

TP 2 : Traiter les données en PHP

Exercice 1 :

1. Créez un tableau contenant les noms de 5 fruits différents. Parcourez le tableau et affichez chaque élément à l'aide d'une boucle for.
2. Créez un tableau contenant les notes de 5 élèves différents (chacun ayant une note différente). Calculez la moyenne des notes et affichez-la.
3. Créez un tableau associatif contenant les noms et les âges de 5 personnes différentes. Parcourez le tableau et affichez les informations de chaque personne sous la forme "Nom : <nom>, Age : <âge>".
4. Créez un tableau associatif contenant les noms, les âges et les adresses de 3 personnes différentes. Parcourez le tableau et affichez les informations de chaque personne sous la forme "Nom : <nom>, Age : <âge>, Adresse : <adresse>".
5. Créez un tableau associatif contenant les noms et les adresses e-mail de 5 personnes différentes. Parcourez le tableau et affichez les adresses e-mail de toutes les personnes dont le nom commence par la lettre "A".
6. Créez un tableau indexé contenant les noms de 5 fruits différents. Utilisez la fonction `array_push()` pour ajouter un nouveau fruit à la fin du tableau.
7. Créez un tableau indexé contenant les noms de 5 villes différentes. Ajoutez une ville supplémentaire au début du tableau et une autre à la fin du tableau. Supprimez la deuxième ville.
8. Créez un tableau indexé contenant les notes d'un élève pour 4 matières différentes. Utilisez la fonction `array_sum()` pour calculer la moyenne de l'élève.
9. Créez un tableau indexé contenant les noms de 5 pays différents. Utilisez la fonction `array_slice()` pour afficher les deux premiers pays de la liste.
10. Créez un tableau associatif contenant les noms et les adresses e-mail de 5 personnes différentes. Utilisez la fonction `array_keys()` pour afficher les noms des personnes dans le tableau.
11. Créez deux tableaux indexés contenant les noms de 5 villes différentes chacun. Utilisez la fonction `array_merge()` pour fusionner les deux tableaux en un seul tableau.

12. Créez un tableau indexé contenant les noms de 5 couleurs différentes. Utilisez la fonction `array_reverse()` pour inverser l'ordre des couleurs dans le tableau.
13. Créez un tableau indexé contenant les noms de 5 pays différents. Triez le tableau par ordre alphabétique et affichez les noms des pays dans l'ordre de tri.
14. Créez un tableau associatif contenant les noms et les âges de 5 personnes différentes. Utilisez la fonction `asort()` pour trier le tableau par ordre croissant d'âge.
15. Créez un tableau indexé contenant les noms de 5 pays différents. Utilisez la fonction `shuffle()` pour mélanger les noms de pays dans le tableau.
16. Créez un tableau associatif contenant les noms et les adresses e-mail de 5 personnes différentes. Utilisez la fonction `array_values()` pour afficher les adresses e-mail des personnes dans le tableau.
17. Créez un tableau indexé contenant les noms de 5 villes différentes. Utilisez la fonction `array_map()` pour ajouter le préfixe "Ville de " à chaque nom de ville dans le tableau.
18. Créez deux tableaux indexés contenant les noms de 3 couleurs différentes chacun. Utilisez la fonction `array_intersect()` pour trouver les couleurs communes entre les deux tableaux.
19. Créez un tableau associatif contenant les noms et les numéros de téléphone de 5 personnes différentes. Utilisez la fonction `array_key_exists()` pour vérifier si le nom "Jean" est présent dans le tableau.
20. Créez un tableau indexé contenant les noms de 5 villes différentes. Utilisez la fonction `array_unique()` pour supprimer les doublons dans le tableau.
21. Créez un tableau associatif contenant les noms et les âges de 5 personnes différentes. Utilisez la fonction `array_column()` pour extraire les âges des personnes dans un nouveau tableau indexé.
22. Créez un tableau associatif contenant les noms et les âges de 5 personnes différentes. Utilisez la fonction `array_filter()` pour afficher les personnes qui ont plus de 30 ans.
23. Créez un tableau indexé contenant les noms de 5 pays différents. Utilisez la fonction `array_search()` pour trouver l'index du pays "France" dans le tableau.
24. Créez un tableau associatif contenant les noms et les salaires de 5 employés différents. Triez le tableau par salaire, du plus petit au plus grand, et affichez le tableau.

Exercice 2 :

Soit le tableau associatif suivant :

```
$notes_apprenants = ("Mohamed" => "16", "Ahmed" => "14", "Rafika" => "13", "Aicha" => "15", "Samir" => "13", "Samar" => "13", "Rafik" => "10", "Samiha" => "09", "Fourat" => "07", "Sami" => "07", "Noura" => "14");
```

1. Utiliser un script php pour afficher ce tableau en deux colonnes Nom et Note.
2. Trier le tableau par clés croissantes, puis par valeurs décroissantes, et afficher les deux tableaux avec des titres successifs "Tableau trié en ordre croissant" et "Tableau trié en ordre décroissant"

Exercice 3 :

Écrire un script PHP contenant un tableau multidimensionnel associatif dont les clés sont des noms de personne et les valeurs des tableaux simples contenant le prénom, la ville de résidence et l'âge de la personne. Afficher le tableau.

Écrire un script PHP tableau multidimensionnel associatif dont les clés sont des noms de personne et les valeurs des tableaux associatifs dont les clés sont le prénom, la ville de résidence et l'âge de la personne avec une série de valeurs associées. Afficher le tableau.

Exercice 4 :

1. Créez un fichier texte et écrivez "Hello World" à l'intérieur. Ensuite, utilisez la fonction `file_get_contents()` pour lire le contenu du fichier et l'afficher.
2. Créez un fichier texte contenant un texte de plusieurs paragraphes. Ensuite, ouvrez le fichier en mode lecture avec `fopen()` et utilisez la fonction `fread()` pour lire les 100 premiers caractères du fichier et les afficher sur la page. Puis, lisez les 100 caractères suivants du fichier et affichez-les également.
3. Créez un fichier texte et écrivez une liste de 5 villes différentes à l'intérieur, chaque ville sur une ligne séparée. Ensuite, utilisez la fonction `fgets()` pour lire chaque ligne du fichier et l'afficher.
4. Créez un fichier texte contenant une phrase. Ensuite, ouvrez le fichier en mode lecture avec `fopen()` et utilisez la fonction `fgetc()` pour afficher chaque caractère de la phrase sur une ligne séparée.
5. Créez un fichier texte et écrivez une liste de 5 noms différents à l'intérieur, chaque nom sur une ligne séparée. Ensuite, utilisez la fonction `file()` pour lire le contenu du fichier et le stocker dans un tableau indexé.

6. Créez un fichier texte et écrivez une liste de 5 noms différents à l'intérieur, chaque nom sur une ligne séparée. Ensuite, utilisez la fonction `fopen()` pour ouvrir le fichier en mode écriture, utilisez la fonction `fwrite()` pour écrire un nouveau nom sur une nouvelle ligne, puis utilisez la fonction `fclose()` pour fermer le fichier.
7. Créez un fichier texte et écrivez une liste de 5 adresses e-mail différentes à l'intérieur, chaque adresse e-mail sur une ligne séparée. Ensuite, utilisez la fonction `file_put_contents()` pour ajouter une nouvelle adresse e-mail à la fin du fichier.
8. Créez un fichier texte et écrivez une liste de 5 noms différents à l'intérieur, chaque nom sur une ligne séparée. Ensuite, utilisez la fonction `file()` pour lire le contenu du fichier et stocker dans un tableau indexé, utilisez la fonction `array_map()` pour ajouter le suffixe "M." ou "Mme" à chaque nom dans le tableau, puis utilisez la fonction `implode()` pour joindre les noms dans une chaîne de caractères séparée par des virgules.
9. Créez un fichier texte et écrivez une liste de 10 nombres différents à l'intérieur, chaque nombre sur une ligne séparée. Ensuite, utilisez la fonction `file()` pour lire le contenu du fichier et le stocker dans un tableau indexé, puis utilisez la fonction `array_sum()` pour calculer la somme des nombres dans le tableau.
10. Créez un fichier texte et écrivez une liste de 5 phrases différentes à l'intérieur, chaque phrase sur une ligne séparée. Ensuite, utilisez la fonction `file()` pour lire le contenu du fichier et le stocker dans un tableau indexé, utilisez la fonction `array_map()` pour convertir toutes les lettres en majuscules dans chaque phrase du tableau, puis utilisez la fonction `file_put_contents()` pour écrire le contenu du tableau modifié dans le fichier.
11. Créez un fichier texte contenant des noms et des adresses e-mail séparés par des virgules, chaque paire nom/e-mail sur une ligne séparée. Ensuite, utilisez la fonction `file()` pour lire le contenu du fichier et le stocker dans un tableau indexé, utilisez la fonction `explode()` pour séparer les noms et les adresses e-mail de chaque ligne du tableau, puis utilisez la fonction `array_combine()` pour créer un tableau associatif où les noms sont les clés et les adresses e-mail sont les valeurs.

Exercice 5 :

Créez un formulaire HTML qui permet à l'utilisateur de télécharger un fichier. Lorsque le formulaire est soumis, vérifiez si un fichier a été sélectionné et s'il n'y a pas d'erreur de téléchargement (la taille ne doit pas dépasser 5Mo). Si tout est correct, déplacez le fichier téléchargé dans un dossier de destination sur le serveur, et affichez un message de réussite.