

TABLE DES MATIERES

Qu'est ce que la session ?	2
Definition simple	2
Concrètement ça va servir à quoi ?	2
Comment utiliser la session	2

QU'EST CE QUE LA SESSION ?

DEFINITION SIMPLE

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

Un peu comme le localStorage d'un navigateur, vous pouvez sauvegarder des données que vous allez utiliser sur vos différentes pages.

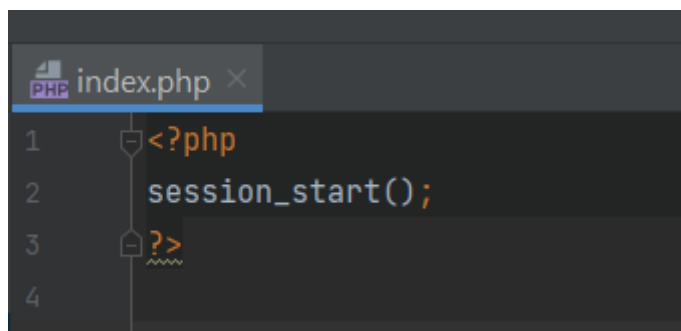
CONCRETEMENT ÇA VA SERVIR A QUOI ?

Avec la session vous allez pouvoir par exemple créer une connexion utilisateur à votre site WEB, ou encore un créer un panier pour des sites e-commerce.

COMMENT UTILISER LA SESSION

Comme avec GET et POST il existe une variable qui nous permet d'utiliser la session « \$_SESSION ».

Au début de chaque page vous allez devoir utiliser la fonction « session_start() » afin de pouvoir avoir accès à notre session.



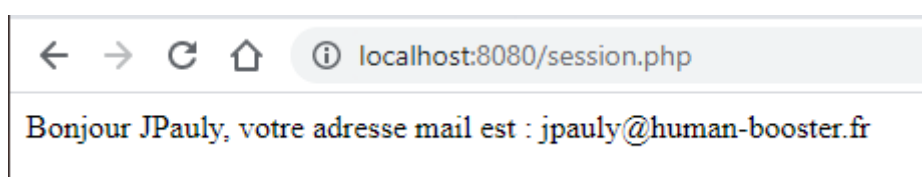
```
1 <?php
2 session_start();
3 ?>
4
```

Une fois `session_start()` effectué, vous pouvez utiliser notre variable `$_SESSION` pour stocker des valeurs, nous allons ici simuler un utilisateur connecté.

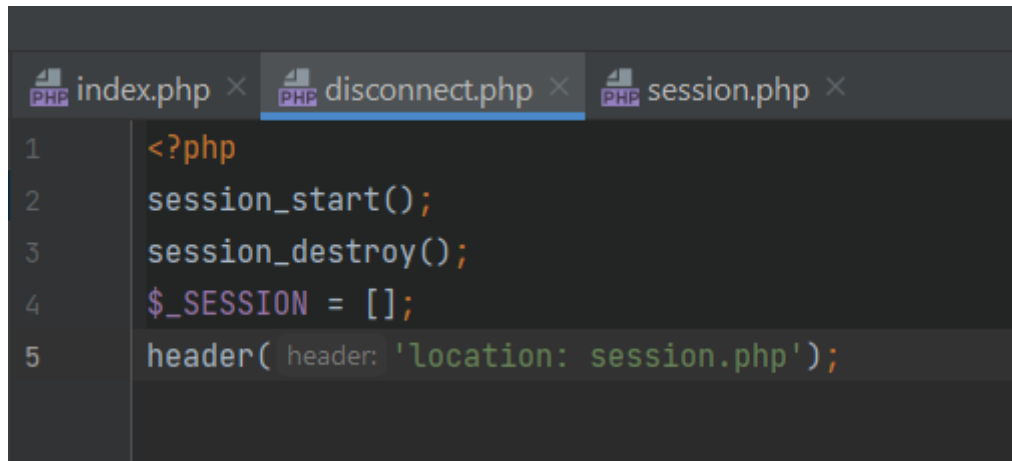
```
index.php x
1  <?php
2  session_start();
3  $_SESSION['pseudo'] = "JPauLy";
4  $_SESSION['mail'] = 'jpauLy@human-booster.fr';
5  ?>
6
7
```

Les variables définies ici dans `$_SESSION` vont être accessibles sur toute nos pages.

```
index.php x session.php x
1  <?php
2  session_start();
3
4  echo 'Bonjour ' . $_SESSION['pseudo'] . ', votre adresse mail est : ' . $_SESSION['mail'];
5  ?>
6
```

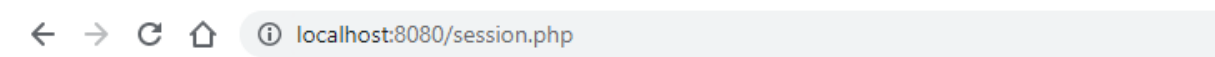


Nous pouvons aussi créer une déconnexion en détruisant la session actuelle avec la fonction « `session_destroy()` », n'oubliez pas d'effacer les variables mises au préalable dans session :



```
1 <?php
2 session_start();
3 session_destroy();
4 $_SESSION = [];
5 header( header: 'location: session.php');
```

Le résultat de la redirection :



Notice: Undefined index: pseudo in C:\Users\paulu\Desktop\cours HB PHP\php exo\session.php on line 4

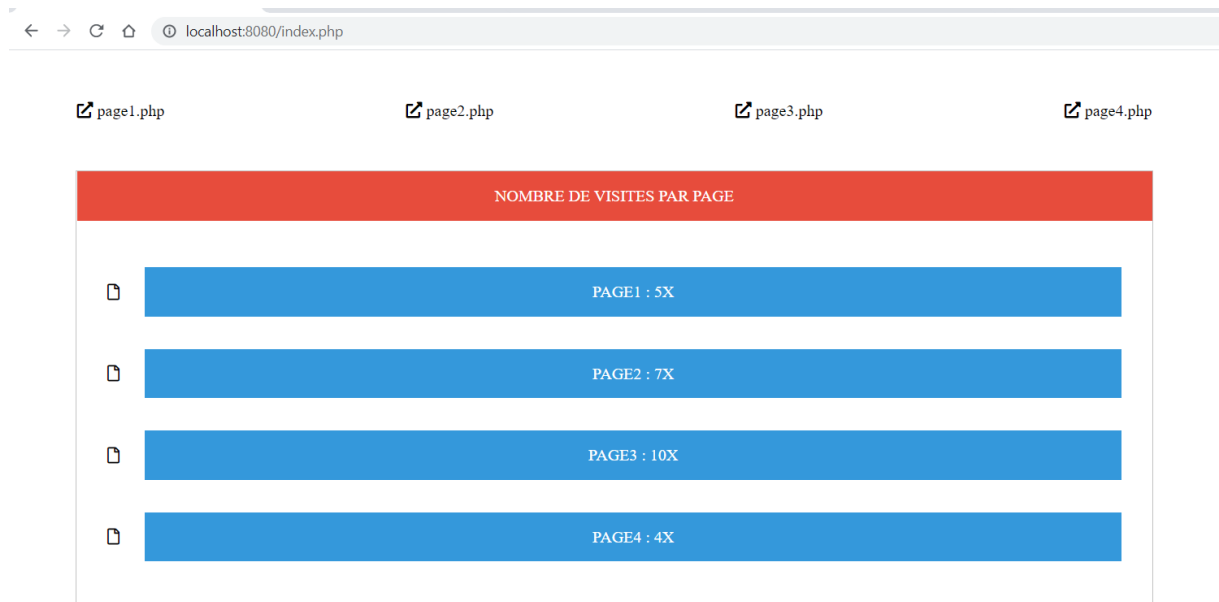
Notice: Undefined index: mail in C:\Users\paulu\Desktop\cours HB PHP\php exo\session.php on line 4

Nous avons forcément **une erreur PHP**, vu que nous essayons d'afficher dans notre page « session.php » les variables contenues dans **`$_SESSION`**.

EXERCICES

1

- Créer un projet avec 4 pages différentes (page1.php, page2.php etc...) + une page index.php.
- Créer des liens vers toutes les pages dans l'index.php.
- Le but est de compter et stocker combien de fois l'utilisateur se déplace sur les pages afin d'afficher ces numéros sur l'index.php.
- Résultat de l'index.php :



2

- Le but va être de simuler une connexion utilisateur.
- Le projet va contenir un index.php un header.php.
- Le header.php est inclus dans notre index.php
- Récupéré aussi sur discord le fichier user.php
- Dans l'index il y aura un formulaire de connexion comprenant pseudo et mot de passe.
- Trouver une solution pour simuler une connexion utilisateur en vous basant sur les données dans le user.php, si l'utilisateur n'est pas connecté, il n'y aura rien affiché dans le header, si l'utilisateur est connecté il y aura écrit son pseudo avec un lien pour se déconnecter.

- Petit plus, si l'utilisateur ou le mot de passe entré dans le formulaire est incorrecte, afficher un message d'erreur à l'utilisateur.
- Dernièrement, faites-en sorte que le formulaire ne soit plus affiché, si l'utilisateur est connecté

3

- Faites marcher votre cerveau !
- Récupérer le fichier item.php sur discord.
- Le but de cet exercice est de trouver un moyen de simuler une sorte de panier.
- Sur votre page d'accueil, vous allez avoir la liste des articles présents dans item.php
- Sur chaque élément de cette liste vous allez avoir un bouton « ajouter au panier »
- Vous pouvez ajouter dans votre panier autant d'élément que vous voulez.
- Vous aurez par la suite une page panier.php qui va lister tous les items que vous avez mis dans votre panier, avec bien entendu l'affichage d'une valeur total qui va représenter le prix total de vos produits choisis
- Si vous êtes partant, réfléchissez à un moyen de supprimer un item de votre panier quand vous êtes sur la page panier.php !

4

Nous allons maintenant utiliser le projet heroes fait précédemment.

Le but va être dans un premier temps d'avoir un formulaire qui nous permet de créer un hero, avec un champ select pour choisir la class, un champ text pour renseigner le nom. Et un champ file qui va nous permettre d'attribuer à notre hero une image.

Lorsqu'un hero est créé le formulaire ne s'affiche plus, si le hero meurt vous pouvez re afficher le formulaire.

Vous allez avoir des liens vers des pages qui vont représenter vos ennemis à affronter, au total 5.

Vous ne pouvez pas aller affronter l'ennemi suivant tant que le précédent n'est pas vaincu.

Comment se déroule un combat : Tant que l'ennemi ou le hero n'a pas ses points de vie à 0, le combat continue. Les entités s'affrontent chacun leur tour.

Si le hero meurt lors d'un de ses combats, il n'existe plus et vous devez en créer un nouveau sur le formulaire.

Avant de commencer le projet, établissez un plan sur papier du fonctionnement final et faites-en part au formateur.

Pour aller plus loin...

- A chaque round d'un combat, le protagoniste qui tape en premier doit être aléatoire.

Il va attaquer l'autre protagoniste, puis l'autre protagoniste attaque à son tour.

Ajouter 3 niveaux de difficultés :

- Explorateur : le hero affrontera à la suite 5 Gobelins ayant 1 niveau de plus que lui
- Aventurier : le hero affrontera à la suite 5 Gobelins ayant 1 niveau de plus que lui et 3 Ogres ayant 1 niveau de plus que lui
- Vétéran : le hero affrontera à la suite 5 Gobelins ayant 1 niveau de plus que lui, 3 Ogres ayant 1 niveau de plus que lui et 2 Dragon dont le dernier a 1 lvl de plus que lui

Le héro ne doit pas commencer chaque niveau en ayant son maximum de point de vie, par contre, il regagnera 50% de ses points de vie perdu et ses manas perdu entre chaque combat.

(Formule : $(hpMax - hp)/2 = hp$ gagné entre chaque combat)