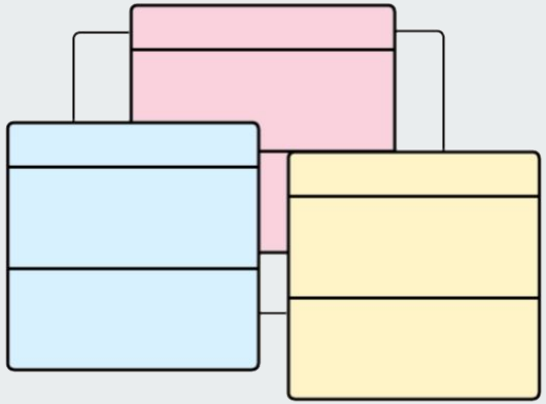
POO

par Jordan Juventin



Introduction

FORMAWAVE

La POO, qu'est-ce que c'est ?

La POO (**Programmation Orientée Objet**) c'est le fait d'utiliser des **objets** pour écrire son code.

Un objet, qu'est-ce que c'est ?

Un **objet**, c'est un type de variable (comme les tableaux, les nombres, …) qui est la représentation d'une structure de données appelée **une classe**.

Une classe, qu'est-ce que c'est ?

Une **classe**, c'est une **structure** de données. Il s'agit donc d'un élément de notre code qui vient indiquer la **forme** que prend une variable, et ce qu'elle peut faire.

Dans notre code

Classe

La structure de données

Une classe se définit avec le mot-clef **class**.

```
class MaClasse {}
```

Objet

La donnée concrète

On crée un **objet** à partir d'une classe.

On crée concrètement quelque chose sur le **modèle** de la classe.

On utilise le mot-clef **new**.

```
$objet = new MaClasse;
```

On dit aussi que `$objet` est une **instance** de `MaClasse`.

Propriété / Attribut

Les composantes de la donnée

Dans une classe, on peut mettre des **attributs** (qu'on appelle aussi **propriétés**).

C'est ce qui va composer chaque objet d'une classe.

```
class MaClasse {
    public $monAttribut;
    public $maPropriete;
}
```

Propriété / Attribut

Les composantes de la donnée

Les attributs sont des variables.

Dans l'objet qui résulte de la classe, on peut y accéder avec la flèche simple.

```
$objet->monAttribut;
```

Dans une classe, on peut également mettre des **méthodes**.

Ce sont des **fonctions**, qui indiquent les actions possibles à la classe (et donc à ses objets).

```php
class MaClasse {
    public function maMethode($param) {
        return 42;
    }
}
```

Méthode

Les actions possibles

Méthode

De la même manière que pour les propriétés, on peut accéder aux méthodes des objets avec la flèche simple.

```
$objet->maMethode('lorem');
```

Les actions possibles

Méthode

Les actions possibles

On a souvent besoin de référencer **l'objet actuel** dans une méthode.

On peut pour cela utiliser la variable spéciale **$this**.

```php
class Voiture {
    public $vitesse;

    public function rouler($destination) {
        echo 'Vroum vroum je roule à '
             . $this->vitesse . ' km/h';
    }
}
```

Constante

Dans une classe, on peut également mettre des **constantes**.

```
class MaClasse {
    const MA_CONSTANTE = 'lorem;
}
```

Constante

Pour accéder aux constantes d’un objet, on utilise l’opérateur “double deux-points” (::).

```
$objet::MA_CONSTANTE;
```

Constante

Puisque la constante ne change pas qu'importe l'objet (elle est *constante*), on peut y accéder directement via la classe.

```
MaClasse::MA_CONSTANTE;
```

En résumé

Classe

Structure de données

Objet / instance

Représentation concrète d'une classe

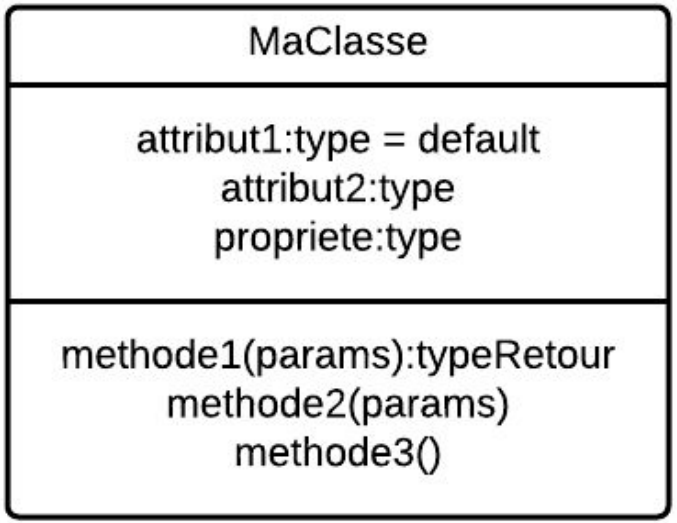
Attribut / Propriété

Une variable dans une classe / un objet

Méthode

Une fonction dans une classe / un objet

Dessiner une classe



Représentation UML d’une classe