Programmation orientée objet en PHP

Le modificateur static

Le modificateur static permet de définir des attributs et méthodes dits "statiques" et qui appartiendront à la classe : nous les appellerons aussi des attributs de classe et des méthodes de classe. Comme pour les constantes, il ne sera pas nécessaire de posséder une instance pour accéder à ces attributs ou méthodes statiques. Pour accéder aux attributs et méthodes de classe, il faudra utiliser le nom de la classe et l'opérateur :: suivi de l'attribut ou de la méthode. Ces attributs et méhtodes de classe posséderont une visibilité et seront accessibles à chaque instance. Par exemple nous allons, dans la classe Humain, créer un attribut de classe public "compteur" qui stockera le nombre d'instances que nous avons créées. Ce compteur sera incrémenté dans le constructeur de la classe : à chaque appel au constructeur nous créerons une nouvelle instance et donc nous incrémenterons le compteur.

```php
<?php

// Humain.php
class Humain {

    const TAILLE_MAX = 2.10;

    public static $nb_instances = 0;
    private $age;
    private $taille;
    private $poids;

    public function __construct($page, $ptaille, $ppoids) {
        $this->age = $page;
        $this->taille = $ptaille;
        $this->poids = $ppoids;
        Humain::$nb_instances++;
    }

    public function __get($name) {
        return $this->$name;
    }

    public function __set($name, $value) {
        $this->$name = $value;
    }

    public function __toString() {
        return "age : $this->age, taille : $this->taille, poids : $this->poids";
    }

}

$h1 = new Humain(40, 1.80, 75);
$h2 = new Humain(35, 1.90, 80);
$h3 = new Humain(42, 1.75, 72);
$h4 = new Humain(39, 1.78, 77);
echo Humain::$nb_instances;
```

Voici un exemple de méthode de classe. Dans une classe Convertisseur, nous créons une constante USD dont la valeur est 1.37 (le taux de change du dollar). La méthode de classe toUSD() prendra en paramètre un double (le montant en euro) et renverra le montant en Dollar US.

```php
<?php

// Convertisseur.php
class Convertisseur {

    const USD = 1.37;

    public static function euroToUSD($amount) {
        return $amount * Convertisseur::USD;
    }

}

$montant = 450;
echo Convertisseur::euroToUSD($montant);
```

Fin

NAVIGATION

Accueil
- Ma page
  - Pages du site
  - Mon profil
  - Cours actuel
    - POO PHP
      - Participants
      - Généralités
      - La programmation orientée objet : premiers pas
      - Introduction
      - Classes et objets
      - T.P. première classe
      - Les classes en PHP
      - T.P. Personne
      - Le constructeur et this
      - T.P. Personne v2
      - Accesseurs et mutateurs
      - Les méthodes magiques
      - T.P. Personne v3
      - Les constantes de classe
      - **Le modificateur static**
      - T.P. Personne v4
      - Une Personne et une Adresse : la relation has-a
      - Histoire de références
      - T.P. formation
      - T.P. formation v2
      - Une Personne et plusieurs Adresse(s)
      - T.P. formation v3
      - Bricolage et dépendance
      - L'héritage
      - Les interfaces
      - Le typage
      - Les namespaces
      - Les exceptions
      - Les bases de données avec PDO
      - Les tests avec PHPUnit
      - Petite application version 2
      - Petitie application version 3
  - Mes cours

ADMINISTRATION

Administration du cours

Réglages de mon profil

Connecté sous le nom « Arnaud Lemais » (Déconnexion)
POO PHP