

## Magic Method

تخيل إنك بتكتب كود عادي في PHP، وفجأة تلاقي فيه **Functions** بتشتغل من غير ما تستدعيها بنفسك

يعني بمجرد ما تنشئ **Object** أو تحاول تستخدم متغير مش موجود، تلاقي فيه حاجة غريبه بتحصل في الخلفية، وبيتنادى كود جاهز يخدمك. السر هنا هو حاجة اسمها **Magic Methods**، وهي واحدة من أقوى المميزات في الـ **OOP** داخل **PHP**. الدوال دي بتديك مرونة رهيبه في التعامل مع **Objects**، وبتخلي الكود أنصف وأسهل في القراءة، وكمان بتوفر طرق غير تقليدية للتحكم في البيانات.

### يعني ايه Magic Methods

هي **Functions** خاصة مسبقه التعريف في **PHP**. كل **Function** بتبدأ بـ **double underscore** `__`.  
أسمائها محجوزة (**Reserved**) ومينفعش تستخدمها كأسماء عادية.  
بتنفذ تلقائياً لما يحصل حدث معين زي: إنشاء **Object**، تدمير **Object**، استدعاء متغير أو دالة غير موجودة

### أهم الـ Magic Methods في PHP

Method Names	Return types	Condition of calling
<code>__construct()</code>	NaN	This method gets called automatically every time the object of a particular class is created. The function of this magic method is the same as the constructor in any OOP language.
<code>__destruct()</code>	NaN	As the name suggests this method is called when the object is destroyed and no longer in use. Generally at the end of the program and end of the function.
<code>__call(\$name,\$parameter)</code>	Not mandatory	This method executes when a method is called which is not defined yet.
<code>__toString()</code>	String	This method is called when we need to convert the object into a string. For example: <code>echo \$obj;</code> The <code>\$obj-&gt;__toString();</code> will be called magically.
<code>__get(\$name)</code>	NaN	This method is called when an inaccessible variable or non-existing variable is used.
<code>__set(\$name , \$value)</code>	NaN	This method is called when an inaccessible variable or non-existing variable is written.
<code>__debugInfo()</code>	array	This magic method is executed when an object is used inside <code>var_dump()</code> for debugging purposes.

## \_\_construct()

يتم استدعاؤه تلقائيًا عند إنشاء **Object**.

```
1 class MagicMethod {
2     function __construct() {
3         echo "This is the construct magic method";
4     }
5 }
6
7 $obj = new MagicMethod();
8
9 // Output => This is the construct magic method
```

## \_\_destruct()

يتم استدعاؤه تلقائيًا لما تعمل **destroy** لل **Object** أو انتهاء البرنامج.

```
1 class MagicMethod2 {
2     function __destruct() {
3         echo "This destruct magic method gets called when object destroys";
4     }
5 }
6
7 $obj = new MagicMethod2();
8
9 // Output => This destruct magic method gets called when object destroys
10
```

### `__call($name, $parameters)`

يتم استدعاؤه عند محاولة استخدام `Function` مش موجودة.

```
1 class MagicMethod3 {
2     function __call($name , $parameters){
3         echo "Name of method =>" . $name."\n";
4         echo "Parameters provided\n";
5         print_r($parameters);
6     }
7 }
8
9 // Instantiating MagicMethod
10 $obj = new MagicMethod3();
11
12 $obj->hello("Magic" , "Method");
13
14 /* Output =>
15 Name of method =>hello
16 Parameters provided
17 Array
18 (
19     [0] => Magic
20     [1] => Method
21 )
22 */
```

### `__toString()`

يتم استدعاؤه لما نحاول نطبع ال `Object` ك `String`.

```
1 class MagicMethod4 {
2     function __toString(){
3         return "You are using MagicMethod object as a String ";
4     }
5 }
6
7 $obj = new MagicMethod4();
8
9 echo $obj;
10
11 // Output => You are using MagicMethod object as a String
12
```

## \_\_get(\$name)

(Private/Protected) يتم استدعاؤه لما نحاول نقرأ متغير غير موجود أو خاص

```
1 class MagicMethod5 {
2     function __get($name){
3         echo "You are trying to get '" . $name .
4             "' which is either inaccessible
5             or non existing member";
6     }
7 }
8
9 $obj = new MagicMethod5();
10 $obj->value;
11
12 /* Output => You are trying to get 'value' which is either
13 inaccessible or non existing member
14 */
```

## \_\_set(\$name, \$value)

يتم استدعاؤه عند محاولة تعديل متغير غير موجود أو خاص.

```
1 class MagicMethod6 {
2     function __set($name , $value) {
3         echo "You are trying to modify '"
4             . $name . "' with '" . $value .
5             "' which is either inaccessible
6             or non-existing member";
7     }
8 }
9 $obj = new MagicMethod6();
10 $obj->value = "Hello";
11
12 /* Output => You are trying to modify 'value' with 'Hello' which is either inaccessible
13 or non-existing member
14 */
```

**\_\_debugInfo()**

بـيستخدم عند `var_dump()` لعرض بيانات مخصصة عن الـ `Object`

```
1 class MagicMethod7 {
2     function __debugInfo(){
3         return array("variable"=> "value");
4     }
5 }
6
7 $obj = new MagicMethod7();
8 var_dump($obj);
9
10 /* Output =>
11 object(MagicMethod)#1 (1) {
12     ["variable"]=>
13     string(5) "value"
14 }
15 */
```

الـ **Magic Methods** وسيلة قوية في PHP علشان تتحكم في سلوك الـ **Objects** بطريقة مرنة وأنيقة. هي بتوفر اختصارات ذكية، وبتخلي الكود أسهل في القراءة والصيانة.

وده بيأكد إن **OOP** في **PHP** مليانة مميزات متطورة بتخليها مناسبة لبناء أنظمة قوية وذكية.

**المصادر:**

<https://www.geeksforgeeks.org/php/what-are-magic-methods-and-how-to-use-them-in-php/>

<https://www.php.net/manual/en/language.oop5.magic.php>

