

# HTML / CSS / PHP

## Todolist - CHAPITRE 4

# Les sessions

## Objectif

Rendre la liste des tâches persistantes

## Points abordés

- Gestion des sessions en PHP
- Focus sur les cookies

## Todo List

### Gestionnaire de tâches

#### Ajouter une tâche

ou

☐

Passer le balai

☒

~~Saluer le boss~~

☐

Couper l'ordi

# PHP

## Documentations

### Présentation des sessions en php

Le support des sessions de PHP est un moyen de préserver des données entre plusieurs accès.

Chaque visiteur accédant à votre page web se voit assigner un identifiant unique, appelé "identifiant de session". Il peut être stocké soit dans un cookie, soit propagé dans l'URL. Le support des sessions vous permet de stocker des données entre les requêtes dans le tableau super-globale **\$\_SESSION**.

Lorsqu'un visiteur accède à votre site, PHP va vérifier automatiquement (si **session.auto\_start** est activé) ou sur demande (explicitement avec **session\_start()**) s'il existe une session du même nom.

Si c'est le cas, l'environnement précédemment sauvé sera recréé.

**Les données de session sont temporaires et sont généralement stockées sur le serveur pendant la durée de la visite de l'utilisateur. Elles peuvent également expirer après une période d'inactivité.**

<https://www.php.net/manual/fr/intro.session.php>

# PHP

## Documentations

### Présentation des sessions en php

#### Usages

**Authentification** : Les sessions sont largement utilisées pour gérer l'authentification des utilisateurs. Après la connexion, l'identité de l'utilisateur est stockée dans une session, ce qui permet de maintenir l'état d'authentification tout au long de sa visite.

**Gestion de panier d'achat** : Les sessions sont utilisées pour stocker temporairement les articles sélectionnés par un utilisateur dans un panier d'achat.

**Personnalisation de l'expérience** : Les données de session peuvent être utilisées pour personnaliser l'expérience utilisateur en stockant des préférences, des paramètres et d'autres informations spécifiques à chaque utilisateur.

**Protection contre la falsification de requête intersite (CSRF)** : Les jetons anti-CSRF sont souvent stockés dans les sessions pour protéger contre les attaques CSRF en vérifiant que les requêtes proviennent de l'utilisateur authentifié.

# PHP

## Documentations

Principe des sessions - création / ajout de données

```
<?php
// Démarrer la session
session_start();

// Vérifier si l'utilisateur a soumis un formulaire pour définir son nom d'utilisateur
if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['username'])) {
    // Stocker le nom d'utilisateur dans une variable de session
    $_SESSION['username'] = $_POST['username'];
}

// Vérifier si le nom d'utilisateur est déjà stocké en session
if (isset($_SESSION['username'])) {
    $username = $_SESSION['username'];
} else {
    $username = "Invité";
}
```

<https://www.php.net/manual/fr/ref.session.php>

# PHP

## Documentations

Pour détruire une session utilisez `session_destroy()` cependant dans la doc de PHP il est indiqué ceci :

**Vous n'avez pas besoin d'appeler `session_destroy()` depuis le programme généralement. Nettoyer le tableau `$_SESSION` plutôt que de détruire les données de session.**

<https://www.php.net/manual/fr/function.session-destroy.php>

# Exercice

Rendre la liste des tâches persistantes

Temps : 2H

## Todo List

### Gestionnaire de tâches

#### Ajouter une tâche

 ou

☐

 Passer le balai

☒

 Saluer le boss

☐

 Couper l'ordi

☐

 Nouvelle tâche