

ตัวดำเนินการ

Operator



ตัวดำเนินการ

ตัวดำเนินการในภาษา PHP

ในบทนี้ คุณจะได้เรียนรู้เกี่ยวกับตัวดำเนินการในภาษา PHP เราจะให้คุณรู้จักตัวดำเนินการแต่ละประเภทและวิธีการใช้งานกับการเขียนโปรแกรมในรูปแบบต่างๆ ตัวดำเนินการ คือเครื่องหมายหรือสัญลักษณ์ที่กำหนดขึ้นสำหรับใช้ในการจัดการกับตัวแปรและค่าคงที่ หรือสร้าง **Expression** ขึ้นและนำไปใช้ในการเขียนโปรแกรม ตัวดำเนินการในภาษา PHP มีหลายรูปแบบซึ่งในบทนี้

เนื้อหาได้แบ่งย่อยออกเป็นดังนี้

- Assignment operator
- Arithmetic operators
- Compound assignment
- Comparison Operators
- Logical operators
- Bitwise operators
- String Concatenation Operator
- Ternary Operator
- Operators' precedence



Assignment operator

ตัวดำเนินการกำหนดค่า (Assignment operator) คือตัวดำเนินการที่ใช้สำหรับกำหนดค่าให้กับตัวแปรหรือค่าคงที่ เครื่องหมายเท่ากับ = ใช้เป็นสัญลักษณ์ของตัวดำเนินการนี้ การทำงานของตัวดำเนินการนี้จะนำค่าจากด้านขวามาใส่ตัวแปรทางด้านซ้าย

```
$x = 1;  
$y = $x + 5;  
$text = "PHP language";
```

ในตัวอย่างเป็นการใช้งานตัวดำเนินการกำหนดค่าในการกำหนดค่าให้กับตัวแปรในภาษา PHP เราได้กำหนดค่า 1 ให้กับตัวแปร \$x เรากำหนดค่าให้กับตัวแปร \$y ในรูปแบบของ Expression และกำหนด String ให้กับตัวแปร \$text

Arithmetic operators

ตัวดำเนินการทางคณิตศาสตร์ (Arithmetic operators)

เป็นตัวดำเนินการในการหาผลลัพธ์จากการกระทำทางคณิตศาสตร์ เช่น การบวก การลบ การคูณ และหารหาร โดยมีตัวเลขเป็น Operand และจะได้ผลลัพธ์สุดท้ายออกมาค่าเดียว นี่เป็นตารางตัวดำเนินการทางคณิตศาสตร์ในภาษา PHP

Symbol	Name	Example
+	Addition	$\$c = \$a + \$b$
-	Subtraction	$\$c = \$a - \$b$
*	Multiplication	$\$c = \$a * \$$
/	Division	$\$c = \$a / \$b$
%	Modulo	$\$c = \$a \% \$b$

ใน 4 ตัวดำเนินการแรกนั้นเป็นตัวดำเนินการพื้นฐานในทางคณิตศาสตร์ แต่ในตัวดำเนินการสุดท้าย Modulo นั้นเป็นการหารแบบเอาเศษ ต่อไปเป็นตัวอย่างการใช้ตัวดำเนินการแบบต่างๆ

```
<?php
$price = 150;
$tax_rate = 10;

$tax_price = $price * ($tax_rate / 100);
$total_price = $price + $tax_price;echo "Item with price $price\n";

echo "Tax rate $tax_rate%\n";
echo "Price increased by tax $tax_price\n";
echo "Total price = $total_price\n";
?>
```

ในตัวอย่างเป็นโปรแกรมในการคำนวณหาภาษามูลค่าเพิ่มจากการ

ซื้อสินค้า

โดยเรามีราคาของสินค้าที่ยังไม่รวมภาษีและอัตราของภาษี

และคำนวณสำหรับราคาทั้งหมดที่ต้องจ่ายหลังจากรวมกับภาษีแล้ว

```
$price = 150;
$tax_rate = 10
```

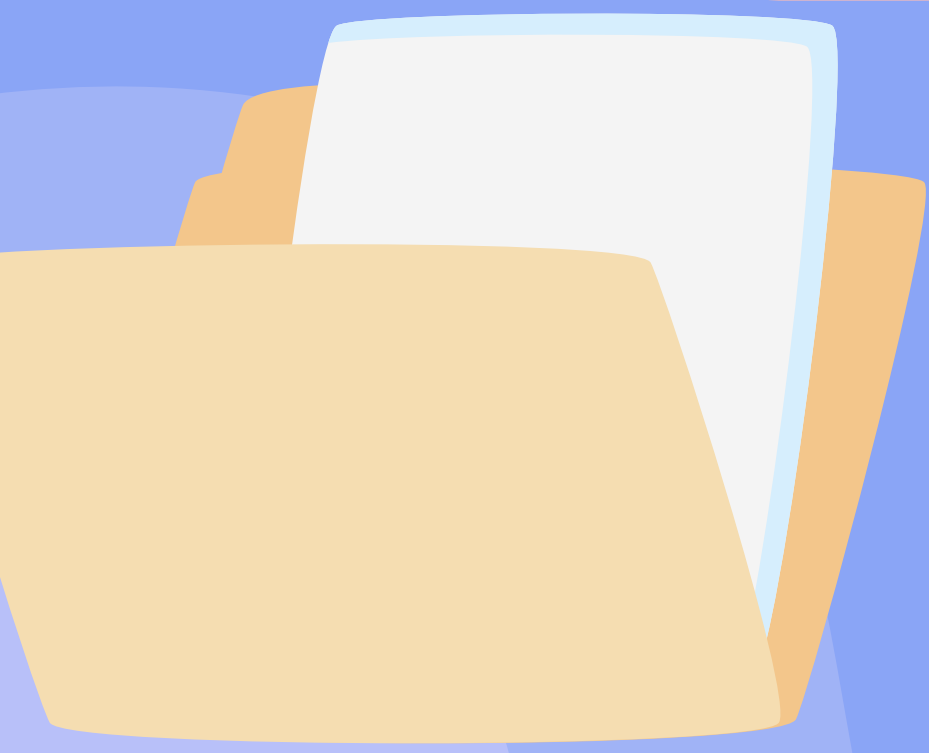
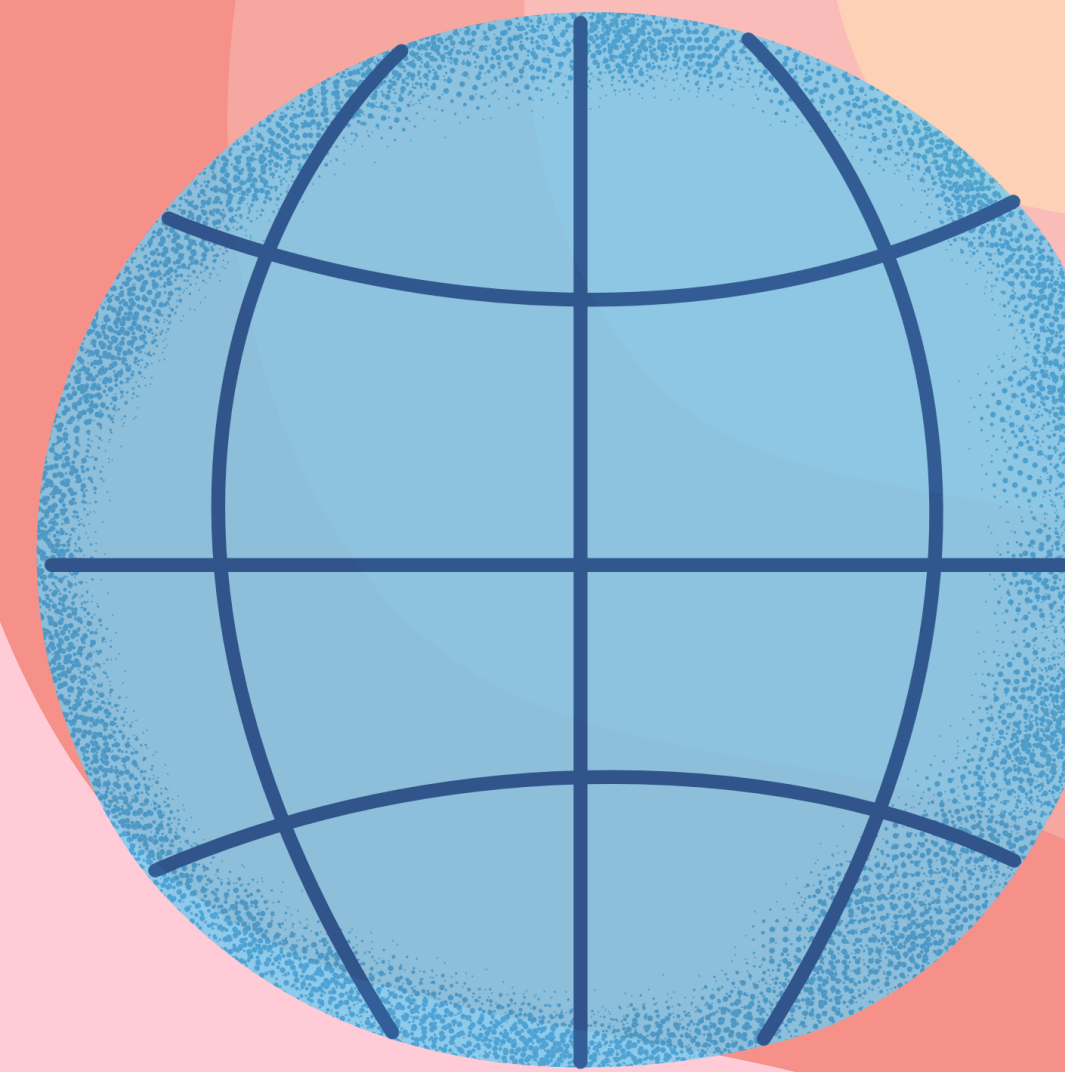
ตัวแปร `$price` นั้นเป็นราคาสินค้าที่ยังไม่ได้รวมกับภาษี และตัวแปร `$tax_rate` เป็นอัตราภาษีต่อ 100% และคุณสามารถเปลี่ยนค่าของตัวแปรเหล่านี้ได้

```
$tax_price = $price * ($tax_rate / 100);  
$total_price = $price + $tax_price;
```

ในตัวแปร `$tax_price` เป็นการคำนวณหาราคาที่เพิ่มขึ้นมาจากภาษี และ `$total_price` เป็นการหาราคาทั้งหมดเมื่อรวมกับภาษีแล้ว

```
Item with price 150  
Tax rate 10%  
Price increased by tax 15  
Total price = 165
```

ผลลัพธ์ของโปรแกรม ซึ่งอธิบายได้ว่าเมื่อคุณซื้อของในราคา 150 และมีภาษี 10% ราคาทั้งหมดที่คุณจะต้องจ่ายหลังจากรวมภาษีแล้วคือ 165



Bitwise operators

ตัวดำเนินการระดับบิต (Bitwise operators) คือตัวดำเนินการที่มีการกระทำกันในระดับบิตของข้อมูลซึ่งมีแค่ **1** และ **0** นี้เป็นโครงสร้างของภาษาที่ถ่ายทอดมาจากภาษา **C** ในการทำงานของตัวดำเนินการระดับบิตเป็นการนำคู่ของบิตมากระทำต่อกันในทางตรรกศาสตร์และได้ผลลัพธ์เป็นบิตใหม่ ส่วนมากแล้วมันมักจะใช้กับการเขียนโปรแกรมในระดับต่ำ

Symbol	Name	Example
&	And	$\$c = \$a \& \$b$
	Or	$\$c = \$a \$b$
^	Xor	$\$c = \$a \wedge \$b$
~	Not	$\$c = \sim \b
<<	Shift left	$\$c = \$a \ll \$b$
>>	Shift right	$\$c = \$a \gg \$b$

Compound assignment

Compound assignment

Compound assignment operators คือตัวดำเนินการที่รวมระหว่างตัวดำเนินการทางคณิตศาสตร์และตัวดำเนินการกำหนดค่า มันใช้สำหรับอัปเดตข้อมูลที่อ้างอิงจากข้อมูลในปัจจุบัน หรือกล่าวอีกนัยหนึ่ง มันเป็นรูปแบบที่สั้นกว่าของตัวดำเนินการทางคณิตศาสตร์

ตัวอย่างการใช้งาน Compound assignment operators ในภาษา PHP

```
$floor = 0;
echo "Building is being constructed.\n";

$floor += 3;
echo "First day finished $floor floors.\n";

$floor += 5;
echo "Second day finished $floor floors.\n";

$floor += 4;
echo "Third day finished $floor floors.\n";

$floor += 2;
echo "Forth day finished $floor floors .\n";

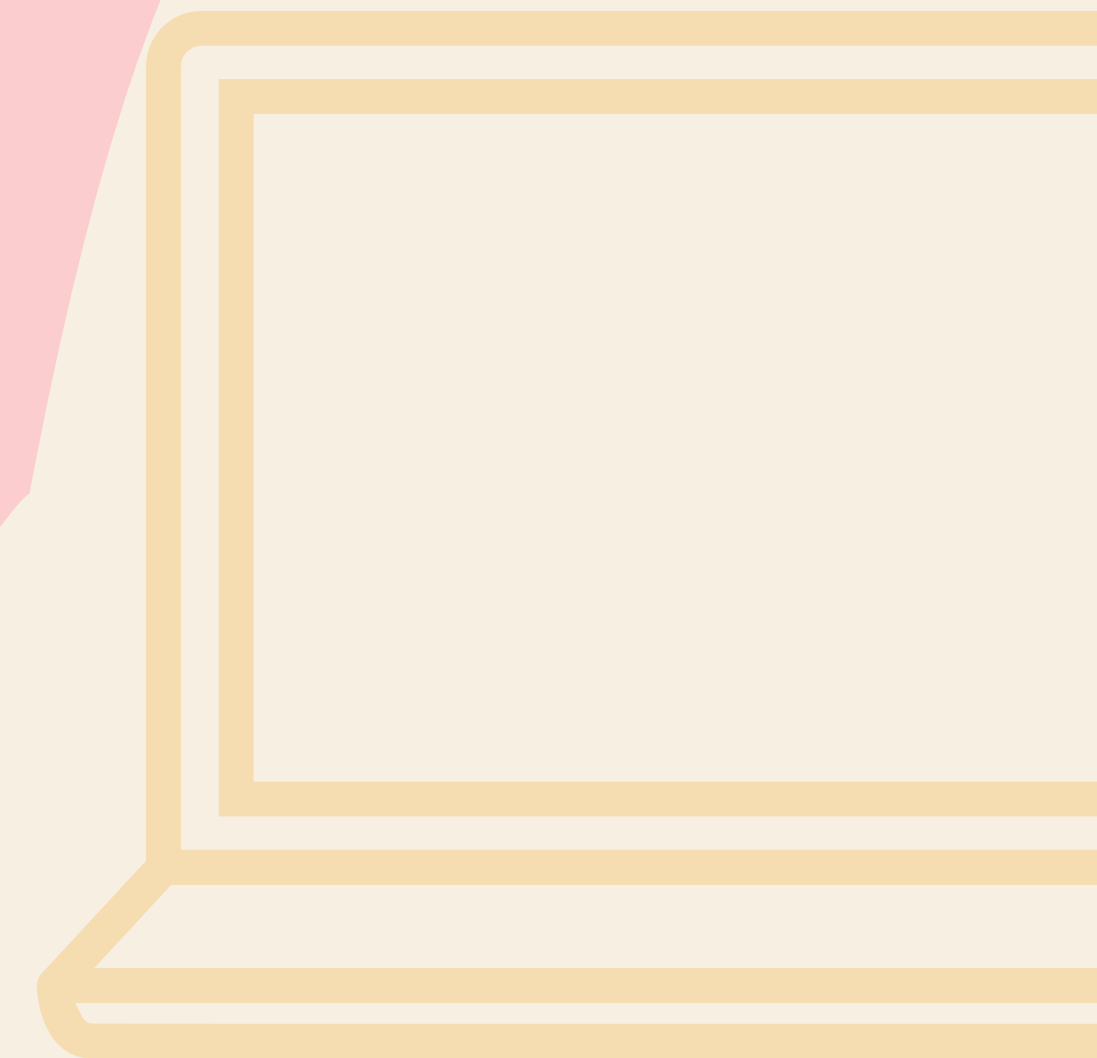
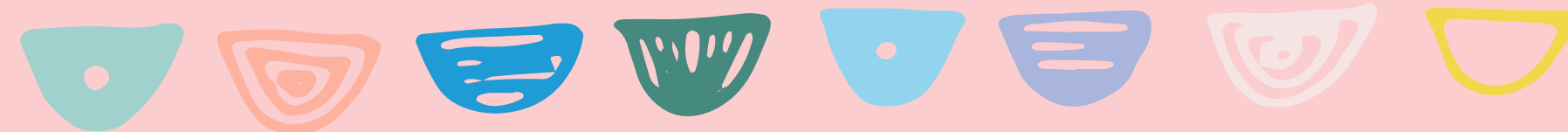
?>
```





ในตัวอย่าง เป็นโปรแกรมแสดงการทำงานของการทำงานของการสร้างตึก
ตัวแปร **\$floor** สำหรับเก็บค่าจำนวนชั้นของตึกที่ทำการสร้าง
เสร็จ แล้วในแต่ละวันที่ทำการสร้างตึกเสร็จ จะนำจำนวนชั้นใหม่ไปบวกเพิ่มกับวันก่อนหน้า เช่น
ในวันแรกสร้างเสร็จไป 3 ชั้นจะได้ **\$floor += 3;** ซึ่งตัวดำเนินการเหล่านี้เป็นรูปแบบที่สั้นขึ้นจาก **\$floor =
\$floor + 3;** นี่คือการลดการใช้ **Operand** ที่ไม่จำเป็นออกไป ในการเขียนคุณสมบัตินี้สามารถใช้แบบไหนก็ได้

```
Building is being constructed.  
First day finished 3 floors.  
Second day finished 8 floors.  
Third day finished 12 floors.  
Forth day finished 14 floors.
```



ในบทนี้ คุณได้เรียนรู้เกี่ยวกับตัวดำเนินการในเบื้องต้นในภาษา **PHP** เช่น ตัวดำเนินการกำหนดค่า ตัวดำเนินการทางคณิตศาสตร์ ตัวดำเนินการระดับบิต ตัวดำเนินการ **Compound assignment** ในบทต่อไป จะเป็นเนื้อหาของตัวดำเนินการที่ใช้ในการสร้าง **Expression** หรือใช้กับคำสั่งควบคุมของโปรแกรม และลำดับความสำคัญของตัวดำเนินการ

