



W1 - Piscine PHP

W-WEB-024

Examen 2

All inclusive



Examen 2

repository name: exam2
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 avez push sur votre repo GitHub se nommant : W-WEB-024-STG-1-1-exam2-prenom.nom
- 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 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 (5 POINTS)

Modalités de rendu : `ex_01.php`

CONSIGNE

Faire une fonction nommée “rev_epur_str”, qui prend en paramètre une chaîne de caractères et qui renvoie une chaîne de caractères, en renversant l'ordre de tous les mots de cette chaîne. Les séparateurs de mots sont les espaces et les tabulations. La fonction ne retourne ni de tabulation, ni plusieurs espaces consécutifs, ni d'espace en début ou en fin de chaîne. Si la chaîne est manquante, la fonction retourne FALSE.

EXERCICE 2 (5 POINTS)

Modalités de rendu : `ex_02.php`

CONSIGNE

Faire une fonction nommée “min_len_array”, qui prend en paramètre un tableau et qui renvoie la valeur dont la clé est la plus courte. Dans le cas où il existe plusieurs clés de taille minimale, c'est la valeur dont la taille est la plus courte qui doit être retournée. Dans le cas où il existe plusieurs valeurs de taille minimale, c'est la valeur de la première clé minimale rencontrée qui doit être retournée. Si le tableau est vide, la fonction doit retourner FALSE.

EXERCICE 3 (5 POINTS)

Modalités de rendu : `ex_03.php`

CONSIGNE

Faire une fonction nommée “resum_join_str”, qui prend en paramètre deux chaînes de caractères et qui renvoie les 14 premiers caractères de la première chaîne, suivi par les 3 derniers caractères de la deuxième chaîne. Si jamais la première chaîne fait moins de 14 caractères ou si la seconde fait moins de 3 caractères,



il faut mettre un point ' pour chaque caractère manquant. Si l'une des deux chaînes est manquante, la fonction doit retourner FALSE.

EXERCICE 4 (5 POINTS)

Modalités de rendu : **ex_04.php**

CONSIGNE

Faire une fonction nommée "return_calls" qui prend une chaîne de caractères en paramètre et un tableau. Elle essaiera d'appeler la fonction nommée grâce au premier paramètre et les arguments seront le tableau. La moindre erreur retournera FALSE. En cas de succès il faudra retourner le nombre de fois que la fonction définit par le premier paramètre est appelé, même si c'est une erreur. Les fonctions pourront être celles de l'examen. Si jamais la fonction retourne quelque chose sans rien afficher ce n'est pas un problème. Vous n'avez pas à inclure les autres fichiers, on le fera à l'aide de la commande suivante si nécessaire :

```
foreach (range(1, 4) as $v) include("ex_0${v}.php");
```

BONUS (5 POINTS)

Modalités de rendu : **bonus.php**

CONSIGNE

Faire une fonction nommée "playWithMe", qui prend en paramètre le nombre de lignes à créer et qui retournera true s'il n'y a pas d'erreur, sinon la fonction retournera false. Vous devez continuer cette suite mathématique:

```
1
11
21
1211
111221
312211
```

Vous devez afficher la ligne correspondant au paramètre avec un saut de ligne.



EXEMPLE

```
playWithMe(5) affichera "312211\n"
```