

PHP – Sessions et Cookies

C. BENSARI

Les cookies

- Un cookie est un fichier qui peut avoir une taille de 4 Ko au maximum. Ce fichier est stocké par l'application dans la machine de l'utilisateur
- Un cookie n'est accessible que par les pages qui appartiennent au même nom de domaine. Un cookie mis-en place par le domaine www.domaine1.com n'est pas accessible depuis le domaine www.domaine2.com

Pourquoi et quand utiliser les cookies ?

- L'idée d'utiliser les cookies vient du fait que HTTP soit un protocole sans état. Un protocole sans état n'enregistre pas l'état de session de communication entre deux requêtes successives (chaque requête est indépendante)
- Les cookies permettent de suivre l'état d'une session en utilisant ces petits fichiers stockés sur la machine de l'utilisateur
- Un exemple connu d'utilisation des cookies est le fait de pouvoir personnaliser l'expérience de l'utilisateur à travers un choix sur ces préférences. Ces préférences sont stockées dans un cookie
- Un autre exemple est de tracer l'historique des pages visitées sur le site

Création de cookies en PHP

- Pour créer un cookie en PHP, il suffit d'utiliser la fonction « **setcookie** »
- La fonction peut prendre plusieurs paramètres dont deux sont obligatoires :
 - Le nom du cookie (obligatoire)
 - La valeur du cookie (obligatoire)
 - Délai d'expiration du cookie (optionnel)
 - Le chemin vers le cookie (optionnel)
 - Le domaine (optionnel)
 - ..
- Pour retrouver la valeur du cookie, il suffit d'utiliser la variable globale :
`$_COOKIE['nom_du_cookie']`
- La fonction `setcookie()` doit être appelée avant la balise ouvrante `<html>`

Exemple d'utilisation d'un cookie

```
// test_insert_cookie.php
```

```
<?php
```

```
    // expire après 60 secondes
```

```
    setcookie("user_mail", "abcde@abc.com", time() + 60, '/');
```

```
?>
```

```
// test_read_cookie.php
```

```
<?php
```

```
    //Lecture du contenu du cookie
```

```
    print_r($_COOKIE); // affichera le contenu du cookie
```

```
    // Array ( [PHPSESSID] => h5onbf7pctbr0t68adugdp2611 [user_mail] =>  
    abcde@abc.com )
```

```
?>
```

Mettre à jour et suppression d'un cookie

```
<?php
```

```
// mettre à jour le cookie
```

```
setcookie("user_mail", "test_update@abc.com", time() + 60, '/');
```

```
// suppression d'un cookie
```

```
setcookie("user_mail", "", time() - 360, '/');
```

```
?>
```

Les sessions PHP

- Une Session est une variable globale (\$_SESSION) stockée sur le serveur. Elle possède un ID unique utilisé pour retrouver les valeurs stockées
- A n'importe quel moment où une session est créée, un cookie contenant l'ID de la session est stocké sur la machine de l'utilisateur et retourné au serveur avec chaque requête HTTP
- Une session peut stocker plus de données par rapport aux cookies
- Les données d'une session sont automatiquement supprimées à la fermeture du navigateur (un stockage temporaire)
- Comme le cas des cookies, une session est créée (initialisée) avant tout élément Html

Pourquoi et quand utiliser une session ?

- Permet de stocker et de passer des données d'une page à une autre
- Peut stocker un volume de données plus important que celui des cookies
- L'utilisation d'une session est une alternative lorsque le navigateur ne supporte pas l'utilisation des cookies
- Stocker des informations importantes dans une session est un moyen sécurisé car l'information sera sur le serveur et donc non accessible aux utilisateurs malveillants

Exemple d'utilisation d'une session

// test_start_session.php

```
<?php
    // initialisation d'une session
    session_start();
    // ajout du mail de l'utilisateur
    $_SESSION['mail_user'] = "abcd@abc.com";
?>
```

// test_read_session_value.php

```
<?php
    // initialisation de la session
    session_start();
    //Lecture du contenu de la session
    echo $_SESSION['mail_user']; // affichera abcd@abc.com
?>
```

Destruction d'une session

- Pour détruite une session entière (tous les éléments), il suffit d'utiliser la fonction «`session_destroy()`». La session est détruite dans les coulisses et les éléments ne seront plus disponibles dans les requêtes suivantes
- Pour détruire un élément d'une session, utiliser la fonction «`unset()`» qui prend en paramètre l'élément à supprimer

// test_destruction_session.php

```
<?php
    session_start();
    // suppression du mail de l'utilisateur
    unset($_SESSION['mail_user']);
    // destruction de toute la session
    session_destroy();
?>
```

Résumé

- Les cookies sont des petits fichiers sur la machine de l'utilisateur. Ils peuvent être lus depuis les pages du même domaine
- Les cookies expirent selon le délai d'expiration précisé lors de la création
- Une session est stockée sur le serveur de l'application et possède un identifiant unique
- Les sessions et cookies doivent être initialisées avant une quelconque balise html