

Les Objets

Arnaud Sallaberry

arnaud.sallaberry@univ-montp3.fr

Introduction

- Un **objet** est une chose quelconque
- Il possède des **attributs/champs** (caractéristiques)
- Il possède des **méthodes** (actions qu'on peut lui appliquer)
- Par exemple, ma voiture est un objet qui possède :
 - Des attributs : couleur bleue, marque ford fiesta, 5 portières, ...
 - Des méthodes : klaxonner, freiner, accélérer, tourner, ...

Plan

- **Classe**
- Attributs/Champs
- Méthodes
- Instances
- Manipulation
- Constructeur

Classe

- **Classe** = modèle d'objets
- Exemple d'une classe d'objets : voiture
- Les voitures ont toutes :
 - des attributs de même type (couleur, marque, nombre de portières, ...)
 - des méthodes de même type (klaxonner, freiner, accélérer, tourner, ...)

Classe

- Déclaration d'une classe en PHP:

```
class voiture {  
    // Les attributs et les  
    méthodes viendront ici  
}
```

Plan

- Classe
- **Attributs/Champs**
- Méthodes
- Instances
- Manipulation
- Constructeur

Attributs/Champs

- Déclaration des attributs en PHP:

```
class voiture {  
    public $couleur;  
    public $marque;  
    private $nbPortieres;  
}
```

- Public = toute les instructions du programme peuvent modifier la valeur
- Private = seule les instructions des méthodes de la classe peuvent modifier la valeur

Plan

- Classe
- Attributs/Champs
- **Méthodes**
- Instances
- Manipulation
- Constructeur

Méthodes

- Déclaration des méthodes en PHP:

```
class voiture {  
  
    public $couleur;  
    public $marque;  
    public $nbPortieres;  
  
    function klaxonner() {  
        echo `Je klaxonne fort !`;  
    }  
}
```

Méthodes

- Déclaration des méthodes en PHP:

```
class voiture {  
  
    public $couleur;  
    public $marque;  
    public $nbPortieres;  
  
    function klaxonner($duree) {  
        echo 'Je klaxonne pendant ' . $duree . '  
secondes';  
    }  
}
```

Méthodes

- Déclaration des méthodes en PHP:

```
class voiture {  
  
    public $couleur;  
    public $marque;  
    public $nbPortieres;  
  
    function klaxonner() {  
        return `Je klaxonne fort !`;  
    }  
}
```

Plan

- Classe
- Attributs/Champs
- Méthodes
- **Instances**
- Manipulation
- Constructeur

Instances

- **Instance** = un objet particulier d'une classe d'objets
- Ma voiture est un objet particulier de la classe voiture
- Création d'un instance en PHP :

```
$maVoiture = new voiture();  
$voitureDuVoisin = new voiture();
```
- Doit être fait après la fin de la déclaration de la classe.

Plan

- Classe
- Attributs/Champs
- Méthodes
- Instances
- **Manipulation**
- Constructeur

Manipulation :

Utilisation d'un attribut

- Chaque instance peut avoir des valeurs différentes pour ses attributs
- Par exemple, ma voiture est verte, celle du voisin est bleue :

```
class voiture {  
    public $couleur;  
}
```

```
$maVoiture = new voiture();  
$maVoiture->couleur = 'bleue';
```

```
$voitureDuVoisin = new voiture();  
$voitureDuVoisin->couleur = 'verte';
```

Manipulation :

Utilisation d'un attribut

- On peut modifier un attribut quand on veut
- Par exemple, je repeints ma voiture après quelques années :

```
$maVoiture = new voiture();  
$maVoiture->couleur = 'bleue';
```

```
// Plein d'années s'écoulent
```

```
$maVoiture->couleur = 'rouge';
```


Manipulation :

Utilisation d'une méthode

- On peut appeler les méthodes d'une classe avec une instance de cette classe
- Une voiture peut klaxonner

```
class voiture {  
    function klaxonner($duree) {  
        echo 'Je klaxonne pendant '.$duree.' secondes';  
    }  
}
```

```
$maVoiture = new voiture();  
$maVoiture->klaxonner(3);
```

Manipulation :

Utilisation d'un attribut dans une méthode

- Utilisation de `this` pour différencier un attribut d'une variable de la fonction

```
class voiture {  
    public $couleur;  
  
    function repeindre ($c) {  
        $this->couleur = $c;  
    }  
}
```

Plan

- Classe
- Attributs/Champs
- Méthodes
- Instances
- Manipulation
- **Constructeur**

Constructeur

- **Constructeur** : méthode particulière appelée lors de l'instanciation
- En PHP, elle s'appelle `__construct`

```
class voiture {  
    public $couleur;  
    public $marque;  
  
    function __construct($c, $m) {  
        $this->couleur = $c;  
        $this->marque = $m;  
    }  
}  
  
$maVoiture = new voiture('bleue', 'ford');  
$voitureDuVoisin = new voiture('verte', 'peugeot');
```