Institut Supérieur des Technologies de l'Informations et de la Communication de Borj Cédria

Unité enseignement	Technologies et programmation web, Techniques Multimédia	**Code U.E**	520080
Elément d'enseignement	Programmation Web Dynamique	**Code E.E**	520101
Niveau	License Fondamentale en Sciences de l'Informatique	**Classe**	LFSI -3

CHAPITRE 3 : LES FONCTIONS ET L'ORIENTE OBJET EN PHP

ELABORE PAR : AZZABI MOHAMED SEIFEDDINE

Année Universitaire 2017-2018

TABLE DES MATIERES

1 LES FONCTIONS EN PHP

1.1 DECLARATION

Une fonction peut être définie en utilisant la syntaxe suivante :

```php
<?php
function foo($arg_1, $arg_2, /* ..., */ $arg_n)
{
    echo "Exemple de fonction.\n";
    return $retval;
}
?>
```

1.1.1 Valeur par défaut des arguments

Vous pouvez définir comme en C++ des valeurs par défaut pour les arguments de type scalaire :

```php
<?php
function servir_cafe ($type = "cappuccino")
{
    return "Servir un $type.\n";
}
echo servir_cafe();
echo servir_cafe(null);
echo servir_cafe("espresso");
?>
```

L'exemple ci-dessus va afficher :

```
Servir un cappuccino.
Servir un .
Servir un espresso.
```

2 PHP ET L'ORIENTE OBJET

2.1 SYNTAXE DE BASE

2.1.1 Déclaration d'une Classe

Une définition de classe basique commence par le mot-clé *class*, suivi du nom de la classe. Suit une paire d'accolades contenant la définition des propriétés et des méthodes appartenant à la classe.

Le nom de la classe peut être quelconque à condition que ce ne soit pas un mot réservé en PHP. Un nom de classe valide commence par une lettre ou un underscore, suivi de n'importe quel nombre de chiffres, ou de lettres, ou d'underscores. Si on devait l'exprimer sous forme d'une expression rationnelle, il s'agirait de *^[a-zA-Z_\x7f-\xff][a-zA-Z0-9_\x7f-\xff]*$*.

Une classe peut contenir ses propres constantes, variables (appelées "propriétés" ou "attributs"), et fonctions (appelées "méthodes").

Exemple #1 Définition typique d'une classe

```php
<?php
class SimpleClass
{
    // déclaration d'une propriété
    public $var = 'une valeur par défaut';

    // déclaration des méthodes
    public function displayVar() {
        echo $this->var;
    }
}
?>
```

2.1.2 Créer une instance d'une classe

Pour créer une instance d'une classe, le mot-clé *new* doit être utilisé. Un objet sera alors systématiquement créé, à moins qu'il ait un constructeur défini qui lance une exception en cas d'erreur. Les classes devraient être définies avant l'instanciation (dans certains cas, c'est impératif).

Si une chaîne de caractères contenant un nom de classe est utilisée avec *new*, une nouvelle instance de cette classe sera créée. Si la classe est dans un espace de noms, son nom pleinement qualifié doit être utilisé.

```php
<?php
$instance = new SimpleClass();

// Ceci peut également être réalisé avec une variable :
$className = 'SimpleClass';
$instance = new $className(); // new SimpleClass()
?>
```

2.2 CONSTRUCTEUR

void **__construct** ([mixed `$args` = "" [, `$...`]])

PHP permet aux développeurs de déclarer des constructeurs pour les classes. Les classes qui possèdent une méthode constructrice appellent cette méthode à chaque création d'une nouvelle instance de l'objet, ce qui est intéressant pour toutes les initialisations dont l'objet a besoin avant d'être utilisé.