

## NOTIONS POO

Classe : C'est une structure regroupant des propriétés et des méthodes pour un type d'objet.

Exemple : `class HUMAIN {}`

Objet : une variable du type de la classe `class. $marcelline = new HUMAIN()` ;

Instance :

Propriété : les attributs de la classe. Exemple : `public $name = « monNom » ;`

Methode : Les fonctions de la classe. Exemple : `public function mySize()`

Constructeur : S'appelle lors de l'instanciation de l'objet. `$marcelline = new HUMAIN()` ;

Statique : Si une variable ou une méthode est statique, on peut y accéder sans avoir besoin d'instancier la classe. On peut y accéder statiquement depuis une instance d'objet.

Exemple : `public static $population ;`

Visibilité : Les propriétés et méthodes ont une visibilité : Publique, Protected ou Privée

- Publique : Elles sont visibles et utilisables de partout dans le code du projet

- Exemple : `public $name ;`

- Protected : Elles sont visibles et utilisables dans la classe ou elles ont été définies et dans les classes qui en hérite

- Exemple : `protected $name ;`

- Privée : Elles sont visibles et utilisables uniquement dans la classe ou elles ont été définies.

- Exemple : `private $name ;`

Getter : Permet de renvoyer une variable dont la visibilité est privée.

- Exemple : `public getSecret() { return $secret ; }`

Setter : Permet de modifier une variable.

- Exemple : `public setSecret($secret) { $this->secret = $secret ; }`

Héritage : Permet de faire héritée des classes entre elles. Les classes filles héritent des classes mères, et donc de tous les attributs et méthodes qui sont de visibilités publique ou protected.

Exemple : `class Homme extends HUMAIN { }`

Overriding : Permet de surcharger une méthode d'une classe parente