Les Cookies en PHP

1. ## Objectifs

    - Etre capable de créer des cookies en PHP

2. ## Définitions

    - Le cookie est un petit fichier qui peut être écrit par un script PHP ou par d’autres langages, tel JavaScript, sur l’ordinateur du visiteur. À l’exception du piratage, c’est le seul cas où un site peut intervenir sur le disque dur d’un utilisateur.
    - Un cookie, c'est un petit fichier que l'on enregistre sur l'ordinateur du visiteur. Ce fichier contient du texte et permet de "retenir" des informations sur le visiteur.
    - Est un petit fichier texte (65 Ko maxi) stocké sur le disque dur du visiteur du site
    - Le cookie est un petit fichier texte, souvent chiffré, qui est situé dans les répertoires des navigateurs. Il est utilisé par les développeurs web pour aider les utilisateurs à naviguer sur leurs sites web de manière efficace et à effectuer certaines actions.

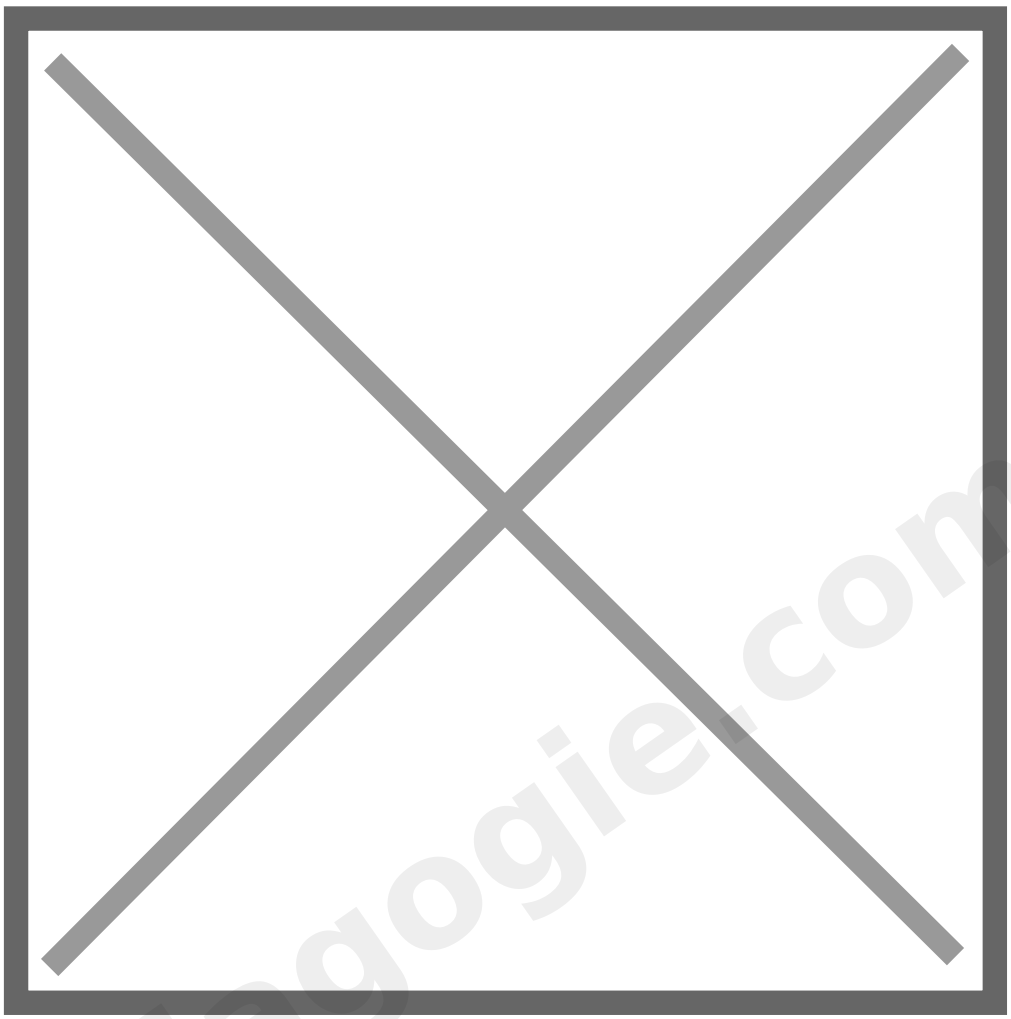
3. ## Emplacement des cookies

    - Cela dépend du navigateur que vous utilisez

        - **Google Chrome**

            - Cliquez sur . Vous le trouverez en haut à droite de la fenêtre de Chrome.
            - Sélectionnez . Vous le verrez vers le bas du menu déroulant.
            - Descendez et appuyez sur . Vous trouverez cette option vers le bas de la page.

apcpedagogie.com



- Vous pouvez également accéder aux paramètres en tapant "chrome://settings/" dans la barre d'adresse et en appuyant sur "Entrée".

- **Mozilla Firefox**

  - Aller dans le menu Outils **>** Options **>** Vie privée **>** Supprimer des cookies spécifiques

- **Microsoft Edge**

  1. Ouvrez Microsoft Edge et sélectionnez **Paramètres et plus** > **Paramètres** > **Confidentialité, recherche et services**.
  2. Effacer les données de navigation, sélectionnez**Choisir les éléments à effacer**.
  3. Sous **Intervalle de temps**, choisissez un intervalle de temps.
  4. Sélectionnez **Cookies et autres données de site**.

apcpedagogie.com

- **Opéra**



4. Créer un cookie en PHP

- **Créer un cookie simple**

  - Pour créer un cookie, utiliser la **fonction `setcookie()`**
  - La fonction `setcookie()` permet d'envoyer un cookie sur le client. Cette fonction doit être appelée avant tout affichage de texte dans la page.

    > - *Cette **fonction** doive être appelée avant d'écrire tout code HTML pour qu'elle fonctionne puisque les cookies doivent être envoyés avant toute autre sortie.*

```php
<?php
$dureeVie = time() + 365*24*3600; // Le cookie a une durée de vie de 1an
setcookie('Nom', 'HAJJI', $dureeVie); // On écrit un cookie
setcookie('Prenom', 'Riadh', $dureeVie); // On en écrit un autre cookie...
// Vous remarquez que tout est bien noté avant le début du code HTML
?>
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>Titre de ma page</title>
</head>
etc ...
```

apcpedagogie.com

- **Syntaxe**

```
boolean setcookie(string nom_cookie,[string valeur, integer datefin,string chemin, string domaine, integer securite] )
```

- **Paramètres**
  - **nom_cookie**
    - **Le nom du cookie**: est une chaîne définissant le nom du cookie.
    - Ce nom obéissant aux mêmes règles de nommage que les variables, il faut éviter les caractères spéciaux. Ce nom sert à identifier le cookie pour les opérations de lecture de leur contenu.
    - **Le nom du cookie créé est obligatoire.**
  - **string valeur**
    - **La valeur du cookie**: Cette valeur est stockée sur l'ordinateur du client.
  - **integer datefin**
    - Le temps après lequel le cookie expire. C'est un **timestamp Unix**, donc, ce sera un nombre de secondes depuis l'époque Unix (1 Janvier 1970).
    - Si ce paramètre est omis, le cookie n'est valable que pendant le temps de connexion du visiteur sur le site.
  - **string chemin**
    - Le chemin sur le serveur sur lequel le cookie sera disponible.
  - **string domaine**
    - Définit le nom entier du domaine à partir duquel vous pouvez accéder au cookie.
  - **integer securite**
    - **securite** est une valeur de type boolean qui vaut TRUE (ou la valeur 1) si le cookie doit être transmis par une connexion sécurisée (avec une adresse du type `https://www.mondomaine.com`) et **FALSE** (ou la valeur **0**) dans le cas contraire, qui est la valeur par défaut.

- **Exemples**

```
<?php //cookie valable uniquement pour la session
setcookie("prenom","Riadh"); //Cookie valable 6 heures
setcookie("nom","HAJJI",time()+21600); ?>
```

apcpedagogie.com

- **Créer un cookie avec plusieurs valeurs sous un même nom**
  - Vous pouvez écrire plusieurs valeurs sous un même nom de cookie en utilisant la notation à crochets des tableaux.

```
<?php
setcookie("identification[prenom]", "Riadh", time()+3600);
setcookie("identification[nom]", "HAJJI", time()+3600);
setcookie("identification[email]", "administrateur@apcpedagogie.com", time()+3600);
?>
```

- Enregistre un cookie nommé "identification" contenant trois valeurs utilisables pendant une heure.

5. **Créer un cookie à partir d'un tableau**
   - Un cookie peut être utilisé avec des tableaux associatifs. Cela à l'avantage de sauvegarder plusieurs valeurs sur un seul cookie.

```
<?php
$tabcookies = array("prenom"=>"Riadh",
"nom"=>"HAJJI","email"=>"administrateur@apcpedagogie.com");
foreach($tabcookies as $cle=>$valeur)
{
setcookie("identification[$cle]",$valeur,time()+7200);
}
?>
```

6. Récupérer la valeur d'un cookie en php

- En PHP, vous pouvez récupérer la valeur d'un cookie en utilisant la variable superglobale **`$_COOKIE`**. Cette variable est un tableau associatif qui contient les valeurs des cookies.
- Voici un exemple de code qui récupère la valeur d'un cookie nommé "**mon_cookie**" :

```
if (isset($_COOKIE['mon_cookie'])) {
    $valeur_cookie = $_COOKIE['mon_cookie'];
    echo "La valeur du cookie est : " . $valeur_cookie;
} else {
    echo "Le cookie n'existe pas";
```

apcpedagogie.com

```
}
```

- Dans cet exemple, on vérifie d'abord si le cookie "mon_cookie" existe en utilisant la fonction isset(). Si le cookie existe, on récupère sa valeur en accédant à l'élément correspondant du tableau **$_COOKIE**. Ensuite, on affiche la valeur du cookie à l'écran. Si le cookie n'existe pas, on affiche un message d'erreur.

- **Cookie simple**

  - Avant de commencer la lecture d'une page, PHP lit les cookies du client pour récupérer toutes les informations qu'ils contiennent. Ces informations sont placées dans la superglobale `$_COOKIE`, sous forme d'`array`.

```
<?php
     echo $_COOKIE['prenom'];
     echo $_COOKIE['nom'];
?>
```

7. **Cookie tableau**

   - La fonction `foreach()` peut servir à parcourir toutes les valeurs et les clés d'un tableau associatif représenté par un cookie de type tableau.

```
foreach($_COOKIE["identification"] as $cle=>$valeur)
{
   echo "Le cookie nommé: $cle contient la valeur : $valeur <br />";
}
```

8. Vous pouvez aussi accéder individuellement à chaque valeur.
9. Pour récupérer la valeur "prenom", par exemple, vous écrivez :

```
$_COOKIE["identification"]["prenom"]
```

10. ## Modifier la valeur d'un cookie en PHP

- **Cookie simple**

  - Pour modifier la valeur d'un cookie, il faut recréer un cookie de même nom mais de valeur différente.

apcpedagogie.com

```
<?php
// Après avoir rechargé la page :
if (isset($_COOKIE['CompteurPage'])) {
  $nbPages = $_COOKIE['CompteurPage']; // on récupère l'ancienne valeur
  $nbPages++; // on incrémente
  setcookie("CompteurPage", $nbPages, time()+365*24*3600); // on modifier le cookie
}
?>
```

11. **Cookie tableau**

    - Pour modifier la valeur d'un cookie tableau, il faut, comme dans le cas d'un cookie simple recréer un cookie avec le même nom de tableau mais de valeur différente.
    - La valeur récupérée doit être traitée comme un tableau.

```
<?php
// Après avoir rechargé la page :
if (isset($_COOKIE['CompteurPage'])) {
  $tabNbPages = $_COOKIE['CompteurPage']; // on récupère l'ancienne valeur
  $tabNbPages['accueil']++; // on incrémente
  setcookie("CompteurPage[accueil]", $tabNbPages, time()+365*24*3600); // on modifier le
cookie
}
?>
```

12. Supprimer un cookie en PHP

- Pour supprimer un cookie Il faut pour créer un cookie du même nom mais sans valeur.

```
<?php
//avant tout affichage de contenu
setcookie("CompteurPage");
```

13. Applications

- **Application 01**

apcpedagogie.com

- **Énoncé**
  - Créer le cookie suivant:
    - **Nom**: "Exemple_Cookies"
    - **Contenu**: "Exemple de Cookie valable 6 heures"
    - **Devrait expirer après**: "6 heures"

```php
<?php
//Cookie valable 6 heures
setcookie("Exemple_Cookies",
"Exemple de Cookie valable 6 heures",
time()+21600);
?>
```



- **Solution**
  - Afficher le cookie "Exemple_Cookies"

```
echo $_COOKIE["Exemple_Cookies"];
```

14. **Application 02**

- **Enoncé**

apcpedagogie.com

- Créer une page nommée **index.php** contenant un formulaire d'identification (nom et prénom dans un seul champs de texte)
- Faire en sorte que si le visiteur vient pour la première fois (ou qu'il a supprimé ses cookies), il aurait alors, la possibilité de saisir son nom et prénom dans le formulaire,
- S'il ne s'agit pas de sa première visite, afficher tout simplement "**Bonjour :**" puis son nom et prénom

- **Solution**

- Essayez de faire l'exercice de votre côté avant de regarder la Solution !

15. **Application 03**

- **Énoncé**

- Supposons que vous avez un site web où les utilisateurs peuvent se connecter. Vous voulez stocker les informations de connexion de l'utilisateur dans un cookie pour que l'utilisateur n'ait pas besoin de se connecter à chaque fois qu'il visite votre site.
- Comment vous pouvez créer un cookie pour stocker les informations de connexion de l'utilisateur.

Solution

```
<?php
// Définir les informations de connexion
$username = "mon_nom_utilisateur";
$password = "mon_mot_de_passe";

// Définir la durée de validité du cookie (1 heure)
$expire = time() + 3600;

// Créer le cookie
setcookie("login", $username . ":" . md5($password), $expire);

echo "Cookie créé !";
?>
```

- Dans cet exemple, nous avons défini le nom d'utilisateur et le mot de passe de l'utilisateur. Nous avons également défini la durée de validité du cookie à une heure. Ensuite, nous avons utilisé la fonction 'set pour créer un cookie nommé

"login" qui stock

16. **Application 04**

- **Énoncé**

  - Il est demandé de concevoir une interface de connexion à l'aide de PHP et de Bootstrap 5 pour objectif de créer une mise en page de base pour la page de connexion. Cette interface devra utiliser un cookie appelé « **maConnexion** » pour stocker les informations de connexion de l'utilisateur.
  - Si ce cookie n'existe pas, il devra être créé automatiquement avec l'adresse e-mail »hajjriadh@gmail.com« et le mot de passe « 2710 » comme identifiants.
  - En cas de succès de la connexion, l'utilisateur devra être redirigé vers une page nommée « page-protegee.php » qui contiendra le chemin du cookie ainsi que son contenu.

Solution

Page: connexion.php

```
<?php
// Vérifier si le formulaire de connexion a été soumis
if (isset($_POST['seConnecter'])) {
    // Récupérer les informations de connexion de l'utilisateur
    $email = $_POST['email'];
    $password = $_POST['password'];

    // Vérifier les informations de connexion de l'utilisateur
    if ($email === 'hajjriadh@gmail.com' && $password === '2710') {
        // Stocker les informations de connexion de l'utilisateur dans un cookie
        setcookie('maConnexion', $email . ':' . md5($password), time() + (86400 * 30),
'/');
        // Rediriger l'utilisateur vers une page protégée
        header('Location: page-protegee.php');
        exit;
    } else {
        // Afficher un message d'erreur si les informations de connexion sont
incorrectes
        echo '<div class="container"><div class="alert alert-danger">Informations de
connexion incorrectes</div></div>';
    }
}
```

apcpedagogie.com

```
// Si l'utilisateur a déjà un cookie de connexion, le rediriger vers la page protégée
if (isset($_COOKIE['maConnexion'])) {
    header('Location: page-protegee.php');
    exit;
}
?>


<!DOCTYPE html>
<html>
<head>
    <title>Page de connexion</title>
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css">
</head>
<body>

    <div class="container">
        <h1>Connexion</h1>
        <form method="post">
            <div class="form-group">
                <label for="email">Adresse email</label>
                <input type="email" class="form-control" id="email" name="email">
            </div>
            <div class="form-group">
                <label for="password">Mot de passe</label>
                <input type="password" class="form-control" id="password" name="password">
            </div>
            <button type="submit" name="seConnecter" class="btn btn-primary">Se connecter</button>
        </form>
    </div>

</body>
</html>
```

Page: page-protege.php

```
<?php
<!DOCTYPE html>
<html>
<head>
```

```
    <title>Page protégée</title>
    <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css">
</head>
<body>

    <div class="container">
        <h1>Page protégée</h1>
        <p>Cette page est protégée et ne peut être accessible qu'après une connexion
réussie.</p>
        <hr>
        <h2>Informations sur le cookie maConnexion</h2>
        <ul>
            <?php
            $cookie_name = "maConnexion";
            if(isset($_COOKIE[$cookie_name])) {
                echo "<li><strong>Contenu :</strong>".$_COOKIE['maConnexion']."</li>";
                echo "<li><strong>Date de création :</strong> ".date('d/m/Y H:i:s',
$_COOKIE['maConnexion_created'])."</li>";
            }
            ?>
        </ul>
    </div>

</body>
</html>
```

17. **Application 05**

    - **Énoncé**

      - Dans ce projet, vous allez implémenter une fonctionnalité de cookie de session qui stockera le nom d’utilisateur connecté. Si l’utilisateur est déjà connecté, le nom d’utilisateur sera extrait du cookie "username" et affiché sur la page d’accueil avec un message de bienvenue.
      - Vous devez créer la page "login.php" qui contiendra un formulaire de connexion en utilisant le framework Bootstrap 5. Le formulaire de connexion doit avoir un champ pour le nom d’utilisateur et un bouton pour soumettre le formulaire. Si le formulaire est soumis, un cookie de session sera créé contenant le nom d’utilisateur et l’utilisateur sera redirigé vers la page "index.php" en Bootstrap 5.

apcpedagogie.com

- La page "index.php" affichera soit le nom d’utilisateur, s’il existe un cookie de session "username", soit un message invitant l’utilisateur à se connecter. Si le cookie de session "username" n’existe pas, un bouton sera affiché permettant à l’utilisateur de se connecter.

Page: index.php

Solution

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <title>Bienvenue</title>
<link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css">
</head>
<body>

<div class="container my-5">
  <h1>Bienvenue</h1>

  <?php
  // Vérifier si le cookie de session "username" existe
  if (isset($_COOKIE['username'])) {
    $username = $_COOKIE['username'];
    echo '<p>Bienvenue, ' . $username . '!</p>';
  } else {
    echo '<p>Veuillez vous connecter pour accéder à cette page.</p>';
    echo '<p><a href="login.php" class="btn btn-primary">Se
connecter</a></p>';
  }
  ?>

</div>
</body>
</html>
```

Page: login.php

Solution

```
<!DOCTYPE html>
<html lang="fr">
<head>
```

```
  <meta charset="UTF-8">
  <title>Connexion</title>
<link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css">
</head>
<body>

<div class="container my-5">
  <h1>Connexion</h1>

  <?php
  // Vérifier si le formulaire de connexion a été soumis
  if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $username = $_POST['username'] ?? '';

    // Créer un cookie de session avec le nom d'utilisateur
    setcookie('username', $username, 0, '/');

    // Rediriger vers la page de bienvenue
    header('Location: index.php');
    exit();
  }
  ?>

  <form method="post">
    <div class="form-group">
      <label for="username">Nom d'utilisateur :</label>
      <input type="text" class="form-control" id="username" name="username"
required>
    </div>
    <button type="submit" class="btn btn-primary mt-1">Se connecter</button>
  </form>

</div>
</body>
</html>
```

Cours PHP

Source : https://apcpedagogie.com/les-cookies-en-php · téléchargé le 08/11/2026 · © APC Pédagogie · apcpedagogie.com