

# PHP et HTML (sorties)

# Table des matières

<b>I. Contexte</b>	<b>3</b>
<b>II. Qu'est-ce que le PHP ?</b>	<b>3</b>
<b>III. Exercice : Appliquez la notion</b>	<b>4</b>
<b>IV. Hello World en PHP</b>	<b>5</b>
<b>V. Exercice : Appliquez la notion</b>	<b>6</b>
<b>VI. Intégrer du PHP dans du HTML</b>	<b>6</b>
<b>VII. Exercice : Appliquez la notion</b>	<b>8</b>
<b>VIII. Créer des pages web dynamiques</b>	<b>8</b>
<b>IX. Exercice : Appliquez la notion</b>	<b>10</b>
<b>X. Auto-évaluation</b>	<b>11</b>
A. Exercice final .....	11
B. Exercice : Défi .....	14
<b>Solutions des exercices</b>	<b>15</b>

## I. Contexte

### Contexte

Dans ce cours, nous allons découvrir ce qu'est PHP, comment l'intégrer dans du HTML et comment il va nous permettre de créer des pages web dynamiques (génération dynamique de code HTML, personnalisation du contenu, etc.).

## II. Qu'est-ce que le PHP ?

### Objectifs

- Apprendre ce qu'est le PHP
- Savoir quelles sont les différences entre les pages web statiques et dynamiques

### Mise en situation

Lorsqu'un internaute consulte une page HTML, il la visualise directement dans son navigateur : ce sont les **pages web statiques**.

Pour des besoins plus avancés, il est possible de créer des **pages web dynamiques** générées par un programme lorsque l'internaute y accède. Le contenu de ces pages est susceptible de varier en fonction de chaque accès. Cela permet, par exemple, de chercher automatiquement les dernières actualités à afficher.

Pour créer des pages web dynamiques, nous allons utiliser le langage de programmation **PHP**.

### Définition PHP

PHP, dont l'acronyme récursif est *PHP Hypertext Preprocessor*, est un langage de programmation *open source* créé en 1994 par Rasmus Lerdorf.

Une majorité des sites utilisent PHP (près de 80 % en 2018), dont de très connus comme Wikipédia, les sites gouvernementaux français, Facebook...

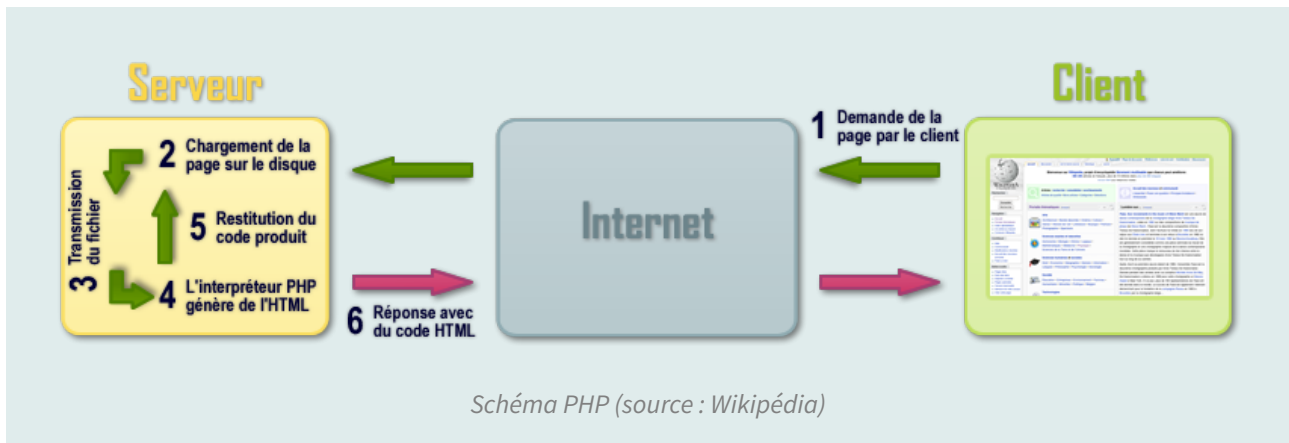
Sa mascotte est un éléphant.



*L'éléPHPant : la mascotte de PHP -*

**PHP est un langage de scripts exécutés côté serveur**, cela signifie que c'est le serveur qui va interpréter le code PHP et générer la page web à renvoyer au navigateur web.

Pour consulter une page PHP via notre navigateur web (le client), celui-ci envoie une requête HTTP au serveur. Ce dernier identifie que l'on demande une page PHP et interprète le code pour générer la page web finale, qu'il renvoie au client au format HTML.



#### Attention

Contrairement au code JavaScript ou HTML, **il n'est à aucun moment possible pour l'internaute de voir le code PHP** qui a été écrit pour générer sa page, ce code étant interprété côté serveur et non côté client.

En plus des pages web, PHP est capable de générer des images, des PDF, etc., et également de se connecter à de nombreux systèmes de gestion de bases de données.

#### Syntaxe

#### À retenir

- **PHP est un langage de scripts exécutés côté serveur** : c'est le serveur qui va interpréter le code PHP et générer la page web renvoyée au navigateur web.
- Le code PHP n'est pas visible par l'internaute.

#### Complément

Qu'est-ce que PHP ?<sup>1</sup>

PHP (Wikipédia)<sup>2</sup>

## Exercice : Appliquez la notion

[solution n°1 p.17]

### Exercice

Qu'est-ce qu'une page dynamique ?

- ☐ Une page web contenant du JavaScript permettant de manipuler son contenu
- ☐ Une page en pur HTML
- ☐ Une page web qui est régénérée à chaque demande

### Exercice

Où est exécuté le code PHP ?

- ☐ Dans le navigateur
- ☐ Sur le serveur
- ☐ Sur la machine locale du client

<sup>1</sup> <https://www.php.net/manual/fr/intro-what-is.php>

<sup>2</sup> <https://fr.wikipedia.org/wiki/PHP>

### Exercice

Depuis son navigateur, un utilisateur peut visualiser :

- ☐ Le code JavaScript
- ☐ La structure HTML
- ☐ Le code PHP
- ☐ Le code CSS

## IV. Hello World en PHP

### Objectif

- Apprendre à créer un fichier PHP et faire afficher un message

### Mise en situation

Nous allons voir comment créer nos fichiers et y faire afficher un message.

#### Exemple Affichage d'un message Hello World

```
1 <?php
2
3 echo 'Hello World !';
```

Si nous exécutons ce code, la seule chose que nous verrons dans notre navigateur au niveau de notre page web est le message : *Hello World !*

Même en inspectant le code source de la page avec les outils du navigateur, nous ne verrons jamais le code PHP qui a été écrit pour arriver à ce résultat, nous ne verrons que *Hello World !*

Le PHP a été interprété côté serveur, qui a ensuite renvoyé le résultat (l'affichage du message) au client (notre navigateur web).

Dans le code ci-dessus, nous faisons afficher grâce à `echo` la chaîne de caractères `'Hello World !'` et nous terminons notre ligne de code par un point-virgule.

Une chaîne de caractères doit être mise entre apostrophes ou guillemets afin d'être interprétée comme telle.

#### Syntaxe Fichier PHP

- L'extension utilisée pour créer les fichiers PHP est **.php** (ex : index.php).
- Pour écrire du code PHP, il faut insérer celui-ci entre une balise ouvrante `<?php` et une balise fermante `?>`. C'est uniquement le code présent entre ces balises qui sera interprété.
- Chaque ligne PHP est terminée par un `;`.

#### Méthode

À noter que, dans un fichier contenant exclusivement du PHP, **il est recommandé de ne pas mettre de balise fermante** : cela permet d'éviter les erreurs en cas, par exemple, d'oubli d'une nouvelle ligne ou d'un espace après la balise fermante, qui pourrait poser problème dans le HTML généré.

Ensuite, chaque ligne de code doit se terminer par un point-virgule (`;`).

### Syntaxe À retenir

- Un fichier PHP a comme extension .php.
- Il commence par une balise <?php.
- S'il contient uniquement du PHP, il est préférable de ne pas mettre de balise fermante ?> afin d'éviter d'éventuelles erreurs.
- Chaque ligne de code se termine par un point-virgule (;).

### Complément

Balises PHP<sup>1</sup>

## V. Exercice : Appliquez la notion

Nous allons maintenant exécuter notre tout premier code PHP ! Pour cela, nous allons utiliser Repl.it, qui est un service mettant à disposition des serveurs qui vont exécuter le code que nous allons lui donner.

Pour réaliser cet exercice, vous aurez besoin de travailler sur l'environnement de travail :



### Question

[solution n°2 p.17]

Rendez-vous sur le site de repl.it<sup>3</sup>. En haut à droite de l'écran, cliquez sur **"new repl"** et sélectionnez **"PHP Web Server"** dans la liste des langages, puis sur **"create repl"**.

Vous arriverez sur un écran qui va vous permettre de saisir du code (sur la moitié gauche de l'écran) et de l'exécuter (le bouton **"Run"**, en haut).

Supprimez tout le code d'exemple qui apparaît par défaut et saisissez le code du *Hello World* en PHP :

```
1 <?php
2
3 echo 'Hello World !';
```

Lancez-le et constatez que le résultat apparaît à droite de l'écran au bout de quelques secondes.

Modifiez le code pour mettre votre prénom à la place de *"World"* et cliquez sur **"Restart"** (là où se situait le bouton **"Run"**) pour rafraîchir la page de droite.

Félicitations, vous venez de lancer votre tout premier script PHP !

## VI. Intégrer du PHP dans du HTML

### Objectif

- Apprendre à intégrer du PHP dans du HTML

### Mise en situation

L'intérêt du PHP est qu'il permet de modifier dynamiquement le contenu d'une page HTML. Il est possible d'intégrer le PHP directement dans une structure HTML et de l'utiliser pour générer des fragments de la page HTML.

1 <https://www.php.net/manual/fr/language.basic-syntax.php-tags.php>

2 <https://repl.it/>

3 <https://repl.it/>

**Méthode**

Pour intégrer du PHP dans du HTML, il faut :

- Créer un fichier PHP.
- Ajouter normalement les parties statiques en HTML, par exemple les éléments *html*, *head* et *body*.
- Pour chaque section dynamique, ajouter le code PHP entre une balise ouvrante `<?php` et une balise fermante `?>`.

**Exemple**    **Intégration d'un fragment PHP dans une structure HTML**

Dans l'exemple ci-dessous, la balise `<h1>` et son contenu sont créés dynamiquement.

Le fichier PHP :

```
1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     <?php echo '<h1>Hello World !</h1>'; ?>
7   </body>
8 </html>
```

Le code HTML généré :

```
1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     <h1>Hello World !</h1>
7   </body>
8 </html>
```

**Exemple**    **Alternance PHP et HTML**

Le fichier PHP :

```
1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     <?php echo '<h1>Hello World !</h1>'; ?>
7     <p>Ici un texte qui n'est pas généré en PHP.</p>
8     <?php echo '<p>Un nouveau texte généré en PHP</p>'; ?>
9   </body>
10 </html>
```

Le code HTML généré :

```
1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     <h1>Hello World !</h1>
7     <p>Ici un texte qui n'est pas généré en PHP.</p>
8     <p>Un nouveau texte généré en PHP</p>
9   </body>
```

```
10 </html>
```

À chaque fois que nous avons eu besoin d'écrire à nouveau du PHP, nous avons remis les balises ouvrante et fermante.

#### Attention Fichier PHP contenant du HTML

La balise de fermeture est **obligatoire**, sinon tout ce qu'il y a après la balise d'ouverture sera considéré comme du PHP.

#### Syntaxe À retenir

Pour intégrer du PHP dans du HTML, il faut créer un fichier PHP, y intégrer le code HTML, ainsi que le code PHP entouré des balises d'ouverture et de fermeture.

## VII. Exercice : Appliquez la notion

Mettons tout cela en pratique ! Nous allons mélanger du PHP avec du HTML sur notre repl.it.

Pour réaliser cet exercice, vous aurez besoin de travailler sur l'environnement de travail :



### Question

[solution n°3 p.17]

Écrivez une page HTML ayant pour titre principal "Bonjour !" généré en PHP et lancez-la dans repl.it. Attention : les balises de titre ne doivent pas faire partie du PHP, seul le contenu "Bonjour !" doit l'être.

#### Indice :

Le titre principal d'une page est entre balises <h1></h1>.

## VIII. Créer des pages web dynamiques

### Objectif

- Connaître la notion de création de pages web dynamiques

### Mise en situation

Nous voyons ici quelques exemples d'usage du PHP pour des cas classiques.

NB: Ces extraits ne sont pas à comprendre pour le moment.

#### Exemple Liste d'animaux

Dans l'exemple ci-dessous, nous avons créé en PHP un tableau contenant une liste d'animaux. Une boucle permet de créer automatiquement toutes les puces de la liste.

Tout le texte situé après la notation // est un **commentaire** (tout ce qui est après "///" est ignoré par l'interpréteur PHP).

1 <https://repl.it/>



```

1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     <?php
7       $animals = ['Lion', 'Chat', 'Chien', 'Cheval', 'Tigre', 'Singe']; // Ici, nous avons un
      tableau de 6 animaux
8       echo '<ul>';
9       foreach ($animals as $animal) { // On construit une balise <li></li> par animal, qui
      contient son nom
10        echo '<li>'.$animal.'</li>';
11      }
12      echo '</ul>';
13    ?>
14  </body>
15 </html>

```

Le HTML final sera donc une liste à puces contenant tous les animaux :

```

1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     <ul>
7       <li>Lion</li>
8       <li>Chat</li>
9       <li>Chien</li>
10      <li>Cheval</li>
11      <li>Tigre</li>
12      <li>Singe</li>
13    </ul>
14  </body>
15 </html>

```

### Exemple

Dans l'exemple ci-dessous, nous avons une boîte de réception d'e-mails : l'utilisateur se connecte, et on affiche ses messages non lus.

```

1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     <?php
7       $userName = 'John'; // Le nom de l'utilisateur
8       $newMessages = [
9         'RDV 12h',
10        'Projet X',
11        'MAJ Document',
12      ]; // La liste des messages non lus
13
14      echo '<h1>Hello ' . $userName . ' !</h1>'; // On affiche "Hello" suivi du nom de
      l'utilisateur
15      echo '<p>Vous avez <strong>' . count($newMessages) . ' nouveaux</strong> messages .
      </p>'; // On utilise PHP pour compter le nombre de messages de notre liste de messages
16
17      echo '<ul>';

```

```

18     foreach($newMessages as $id => $message) {
19         echo '<li><a href="/readMessage/' . $id . '">' . $message . '</a></li>'; // On affiche
une liste de messages
20     }
21     echo '</ul>';
22     ?>
23 </body>
24 </html>

```

Le rendu HTML :

```

1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     <h1>Hello John !</h1>
7     <p>Vous avez <strong>3 nouveaux</strong> messages.</p>
8     <ul>
9       <li><a href="/readMessage/0">RDV 12h</a></li>
10      <li><a href="/readMessage/1">Projet X</a></li>
11      <li><a href="/readMessage/2">MAJ Document</a></li>
12    </ul>
13  </body>
14 </html>

```

## Syntaxe À retenir

PHP permet de faire énormément de choses, du simple ajout de texte (pour un nom d'utilisateur, par exemple), jusqu'à la génération de morceaux entiers de HTML (comme pour les listes).

## IX. Exercice : Appliquez la notion

Dans cet exercice, nous allons manipuler un script permettant d'afficher les numéros gagnants d'un loto !

Pour réaliser cet exercice, vous aurez besoin de travailler sur l'environnement de travail :



### Question 1

[solution n°4 p.17]

Dans un repl.it, lancez le code suivant :

```

1 <html>
2   <head>
3     <title>PHP Test</title>
4     <style>
5       .ball {
6         width: 100px;
7         height: 100px;
8         border-radius: 50px;
9         background: radial-gradient(circle at 50% 120%, #5555FF, #0a0a0a 80%, #AAAAFF 100%);
10      }
11

```

1 <https://repl.it/>

```

12     .ball-content {
13         width: 50px;
14         height: 50px;
15         border-radius: 25px;
16         background: radial-gradient(circle at 50% 120%, #FFFFFF, #EFEFEF 60%, #AEAEAE 100%);
17         text-align: center;
18         position: relative;
19         top: 10px;
20         left: 25px;
21         font-size: 40px;
22     }
23 </style>
24 </head>
25 <body>
26     <h1>Numéros gagnants de notre grand loto :</h1>
27     <?php
28         $winningNumbers = [5];
29         foreach ($winningNumbers as $winningNumber) {
30             echo '<div class="ball"><div class="ball-content">'.$winningNumber.'</div></div>';
31         }
32     ?>
33 </body>
34 </html>

```

Combien de numéros gagnants apparaissent ?

### Question 2

[solution n°5 p.18]

Dans le code ci-dessus, remplacez la ligne `$winningNumbers = [5];` par `$winningNumbers = [5, 14];`.

Que remarquez-vous ?

### Question 3

[solution n°6 p.18]

Sachant que la variable `$winningNumbers` contient une liste de nombres séparés par des virgules et entre crochets, ajoutez trois autres nombres de votre choix pour avoir un tirage complet.

## X. Auto-évaluation

### A. Exercice final

#### Exercice 1

[solution n°7 p.18]

Exercice

À quoi sert le PHP ?

- ☐ Générer du HTML
- ☐ Manipuler le DOM
- ☐ Donner un style à chaque bloc
- ☐ Insérer du contenu variable dans des balises

Exercice

Le PHP est le seul langage permettant de faire des sites dynamiques.

- ☐ Vrai
- ☐ Faux

Exercice

À quoi sert l'instruction `echo` ?

- ☐ Répéter plusieurs fois une instruction
- ☐ Afficher du contenu
- ☐ Délimiter le code PHP du HTML

Exercice

Quelle extension doit être utilisée pour les fichiers PHP ?

- ☐ .php
- ☐ .echo
- ☐ .html
- ☐ .phpfile

Exercice

Ce fichier est un fichier PHP valide.

```
1 <?php
2
3 echo 'Bonjour !';
```

- ☐ Vrai
- ☐ Faux

Exercice

Ce fichier est un fichier PHP valide.

```
1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     <?php echo '<h1>Bienvenue sur mon site !</h1>'; ?>
7   </body>
8 </html>
```

- ☐ Vrai
- ☐ Faux

Exercice

Ce fichier est un fichier PHP valide.

```
1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     <?php echo '<h1>Bienvenue sur mon site !</h1>';
7   </body>
8 </html>
```

- ☐ Vrai
- ☐ Faux

## Exercice

Ce fichier est un fichier PHP valide.

```
1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     echo '<h1>Bienvenue sur mon site !</h1>';
7   </body>
8 </html>
```

- ☐ Vrai
- ☐ Faux

## Exercice

Ce fichier est un fichier PHP valide.

```
1 <html>
2   <head>
3     <title>Mes courses</title>
4   </head>
5   <body>
6     $listeCourses = ['fraise', 'banane', 'oeufs', 'poireaux'];
7     <h1>Ma liste de courses :</h1>
8     <ul>
9       <?php
10         foreach($listeCourses as $element) {
11           echo '<li>'.$element.'</li>';
12         }
13       ?>
14     </ul>
15   </body>
16 </html>
```

- ☐ Vrai
- ☐ Faux

## Exercice

Ce fichier est un fichier PHP valide.

```
1 <html>
2   <head>
3     <title>Mon blog</title>
4   </head>
5   <body>
6     <h1>Bienvenue sur mon blog sur le HTML!</h1>
7     <h2>Mes articles sur le HTML :</h2>
8     <?php
9       $articles = ['La balise p', 'Tout savoir sur la balise a', 'Les nouveautés HTML 5',
10        '5 raisons de faire du HTML (la 6 ème va vous étonner !)']
11       foreach($articles as $id => $title) {
12         echo '<a href="article?id='.$id.'">'.$title.'</a><br>'
```

```
12     }
13     ?>
14     </body>
15 </html>
```

- ☐ Vrai
- ☐ Faux

## B. Exercice : Défi

Pour réaliser cet exercice, vous aurez besoin de travailler sur l'environnement de travail :



### Question

[solution n°8 p.21]

Nous allons intégrer un script qui va permettre de faire un petit jeu avec l'ordinateur : vous allez penser à un nombre entre 0 et 50 et l'ordinateur va essayer de le trouver. Pour cela, créez un repl.it<sup>2</sup> "**PHP Web Server**" et collez le code suivant :

```
1 <html>
2   <head>
3     <title>Devine mon nombre !</title>
4   </head>
5   <body>
6     <h1>Pense à un nombre entre 1 et 50 !</h1>
7   <?php
8     $nombre = 30; // Le nombre à trouver
9     $nombreDeTentatives = 50; // Le nombre de tentatives autorisées
10
11     if ($nombre < 0 || $nombre > 50) {
12       echo 'Hey, tu triches ! On avait dit entre 0 et 50 !';
13     } else {
14       ?>
15       <p>J'essaye de le deviner...</p>
16     <?php
17       $essais = 0;
18       do {
19         $tentative = rand(0, 50);
20         echo 'Est-ce que c\'est ' . $tentative . ' ? ';
21
22         if ($tentative == $nombre) {
23           echo '<h2>OUI !</h2>';
24           break;
25         } else {
26           echo 'Non...<br>';
27         }
28         $essais++;
29       } while ($essais < $nombreDeTentatives);
30       if ($tentative != $nombre) {
31         echo '<br>Je n\'ai pas trouvé =(';
32       }
33     }
34   ?>
```

<sup>1</sup> <https://repl.it/>

<sup>2</sup> <https://repl.it/>

```
35 </body>
36 </html>
```

Par défaut, l'ordinateur va chercher le nombre "30" et réaliser jusqu'à 50 tentatives. Lancez le programme plusieurs fois pour constater que le HTML généré est différent à chaque fois.

Manipulez ensuite le PHP en modifiant le nombre à trouver (la ligne 8 : `$nombre = 30;`) en 25, et le nombre de tentatives (la ligne 9 : `$nombreDeTentatives = 50;`) à 20.

### Indice :

Pour créer un repl.it, allez sur le site <https://repl.it>, cliquez sur "**New repl**" en haut à droite, sélectionnez "**PHP Web Server**" dans la liste des langages et cliquez sur "**Create Repl**". Pour lancer un script, cliquez sur "**Run**" (ou "**Restart**" si vous l'avez déjà lancé une fois).


## Solutions des exercices






**Exercice p. 4 Solution n°1****Exercice**

Qu'est-ce qu'une page dynamique ?

- ☐ Une page web contenant du JavaScript permettant de manipuler son contenu
- ☐ Une page en pur HTML
- ☒ Une page web qui est régénérée à chaque demande
-  Une page dynamique est régénérée à chaque demande : son contenu pourra donc être différent à chaque visite !


**Exercice**

Où est exécuté le code PHP ?

- ☐ Dans le navigateur
- ☒ Sur le serveur
- ☐ Sur la machine locale du client
-  Le code PHP est exécuté côté serveur.

**Exercice**

Depuis son navigateur, un utilisateur peut visualiser :

- ☒ Le code JavaScript
- ☒ La structure HTML
- ☐ Le code PHP
- ☒ Le code CSS
-  Le navigateur permet de visualiser tout ce qui est exécuté "côté client", donc le HTML, le CSS et le JavaScript. Le code PHP ne quitte jamais le serveur.

**p. 6 Solution n°2**

```
1 <?php
2
3 echo 'Hello John !';
```

**p. 8 Solution n°3**

```
1 <h1><?php echo 'Bonjour !'; ?></h1>
```

**p. 10 Solution n°4**

Un seul : le numéro 5.

**p. 11 Solution n°5**

Un deuxième numéro est apparu : le numéro 14. Ainsi, en ne modifiant qu'une seule valeur, toute l'architecture HTML s'est répétée pour former le design de la boule de loto.

**p. 11 Solution n°6**


```
1 <?php
2 $winningNumbers = [5, 14, 24, 54, 78];
```

**Exercice p. 11 Solution n°7**

**Exercice**

À quoi sert le PHP ?


- ☒ Générer du HTML
- ☐ Manipuler le DOM
- ☐ Donner un style à chaque bloc
- ☒ Insérer du contenu variable dans des balises

 Le PHP permet de générer du HTML et d'ajouter du contenu variable entre deux balises, comme un nom d'utilisateur, par exemple. C'est le JavaScript qui s'occupe de manipuler le DOM, et le CSS qui donne le style.

**Exercice**

Le PHP est le seul langage permettant de faire des sites dynamiques.


- ☐ Vrai
- ☒ Faux

 PHP est le langage le plus populaire, mais il est possible de faire des sites dynamiques dans n'importe quel langage. Ce qui importe, c'est que le serveur HTTP demande au bon langage de traiter la requête HTTP. PHP a été créé spécialement pour cela, d'où sa popularité, mais d'autres langages existent !

**Exercice**


À quoi sert l'instruction `echo` ?

- ☐ Répéter plusieurs fois une instruction
- ☒ Afficher du contenu
- ☐ Délimiter le code PHP du HTML

 `echo` est l'instruction PHP qui permet d'afficher quelque chose. Tout ce qui est "affiché" fera partie du HTML généré final.

**Exercice**

Quelle extension doit être utilisée pour les fichiers PHP ?


- ☒ .php
  - ☐ .echo
  - ☐ .html
  - ☐ .phpfile
-  C'est .php qui doit être utilisée.

### Exercice

---

Ce fichier est un fichier PHP valide.

```
1 <?php
2
3 echo 'Bonjour !';
```


- ☒ Vrai
  - ☐ Faux
-  Dans les fichiers en pur PHP, on a le droit d'omettre la balise fermante ?>.

### Exercice

---

Ce fichier est un fichier PHP valide.

```
1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     <?php echo '<h1>Bienvenue sur mon site !</h1>'; ?>
7   </body>
8 </html>
```

- ☒ Vrai
  - ☐ Faux
-  Ce fichier est parfaitement valide.

### Exercice

---

Ce fichier est un fichier PHP valide.

```
1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     <?php echo '<h1>Bienvenue sur mon site !</h1>';
7   </body>
8 </html>
```

- ☐ Vrai
- ☒ Faux

Q Il manque la balise PHP fermante `?>`. Ici, le PHP va donc essayer d'interpréter le reste du HTML, ce qui va provoquer une erreur.

### Exercice

Ce fichier est un fichier PHP valide.

```
1 <html>
2   <head>
3     <title>PHP Test</title>
4   </head>
5   <body>
6     echo '<h1>Bienvenue sur mon site !</h1>';
7   </body>
8 </html>
```

☐ Vrai

☒ Faux

Q Ce fichier est du pur HTML et non pas du PHP : si les instructions PHP ne sont pas contenues dans les balises `<?php ?>`, alors il sera interprété comme du texte simple.

### Exercice

Ce fichier est un fichier PHP valide.

```
1 <html>
2   <head>
3     <title>Mes courses</title>
4   </head>
5   <body>
6     $listeCourses = ['fraise', 'banane', 'oeufs', 'poireaux'];
7     <h1>Ma liste de courses :</h1>
8     <ul>
9       <?php
10         foreach($listeCourses as $element) {
11           echo '<li>'.$element.'</li>';
12         }
13       ?>
14     </ul>
15   </body>
16 </html>
```

☐ Vrai

☒ Faux

Q La ligne 6, qui est de toute évidence du code PHP, n'est pas placée entre les balises d'ouverture et de fermeture PHP. Son code n'est donc pas interprété.

### Exercice

Ce fichier est un fichier PHP valide.

```
1 <html>
2   <head>
3     <title>Mon blog</title>
4   </head>
5   <body>
6     <h1>Bienvenue sur mon blog sur le HTML!</h1>
```


```

7     <h2>Mes articles sur le HTML :</h2>
8     <?php
9         $articles = ['La balise p', 'Tout savoir sur la balise a', 'Les nouveautés HTML 5',
10            '5 raisons de faire du HTML (la 6 ème va vous étonner !)']
11         foreach($articles as $id => $title) {
12             echo '<a href="article?id='.$id.'">'.$title.'</a><br>'
13         }
14     ?>
15 </body>
16 </html>

```

☐ Vrai

☒ Faux

 Il n'y a pas un seul point-virgule à l'intérieur des instructions PHP, ce code a peu de chances de fonctionner !

#### p. 14 Solution n°8

Voici un exemple d'affichage que vous devriez obtenir :

## Pense à un nombre entre 1 et 50 !

J'essaye de le deviner...

Est-ce que c'est 14 ? Non...

Est-ce que c'est 11 ? Non...

Est-ce que c'est 50 ? Non...

Est-ce que c'est 34 ? Non...

Est-ce que c'est 10 ? Non...

Est-ce que c'est 39 ? Non...

Est-ce que c'est 32 ? Non...

Est-ce que c'est 39 ? Non...

Est-ce que c'est 42 ? Non...

Est-ce que c'est 35 ? Non...

Est-ce que c'est 7 ? Non...

Est-ce que c'est 0 ? Non...

Est-ce que c'est 30 ?

## OUI !

Avec les modifications, le code devrait être :

```

1 <html>
2 <head>
3     <title>Devine mon nombre !</title>
4 </head>
5 <body>
6     <h1>Pense à un nombre entre 1 et 50 !</h1>
7     <?php

```

```

8      $nombre = 30; // Le nombre à trouver
9      $nombreDeTentatives = 20; // Le nombre de tentatives autorisées
10
11     if ($nombre < 0 || $nombre > 50) {
12         echo 'Hey, tu triches ! On avait dit entre 0 et 50 !';
13     } else {
14     ?>
15     <p>J'essaye de le deviner...</p>
16     <?php
17         $essais = 0;
18         do {
19             $tentative = rand(0, 50);
20             echo 'Est-ce que c\'est ' . $tentative . ' ? ' ;
21
22             if ($tentative == $nombre) {
23                 echo '<h2>OUI !</h2>';
24                 break;
25             } else {
26                 echo 'Non...<br>';
27             }
28             $essais++;
29         } while ($essais < $nombreDeTentatives);
30         if ($tentative != $nombre) {
31             echo '<br>Je n\'ai pas trouvé =(';
32         }
33     }
34     ?>
35     </body>
36 </html>

```

N'hésitez pas à modifier les valeurs pour voir ce qu'il se passe !

On voit bien le caractère dynamique des sites au travers de cet exercice : à chaque nouvel affichage de la page, le contenu change. On ne sait pas combien de lignes est-ce qu'il va y avoir, ni si l'ordinateur va trouver le nombre ou pas.