น

UNSIGNED หรือไม่คิดเครื่องหมาย) ซึ่งสามารถเลือก Attribute เป็น UNSIGNED และ UNSIGNED ZEROFILL ได้เช่นเดียวกับ TINYINT

FLOAT[(M,D)] : ที่กล่าวถึงไปทั้งหมด ในตระกูล INT นั้นจะเป็นเลขจำนวนเต็ม หากเราบันทึกข้อมูลที่มีเศษทศนิยม มันจะถูกปัดท้นที่ ดังนั้นหากต้องการจะเก็บค่าที่เป็นเลขทศนิยม ต้องเลือกชนิดของฟิลด์ เป็น FLOAT โดยจะเก็บข้อมูลแบบ 32 บิต คือมีค่าตั้งแต่ -3.402823466E+38 ไปจนถึง -1.175494351E-38, 0 และ 1.175494351E-38 ถึง 3.402823466E+38

DOUBLE[(M,D)] : สำหรับเก็บข้อมูลประเภทตัวเลขทศนิยม เช่นเดียวกับ FLOAT แต่มีขนาดเป็น 64 บิต สามารถเก็บได้ตั้งแต่ -1.7976931348623157E+308 ถึง -2.2250738585072014E-308, 0 และ 2.2250738585072014E-308 ถึง 1.7976931348623157E+308

DECIMAL[(M,D)] : สำหรับเก็บข้อมูลประเภทตัวเลขทศนิยม เช่นเดียวกับ FLOAT แต่ใช้กับข้อมูลที่ต้องการความละเอียดและถูกต้องของข้อมูลสูง ข้อสังเกต เกี่ยวกับข้อมูลประเภท FLOAT, DOUBLE และ DECIMAL ก็คือ เวลากำหนดความ ยาวของข้อมูลในฟิลด์ จะถูกกำหนดอยู่ในรูปแบบ (M,D) ซึ่งหมายความว่า ต้องมีการระบุว่าจะให้มี ตัวเลขส่วนที่เป็นจำนวนเต็มกี่หลัก และมีเลขทศนิยมกี่หลัก เช่น ถ้าเรากำหนดว่า FLOAT(5,2) จะ หมายความว่า เราจะเก็บข้อมูลเป็นตัวเลขจำนวนเต็ม 5 หลัก และทศนิยม 2 หลัก ดังนั้น หากทำการใส่ ข้อมูล 12345.6789 เข้าไป สิ่งที่จะเข้าไปอยู่ในข้อมูลจริงๆ ก็คือ 12345.68 (พิเศษให้มีจำนวนหลัก ตามที่กำหนดไว้)

DATE : สำหรับเก็บข้อมูลประเภทวันที่ โดยเก็บได้จาก 1 มกราคม ค.ศ. 1000 ถึง 31 ธันวาคม ค.ศ. 9999 โดยจะแสดงผลในรูปแบบ YYYY-MM-DD

DATETIME : สำหรับเก็บข้อมูลประเภทวันที่ และเวลา โดยจะเก็บได้ตั้งแต่ 1 มกราคม ค.ศ. 1000 เวลา 00:00:00 ไปจนถึง 31 ธันวาคม ค.ศ. 9999 เวลา 23:59:59 โดยรูปแบบการแสดงผล เวลาที่ทำการสืบค้น

(query) ออกมา จะเป็น YYYY-MM-DD HH:MM:SS

TIMESTAMP[(M)] : สำหรับเก็บข้อมูลประเภทวันที่ และเวลาเช่นกัน แต่จะเก็บในรูปแบบของ YYYYMMDDHHMMSS หรือ YMMDDHHMMSS หรือ YYYYMMDD หรือ YYMMDD แล้วแต่ว่าจะระบุค่า M เป็น 14, 12, 8 หรือ 6 ตามลำดับ สามารถเก็บได้ตั้งแต่วันที่ 1 มกราคม ค.ศ. 1000 ไป จนถึงประมาณปี ค.ศ. 2037

TIME : สำหรับเก็บข้อมูลประเภทเวลา มีค่าได้ตั้งแต่ -838:59:59 ไปจนถึง 838:59:59 โดยจะแสดงผล ออกมาในรูปแบบ HH:MM:SS YEAR[(2/4)] : สำหรับเก็บข้อมูลประเภทปี ในรูปแบบ YYYY หรือ YY แล้วแต่ว่าจะเลือก 2 หรือ 4 (หากไม่ระบุ จะถือว่าเป็น 4 หลัก) โดยหากเลือกเป็น 4 หลัก จะเก็บค่าได้ตั้งแต่ ค.ศ. 1901 ถึง 2155 แต่ หากเป็น 2 หลัก จะเก็บตั้งแต่ ค.ศ. 1970 ถึง 2069 ข้อสังเกต ค่าที่เก็บใน

ข้อมูลประเภท TIMESTAMP และ YEAR นั้นจะมีความสามารถพอ ๆ กับ การเก็บข้อมูลวันเดือนปี และเวลา ด้วยฟิลด์ชนิด VARCHAR แต่ต่างกันตรงที่ จะใช้เนื้อที่เก็บข้อมูล น้อยกว่า

TINYBLOB : สำหรับเก็บข้อมูลประเภทไบนารี ได้แก่ ไฟล์ข้อมูลต่างๆ, ไฟล์รูปภาพ, ไฟล์มัลติมีเดีย เป็นต้น คือไฟล์อะไรก็ตามที่อัปโหลดผ่านฟอร์มอัปโหลดไฟล์ในภาษา HTML โดย TINYBLOB นั้น จะมีเนื้อที่ให้เก็บข้อมูลได้ 256 ไบต์

BLOB : สำหรับเก็บข้อมูลประเภทไบนารี เช่นเดียวกับ TINYBLOB แต่สามารถเก็บข้อมูลได้ 64KB

MEDIUMBLOB : สำหรับเก็บข้อมูลประเภทไบนารี เช่นเดียวกับ TINYBLOB แต่เก็บข้อมูลได้ 16MB

LOB : สำหรับเก็บข้อมูลประเภทไบนารี เช่นเดียวกับ TINYBLOB แต่เก็บข้อมูลได้ 4GB

ข้อสังเกต ข้อมูลประเภท BLOB นั้น แม้จะมีประโยชน์ในเรื่องของการเก็บข้อมูลประเภท BINARY ให้อยู่กับตัวฐานข้อมูล ทำให้สะดวกเวลาสืบค้นก็ตาม แต่มันก็ทำให้ฐานข้อมูลมีขนาดใหญ่ เกินความจำเป็นด้วย ทำให้เกิดความไม่สะดวกในการสำรองฐานข้อมูลในกรณีที่ มีข้อมูลอัปโหลดไป เก็บมาก ๆ โดยปกติแล้ว จะใช้วิธีการอัปโหลดไปเก็บไว้ในโฟลเดอร์ แล้วเก็บลิงค์ไปยังไฟล์เหล่านั้น เป็นฟิลด์ชนิด VARCHAR มากกว่า

SET : สำหรับเก็บข้อมูลที่เป็นกลุ่มของข้อมูลที่ยอมให้เลือกได้ 1 ค่าหรือหลาย ๆ ค่า ซึ่งสามารถกำหนด ได้ถึง 64 ค่า

Bootstrap Framework

Bootstrap คืออะไร

Bootstrap คือชุดคำสั่งที่ประกอบด้วยภาษา CSS, HTML และ JavaScript เป็นชุดคำสั่งที่ถูกพัฒนาขึ้นมาเพื่อกำหนดกรอบหรือรูปแบบการพัฒนาเว็บไซต์ในส่วนของการปฏิสัมพันธ์กับผู้ใช้งานเว็บไซต์ (User Interface) เราจึงสามารถเรียก Bootstrap ว่าเป็น Front-end framework คือใช้สำหรับ พัฒนาเว็บไซต์ส่วนการแสดงผล ซึ่งแตกต่างจากภาษาประเภท Server Side Script อย่าง PHP, Python หรือ ภาษาอื่น ๆ

รู้จักทีมผู้พัฒนา

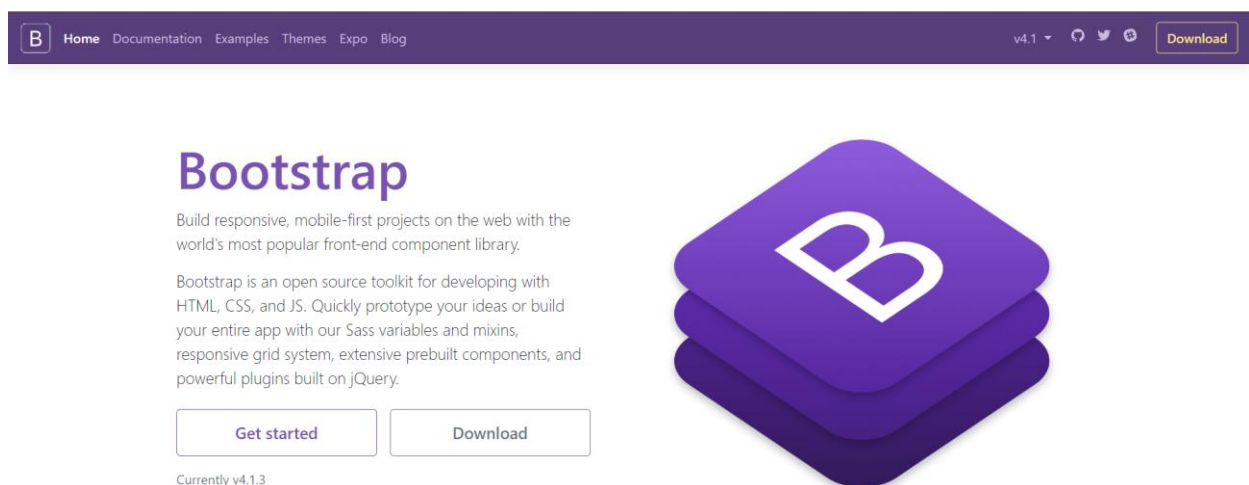
Bootstrap ถูกพัฒนาขึ้นโดย Mark Otto และ Jacob Thornton ทีมพัฒนาของ Twitter Inc. ก่อนหน้านี้ใช้ชื่อว่า Twitter Blueprint และเปิดให้นักพัฒนาสามารถนำไปใช้งานพัฒนาเว็บไซต์ได้แบบฟรี (Open Source) ในชื่อว่า Bootstrap Framework

จุดเด่นของ Bootstrap Framework

- มี UI เริ่มต้นแบบที่สวยงามและใช้งานง่าย
- มีการปรับปรุงและพัฒนาอย่างต่อเนื่อง ปัจจุบันเป็นเวอร์ชัน 4.1.3
- เป็นที่นิยมของนักพัฒนาทั่วโลก ทำให้สามารถเรียนรู้และแก้ปัญหาได้ง่าย
- โค้ดหรือชุดคำสั่งต่าง ๆ ค่อนข้างสะอาดมีโฟลเดอร์ต้นแบบแค่ 3 ส่วนคือ js, css, fonts
- ประหยัดเวลาในการพัฒนาเว็บไซต์และนำไปพัฒนาต่อได้ง่าย
- เป็น Responsive Framework พัฒนาเว็บไซต์ที่รองรับการแสดงผลได้หลากหลาย Device

เริ่มต้นใช้งาน

เข้าไปที่เว็บไซต์ของผู้พัฒนาที่ <https://getbootstrap.com/> เมื่อเข้าไปถึงจะพบหน้าจอต้อนรับประมาณนี้



การติดตั้งและเรียกใช้งาน

วิธีการติดตั้ง สามารถ Download เป็นไฟล์มาเก็บไว้ในเครื่องเรา หรือ หากใครไม่อยาก Download จะใช้งาน แบบ Online โดยใช้บริการจาก CDN ที่ Bootstrap เตรียมไว้ <https://getbootstrap.com/> ไฟล์ที่ต้องเรียกใช้งานมีดังนี้

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

สำหรับตัวอย่างทั้งหมดใน E-book เล่มนี้ จะเรียกใช้งาน จาก CDN ครับ

ตัวอย่างการเรียกใช้งาน

```
1. <!doctype html>
2. <html lang="en">
3.   <head>
4.     <!-- Required meta tags -->
5.     <meta charset="utf-8">
6.     <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
7.
8.     <!-- Bootstrap CSS -->
9.     <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css">
10.
11.    <title>Hello, world!</title>
12.  </head>
13.  <body>
14.    <h1>Hello, world!</h1>
15.
16.    <!-- Optional JavaScript -->
17.    <!-- jQuery first, then Popper.js, then Bootstrap JS -->
18.    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"></script>
19.    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js"></script>
20.    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js"></script>
21.  </body>
22. </html>
```

Grid System

ระบบ Grid System คือ หลักการแบ่งหน้าจอ ออกเป็น 12 columns เท่า ๆ กัน โดยในแต่ละ แถว (row) จะมีกี่ columns ก็ได้ แต่รวมกันจะต้องไม่เกิน 12 ช่อง มาลอง

.col-xs-1	.col-xs-1	.col-xs-1	.col-xs-1	.col-xs-1	.col-xs-1	.col-xs-1	.col-xs-1	.col-xs-1	.col-xs-1	.col-xs-1	.col-xs-1
.col-xs-2		.col-xs-3			.col-xs-7						
.col-xs-4				.col-xs-4				.col-xs-4			
.col-xs-5					.col-xs-7						
.col-xs-6						.col-xs-6					
.col-xs-12											

นอกจากนั้นเรายังสามารถกำหนดได้อีกว่าอยากให้ columns โหนดแสดงใน หน้าจอแต่ละขนาดได้ด้วย อธิบายตามความเข้าใจอีกครั้งก็คือ อยากให้บน Desktop แสดงเป็น 6 columns แต่ถ้าไปเปิดหน้าเว็บบน Mobile ให้แสดงเป็น 12 columns ก็สามารทำได้ โดย Bootstrap 4 ได้มีการแบ่งหน้าจอขนาดต่าง ๆ ดังนี้

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

- 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

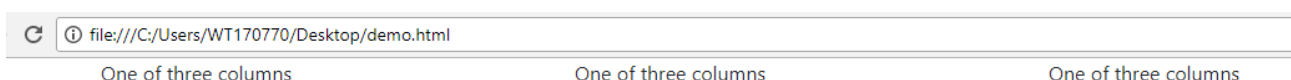
การเรียกใช้งาน Grid เราจะสร้าง class row ขึ้นมาก่อน แล้ว เรียก class="col-ขนาดหน้าจอ-จำนวน Grid"

ตัวอย่างการเรียกใช้งาน

```
1. <div class="container">
2.   <div class="row">
3.     <div class="col-sm">
4.       One of three columns
5.     </div>
6.     <div class="col-sm">
7.       One of three columns
8.     </div>
9.     <div class="col-sm">
10.      One of three columns
11.    </div>
12.  </div>
13.</div>
```

```
1. <!-- Stack the columns on mobile by making one full-width and the other half-
   width -->
2. <div class="row">
3.   <div class="col-12 col-md-8">.col-12 .col-md-8</div>
4.   <div class="col-6 col-md-4">.col-6 .col-md-4</div>
5. </div>
6.
7. <!-- Columns start at 50% wide on mobile and bump up to 33.3% wide on desktop -->
8. <div class="row">
9.   <div class="col-6 col-md-4">.col-6 .col-md-4</div>
10.  <div class="col-6 col-md-4">.col-6 .col-md-4</div>
11.  <div class="col-6 col-md-4">.col-6 .col-md-4</div>
12.</div>
13.
14.<!-- Columns are always 50% wide, on mobile and desktop -->
15.<div class="row">
16.  <div class="col-6">.col-6</div>
17.  <div class="col-6">.col-6</div>
18.</div>
```

ตัวอย่างการแสดงผล จะเห็นว่า ได้มีการแบ่งหน้าจอออกเป็น 3 ส่วน ตามที่เรากำหนดไว้



Auto-layout columns

เราสามารถให้ Bootstrap กำหนดขนาดของ columns ให้เราอัตโนมัติ โดยใช้ class .col

```
1. <div class="container">
2.   <div class="row">
3.     <div class="col">
4.       1 of 2
5.     </div>
6.     <div class="col">
7.       2 of 2
8.     </div>
9.   </div>
10.  <div class="row">
11.    <div class="col">
12.      1 of 3
13.    </div>
14.    <div class="col">
15.      2 of 3
16.    </div>
17.    <div class="col">
18.      3 of 3
19.    </div>
20.  </div>
21. </div>
```

ตัวอย่างการแสดงผล

1 of 2	2 of 2	
1 of 3	2 of 3	3 of 3

Tables

สำหรับ Table ปกติ เวลาที่เราเขียน HTML เวลาต้องแสดงข้อมูลที่เป็น ตาราง เราอาจต้องมาเขียนในส่วนของการจัดรูปแบบตาราง เช่น ใส่เส้นขอบ(border), ใส่สี (background-color), หรือ จักระยะของตาราง (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

การใช้งานเพียงแค่นี้เพิ่ม class ที่ชื่อว่า table ลงไปก็สามารถใช้งานได้ทันที

```
1. <table class="table">
2.   <thead>
3.     <tr>
4.       <th scope="col">#</th>
5.       <th scope="col">First</th>
6.       <th scope="col">Last</th>
7.       <th scope="col">Handle</th>
8.     </tr>
9.   </thead>
10.  <tbody>
11.    <tr>
12.      <th scope="row">1</th>
13.      <td>Mark</td>
14.      <td>Otto</td>
15.      <td>@mdo</td>
16.    </tr>
17.    <tr>
18.      <th scope="row">2</th>
19.      <td>Jacob</td>
20.      <td>Thornton</td>
21.      <td>@fat</td>
22.    </tr>
23.    <tr>
24.      <th scope="row">3</th>
25.      <td>Larry</td>
26.      <td>the Bird</td>
27.      <td>@twitter</td>
28.    </tr>
29.  </tbody>
30.</table>
```

ตัวอย่างการแสดงผล เมื่อใช้ bootstrap table

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

Striped rows

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

ตัวอย่าง การเพิ่ม class table-striped ใน table

```
1. <table class="table table-striped">
2.   <thead>
3.     <tr>
4.       <th scope="col">#</th>
5.       <th scope="col">First</th>
6.       <th scope="col">Last</th>
7.       <th scope="col">Handle</th>
8.     </tr>
9.   </thead>
10.  <tbody>
11.    <tr>
12.      <th scope="row">1</th>
13.      <td>Mark</td>
14.      <td>Otto</td>
15.      <td>@mdo</td>
16.    </tr>
17.    <tr>
18.      <th scope="row">2</th>
19.      <td>Jacob</td>
20.      <td>Thornton</td>
21.      <td>@fat</td>
22.    </tr>
23.    <tr>
24.      <th scope="row">3</th>
25.      <td>Larry</td>
26.      <td>the Bird</td>
27.      <td>@twitter</td>
28.    </tr>
29.  </tbody>
30.</table>
```

ตัวอย่างการแสดงผล

#	First	Last	Handle
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

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

ตัวอย่างการแสดงผล จะมีการเพิ่มเส้นขอบให้กับตารางให้มีความเรียบร้อยมากยิ่งขึ้น

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

Hover Table

เมื่อนามาส์ไปชี้ที่ตาราง จะมี hover สีเทาที่ตารางเพื่อให้เห็นว่าเรากำลังชี้ที่ข้อมูลในแถวโดยอยู่ โดยเพิ่ม class .table-hover

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

ตัวอย่างการแสดงผล

#	First	Last	Handle
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

<https://getbootstrap.com/docs/4.1/content/images/>

สำหรับในส่วนของรูปภาพ ทาง bootstrap ได้เตรียมรูปแบบ ต่าง ๆ ให้เราใช้งาน ดังนี้

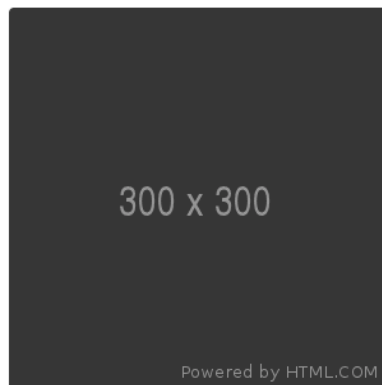
- **.rounded** คือการเพิ่มขอบมนให้กับรูปภาพ
- **.rounded-circle** คือกำหนดให้รูปภาพเป็นวงกลม
- **.img-thumbnail** คือการแสดงผลภาพตัวอย่าง (Thumbnail) แบบมีเส้นขอบรอบรูปภาพ
- **.img-fluid** คือ การกำหนดให้รูปภาพแสดงผลแบบ responsive

ตัวอย่างการใช้งาน แต่ละแบบ เพิ่ม class ที่เราต้องการให้กับ tag img ดังนี้

rounded เพิ่มขอบมนให้กับภาพ

```
1. 
```

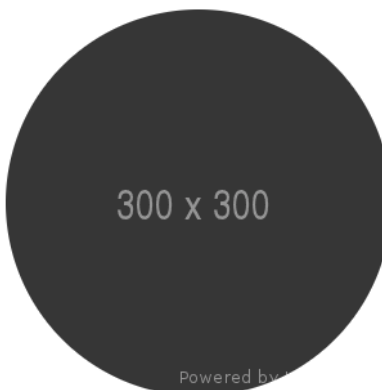
ตัวอย่างการแสดงผล



rounded-circle กำหนดให้รูปภาพเป็นวงกลม

```
1. 
```

ตัวอย่างการแสดงผล



img-thumbnail แสดงภาพตัวอย่าง (Thumbnail) แบบมีเส้นขอบ

1. ``

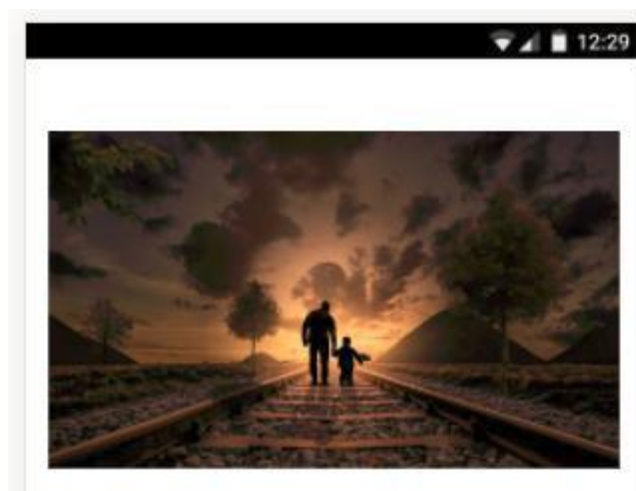
ตัวอย่างการแสดงผล



img-fluid คือ การกำหนดให้รูปภาพแสดงผลแบบ responsive

1. ``

ตัวอย่างการแสดงผล



Alert

<https://getbootstrap.com/docs/4.1/components/alerts/>

คือกล่องข้อความแจ้งเตือนต่าง ๆ ในระบบ เรียกใช้งานได้โดยใช้ class .alert alert-สีของกล่องข้อความ

```
1. <div class="alert alert-primary" role="alert">
2.   A simple primary alert-check it out!
3. </div>
4. <div class="alert alert-secondary" role="alert">
5.   A simple secondary alert-check it out!
6. </div>
7. <div class="alert alert-success" role="alert">
8.   A simple success alert-check it out!
9. </div>
10. <div class="alert alert-danger" role="alert">
11.   A simple danger alert-check it out!
12. </div>
13. <div class="alert alert-warning" role="alert">
14.   A simple warning alert-check it out!
15. </div>
16. <div class="alert alert-info" role="alert">
17.   A simple info alert-check it out!
18. </div>
19. <div class="alert alert-light" role="alert">
20.   A simple light alert-check it out!
21. </div>
22. <div class="alert alert-dark" role="alert">
23.   A simple dark alert-check it out!
24. </div>
```

ตัวอย่างการแสดงผล

A simple primary alert—check it out!

A simple secondary alert—check it out!

A simple success alert—check it out!

A simple danger alert—check it out!

A simple warning alert—check it out!

A simple info alert—check it out!

A simple light alert—check it out!

A simple dark alert—check it out!

Button

<https://getbootstrap.com/docs/4.1/components/buttons/>

ในส่วนของปุ่ม นั้นทาง Bootstrap ได้เตรียมรูปแบบไว้ให้เราใช้ หลายแบบ ตัวอย่างการแสดงผล ดังนี้



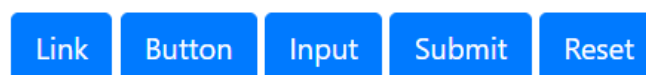
ตัวอย่าง code การใช้งาน button เพิ่ม class btn btn-ชื่อปุ่มที่ต้องการ เช่น btn btn-primary เป็นต้น

```
1. <button type="button" class="btn btn-primary">Primary</button>
2. <button type="button" class="btn btn-secondary">Secondary</button>
3. <button type="button" class="btn btn-success">Success</button>
4. <button type="button" class="btn btn-danger">Danger</button>
5. <button type="button" class="btn btn-warning">Warning</button>
6. <button type="button" class="btn btn-info">Info</button>
7. <button type="button" class="btn btn-light">Light</button>
8. <button type="button" class="btn btn-dark">Dark</button>
9.
10. <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

```
1. <a class="btn btn-primary" href="#" role="button">Link</a>
2. <button class="btn btn-primary" type="submit">Button</button>
3. <input class="btn btn-primary" type="button" value="Input">
4. <input class="btn btn-primary" type="submit" value="Submit">
5. <input class="btn btn-primary" type="reset" value="Reset">
```

ตัวอย่างการแสดงผล



Button Outline

<https://getbootstrap.com/docs/4.1/components/buttons/#outline-buttons>

คือ การแสดงผล button แบบมีเฉพาะเส้นขอบ การใช้งานเพียงแค่ เพิ่ม outline ลงไปใน class เช่น

```
1. <button type="button" class="btn btn-outline-primary">Primary</button>
2. <button type="button" class="btn btn-outline-secondary">Secondary</button>
3. <button type="button" class="btn btn-outline-success">Success</button>
4. <button type="button" class="btn btn-outline-danger">Danger</button>
5. <button type="button" class="btn btn-outline-warning">Warning</button>
6. <button type="button" class="btn btn-outline-info">Info</button>
7. <button type="button" class="btn btn-outline-light">Light</button>
8. <button type="button" class="btn btn-outline-dark">Dark</button>
```

ตัวอย่างการแสดงผล



Button Size

<https://getbootstrap.com/docs/4.1/components/buttons/#sizes>

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

```
1. <button type="button" class="btn btn-primary btn-lg">lage button</button>
2. <button type="button" class="btn btn-primary btn-sm">Small button</button>
```

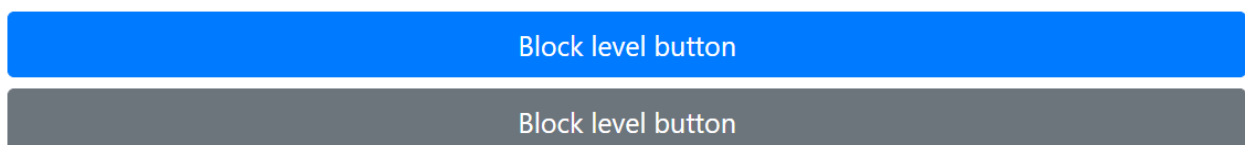
ตัวอย่างการแสดงผล



เราสามารถกำหนดให้ button ขยายความกว้างเต็มขนาด ได้โดยการเพิ่ม .btn-block

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

ตัวอย่างการแสดงผล



Card

<https://getbootstrap.com/docs/4.1/components/card/>

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

Basic Card

```
1. <div class="card">
2.   <div class="card-body">
3.     This is some text within a card body.
4.   </div>
5. </div>
```

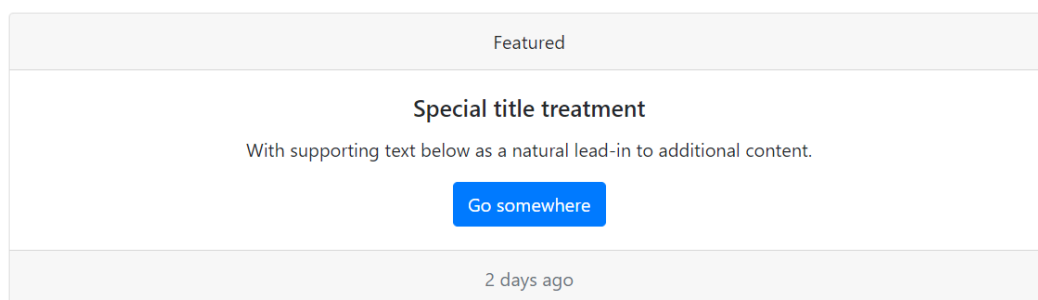
การแสดงผล จะเห็นว่ามีกรอบรอบข้อความทำให้มีความสวยงามมากยิ่งขึ้น

This is some text within a card body.

Header and footer

```
1. <div class="card text-center">
2.   <div class="card-header">
3.     Featured
4.   </div>
5.   <div class="card-body">
6.     <h5 class="card-title">Special title treatment</h5>
7.     <p class="card-text">With supporting text below as a natural lead-
      in to additional content.</p>
8.     <a href="#" class="btn btn-primary">Go somewhere</a>
9.   </div>
10.  <div class="card-footer text-muted">
11.    2 days ago
12.  </div>
13. </div>
```

ตัวอย่างการแสดงผล



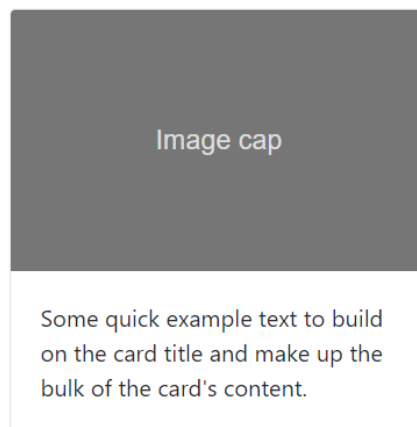
Card Images

<https://getbootstrap.com/docs/4.1/components/card/#images>

ใน card เราสามารถใส่รูปภาพแล้วมีข้อความประกอบด้านล่างได้ สามารถนำไปประยุกต์ใช้งานได้หลายแบบ เช่น แสดงข้อมูลสินค้า เป็นต้น

```
1. <div class="card" style="width: 18rem;">
2.   
3.   <div class="card-body">
4.     <p class="card-text">Some quick example text to build on the card title and make up the bulk of the card's content.</p>
5.   </div>
6. </div>
```

ตัวอย่างการแสดงผล

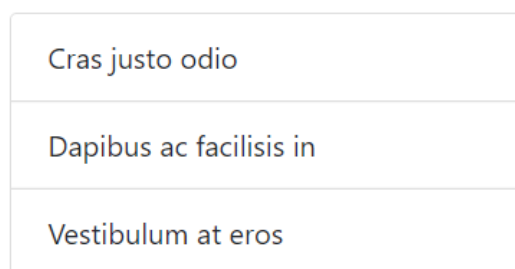


List groups

<https://getbootstrap.com/docs/4.1/components/list-group/>

```
1. <div class="card" style="width: 18rem;">
2.   <ul class="list-group list-group-flush">
3.     <li class="list-group-item">Cras justo odio</li>
4.     <li class="list-group-item">Dapibus ac facilisis in</li>
5.     <li class="list-group-item">Vestibulum at eros</li>
6.   </ul>
7. </div>
```

ตัวอย่างการแสดงผล สามารถแสดงข้อมูลในรูปแบบ List เรียงลงมาใน Card ได้



Dropdown

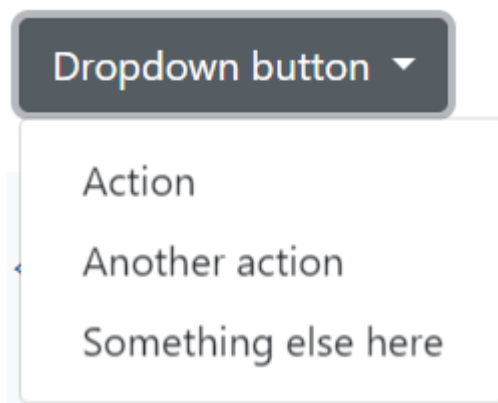
<https://getbootstrap.com/docs/4.1/components/dropdowns/>

คือ ปุ่มที่สามารถย่อ หรือ แสดงเมนูที่อยู่ด้านในได้ เหมาะสำหรับในกรณีที่เรามีเมนูจำนวนมาก เมื่อใช้งาน Dropdown จะช่วยลดพื้นที่ของหน้าเว็บในการแสดงเมนูได้

ตัวอย่างการแสดงผล สามารถคลิกเพื่อแสดงหรือย่อเมนูได้

```
1. <div class="dropdown">
2.   <button class="btn btn-secondary dropdown-
   toggle" type="button" id="dropdownMenuButton" data-toggle="dropdown" aria-
   haspopup="true" aria-expanded="false">
3.     Dropdown button
4.   </button>
5.   <div class="dropdown-menu" aria-labelledby="dropdownMenuButton">
6.     <a class="dropdown-item" href="#">Action</a>
7.     <a class="dropdown-item" href="#">Another action</a>
8.     <a class="dropdown-item" href="#">Something else here</a>
9.   </div>
10. </div>
```

ตัวอย่างการแสดงผล



Dropdowns Sizing

<https://getbootstrap.com/docs/4.1/components/dropdowns/#sizing>

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



Forms

<https://getbootstrap.com/docs/4.1/components/forms/>

สำหรับ Forms คือส่วนที่ใช้สำหรับการรับข้อมูลจากผู้ใช้ เช่น ช่องกรอกข้อความ ต่าง ๆ เป็นต้น ใน Bootstrap ได้เตรียมรูปแบบ Forms ให้เราใช้งานดังนี้

Form-Group และ Form controls

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

```
1. <form>
2.   <div class="form-group">
3.     <label for="exampleInputEmail1">Email address</label>
4.     <input type="email" class="form-control" id="exampleInputEmail1" aria-
      describedby="emailHelp" placeholder="Enter email">
5.     <small id="emailHelp" class="form-text text-
      muted">We'll never share your email with anyone else.</small>
6.   </div>
7.   <div class="form-group">
8.     <label for="exampleInputPassword1">Password</label>
9.     <input type="password" class="form-
      control" id="exampleInputPassword1" placeholder="Password">
10.  </div>
11.  <div class="form-group form-check">
12.    <input type="checkbox" class="form-check-input" id="exampleCheck1">
13.    <label class="form-check-label" for="exampleCheck1">Check me out</label>
14.  </div>
15.  <button type="submit" class="btn btn-primary">Submit</button>
16. </form>
```

ตัวอย่างการแสดงผล

Email address

We'll never share your email with anyone else.

Password

☐ Check me out

Form Sizing

<https://getbootstrap.com/docs/4.1/components/forms/#sizing>

เราสามารถ ปรับขนาดของ input โดยใช้ class `.form-control-lg` (ขนาดใหญ่) และ class `.form-control-sm` (ขนาดเล็ก)

```
1. <input class="form-control form-control-lg" type="text" placeholder=".form-control-lg">
2. <input class="form-control" type="text" placeholder="Default input">
3. <input class="form-control form-control-sm" type="text" placeholder=".form-control-sm">
```

ตัวอย่างการแสดงผล

.form-control-lg

Default input

.form-control-sm

Read only

<https://getbootstrap.com/docs/4.1/components/forms/#readonly>

ในบางครั้งเราต้องการที่จะแสดงข้อมูลในช่อง input แต่ไม่ต้องการให้ผู้แก้ไขข้อมูลในช่องได้ เราสามารถกำหนดให้ ช่อง input อ่านได้อย่างเดียวเท่านั้น โดยใช้ attribute `readonly`

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

ตัวอย่างการแสดงผล จะเห็นว่าในช่อง input จะเป็นสีเทา และ จะไม่สามารถแก้ไขข้อมูลได้

Readonly input here...

Read only plain text

<https://getbootstrap.com/docs/4.1/components/forms/#readonly-plain-text>

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

```
1. <form>
2.   <div class="form-group row">
3.     <label for="staticEmail" class="col-sm-2 col-form-label">Email</label>
4.     <div class="col-sm-10">
5.       <input type="text" readonly class="form-control-
        plaintext" id="staticEmail" value="email@example.com">
6.     </div>
7.   </div>
8.   <div class="form-group row">
9.     <label for="inputPassword" class="col-sm-2 col-form-label">Password</label>
10.    <div class="col-sm-10">
11.      <input type="password" class="form-
        control" id="inputPassword" placeholder="Password">
12.    </div>
13.  </div>
14. </form>
```

ตัวอย่างการแสดงผล

Email email@example.com

Password Password

```
1. <form class="form-inline">
2.   <div class="form-group mb-2">
3.     <label for="staticEmail2" class="sr-only">Email</label>
4.     <input type="text" readonly class="form-control-
        plaintext" id="staticEmail2" value="email@example.com">
5.   </div>
6.   <div class="form-group mx-sm-3 mb-2">
7.     <label for="inputPassword2" class="sr-only">Password</label>
8.     <input type="password" class="form-
        control" id="inputPassword2" placeholder="Password">
9.   </div>
10.  <button type="submit" class="btn btn-primary mb-2">Confirm identity</button>
11. </form>
```

ตัวอย่างการแสดงผล

email@example.com

Password

Confirm identity

Form Grid

<https://getbootstrap.com/docs/4.1/components/forms/#form-grid>

สำหรับ Form ใน Bootstrap เราสามารถแบ่งการแสดงผล Form โดยใช้ grid ได้ดังนี้

```
1. <form>
2.   <div class="row">
3.     <div class="col">
4.       <input type="text" class="form-control" placeholder="First name">
5.     </div>
6.     <div class="col">
7.       <input type="text" class="form-control" placeholder="Last name">
8.     </div>
9.   </div>
10. </form>
```

ตัวอย่างการแสดงผล

Form row

<https://getbootstrap.com/docs/4.1/components/forms/#form-row>

```
1. <form>
2.   <div class="form-row">
3.     <div class="form-group col-md-6">
4.       <label for="inputCity">City</label>
5.       <input type="text" class="form-control" id="inputCity">
6.     </div>
7.     <div class="form-group col-md-4">
8.       <label for="inputState">State</label>
9.       <select id="inputState" class="form-control">
10.        <option selected>Choose...</option>
11.        <option>...</option>
12.      </select>
13.    </div>
14.    <div class="form-group col-md-2">
15.      <label for="inputZip">Zip</label>
16.      <input type="text" class="form-control" id="inputZip">
17.    </div>
18.  </div>
19. </form>
```

ตัวอย่างการแสดงผล

Horizontal form

<https://getbootstrap.com/docs/4.1/components/forms/#horizontal-form>

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

```
1. <form>
2.   <div class="form-group row">
3.     <label for="inputEmail3" class="col-sm-2 col-form-label">Email</label>
4.     <div class="col-sm-10">
5.       <input type="email" class="form-
control" id="inputEmail3" placeholder="Email">
6.     </div>
7.   </div>
8.   <div class="form-group row">
9.     <label for="inputPassword3" class="col-sm-2 col-form-
label">Password</label>
10.    <div class="col-sm-10">
11.      <input type="password" class="form-
control" id="inputPassword3" placeholder="Password">
12.    </div>
13.  </div>
14.  <div class="form-group row">
15.    <div class="col-sm-10">
16.      <button type="submit" class="btn btn-primary">Sign in</button>
17.    </div>
18.  </div>
19. </form>
```

ตัวอย่างการแสดงผล

Email	<input type="text" value="Email"/>
Password	<input type="password" value="Password"/>
<input type="button" value="Sign in"/>	

Inline forms

<https://getbootstrap.com/docs/4.1/components/forms/#inline-forms>

คือ ฟอรม ที่มี input เรียงต่อกันไปบนแถวเดียว โดยเราจะเพิ่ม class .form-inline ใน form

```
1. <form class="form-inline">
2.   <label class="sr-only" for="inlineFormInputName2">Name</label>
3.   <input type="text" class="form-control mb-2 mr-sm-
4.     2" id="inlineFormInputName2" placeholder="Jane Doe">
5.   <label class="sr-only" for="inlineFormInputGroupUsername2">Username</label>
6.   <div class="input-group mb-2 mr-sm-2">
7.     <div class="input-group-prepend">
8.       <div class="input-group-text">@</div>
9.     </div>
10.    <input type="text" class="form-
11.      control" id="inlineFormInputGroupUsername2" placeholder="Username">
12.    </div>
13.    <div class="form-check mb-2 mr-sm-2">
14.      <input class="form-check-input" type="checkbox" id="inlineFormCheck">
15.      <label class="form-check-label" for="inlineFormCheck">
16.        Remember me
17.      </label>
18.    </div>
19.
20.    <button type="submit" class="btn btn-primary mb-2">Submit</button>
21. </form>
```

ตัวอย่างการแสดงผล

☐ Remember me

NavBar

<https://getbootstrap.com/docs/4.1/components/navbar/>

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

```
1. <nav class="navbar navbar-expand-lg navbar-light bg-light">
2.   <a class="navbar-brand" href="#">Navbar</a>
3.   <button class="navbar-toggler" type="button" data-toggle="collapse" data-
4.     target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-
5.     expanded="false" aria-label="Toggle navigation">
6.     <span class="navbar-toggler-icon"></span>
7.   </button>
8.   <div class="collapse navbar-collapse" id="navbarSupportedContent">
9.     <ul class="navbar-nav mr-auto">
10.      <li class="nav-item active">
11.        <a class="nav-link" href="#">Home <span class="sr-
12.          only">(current)</span></a>
13.      </li>
14.      <li class="nav-item">
15.        <a class="nav-link" href="#">Link</a>
16.      </li>
17.      <li class="nav-item dropdown">
18.        <a class="nav-link dropdown-
19.          toggle" href="#" id="navbarDropdown" role="button" data-toggle="dropdown" aria-
20.          haspopup="true" aria-expanded="false">
21.          Dropdown
22.        </a>
23.        <div class="dropdown-menu" aria-labelledby="navbarDropdown">
24.          <a class="dropdown-item" href="#">Action</a>
25.          <a class="dropdown-item" href="#">Another action</a>
26.          <div class="dropdown-divider"></div>
27.          <a class="dropdown-item" href="#">Something else here</a>
28.        </div>
29.      </li>
30.      <li class="nav-item">
31.        <a class="nav-link disabled" href="#">Disabled</a>
32.      </li>
33.    </ul>
34.    <form class="form-inline my-2 my-lg-0">
35.      <input class="form-control mr-sm-
36.        2" type="search" placeholder="Search" aria-label="Search">
37.      <button class="btn btn-outline-success my-2 my-sm-
38.        0" type="submit">Search</button>
39.    </form>
40.  </div>
41.</nav>
```

ตัวอย่างการแสดงผล

Navbar Home Link Dropdown ▾ Disabled

Search

Search

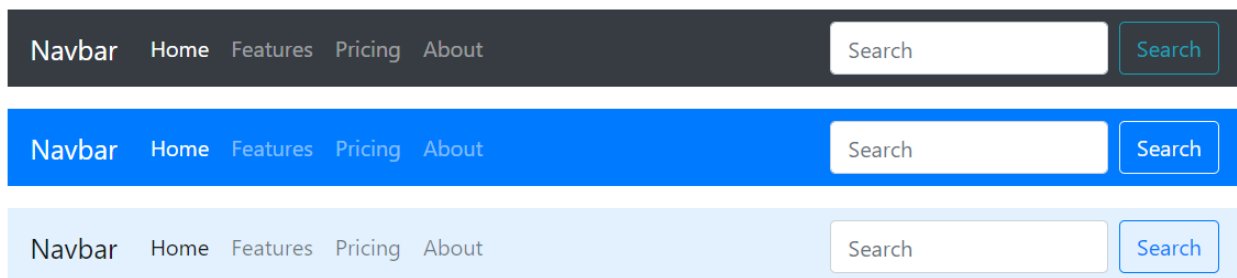
Navbar Color

<https://getbootstrap.com/docs/4.1/components/navbar/#color-schemes>

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

```
1. <nav class="navbar navbar-dark bg-dark">
2.   <!-- Navbar content -->
3. </nav>
4.
5. <nav class="navbar navbar-dark bg-primary">
6.   <!-- Navbar content -->
7. </nav>
8.
9. <nav class="navbar navbar-light" style="background-color: #e3f2fd;">
10.  <!-- Navbar content -->
11. </nav>
```

ตัวอย่างการแสดงผล



Navbar Placement

<https://getbootstrap.com/docs/4.1/components/navbar/#placement>

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

Fixed top กำหนดให้ Navbar ยึดติดอยู่กับด้านบนสุดของหน้าจอตลอดเวลา

ตัวอย่างการเรียกใช้งาน

```
1. <nav class="navbar navbar-light bg-light">
```

Fixed bottom กำหนดให้ Navbar ติดอยู่ด้านล่างสุดของจอ

ตัวอย่างการเรียกใช้งาน

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

Popovers

<https://getbootstrap.com/docs/4.1/components/popovers/>

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

```
1. $(function () {  
2.   $('[data-toggle="popover"]').popover()  
3. })
```

ตัวอย่างการเรียกใช้งาน Popovers

```
1. <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.">  
2.   Popover on top  
3. </button>  
4.  
5. <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.">  
6.   Popover on right  
7. </button>  
8.  
9. <button type="button" class="btn btn-secondary" data-container="body" data-  
   toggle="popover" data-placement="bottom" data-content="Vivamus  
10. sagittis lacus vel augue laoreet rutrum faucibus.">  
11.   Popover on bottom  
12. </button>  
13.  
14. <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.">  
15.   Popover on left  
16. </button>
```

ตัวอย่างการแสดงผล



Tooltips

<https://getbootstrap.com/docs/4.1/components/tooltips/>

เราสามารถแสดงข้อความแบบ Tooltips ให้กับ button ได้ โดยจะทำงานเมื่อนาเมาส์ไปวางที่ button หรือ Tag Html ที่เรากำหนดค่า tooltip ได้ เหมาะสำหรับการแสดงข้อความแนะนำการใช้งานปุ่มต่าง ๆ เป็นต้น

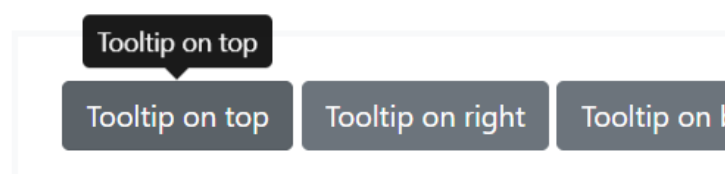
ก่อนการใช้งานเราต้องมีการเรียกใช้งาน jQuery เพื่อให้ tooltip ทำงานดังนี้

```
1. $(function () {  
2.   $('[data-toggle="tooltip"]').tooltip()  
3. })
```

ตัวอย่างการเรียกใช้งาน

```
1. <button type="button" class="btn btn-secondary" data-toggle="tooltip" data-  
   placement="top" title="Tooltip on top">  
2.   Tooltip on top  
3. </button>  
4. <button type="button" class="btn btn-secondary" data-toggle="tooltip" data-  
   placement="right" title="Tooltip on right">  
5.   Tooltip on right  
6. </button>  
7. <button type="button" class="btn btn-secondary" data-toggle="tooltip" data-  
   placement="bottom" title="Tooltip on bottom">  
8.   Tooltip on bottom  
9. </button>  
10. <button type="button" class="btn btn-secondary" data-toggle="tooltip" data-  
    placement="left" title="Tooltip on left">  
11.   Tooltip on left  
12. </button>
```

ตัวอย่างการแสดงผล เมื่อนาเมาส์ไปวางบน Button จะมี Tooltip แสดงข้อความขึ้นมาทันที



เราสามารถกำหนดให้ Tooltip แสดงข้อมูลตรงตำแหน่ง ด้าน บน ล่าง ซ้าย หรือ ขวา โดยกำหนดได้ที่ data-placement

ตัวอย่าง กำหนดให้ Tooltip แสดงด้านซ้ายของ Button

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

Modal

<https://getbootstrap.com/docs/4.1/components/modal/>

สำหรับ Modal เราสามารถแสดงข้อความ เป็นกล่องข้อความ ดังขึ้นมาอยู่เหนือหน้าเว็บของเรา ซึ่งเป็นที่นิยมในการแสดงข้อความหรือ ประยุกต์ใช้งานได้อีกหลากหลายรูปแบบ

ตัวอย่างการเรียกใช้งาน

สร้าง Button สำหรับเรียกใช้งาน modal

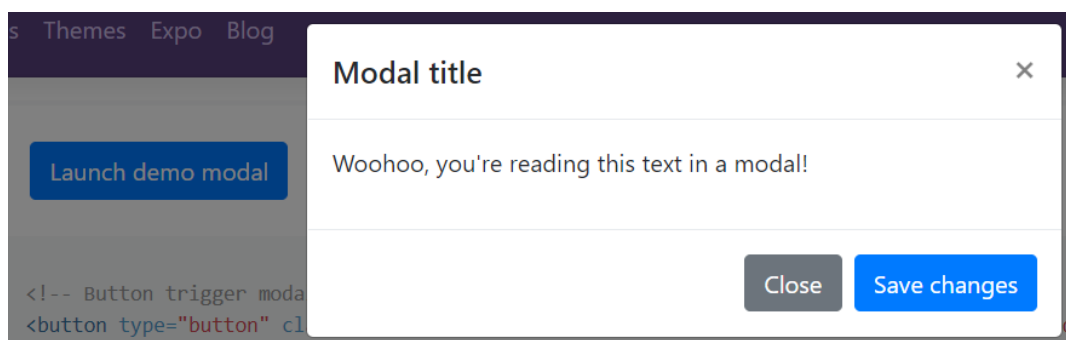
```
1. <!-- Button trigger modal -->
2. <button type="button" class="btn btn-primary" data-toggle="modal" data-
   target="#exampleModal">
3.   Launch demo modal
4. </button>
```

สร้าง class .modal สำหรับแสดงข้อความใน modal

โดยเราสามารถ ใส่ข้อความ ได้ภายในส่วนของ class .modal-body

```
1. <!-- Modal -->
2. <div class="modal fade" id="exampleModal" tabindex="-1" role="dialog" aria-
   labelledby="exampleModalLabel" aria-hidden="true">
3.   <div class="modal-dialog" role="document">
4.     <div class="modal-content">
5.       <div class="modal-header">
6.         <h5 class="modal-title" id="exampleModalLabel">Modal title</h5>
7.         <button type="button" class="close" data-dismiss="modal" aria-label="Close">
8.           <span aria-hidden="true">x</span>
9.         </button>
10.      </div>
11.      <div class="modal-body">
12.        ...
13.      </div>
14.      <div class="modal-footer">
15.        <button type="button" class="btn btn-secondary" data-
   dismiss="modal">Close</button>
16.        <button type="button" class="btn btn-primary">Save changes</button>
17.      </div>
18.    </div>
19.  </div>
20. </div>
```

ตัวอย่างการแสดงผล



Breadcrumb

<https://getbootstrap.com/docs/4.1/components/breadcrumb/>

Breadcrumb เป็นปุ่มที่มีไว้บอกตำแหน่งหน้าที่เรายู่ปัจจุบัน จะแสดงในลักษณะเหมือนเป็นลำดับชั้นของpathมายังหน้าที่เรายู่ปัจจุบัน ซึ่งทำให้เราสามารถคลิกไปที่หน้าก่อนหน้านี้ทั้งหมดได้ดังภาพข้างล่างครับ

```
1. <nav aria-label="breadcrumb">
2.   <ol class="breadcrumb">
3.     <li class="breadcrumb-item active" aria-current="page">Home</li>
4.   </ol>
5. </nav>
6.
7. <nav aria-label="breadcrumb">
8.   <ol class="breadcrumb">
9.     <li class="breadcrumb-item"><a href="#">Home</a></li>
10.    <li class="breadcrumb-item active" aria-current="page">Library</li>
11.  </ol>
12. </nav>
13.
14. <nav aria-label="breadcrumb">
15.   <ol class="breadcrumb">
16.     <li class="breadcrumb-item"><a href="#">Home</a></li>
17.     <li class="breadcrumb-item"><a href="#">Library</a></li>
18.     <li class="breadcrumb-item active" aria-current="page">Data</li>
19.   </ol>
20. </nav>
```

ตัวอย่างการแสดงผล

Home

Home / Library

Home / Library / Data

ส่งท้าย

สำหรับใน เนื้อหาที่แนะนำทั้งหมดใน เอกสารชุดนี้เป็นเพียงแค่การแนะนำบางส่วนทั้งหมดซึ่งสามารถดูตัวอย่างและวิธีการใช้งานเพิ่มเติมจาก <https://getbootstrap.com/> ซึ่งผู้เขียนได้เขียนเอกสารนี้ขึ้นเพื่อเป็นแนวทางในการเริ่มต้นสำหรับผู้เริ่มต้น หรือ ยังไม่เคยใช้งานเท่านั้น ผู้เขียนหวังเป็นอย่างยิ่งว่าข้อมูลในเอกสารชุดนี้จะมีประโยชน์บ้างสำหรับผู้อ่าน หากมีข้อผิดพลาดประการใดทางผู้เขียนต้องขออภัยมา ณ ที่นี้ด้วยครับ

เซสชัน (SESSION)

เซสชัน (Session) คือตัวแปรในภาษา PHP ซึ่งมีคุณสมบัติทุกอย่างเหมือนกับตัวแปรปกติทั้งเก็บค่าข้อความหรือตัวเลข ตลอดทั้งตัวแปรที่เป็น Object ต่างๆได้ แต่จะมีความพิเศษกว่าตัวแปรปกติคือ ตัวแปรเซสชันและค่าตัวแปรจะยังคงอยู่ไม่ว่าเราจะเปลี่ยนหน้าจากหน้าหนึ่งไป สู่อีกหน้าหนึ่งซึ่งเป็นข้อดีของตัวแปรนี้ เช่น การกำหนดค่าตัวแปรเซสชันในหน้า sample1.php แล้วเอาค่าที่ได้มาแสดงผลในหน้า sample2.php โดยจะได้ค่าที่กำหนดจากหน้า sample1.php มาทั้งหมด ตัวแปรเซสชันจะยังมีค่าคงอยู่เสมอจนกระทั่งตัวแปรหมดอายุหรือถูกทำการลบค่า ทั้ง ตัวแปรเซสชันนิยมนำมาใช้ในการรับส่งค่าข้อมูลที่เก็บเป็นความลับที่ต้องใช้ในการระบุตัวตนใช้สิทธิของระบบเช่น การยืนยันตัวตนด้วย Username และ Password เพื่อให้ได้ค่าตัวแปรนี้มาระบุตัวตน ตัวแปรเซสชันจึงมีความปลอดภัยในการเก็บรักษาข้อมูลได้อย่างมาก

เริ่มต้นใช้เซสชัน

ก่อนการใช้งานตัวแปรเซสชัน ต้องใช้คำสั่งเพื่อเปิดเซสชันดังนี้

```
1. <?php
2. session_start();
3. ob_start();
4. ?>
```

หน้ากำหนดตัวแปร session

สร้างไฟล์ sample1.php

```
1. <?php
2. session_start();
3. ob_start();
4.
5. $_SESSION['animal']="Cat";
6. $_SESSION['color']="Brown";
7. ?>
```

หน้าแสดงผลตัวแปรเซสชัน

สร้างไฟล์ sample2.php

```
1. <?php
2. session_start();
3. ob_start();
4. echo "Hello, Your animal is ".$_SESSION['animal'].' , It has '.$_SESSION['color']
   ].' color.';
5. // ผลลัพธ์คือแสดงข้อความ Hello, Your animal is Cat, It has Brown color.
6. ?>
```

เซสชันและอาเรย์

กำหนดตัวแปร สร้างไฟล์ init.php

```
1. <?php
2.     session_start();
3.     ob_start();
4.
5.     $_SESSION['arrData']=array('Joe','Alex','Anna','Sam','Dojo');
6. ?>
```

แสดงผลตัวแปรอาเรย์ สร้างไฟล์ display.php

```
1. <?php
2.     session_start();
3.     ob_start();
4.
5.     echo 'Contact list ';
6.     $count=1;
7.     foreach($_SESSION['arrData'] as $val){
8.         echo $count.') '.$val;
9.         $count+=1;
10.    }
11. ?>
```

ผลลัพธ์

```
Contact List
1.) Joe
2.) Alex
3.) Anna
4.) Sam
5.) Dojo
```

การลบค่าเซสชัน

การลบค่าเซสชันใช้คำสั่ง `unset()`; ในการลบเฉพาะตัวแปรเดียว และ `session_destroy()`; ในการลบตัวแปร session ทั้งหมด

ตัวอย่าง unset สร้างไฟล์ `unset.php`

```
1. <?php
2.     session_start();
3.     ob_start();
4.
5.     $_SESSION['sample']="Nakomah Studio. Full Stack Developer";
6.     unset($_SESSION['sample']);
7. ?>
```

ตัวอย่าง `session_destroy()`;

```
1. <?php
2.     session_start();
3.     ob_start();
4.
5.     $_SESSION['country']="Thailand";
6.     session_destroy();
7. ?>
```

การใช้งาน PHP Function และการสร้างฟังก์ชันขึ้นมาใช้เอง

- **include & require** (ความแตกต่างระหว่าง include & require) สำหรับข้อแตกต่างคือ include เป็นคำสั่งที่เพียงต้องการเรียกไฟล์ภายนอกแต่ไม่ได้ให้ความสำคัญมาก คือ ถ้าไม่มีการพบไฟล์ดังกล่าว โปรแกรมจะยังสามารถทำงานต่อไปได้นิยมใช้กับเอกสาร .html ส่วน require เป็นคำสั่งที่เรียกใช้งานแล้วไฟล์ดังกล่าวจะต้องมีอยู่จริง ถ้าไม่พบไฟล์โปรแกรมก็จะหยุดทำงานในทันที นิยมใช้กับเอกสาร .php, .inc

Syntax

```
1. <?php include("file.php");?>
2. <?php require("file.php");?>
```

อ้างอิงจาก <https://www.thaicreate.com/php/php-include-compare-require.html>

-
- **getdate()** เป็น Function อ่านค่าเวลาและวันที่ปัจจุบันของเครื่อง Server

Syntax

```
1. <?php
2.     getdate($format);
3. ?>
```

\$format Parameters

<https://www.thaicreate.com/php-manual/function.date.html>

ตัวอย่าง

```
1. <?php
2.     $today = getdate();
3.     echo "Day = ".$today["mday"]."<br>";
4.     echo "Month = ".$today["mon"]."<br>";
5.     echo "Year = ".$today["year"]."<br>";
6.     echo "<hr>";
7.     foreach($today as $key => $value) {
8.         echo "Key: $key; Value: $value<br>";
9.     }
10. ?>
```

ผลลัพธ์ที่ได้

```
Day = 14  
Month = 8  
Year = 2008
```

```
Key: seconds; Value: 21  
Key: minutes; Value: 32  
Key: hours; Value: 14  
Key: mday; Value: 14  
Key: wday; Value: 4  
Key: mon; Value: 8  
Key: year; Value: 2008  
Key: yday; Value: 226  
Key: weekday; Value: Thursday  
Key: month; Value: August  
Key: 0; Value: 1218695541
```

-
- **Strptime()** ฟังก์ชันในการอ่านวันที่และเวลาในรูปแบบของตัวเลข

Syntax

```
1. <?php  
2.     strptime($date,$format);  
3. ?>
```

\$format Parameters

<https://www.thaicreate.com/php-manual/function.date.html>

ตัวอย่าง

```
1. <?php  
2. $format = "%d/%m/%Y %H:%M:%S";  
3. $strDateTime = strftime($format);  
4. print_r(strptime($strf, $format));  
5.  
6. /**  
7. Out put Array  
8. Array  
9. (  
10.     [tm_sec] => 19  
11.     [tm_min] => 54  
12.     [tm_hour] => 15  
13.     [tm_mday] => 3  
14.     [tm_mon] => 9  
15.     [tm_year] => 104  
16.     [tm_wday] => 0  
17.     [tm_yday] => 276  
18.     [unparsed] =>  
19. )  
20. **/  
21. ?>
```

■ Compare Date/Time ฟังก์ชันการเปรียบเทียบเวลา

```
1. <?php
2.     function compareDate($date1,$date2) {
3.         $arrDate1 = explode("-", $date1);
4.         $arrDate2 = explode("-", $date2);
5.         $timStmp1 = mktime(0,0,0,$arrDate1[1],$arrDate1[2],$arrDate1[0]);
6.         $timStmp2 = mktime(0,0,0,$arrDate2[1],$arrDate2[2],$arrDate2[0]);
7.
8.         if ($timStmp1 == $timStmp2) {
9.             echo "\$date = \$date2";
10.        } else if ($timStmp1 > $timStmp2) {
11.            echo "\$date > \$date2";
12.        } else if ($timStmp1 < $timStmp2) {
13.            echo "\$date < \$date2";
14.        }
15.    }
16.    echo compareDate("2004-01-06", "2004-05-06");
17. ?>
```

ผลลัพธ์

```
$date < $date2
```

■ Thai Date/Time Function Thai Date ตัวอย่างนี้ผมจะยกตัวอย่างการเขียน function ให้แสดงวันที่และเวลาในรูปแบบของภาษาไทย

```
1. <?php
2.     function DateThai($strDate)
3.     {
4.         $strYear = date("Y",strtotime($strDate))+543;
5.         $strMonth= date("n",strtotime($strDate));
6.         $strDay= date("j",strtotime($strDate));
7.         $strHour= date("H",strtotime($strDate));
8.         $strMinute= date("i",strtotime($strDate));
9.         $strSeconds= date("s",strtotime($strDate));
10.        $strMonthCut = Array("", "ม.ค.", "ก.พ.", "มี.ค.", "เม.ย.", "พ.ค.", "มิ.ย.", "ก.ค.", "ส.ค.", "ก.ย.", "ต.ค.", "พ.ย.", "ธ.ค.");
11.        $strMonthThai=$strMonthCut[$strMonth];
12.        return "$strDay $strMonthThai $strYear, $strHour:$strMinute";
13.    }
14.
15.    $strDate = "2018-08-14 13:42:44";
16.    echo "Nakomah.com Time now : ".DateThai($strDate);
17. ?>
```

ผลลัพธ์

```
Nakomah.com Time now : 14 ส.ค. 2561, 13:42
```

- `date()` แสดงค่าวันที่และเวลา

Syntax

1. `date(format,timestamp)`

Parameter	Description
format	<p>รูปแบบการแสดงผลมีตัวเลือกต่อไปนี้</p> <p>อักขระที่ใช้แทน "วัน" และ "สัปดาห์"</p> <ul style="list-style-type: none">• d - ลำดับวันในรอบเดือนแบบมี 0 นำหน้า (มีค่าตั้งแต่ 01 - 31)• D - วันในรอบสัปดาห์แบบตัวอักษรย่อ 3 ตัว• j - วันที่ของเดือนแบบไม่มี 0 นำหน้า (มีค่าตั้งแต่ 1 - 31)• l (ตัว 'L' เล็ก) - ชื่อวันในรอบสัปดาห์แบบเต็ม• N - ลำดับวันในสัปดาห์ตามมาตรฐาน ISO-8601 (1 คือ Monday จนถึง 7 คือ Sunday)• S - ตัวย่อแสดงลำดับของวันคือวันแรกเป็น st,วันที่ 2 เป็น nd,วันที่ 3 เป็น rd ส่วนวันอื่นๆจะเป็น th.)• w - ลำดับวันเป็นตัวเลขในรอบสัปดาห์ (0 คือ Sunday จนถึง 6 คือ Saturday)• z - วันที่เท่าใดของปีนับจากต้นปี (จาก 0 ถึง 365)• W - แสดงค่านับสัปดาห์ที่ตามมาตรฐาน ISO-8601 <p>อักขระที่ใช้แทน "เดือน"</p> <ul style="list-style-type: none">• F - แสดงชื่อเดือนแบบเต็ม (มีค่าตั้งแต่ January - December)• m - รูปแบบเดือนเป็นตัวเลขมี 0 นำหน้า (มีค่าตั้งแต่ 01 - 12)• M - แสดงตัวอักษร 3 ตัวแรกของชื่อเดือน• n - รูปแบบเดือนเป็นตัวเลขไม่มี 0 นำหน้า (มีค่าตั้งแต่ 1 - 12)• t - จำนวนวันในเดือนนี้ <p>อักขระที่ใช้แทน "ปี"</p> <ul style="list-style-type: none">• L - ตรวจสอบปีอธิกมาส (มีค่าเป็น 1 ปีอธิกมาส หรือ 0 ถ้าไม่ใช่ปีอธิกมาส)• o - รูปแบบปีตามมาตรฐาน ISO-8601• Y - ปี ค.ศ. 4 หลัก• y - ปี ค.ศ. 2 หลัก <p>อักขระที่ใช้แทน "เวลา"</p> <ul style="list-style-type: none">• a - แสดงคำว่า am หรือ pm• A - แสดงคำว่า AM หรือ PM• B - เวลาในอินเทอร์เน็ตที่สร้างมาเพื่อเลี่ยงเวลาในเขตโซนเวลาที่ต่างกัน โดยแบ่งเวลาแต่ละวันเป็น 1000 ส่วนเรียกแต่ละส่วนว่า Beats ซึ่งแต่ละ Beats มีค่าเท่ากับ 86.4 วินาที (มีค่าตั้งแต่ 000 - 999)• g - รูปแบบชั่วโมงไม่มี 0 นำหน้าแบบ 12 ชั่วโมง (มีค่าตั้งแต่ 1 - 12)

	<ul style="list-style-type: none"> • G - รูปแบบชั่วโมงไม่มี 0 นำหน้าแบบ 12 ชั่วโมง (มีค่าตั้งแต่ 0 - 23) • h - รูปแบบชั่วโมง 2 หลักแบบ 12 ชั่วโมง (มีค่าตั้งแต่ 01 - 12) • H - รูปแบบชั่วโมง 2 หลักแบบ 24 ชั่วโมง (มีค่าตั้งแต่ 00 - 23) • i - เวลาในหน่วยนาที่แบบมีเลข 0 นำหน้า (มีค่าตั้งแต่ 00 - 59) • s - เวลาหน่วยวินาทีที่เป็นเลข 2 หลัก (มีค่าตั้งแต่ 00 - 59) • e - สัญลักษณ์เขต Timezone • l (i ตัวใหญ่) - มีค่าเป็น 1 ถ้าใช้ระบบ Daylight Savings Time : DST หรือ 0 ถ้าไม่ใช้ระบบ DST) • O - เวลาที่แตกต่างจากมาตรฐานกรีนิช (GMT) หน่วยเป็นชั่วโมง (ตัวอย่างเช่น: +0100) • T - เขตเวลา • Z - เขตเวลา Timezone ในหน่วยวินาที (มีค่าตั้งแต่ -43200 ถึง 43200) • c - เวลาตามมาตรฐาน ISO-8601 (รูปแบบคือ ปี-เดือน-วันTชั่วโมง:นาที่:วินาที(+,-)GMT ซึ่งจะใช้ได้กับ PHP เวอร์ชัน 5 เป็นต้นไป ตัวอย่างเช่น 2004-02-12T15:19:21+00:00) • r - วันเวลาตามมาตรฐาน RFC 2822 (ตัวอย่างเช่น Thu, 21 Dec 2000 16:01:07 +0200) • U - เวลา Timestamp อาจได้มาจาก time()
timestamp	ค่าของ Timestamp

ตัวอย่าง

```

1. <?php
2.     echo(date("l") . "<br />");
3.     echo(date("l dS \of F Y h:i:s A") . "<br />");
4.     echo("Oct 3,1975 was on a ".date("l", mktime(0,0,0,10,3,1975))."<br />");
5.     echo(date(DATE_RFC822) . "<br />");
6.     echo(date(DATE_ATOM,mktime(0,0,0,10,3,1975)) . "<br /><br />");
7. ?>

```

ผลลัพธ์

```

Thursday
Thursday 09th of August 2012 05:43:01 PM
Oct 3,1975 was on a Friday
Thu, 09 Aug 12 17:43:01 +0700
1975-10-03T00:00:00+07:00

```

- **sprintf()** เป็นฟังก์ชันใส่สตริงให้ตัวแปร ตามรูปแบบที่กำหนด

ตัวอย่างเช่น ต้องการใส่ 0 ข้างหน้าตัวเลขเพื่อให้ตัวเลขมี 5 หลัก 00001 , 00009 , 00010

Syntax

1. `sprintf(format , arg)`

format คือ รูปแบบที่ต้องการ

รูปแบบที่น่าสนใจคือ

%% - รูปแบบ %

%b - เลขฐานสอง

%c - ตัวอักษรที่สอดคล้องกับ ASCII value

%d - เลขฐาน 10 แบบมีเครื่องหมาย

%e - สัญลักษณ์แบบวิทยาศาสตร์ เช่น

1.2e+2

%u - เลขฐาน 10 แบบไม่มีเครื่องหมาย

%f - เลขทศนิยมแบบมี local settings

%F - เลขทศนิยมแบบไม่มี local settings

%o - เลขฐาน 8

%s - String

%x - เลขฐาน 16 ตัวอักษรเล็ก

%X - เลขฐาน 16 ตัวอักษรใหญ่

arg คือ ข้อความที่ใส่ให้ format

ตัวอย่าง

```
1. <?php
2.   $id = 9; // ตัวเลข
3.   $h = 5; // จำนวนหลัก
4.   echo sprintf("%0".$h."d", $id);
5.   //output = 00009
6. ?>
```

- **number_format()** เป็น function ที่ทำงานเกี่ยวกับการจัดรูปแบบตัวเลข เช่น การใส่ตัวคั่นหลักพัน จำนวนจุดทศนิยม เป็นต้น

Syntax

```
1. string number_format ( float $number , int $decimals = 0 , string $dec_point = "."  
    , string $thousands_sep = "," )
```

\$number คือ ตัวเลขที่อาจมีหรือไม่มีทศนิยมก็ได้ เช่น 12345.67

\$decimals คือ จำนวนทศนิยม ค่าเริ่มต้นคือ 0

\$dec_point คือ สัญลักษณ์จุดทศนิยม ค่าเริ่มต้นคือ .

\$thousands_sep คือ ตัวคั่นหลักพัน ค่าเริ่มต้นคือ ,

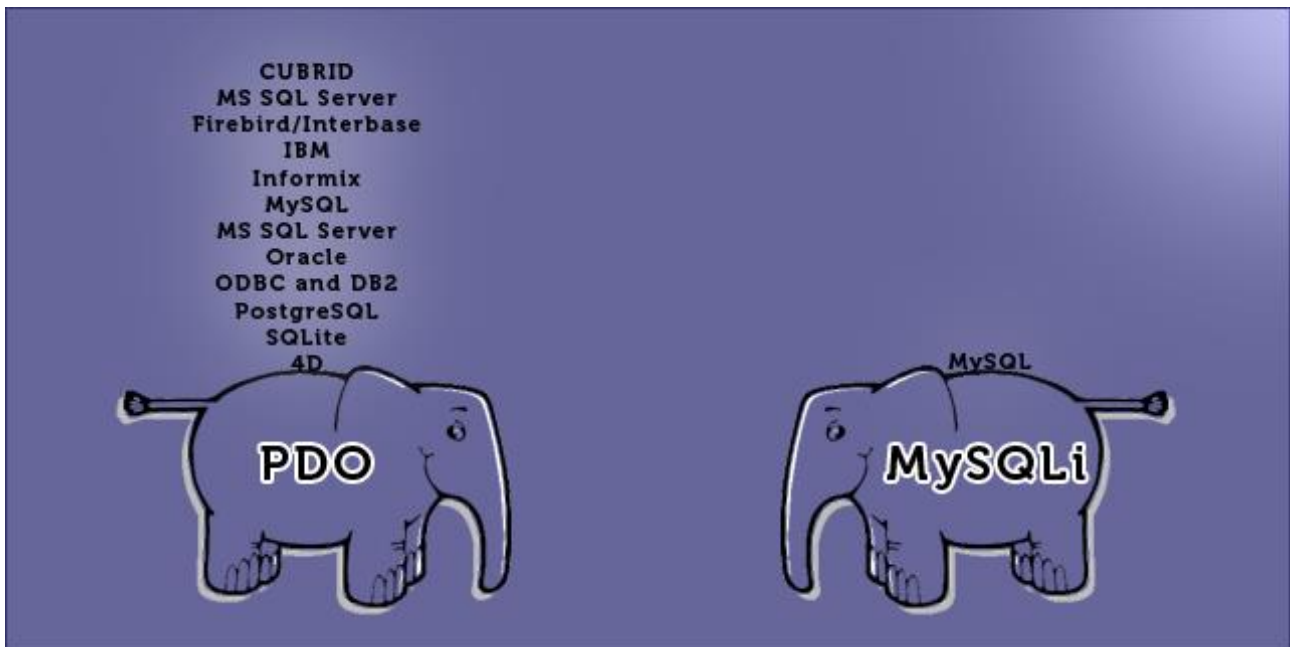
ตัวอย่าง

```
1. <?php  
2. $number = 1234567.89;  
3. echo number_format($number)."<br />";  
4. echo number_format($number, 2)."<br />";  
5. echo number_format($number, 3)."<br />";  
6. echo number_format($number, 2, '-', '-')."<br />";  
7. ?>
```

ผลลัพธ์

```
1,234,567  
1,234,567.89  
1,234,567.890  
1-234-567-89
```

PHP Data Object (PDO)



PDO คือ Extension หรือส่วนเสริมของ PHP ย่อมาจากคำว่า PHP Data Object ซึ่งเป็น Object ที่ใช้ในการเชื่อมต่อกับฐานข้อมูลได้หลากหลายโดยที่เราไม่ต้องมานั่งเรียนรู้การใช้งานฐานข้อมูลแต่ละฐาน ที่สำคัญคือใช้คำสั่งเดียวสามารถทำงานได้กับฐานข้อมูลหลายๆ แบบได้นั่นเอง เช่นคำสั่งเรียกข้อมูล `$obj->fetch()` หากมีการเปลี่ยนฐานข้อมูลก็ยังคงใช้คำสั่งเดิมไม่ได้เปลี่ยน จึงทำให้ PDO มีความยืดหยุ่นและสะดวกมากยิ่งขึ้น

PHP เวอร์ชัน 5.5 ขึ้นไปได้ยกเลิกการเชื่อมต่อ MySQL แบบเดิมแล้วคือการใช้ Function `mysql_connect()` และ `mysql_xxxx()` ทั้งหมด แต่จะเปลี่ยนไปใช้ MySQLi หรือ MySQL Improved แทนซึ่งมีความปลอดภัยมากกว่า

แต่อย่างไรก็ตามเราควรมาเริ่มใช้งาน PDO มากกว่า เนื่องจากการทำงานเป็นลักษณะ Object ซึ่ง PHP ก็รองรับการเขียนโปรแกรมแบบ Object มากยิ่งขึ้น และนอกจากนั้นจำทำให้เราพัฒนาตัวเองให้สามารถเขียนโปรแกรมในระดับที่สูงขึ้นได้เช่นกัน

PHP Data Object (Syntax)

<https://nakomah.com/article/view/php-dataobject-syntax>

Connect Database – เชื่อมต่อฐานข้อมูล

```
1. ob_start();
2. session_start(); //เริ่มต้น session
3.
4. $db_name = "database_name"; //ชื่อฐานข้อมูล
5. $db_host = "localhost";
6. $db_user = "root"; //ชื่อuser
7. $db_pass = ""; //ชื่อรหัสผ่าน
8.
9. try{
10.  $db_con = new PDO("mysql:host={$db_host};dbname={$db_name}", $db_user, $db_pass);
11.  $db_con->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
12.  $db_con->exec("set names utf8");
13. }
14. catch(PDOException $e){
15.  echo $e->getMessage();
16. }
```

Insert Syntax – เพิ่มข้อมูล

```
1. $stm = $db_con-
  >prepare("INSERT INTO table_name (col_1, col_2) VALUES (:col_1, :col_2)");
2. $stm->bindParam("col_1", $_POST['col_1']);
3. $stm->bindParam("col_2", $_POST['col_2']);
4.
5. $stm->execute();
```

Update Syntax – เปลี่ยนแปลงข้อมูล

```
1. $stm = $db_con-
  >prepare("UPDATE table_name SET col_1 = :col_1, col_2 = :col_2 WHERE id = :id");
2. $stm->bindParam("col_1", $_POST['col_1']);
3. $stm->bindParam("col_2", $_POST['col_2']);
4. $stm->bindParam("id", $_GET['id']);
5.
6. $stm->execute();
```

Delete Syntax

```
1. $stm = $db_con->prepare("DELETE FROM table_name WHERE id = :id ");
2. $stm->bindParam(':id', $_GET["id"]);
3.
4. $result = $stm->execute();
```

Select Syntax – ดึงข้อมูลแบบไม่มีเงื่อนไข

```
1. $stmt = $db_con->prepare("SELECT * FROM table_name ORDER BY id DESC");
2. $stmt->execute();
3. while ($rows = $stmt->fetch(PDO::FETCH_ASSOC)) {
4.     echo $rows['id'];
5. }
```

Select Syntax – ดึงข้อมูลแบบมีเงื่อนไข

```
1. $stmt = $db_con->
    >prepare("SELECT * FROM table_name WHERE id = :id ORDER BY id DESC");
2. $stm->bindParam(':id', $_GET["id"]);
3.
4. $stmt->execute();
5. while ($rows = $stmt->fetch(PDO::FETCH_ASSOC)) {
6.     echo $rows['id'];
7. }
```

Select Syntax - การเรียกข้อมูล 1 แถว

```
1. $stmt = $db_con->prepare("SELECT * FROM table_name WHERE id = :id ");
2. $stmt->bindParam(':id', $_GET["id"]);
3.
4. $stmt->execute();
5. $row = $stmt->fetch(PDO::FETCH_ASSOC);
6. echo $rows['id'];
```

Select Syntax - ดึงข้อมูลแสดงผลในรูปแบบ JSON (จะใช้สำหรับการส่งข้อมูลในรูปแบบของ API)

```
1. $response = array();
2. $stmt = $db_con->prepare("SELECT * FROM table_name ORDER BY id DESC");
3. $stmt->execute();
4. while ($rows = $stmt->fetch(PDO::FETCH_ASSOC)) {
5.     $response[] = $rows;
6. }
7.
8. echo json_encode($response);
```

การติดตั้ง Git

- Windows ดาวน์โหลดตัวติดตั้งได้จาก <https://git-scm.com/downloads>
- Mac OS X Yosemite or later ติดตั้งผ่าน Terminal ให้พิมพ์ `$ git version`
- Linux ติดตั้งผ่าน Terminal ให้พิมพ์ `$ sudo apt-get install git`

Git Configurations

ก่อนจะเริ่มใช้งาน Git ได้นั้น จำเป็นต้องตั้งค่าการใช้งานเบื้องต้นก่อน ในบทความนี้จะใช้โปรแกรม Git Bash ที่ติดตั้งมาพร้อม Git for Windows

- ตั้งชื่อผู้ใช้งาน "user.name"

```
$ git config --global user.name "Taveevut Nakomah"
```

- ตั้งอีเมลของผู้ใช้งาน "user.email"

```
$ git config --global user.email "nakomah.web@gmail.com"
```

- ตรวจสอบว่าที่ตั้งค่าถูกต้องหรือไม่

```
$ git config --global --list
```

ผลลัพธ์

```
user.name=Taveevut Nakomah
user.email=nakomah.web@gmail.com
```

หรือ

```
$ cat ~/.gitconfig
```

ผลลัพธ์

```
[user]
user.name=Taveevut Nakomah
user.email=nakomah.web@gmail.com
```