



# W1 - Piscine PHP

---

W-WEB-024

## Examen 3

---

La PPOO



# Examen 3

repository name: exam3  
language: PHP, CLI



- The totality of your source files, except all useless files (binary, temp files, obj files,...), must be included in your delivery.

## DÉTAILS ADMINISTRATIFS

- Il est interdit de faire du bruit, de perturber ou de nuire, de quelque manière que ce soit, au bon déroulement de l'examen.
- Vous avez accès à internet mais il est recommandé de ne pas s'en servir afin de réussir soi-même les exercices, faites l'effort de réflexion avant de chercher à savoir si vous avez juste.
- Toute sortie est définitive, interdiction donc de sortir pour aller boire, manger, fumer ou autre.
- Il est inutile de tricher, vous ne comprendrez pas mieux vos exercices et la correction (humaine ou automatique) le remarquera immédiatement.
- Vous devez créer un dépôt git nommé `Piscine_PHP_Exam_1` dans lequel vous rendrez tous vos exercices.
- Votre dépôt sera automatiquement ramassé (et donc corrigé) uniquement si vous appliquez la règle suivante :  
Vous devez donner les droits de lecture à :
  - `ramassage-tek`
  - `prenom.nom@epitech.eu` (si votre équipe pédagogique l'a fourni)

Par exemple : `bli -u mail repository setacl Piscine_PHP_Exam_1 jean.dupont@epitech.eu r`

- Vous devez rendre les exercices tels qu'ils vous sont demandés. Ne faites appel à aucun "include" ou fonction analogue qui permettrait l'inclusion d'un fichier.
- Le temps d'exécution est limité à 1 seconde par fonction. Au delà, les exercices seront faux. Vous pourrez essayer "timeout 1 php correction.php".

- Vous ne devez pas écrire de code en dehors des fonctions demandées. Vous ne serez pas corrigés si vous laissez la moindre ligne en dehors de ces dernières.



Nous vous conseillons fortement, afin d'optimiser la valeur pédagogique de cet examen, de ne pas tenter de chercher d'exemples de code en rapport avec les exercices et de vous contenter de vous référer au manuel PHP qui est amplement suffisant pour résoudre toutes les problématiques exposées.



Pour avoir accès au manuel complet de PHP, vous devez décompresser l'archive `php_manual_fr.tar.gz` présente dans le dossier du sujet et ouvrir la page `php-chunked-xhtml/index.html` dans un navigateur.



Pour chacun des exercices, vous pouvez trouver un ou plusieurs tests qui vous aideront à comprendre comment réaliser ces derniers. Vous devriez jeter un coup d'œil au dossier "misc" !



Afin de devenir de bons développeurs il vous faut être rigoureux et méthodique. Cela s'applique aussi à la qualité du code que vous allez produire. Pour cela, vous devez respecter des normes de code ! Pour le PHP, par exemple, vous devez respecter la norme **PSR-12**.



Toute erreur de norme entraînera un malus conséquent sur votre note finale.



Votre répertoire ne doit pas contenir de fichiers inutiles (fichiers temporaires, ...). N'oubliez pas de push régulièrement vos fichiers, sans cela, pas de correction.



## EXERCICE 1 - 4 POINTS

---

Modalités de rendu : `Piscine_PHP_Exam_3/ex_01.php`

### CONSIGNES

---

Faire une classe nommée `hero` qui hérite de la classe `personnage`, avec les propriétés privées **“vie”**, **“force”**, **“magie”**, **“agilite”** et **“intelligence”** qui sont tous des nombres, avec les méthodes publiques dites finales **“getVie”**, **“getForce”**, **“getMagie”**, **“getAgilite”** et **“getIntelligence”**, qui renvoient respectivement les valeurs de chacune des propriétés, avec une constante nommée **“MY\_HERO”** qui a pour valeur la chaîne de caractères **“Pango”** et avec un constructeur qui prend 5 paramètres, pour initialiser chacune des propriétés déjà listées. Attention, le fichier rendu ne doit pas comporter la classe `personnage`.

## EXERCICE 2 - 4 POINTS

---

Modalités de rendu : `Piscine_PHP_Exam_3/ex_02.php`

### CONSIGNES

---

Faire une interface nommée **“iHero”** qui implémente tout ce qui est décrit dans l'exercice précédent. Attention, le fichier de rendu ne doit pas comporter de classe.

## EXERCICE 3 - 4 POINTS

---

Modalités de rendu : `Piscine_PHP_Exam_3/ex_03.php`

### CONSIGNES

---

Faire une fonction nommée **“count\_rec\_str”**, qui prend en paramètre un tableau et une chaîne de caractères, et qui retourne un tableau. Le tableau retourné doit avoir pour chaque profondeur du tableau d'origine, le numéro de cette profondeur en clé, et en valeur le nombre de fois où la chaîne de caractères a été trouvée. Si la chaîne est vide (**“ ”**), la fonction doit retourner **FALSE**.

## EXERCICE 4 - 4 POINTS

---

Modalités de rendu : `Piscine_PHP_Exam_3/ex_04.php`

### CONSIGNES

---

Faire une fonction nommée "ord\_alphlong", qui prend en paramètre une chaîne de caractères et qui affiche les mots de cette chaîne par ordre de longueur puis dans l'ordre alphabétique. Les majuscules et minuscules sont identiques dans l'ordre alphabétique. En cas d'égalité alphabétique (aA et Aa), les mots doivent rester dans l'ordre d'apparition de la chaîne d'origine. Si aucun paramètre n'est donné, la fonction doit écrire "n". Dans la chaîne, il n'y aura que des espaces, des tabulations et des caractères alphabétiques. Vous n'afficherez qu'un espace entre les mots. Aucun avant le premier ni après le dernier de chaque ligne.

## EXERCICE 5 - 4 POINTS

---

Modalités de rendu : `Piscine_PHP_Exam_3/ex_05.php`

### CONSIGNES

---

Faire une fonction nommée "count\_island", qui prend en paramètre le nom d'un fichier qui contient une série de lignes de longueurs égales qui comportent soit le caractère "." soit le caractère 'w'. Ainsi, des îlots de 'w' sont formés par des suites de caractères contigus. La fonction doit afficher le contenu du fichier à l'écran en remplaçant tous les 'w' des îlots par leur numéro d'apparition dans le fichier et renvoyer TRUE. Si le fichier est vide, qu'une erreur se produit ou qu'aucun fichier n'est passé en paramètre, la fonction renvoie FALSE et rien ne s'affiche. Le fichier comporte au maximum 9 îlots.



## EXEMPLE

---

```
Terminal
~/W-WEB-024>
cat count_island.config
.....W.....

...WW.....

...WW...WWW.....

...WW...WWW.....

.....WWW..WW...

.....W.....

~/W-WEB-024> php ex_05.php
.....1.....

...22.....

...22...333.....

...22...333.....

.....333..44...

.....5.....
```

## BONUS - 1 POINT

---

Modalités de rendu : [Piscine\\_PHP\\_Exam\\_3/bonus.php](#)



## CONSIGNES

---

Rendre un fichier qui contient tous les nombres premiers de 2 à 100 inclus, séparés par des espaces et suivi d'un retour à la ligne. Aucun espace ni avant, ni après le dernier nombre.