***EXO-7 :***

**LES FONCTIONS**

* ***Qu’est ce qu’une fonction ?***

Une fonction est un bloc de code réutilisable qu’on peut appeler à tout moment dans le programme. Elle permet :

* de ne pas dupliquer du code,
* de structurer le programme,
* de le rendre plus lisible et maintenable.
* ***Déclarer une fonction***

|  |
| --- |
| function saluer() {  echo "Bonjour !"; } |

Ici, on crée une fonction saluer qui n’attend rien et affiche un message.

* ***Appel d’une fonction***

|  |
| --- |
| saluer(); // affiche "Bonjour !" |

* ***Les paramètres***

Une fonction peut recevoir une ou plusieurs **valeurs** (appelées **paramètres**).

|  |
| --- |
| function direBonjour($prenom) {  echo "Bonjour " . $prenom . " !"; }  direBonjour("Alice"); // Affiche : Bonjour Alice ! |

Il est possible de prévoir une valeur par défaut des paramètres afin de les rendre facultatifs.

|  |
| --- |
| function direAurRevoir($prenom = "invité") {  echo "Au revoir " . $prenom; }  direBonjour(); // Au revoir invité |

***6) Valeur de retour***

Une fonction peut retourner un résultat grâce à return :

|  |
| --- |
| function carre($x) {  return $x \* $x; }  $resultat = carre(4); // $resultat vaut 16 |

***7) La portée des variables***

En PHP, **la portée** d'une variable définit **où** cette variable peut être utilisée. Il existe principalement trois types de portée :

* *Portée locale*

Une variable déclarée **dans une fonction** n’est accessible **que dans cette fonction**.

|  |
| --- |
| function direBonjour() {  $message = "Bonjour";  echo $message; }  direBonjour(); // Affiche "Bonjour" // echo $message; // Erreur : variable non accessible en dehors de la fonction |

* *Portée globale*

Une variable déclarée **en dehors de toute fonction** est dite **globale**, mais elle **n’est pas accessible directement à l’intérieur** d’une fonction.

|  |
| --- |
| $nom = "Alice";  function saluer() {  // echo $nom; // Erreur : $nom n'est pas accessible ici }  saluer(); |

SOLUTION : utiliser global

|  |
| --- |
| $nom = "Alice";  function saluer() {  global $nom;  echo "Bonjour $nom"; }  saluer(); // Bonjour Alice |

* *Variables statiques*

Une variable déclarée avec static **conserve sa valeur entre les appels à la fonction**, mais **reste locale** à cette fonction.

|  |
| --- |
| function compteur() {  static $i = 0;  $i++;  echo "Appel numéro : $i<br>"; }  compteur(); // Appel numéro : 1 compteur(); // Appel numéro : 2 |

**Exercice 1 :**

Créer une fonction qui affiche “Hello world !”

**Exercice 2 :**

Créer une fonction “afficherEtoile($nbr)” qui affiche un certain nombre de “ \* ” en fonction du nombre inscrit en paramètre.

Puis, à l’aide de cette fonction, affiche ceci :

|  |
| --- |
| \* \*\* \*\*\* \*\*\*\* \*\*\*\*\* |

**Exercice 3 :**

Écrire une fonction qui calcul l’impôt sur le revenu en fonction du revenu ( calculImpots($revenu) ).

Barème fictif de l’impôt sur le revenu :

* de 0 à 9 999€ => 0%
* de 10 000€ à 19 999€ => 10%
* de 20 000€ à 34 999€ => 18%
* de 35 000 à 49 999€ => 25%
* à partir de 50 000€ => 35%

NB => l’IR est un impôt progressif donc si jamais j’ai 12 000€ de revenu je vais devoir payer ((12 000 - 10 000) \* 10%) 200€ d’impôt.

* Calcul l’impôt si le revenu est de 16 000€
* Calcul l’impôt si le revenu est de 38 000€
* Calcul l’impôt si le revenu est de 64 000€

**Exercice 4 :**

|  |
| --- |
| $classe = [  [  "prenom" => "Lucie",  "maths" => 12,  "francais" => 14,  "histoire" => 9  ],  [  "prenom" => "Yann",  "maths" => 8,  "francais" => 10,  "histoire" => 7  ],  [  "prenom" => "Sophie",  "maths" => 16,  "francais" => 13,  "histoire" => 15  ] ]; |

Ecrire une ou plusieurs fonctions permettant :

* De calculer la moyenne d’un élève
* Déterminer s’il est admis (moyenne >= 10)
* Afficher pour chaque élève : son prénom, sa moyenne et un message d’admission ou non.

**Exercice 5 :**

Écrire une fonction qui affiche tous les nombres pairs ou impairs entre deux valeurs données ($debut et $fin).

1- A l’aide de cette fonction affiche tous les nombres pairs de 7 à 99.

2- A l’aide de cette même fonction affiche tous les nombres impairs de 59 à 169.

**Exercice 6 :**

Écrire une fonction qui permet d’afficher la table de multiplication pour n’importe quel nombre. Par exemple, pour la table 1 je veux obtenir :

|  |
| --- |
| 1 x 1 = 1 1 x 2 = 2 1 x 3 = 3 ... |