

# OOP in PHP ตอนที่ 2

สวัสดีครับ สำหรับในตอนี่ 2 นี้เราจะนำเข้าสู่เรื่องที่ยากขึ้นอีกระดับจากตอนที่แล้ว เพราะเราได้ปรับพื้นฐานในการเขียนโปรแกรมแบบเชิงวัตถุมาบ้างแล้ว และได้ทราบถึงคำที่มักถูกใช้ในการเขียนโปรแกรมแบบเชิงวัตถุซึ่งได้แก่

- แม่แบบ (Class)
- ลูกแบบ (Sub-Class)
- วัตถุ (Object)
- การส่งข้อความ (Messaging)
- ข้อความ (Message)
- การห่อหุ้มข้อมูล(Encapsulation)
- การสืบทอด (Inheritance)
- การทำหลายรูปแบบ (Polymorphism)
- คุณสมบัติ (Properties)
- แบบแผนพฤติกรรม (Method Behavior)

โดยจากตอนที่แล้วนั้นก็เขียนโปรแกรมแบบเบื้องต้นไปแล้วบ้างบางส่วน แต่หลาย ๆ ท่านที่ได้ลองเขียนดูตามตัวอย่างแล้วนั้นอาจจะยังไม่ทราบถึงประโยชน์ และเห็นภาพในการเอาไปใช้งานมากนัก ในตอนนี้ผู้เขียนจะนำเข้าสู่การนำไปใช้งานจริงมากขึ้น

## พฤติกรรมพื้นฐาน (Constructor Method Behavior)

ก่อนที่เราจะนำไปใช้งานจริง เราควรจะรู้ถึงการสร้างพฤติกรรมพื้นฐานของแม่แบบนั้น ๆ ในบางครั้งแม่แบบนี้จะถูกนำมาสร้างวัตถุได้ ต้องกำหนดคุณสมบัติบางประการให้อยู่ในค่าที่ถูกต้องเสมอตั้งแต่วัตถุถูกสร้างขึ้นมา

การสร้างพฤติกรรมพื้นฐานบน PHP นั้นก็เหมือนกับการสร้างพฤติกรรมโดยทั่วไปเพียงแต่ใช้ชื่อพฤติกรรมนั้นให้เหมือนกับชื่อแม่แบบ ดังตัวอย่างชุดคำสั่งต่อไปนี้

```
1. <?php
2. class class_name {
3.     function class_name ([parameter]) {
4.         ....
5.         [return ....];
6.     }
7. }
8. ?>
```

ชุดคำสั่งที่ 1

ผู้เขียนจึงขอยกตัวอย่างจากตอนที่แล้วว่า เมื่อเราสร้างวัตถุจากแม่แบบมนุษย์เราต้องกำหนดชื่อและเพศ รวมถึงกำหนดค่าเริ่มต้นของความสูง, น้ำหนัก และอายุให้มีค่าความสูง 170 ซม., น้ำหนัก 60 กก. และมีอายุ 21 ปีก่อนเสมอจากชุดคำสั่งตอนที่แล้ว เรานำมาปรับปรุงใหม่ได้ดังนี้

```
1. <?php
2. class human {
3.     var $name = "human"; // default name = "human"
4.     var $Sex= 1; // default = 1 and 1 = male, 2 = female
5.     var $height = 0; // height > 0
6.     var $weight = 0; // weight > 0
7.     var $age = 0; // age > 0
8.     function human($name, $sex){
```

```

9.     $this->setName($name);
10.    $this->setSex($sex);
11.    $this->setHeight(170);
12.    $this->setWeight(60);
13.    $this->setAge(21);
14.    }
15.    function setName($value = 'human'){
16.        $this->name = $value;
17.    }
18.    function setSex($value = 1){
19.        $this->Sex = ($value==1 || $value==2 ? $value : 1);
20.    }
21.    function setHeight($value = 1){
22.        $this->height = ($value > 0 ? $value : 1 );
23.    }
24.    function setWeight($value = 1){
25.        $this->weight = ($value > 0 ? $value : 1 );
26.    }
27.    function setAge($value = 1){
28.        $this->age = ($value > 0 ? $value : 1 );
29.    }
30.    function getName(){
31.        return $this->name;
32.    }
33.    function getSex(){
34.        return ($this->Sex == 1 ? 'male' : 'female');
35.    }
36.    function getHeight(){
37.        return $this->height;
38.    }
39.    function getWeight(){
40.        return($this->Sex == 2 ? 'No Answer':$this->weight);
41.    }
42.    function getAge(){
43.        return($this->Sex == 2 ? 'No Answer':$this->age);
44.    }
45.}
46.??>

```

### ชุดคำสั่งที่ 2

จากชุดคำสั่งที่ 1 นั้นบรรทัดที่ 8 - 14 เราได้ทำการเพิ่มพฤติกรรมมาตรฐาน โดยเมื่อเราสามารถวัตถุขึ้นมาเราต้องใส่ข้อมูลมาตรฐานลงไปด้วย ซึ่งในที่นี้ก็คือชื่อและเพศ ตามลำดับ โดยการนำไปใช้งานก็จะได้นี้

```

1. $somsak = new human('Somsak', 1);
2. echo 'Name : '.$somsak->getName().'<br />';
3. echo 'Sex : '.$somsak->getSex().'<br />';
4. echo 'Height : '.$somsak->getHeight().'<br />';
5. echo 'Weight : '.$somsak->getWeight().'<br />';
6. echo 'Age : '.$somsak->getAge().'<br />';

```

### ชุดคำสั่งที่ 3

ซึ่งจากตัวอย่างชุดคำสั่งที่ 3 เราจะได้ผลการทำงานที่ 1

Name : Somsak  
Sex : male  
Height : 170  
Weight : 60  
Age : 21

### ผลการทำงานที่ 1

จากชุดคำสั่งที่ 2 และ 3 และผลการทำงานที่ 1 นั้นจะเห็นได้ว่าเมื่อเราสร้างวัตถุขึ้นมา 1 วัตถุตัวพฤติกรรมมาตรฐานจะทำการกำหนดค่าเริ่มต้นของวัตถุนั้น ๆ ให้กับที่ รวมไปถึงค่าบางค่าที่จำเป็นต้องมีในการสร้างวัตถุขึ้นมา ซึ่งทำให้เราสามารถนำไปประยุกต์ใช้ได้หลาย ๆ งานที่ต้องการใช้ค่าพื้นฐานที่ซ้ำ ๆ กันตลอดเวลา ซึ่งช่วยให้เราไม่ต้องมาใส่ค่าต่าง ๆ ตลอดเวลาให้เป็นค่ามาตรฐานเสมอ ๆ

## นำแนวคิดเชิงวัตถุไปใช้งานจริง

จากตัวอย่างที่แล้วที่เราเขียนโปรแกรมเล็ก ๆ แล้วสร้างวัตถุขึ้นมา 2 วัตถุให้มีคุณสมบัติแตกต่างกันในเรื่องเพศ ทำให้มีพฤติกรรมในการแสดงออกของตัววัตถุแตกต่างกันตามที่ได้กล่าวไปแล้วในตอนที่แล้ว

คราวนี้เรามาดูเขียนโปรแกรมเล็ก ๆ 1 ตัวที่เอาไว้ใช้ในการบอกวันและเวลาโดยอาศัยฟังก์ชันพื้นฐานของ PHP เป็นคุณสมบัติภายใน โดยที่เราสร้างวัตถุแล้ว เราไม่จำเป็นต้องรู้ว่าภายในทำงานอย่างไร เพียงแต่สั่งให้ส่งค่านั้นออกมาก็เพียงพอต่อการนำไปใช้งานได้

เรามาดูกำหนดลักษณะของแม่แบบกันเสียก่อนว่ามีลักษณะแบบใด โดยแม่แบบเราจะให้ชื่อว่า **thai\_datetime** โดยเป็นแม่แบบที่ใช้ในการแสดงผลวันที่ และเวลา โดยที่แสดงผลให้เป็นไปตามแบบไทย คือชื่อเดือนเป็นภาษาไทยและเป็นปี พ.ศ. โดยพื้นฐานของวันที่ยังคงเป็นปี ค.ศ. ทำให้ง่ายต่อการนำไปใช้ โดยที่มีพฤติกรรมต่าง ๆ ดังต่อไปนี้

- |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ol style="list-style-type: none"><li><b>พฤติกรรมพื้นฐาน (thai_datetime)</b><br/>จะมีการเรียกใช้พฤติกรรมตั้งค่าวันและเวลาตั้งแต่เริ่มสร้างวัตถุทันทีโดยในพฤติกรรมพื้นฐานนี้มีการรับค่าเริ่มต้นเข้ามาด้วยหรือไม่ก็ได้</li><li><b>ตั้งค่าวันและเวลา (set_datetime)</b><br/>ถ้าไม่ใส่วันที่เข้ามาจะให้ผลเป็นวันที่และเวลาปัจจุบัน แต่ถ้ามีการใส่วันที่เข้ามาจะแสดงผลวันที่และเวลาในขณะนั้นเป็นวันที่แบบไทย</li><li><b>แปลงค่าวันที่และเวลา (cast_datetime)</b><br/>เป็นการแปลงค่าตัวเลขที่เป็นรูปแบบของวันที่ต่าง ๆ ให้เป็นค่าของวันที่และเวลาตามที่กำหนดไว้ซึ่งได้แก่<br/><br/><b>d</b> คือวันที่    <b>m</b> คือเดือน    <b>y</b> คือปี<br/><b>H</b> คือชั่วโมง    <b>M</b> คือนาที    <b>S</b> คือวินาที</li></ol> | <ol style="list-style-type: none"><li><b>ชื่อย่อของเดือน (short_month)</b><br/>เมื่อใส่ลำดับของเดือนเข้าไปในพฤติกรรมนี้แล้วส่งชื่อย่อของเดือนออกมาให้เรา</li><li><b>ชื่อเต็มของเดือน (long_month)</b><br/>เมื่อใส่ลำดับของเดือนเข้าไปในพฤติกรรมนี้แล้วส่งชื่อเต็มของเดือนออกมาให้เรา</li><li><b>ส่งค่าปี (year)</b></li><li><b>ส่งค่าเดือน (month)</b></li><li><b>ส่งค่าวัน (day)</b></li><li><b>ส่งค่าชั่วโมง (hour)</b></li><li><b>ส่งค่านาที (minute)</b></li><li><b>ส่งค่าวินาที (second)</b></li></ol> |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

```

1. <?php
2. class thai_datetime {
3.     var $datetime;
4.     function thai_datetime($valuedate = NULL) {
5.         $this->set_datetime($valuedate);
6.     }
7.     function set_datetime($valuedate) {
8.         if(empty($valuedate))
9.             $this->datetime = date("Y-m-d H:i:s");
10.        else
11.            $this->datetime = $valuedate;
12.    }
13.    function cast_datetime($value, $type) {
14.        $value = strtotime($value);
15.        switch($type) {
16.            case "d": return date("j", $value);
17.            case "m": return date("m", $value);
18.            case "Y": return date("Y", $value);
19.            case "H": return date("H", $value);
20.            case "M": return date("M", $value);
21.            case "S": return date("S", $value);
22.        }
23.    }
24.    function short_month($m_val = NULL) {
25.        if(empty($m_val))
26.            $m_val = $this->cast_datetime($this->datetime, "m");
27.        switch($m_val) {
28.            case 1: return "ม.ค."; break;
29.            case 2: return "ก.พ."; break;
30.            case 3: return "มี.ค."; break;
31.            case 4: return "เม.ย."; break;
32.            case 5: return "พ.ค."; break;
33.            case 6: return "มิ.ย."; break;
34.            case 7: return "ก.ค."; break;
35.            case 8: return "ส.ค."; break;
36.            case 9: return "ก.ย."; break;
37.            case 10: return "ต.ค."; break;
38.            case 11: return "พ.ย."; break;
39.            case 12: return "ธ.ค."; break;
40.            default : return "ม.ค."; break;
41.        }
42.    }
43.    function long_month($m_val = NULL) {
44.        if(empty($m_val))
45.            $m_val = $this->cast_datetime($this->datetime, "m");
46.        switch($m_val) {
47.            case 1: return "มกราคม"; break;
48.            case 2: return "กุมภาพันธ์"; break;
49.            case 3: return "มีนาคม"; break;
50.            case 4: return "เมษายน"; break;
51.            case 5: return "พฤษภาคม"; break;
52.            case 6: return "มิถุนายน"; break;
53.            case 7: return "กรกฎาคม"; break;
54.            case 8: return "สิงหาคม"; break;
55.            case 9: return "กันยายน"; break;
56.            case 10: return "ตุลาคม"; break;
57.            case 11: return "พฤศจิกายน"; break;
58.            case 12: return "ธันวาคม"; break;
59.            default : return "มกราคม"; break;

```

```

60.     }
61. }
62. function year($value = NULL) {
63.     if(empty($value))
64.         return $this->cast_datetime($this->datetime,"y")+543;
65.     else return $value+543;
66. }
67. function month() {
68.     return ($this->cast_datetime($this->datetime,"m"));
69. }
70. function day() {
71.     return $this->cast_datetime($this->datetime,"d");
72. }
73. function hour() {
74.     return $this->cast_datetime($this->datetime,"H");
75. }
76. function minute() {
77.     return $this->cast_datetime($this->datetime,"M");
78. }
79. function second() {
80.     return $this->cast_datetime($this->datetime,"S");
81. }
82. }
83. ?>

```

#### ชุดคำสั่งที่ 4

จากชุดคำสั่งที่ 4 นั้นจะเห็นว่าเรานำเข้าฟังก์ชันพื้นฐานของ PHP มาใช้งานในแม่แบบอยู่มากมาย ซึ่งในงานที่มีการใช้ฟังก์ชันต่าง ๆ มาก ๆ จะทำให้ชุดคำสั่งของเรานั้นอ่านได้ยาก การนำเอามาปรับปรุงโดยการใส่ลงในแม่แบบและสร้างวัตถุนั้นก็ทำให้ดูเข้าใจง่ายมากขึ้นไปอีก โดยการใช้งานนั้นดูได้จากชุดคำสั่งที่ 5

```

1. <?php
2. $x = new thai_datetime();
3. echo $x->day() , " " , $x->short_month() , " " , $x->year();
4. echo '<br />';
5. $y = new thai_datetime("2005-1-20");
6. echo $y->day() , " " , $y->short_month() , " " , $y->year();
7. ?>

```

#### ชุดคำสั่งที่ 5

```

11 มี.ค. 2550
12 20 ม.ค. 2548

```

#### ผลการทำงานที่ 2

จากชุดคำสั่งที่ 5 นั้นจะเห็นได้ว่าเราได้สร้างวัตถุมา 2 วัตถุคือ **x** และ **y** โดยที่วัตถุ **x** นั้นไม่ได้กำหนดพฤติกรรมมาตรฐาน แต่ผลการทำงานก็ออกมาเป็นวันที่ปัจจุบัน (ณ.วันที่เขียนบทความ) ส่วนวัตถุ **y** นั้นได้กำหนดพฤติกรรมพื้นฐานไว้ว่าเป็นวันที่ “20 เดือนที่ 1 ปี 2005” ผลที่ได้ออกมาคือวันที่ “20 ม.ค. 2548”

จากตัวอย่างการนำเอาแนวคิดการเขียนโปรแกรมเชิงวัตถุมาเขียนแม่แบบวันที่นั้นจะเห็นได้ว่าเราสามารถนำเอาสิ่งที่เราต้องเขียนซ้ำ ๆ มาเขียนให้อยู่รูปที่เอาไปใช้งานได้ง่ายมากขึ้น

ในตอนต่อไปเราจะมาพูดถึงการสืบทอดแม่แบบให้มีลูกแบบเป็นทอด ๆ และการนำเอาแนวคิดนี้ไปใช้งานในด้าน การเชื่อมต่อฐานข้อมูลอย่าง MySQL ให้สามารถนำเอาวัตถุที่เป็นแม่แบบของการติดต่อกับฐานข้อมูล MySQL นั้นเอามาช่วยในการทำงานกับฐานข้อมูลให้ง่ายมากขึ้น และชุดคำสั่งที่เราใช้ดูสั้นลงอย่างเห็นได้ชัดเลยทีเดียว

# เอกสารอ้างอิง

- **PHP Manual**, <http://www.php.net>, January 2007
- **Zend PHP Certification Study Guide**, Zend Technologies, August 2004
- **Concepts of Programming Languages (7th Edition)**, Robert W. Sebesta, April 2005