



Late Static Binding

في الـ (OOP) بلغة PHP ، لما تيجي تستخدم static ، في العادي لو عندك صنف (class) وفيه دوال أو خصائص static ، لما تستخدم self :: جوا الدالة، هتلاقى إن PHP بتدور على الخاصية أو الدالة static في الصنف اللي جوا الدالة نفسها، مش في أي صنف ثاني أو صنف موروث (child class).

لكن مع Late Static Binding ، باستخدام PHP static :: بتقدر تشوف الصنف اللي فعلاً استدعى الدالة في وقت التشغيل (runtime) ، يعني لو في وراثة بين الأصناف، بيحصل تحسن في الوصول للخصائص والدوال static في الصنف المشتق.

الفرق بين self و static::

- self :: دا بيخليك دائماً تشير للصنف اللي جوا الدالة نفسها، حتى لو كنت بتستخدم الصنف المشتق (Child Class).
- static :: دا ببشاور على الصنف الفعلي اللي استدعى الدالة في وقت التشغيل. يعني لو استدعيت الدالة من صنف مشتق، هتستخدم الخصائص والدوال الخاصة بالصنف المشتق.

شوف المثال ده :

```
1
2 class ParentClass {
3     protected static $name = "ParentClass";
4     // Static property
5
6     // Static method using self::
7     public static function getClass_name() {
8         return self::$name;
9     }
10    // Static method using static::
11    public static function getClass_nameLateBinding() {
12        return static::$name;
13    }
14    // This will return "ParentClass" or "ChildClass" depending
15    // on the class that calls the method
16 }
```

```

15
16 class ChildClass extends ParentClass {
17     protected static $name = "ChildClass";
18     // Different static property
19 }
20 // Tests
21 echo ParentClass::getClassName() . "<br>";
22 // Will return "ParentClass"
23 echo ChildClass::getClassName() . "<br>";
24 // Will return "ParentClass" because we used self::
25 echo ParentClass::getClassNameLateBinding() . "<br>";
26 // Will return "ParentClass"
27 echo ChildClass::getClassNameLateBinding() . "<br>";
28 // Will return "ChildClass" because we used static::

```

إليه اللي حصل؟

دالة `getClassName()`:

- استخدمنا `self::` جوا الدالة، علشان كده مهما استدعيت الدالة من `ChildClass`، هيرجع "ParentClass" لأن `self::` بتشير للصنف اللي جوا الدالة نفسها.

دالة `getClassNameLateBinding()`:

- استخدمنا `static::` هنا. وال `static::` هتشوف في وقت التنفيذ إنت جايها من إيه صنف، فلو استدعيت الدالة من `ChildClass` هيرجع "ChildClass"، ولو استدعيت من `ParentClass` هيرجع "ParentClass".

إيه فائدة دا؟

- الفائدة إنك لما تستخدم `static::` بدلاً من `self::`، تقدر تتحكم أكثر في كيفية الوصول للخصائص والدوال `static` في حالة الوراثة.