

TD 2

PHP

Table des matières

Exercice 1	2
Exercice 2	2
Exercice 3	2
Exercice 4	3
Exercice 5	4
Exercice 6	4
Exercice 7	4
Exercice 8	4
Exercice 9	5
Exercice 10	5

Exercice 1

Le tableau suivant donne les noms des étudiants et leur note.

```
$tab = [  
  'Martin' => 13,  
  'Thomas' => 5,  
  'Marie' => 19,  
  'Nicolas' => 19,  
  'Louise' => 13,  
  'Julien' => 13,  
  'Olivier' => 11  
];
```

- Afficher sous forme d'une liste HTML les noms des différents étudiants.
- Afficher sous forme d'une table HTML les noms et les notes des étudiants.
- Créer un autre tableau \$tab2 dans lequel les notes sont maintenant les clés. Afficher sous forme d'une liste les noms des différents étudiants du nouveau tableau.

Exercice 2

```
$schoolCalendar = [  
    "Rentrée" => [1,9,2020],  
    "Vacances Toussaint" => [  
        "début" => [17,10,2020],  
        "fin" => [2,11,2020]  
],  
    "Vacances Noël" => [  
        "début" => [19,12,2020],  
        "fin" => [4,1,2021]  
],  

```

Écrire le code permettant d'obtenir l'affichage suivant.

- Rentrée: 1/9/2020
- Vacances Toussaint: du 17/10/2020 au 2/11/2020
- Vacances Noël: du 19/12/2020 au 4/1/2021
- Vacances d'hiver: du 6/2/2021 au 22/2/2021
- Vacances de printemps: du 10/4/2021 au 26/4/2021
- Fin des cours: 2/6/2021

Condition: Uniquement utiliser deux boucles `foreach` ainsi que les fonctions `count` et `implode`.

Exercice 3

```
$players = [  
    ['name' => 'Martin', 'score' => 150],  
    ['name' => 'Thomas', 'score' => 120],  
    ['name' => 'Marie', 'score' => 98],  
    ['name' => 'Nicolas', 'score' => 153],  
    ['name' => 'Louise', 'score' => 118]  
];
```

Écrire une fonction `bestPlayer()` prenant en paramètre un tableau ayant la même structure que le tableau `$players` et retournant un tableau contenant le nom et le score du meilleur joueur.

Exercice 4

- Donner l'expression régulière correspondant à un nombre entier. Il est possible d'avoir un nombre négatif.
- Donner l'expression régulière correspondant à un nombre décimal. On utilisera le symbole `.` (point) comme séparateur décimal. Pour simplifier, on supposera qu'il y a forcément un chiffre après le séparateur décimal si celui-ci est donné. Les valeurs 10.0, 3.14, -42, 0.001, .001 doivent satisfaire l'ER mais 10. non.
- Donner l'expression régulière correspondant aux dates au format JJ/MM/AAAA. Il est possible de ne spécifier qu'un chiffre pour le jour et le mois.

Exercice 5

Créer un script `number.php` qui teste s'il existe un paramètre (url) de nom `number`. Si c'est le cas, alors le script teste si sa valeur est un nombre ou pas et affiche cette information.

Exercice 6

Créer un formulaire avec un champ de type texte (pour saisir un nombre) et un bouton submit. La soumission du formulaire appellera le fichier `number.php` défini dans l'exercice précédent.

Exercice 7

On suppose le formulaire précédent de l'exercice 6.

- Utiliser les sessions pour faire la multiplication du nombre précédent du formulaire (par défaut 1).
- Ajouter un bouton pour réinitialiser le produit des nombres.
- À l'aide des cookies, modifier le script pour qu'à chaque fois que l'on réinitialise un produit, la valeur de ce produit (s'il est différent de 1) soit stockée dans un cookie. Afficher à la fin du script la valeur du dernier produit calculé avant réinitialisation si cette information est disponible.

Exercice 8

- Créer un formulaire de méthode 'POST', redirigeant vers un autre fichier, avec pour champs:
 - Nom (Obligatoire, non vide)
 - Prénom (Obligatoire, non vide)
 - Date de naissance (Obligatoire, non vide, la personne doit être majeur)
 - Mail (Doit correspondre à la syntaxe d'un mail)
 - Mot de passe (Doit contenir une majuscule, une minuscule, et un chiffre allant de 8 à 12 caractères)
- Lors du traitement des réponses, vérifier qu'elles respectent les conditions.
- Une fois l'inscription effectuée, afficher à l'utilisateur ses informations.
- Ajouter un bouton de déconnexion qui ramène l'utilisateur sur la page du formulaire.