บทที่ 8
คอนสตรัคเตอร์ (Constructor) และ ดีสตรัคเตอร์ (Destructor)

ในการเขียนโปรแกรมเชิงวัตถุ เมธอดที่มีความสำคัญในการใช้งานอีกประเภท ที่เมื่อมีการใช้สร้างเมธอดแบบคอนสตรัคเตอร์ จะมีการเรียกใช้ก่อนเสมอ หรือดีสตรัคเตอร์ จะมีการเรียกใช้เมธอดสุดท้ายเสมอ เมื่อมีการใช้งานผ่านออบเจ็กต์ หรืออินสแตนต์ นั่นเอง

8.1 คอนสตรัคเตอร์

คอนสตรัคเตอร์ (constructor) เป็นเมธอดที่มีชื่อเหมือนกับชื่อคลาส ในการพัฒนาโปรแกรมนั้น จะใช้คอนสตรัคเตอร์สำหรับเป็นตัวกำหนดค่าเริ่มต้นให้กับอินสแตนท์โดยเฉพาะ เมื่อใดก็ตามที่มีการสร้างตัวอินสแตนท์ด้วยคำสั่ง new คอนสตรัคเตอร์จะถูกเรียกให้ทำงานโดยอัตโนมัติ ซึ่งมีโครงสร้างการทำงานดังตัวอย่าง ตารางที่ 8.1 หรือตารางที่ 8.2

ตารางที่ 8.1 การสร้างคอนสตรัคเตอร์

แถวที่	ประโยคคำสั่ง
1	<?
2	class MyCon
3	{
4	public function MyCon()
5	{
6	echo "Constructor OK";
7	}
8	}
9	$obj=new MyCon;
10	?>

ตารางที่ 8.2 การสร้างคอนสตรัคเตอร์ด้วยคำสั่ง __construct

แถวที่	ประโยคคำสั่ง
1	<?
2	class MyCon
3	{
4	public function __construct()
5	{
6	echo "Constructor OK";
7	}
8	}
9	$obj=new MyCon;
10	?>

ผลลัพธ์ จากตารางรูปที่ 8.1 และ รูปที่ 8.2

Constructor OK

8.1.1 การทำงานของคอนสตรัคเตอร์

เมื่อมีการสร้างคลาสขึ้นมา และเรียกใช้งานผ่านออบเจ็กต์ หรือเรียกอีกแบบหนึ่งว่า อินสแตนท์ คอนสตรัคเตอร์จะถูกเรียกใช้งานโดยอัตโนมัติ โดยไม่ต้องเรียกใช้เมธอดขึ้นมาทุกครั้ง ผ่านออบเจ็กต์ หรืออินสแตนท์นั่นเอง ดังตัวอย่างที่ 8.1

ตารางที่ 8.2 การสร้างคอนสตรัคเตอร์

แถวที่	ประโยคคำสั่ง
1	class MyCon
2	{
3	public $name;
4	public $surname;
5	
6	public function MyCon()
7	{
8	$this->name="นายกอเอ๋ย";
9	echo $this->name;

```
10          }
11  }
12  $obj=new MyCon;
13  ?>
```

ผลลัพธ์

```
นายกอเอ๋ย
```

การสร้างคอนสตรัคเตอร์ต้องกำหนดชื่อเป็นชื่อเดียวกันกับคลาส (บรรทัดที่ 4) และเมื่อมีการสร้างอินสแตนท์ (บรรทัดที่ 12) และทำการรันโปรแกรมจะได้ผลลัพธ์ดังตัวอย่าง ซึ่งจะเห็นได้ว่าไม่จำเป็นต้องเรียกใช้เมธอด ตัวคอนสตรัคเตอร์จะทำงานโดยอัตโนมัติทันที

ในการสร้างคอนสตรัคเตอร์สามารถกำหนดชื่อเดียวกันกับคลาส หรือจะใช้คำสั่ง __construct (อันเดอร์สกอร์ 2 ครั้ง และตามด้วยคำว่า construct) ในภาษา php ก็สามารถเข้าใจว่าเป็นคอนตรัคเตอร์เหมือนกัน ดังตัวอย่างที่ 8.3

8.1.2 การส่งผ่านค่าพารามิเตอร์ของคอนสตรัคเตอร์

จากคุณลักษณะของคอนสตรัคเตอร์ นั่นคือ เป็นเมธอดอย่างหนึ่ง แต่จะแตกต่างกับเมธอดทั่วไปคือ จะถูกเรียกใช้โดยอัตโนมัติ และชื่อจะต้องเป็นชื่อเดียวกับชื่อคลาส หรือ __construct แต่เมธอดทั่วไปจะถูกกำหนดขึ้นมาใช้งานเอง ตั้งชื่อตามที่ต้องการ

จากข้างต้นได้คอนตรัคเตอร์ก็คือ เมธอด ดังนั้นสามารถที่จะมีการส่งผ่านค่าพารามิเตอร์ได้เหมือนกันเมธอดทั่วไป (ซึ่งกล่าวไว้ในบทที่ 6) ดังตัวอย่างที่ 8.3

ตารางที่ 8.3 การส่งผ่านค่าพารามิเตอร์ในคอนสตรัคเตอร์

แถวที่	ประโยคคำสั่ง
1	<?
2	class MyCon
3	{
4	private $name;
5	private $surname;
6	
7	public function __construct($p_name="",$p_surname="")
8	{

```
9                       $this->name=$p_name;
10                      $this->surname=$p_surname;
11              }
12              public function getValue()
13              {
14                      return $this->name." ".$this->surname;
15              }
16      }
17      $obj=new MyCon("กอเอ๋ย","กอไก่");
18      echo $obj->getValue();
19      ?>
```

ผลลัพธ์

```
กอเอ๋ย กอไก่
```

จากบรรทัดที่ 17 เมื่อมีการสร้างอินสแตนท์ขึ้นมา สามารถที่จะกำหนดค่าเริ่มต้นส่งไปยังคอนสตรัคเตอร์ได้ทันที

8.2 ดีสตรัคเตอร์ (Destructor)

คอนสตรัคเตอร์จะทำงานทันทีเมื่อมีการเรียกใช้งานคลาสผ่านอินสแตนท์ และเมื่อมีการใช้งานผ่านเมธอดต่างเรียบร้อยแล้ว จะต้องมีการปิดการทำงานของการใช้งานคลาส (ในบางกรณี ปิดค่าต่างๆ หลังจากใช้งานต่างๆ เสร็จเรียบร้อยแล้ว) จะมี ดีสตรัคเตอร์ ทำหน้าที่จะปิดการทำงาน ดังตัวอย่างที่ 8.4

ตารางที่ 8.4 การทำงานของดีสตรัคเตอร์

แถวที่	ประโยคคำสั่ง
1	<?
2	class MyCon
3	{
4	private $Name;
5	private $Surname;

```
6
7           public function __construct($name="",$surname="")
8           {
9                   $this->Name=$name;
10                  $this->Surname=$surname;
11          }
12
13          public function getValue()
14          {
15                  return $this->Name." ".$this->Surname;
16          }
17
18          public function __destruct()
19          {
20                  echo " END";
21          }
22
23  }
24  $obj=new MyCon("กอเอ๋ย","กอไก่");
25  echo $obj->getValue();
26  ?>
```

ผลลัพธ์

```
กอเอ๋ย กอไก่ END
```