

Module 6 : Exploitation d'une BdD

Pré-requis

Notions de base des langages HTML5 / CSS3 / JS acquises durant les TP précédents.

Le respect des noms des fichiers et dossier est obligatoire, ainsi que le nom des fonctions et des variables.

Description

Pour compléter un jeu web, il faut l'alimenter avec une base de données. Cette base de données permettra aux utilisateurs de créer un compte ou de s'y connecter, d'enregistrer son score et de le comparer à ses amis.

Objectifs

- Créer et/ou se connecter à un compte
- Afficher ou modifier son profile
- Rechercher des amis. Visualiser et comparer leurs scores

Ressources

Supports de cours	HTML5, CSS3, JS, PHP,MySQL	Documents support de cours HTML5, CSS3, JS, PHP et MySQL
Evaluation		
Sujet de TD/TP	Tous les TP précédents	
Doc. ou sites de référence		http://www.w3schools.com http://www.w3c.com http://www.code-couleur.com http://php.developpez.com/ cours/ http://phpdebutant.org/

Supports

Matériel(s) : Poste de développement sous Windows

Logiciel(s) : Notepad++, Mozilla Firefox

Légendes

- Partie à coder
- Partie à rédiger

Prolongement(s) possible(s)

Améliorer le site en passant par l'architecture MVC

Compétences:

	BdD	Je suis capable de:
<input type="checkbox"/>	Création d'un Modèle Conceptuel de Données	<input type="checkbox"/> Créer des tables <input type="checkbox"/> Créer des clés primaires <input type="checkbox"/> Créer des clés étrangères <input type="checkbox"/> Créer des champs
<input type="checkbox"/>	Synchronisation avec un serveur	<input type="checkbox"/> Créer une nouvelle connexion SQL <input type="checkbox"/> Synchroniser vers le serveur <input type="checkbox"/> Synchroniser depuis le serveur
<input type="checkbox"/>	Insérer des données	<input type="checkbox"/> Insérer une données via phpmyadmin <input type="checkbox"/> Insérer un fichier de données csv ou sql

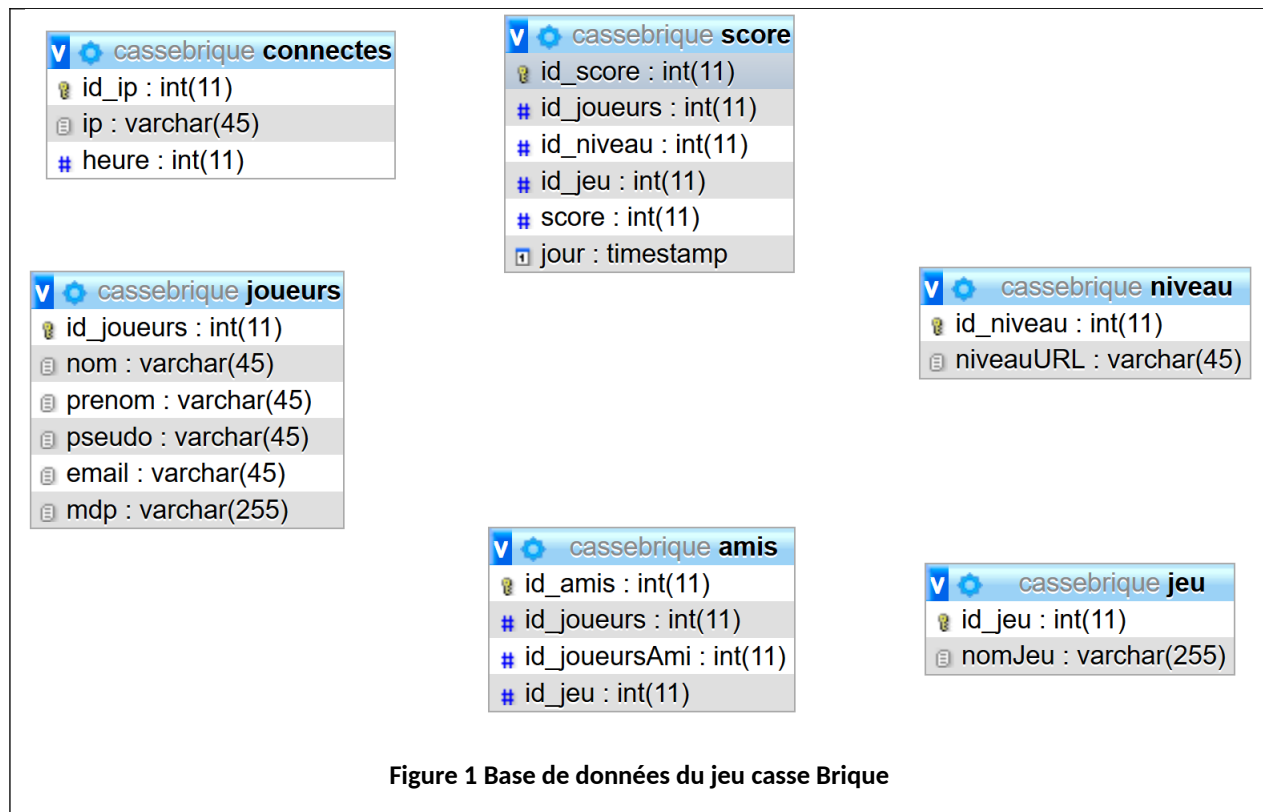
	Langage: PHP	Je suis capable de:
<input type="checkbox"/>	Formulaire	<input type="checkbox"/> Récupérer les données d'un formulaire <input type="checkbox"/> Extraire les données d'un formulaire en GET/POST <input type="checkbox"/>
<input type="checkbox"/>	COOKIE	<input type="checkbox"/> Installer un gestionnaire de COOKIE sur un navigateur <input type="checkbox"/> Visualiser un COOKIE quelconque d'une page web <input type="checkbox"/> Créer un COOKIE <input type="checkbox"/> Utiliser un COOKIE <input type="checkbox"/> Supprimer un COOKIE

Langage: SQL		Je suis capable de:	
<input type="checkbox"/>	Ecrire une requête	<input type="checkbox"/>	Sélectionner une information
		<input type="checkbox"/>	Insérer de nouvelles données
		<input type="checkbox"/>	Mettre à jour une donnée
		<input type="checkbox"/>	Supprimer une donnée
<input type="checkbox"/>	Executer une requête	<input type="checkbox"/>	Créer une connexion au serveur
		<input type="checkbox"/>	Exécuter la requête
		<input type="checkbox"/>	Vérifier que la réponse existe
<input type="checkbox"/>	Traiter la réponse d'une requête	<input type="checkbox"/>	Extraire les données d'une réponse à une requête SQL
		<input type="checkbox"/>	Traiter ou afficher les données

Partie 1.Créer la base de données

Objectif :

Permettre à un joueur de se créer et de jouer sur un compte personnel. Ce compte, lié à une base de données, qui enregistrera le score et le niveau atteint. Chaque niveau est construit à partir d'un fichier « *.txt » dont le chemin d'accès est stocké dans une table de données. Le compte contiendra un pseudo, le nom, le prénom, l'email et le mot de passe. La table "amis" contiendra les différents amis classés par ordre de score décroissant.



Procédure :

- ☒ Construire toutes les tables, avec MysqlWorkbench, contenant les champs de données nécessaires
- ☒ Renommer la base de données "jeuHTML" et non MySQL
- ☒ Transférer la base de données vers votre serveur MySQL
- ☒ Compléter la BdD avec les fichiers "csv" fournis par vos professeurs.
- ? Lister les clés primaires et les clés étrangères

Partie 2.Connexion / Inscription

Objectif :

Gérer la connexion et l'inscription du site.

Procédure :

Afin de gérer l'évolution d'un joueur, il est très pratique d'avoir un compte. Pour cela vous devez utiliser un navigateur avec la balise « <nav> » qui contient deux formulaires : connexion et inscription.

N.B : Pour l'instant nous allons uniquement gérer le jeu "pong"

Pour toute connexion à un compte, il faudra tout d'abord passer par le fichier `compte.php` qui validera ou non les informations puis renverra vers le fichier **`index.php` (page d'accueil de votre site)**

A. Code commun (`index.php`)

☞ Pour connecter un nouveau joueur, les données du formulaire seront envoyées vers le fichier "**`compte.php`**". **Assurez-vous que l'action du formulaire soit "`compte.php`". Attention le formulaire a été écrit en JS**

? Rappelez les deux méthodes d'envoies d'un formulaire ?

En vous aidant du cours de PHP et MySQL donnez le nom des fonctions permettant de :

? Se connecter au serveur et à la base de données : **`jeuHTML`**

Ecrire les codes, dans le fichier **`compte.php`**, permettant de :

☞ Se connecter au serveur et à la base de données : **`jeuHTML`**

Afin de faire la différence entre le formulaire de connexion et d'inscription, nous allons utiliser les valeurs envoyées par le "**`submit`**"

? Donner le nom du fichier contenant le code des formulaires.

? Dans quel cas le code suivant retournera vrai : `isset($_GET['choixCompte'])` ? Pour rappel, « `choixCompte` » est le « `name` » de la balise « `input` » de type « `submit` » dans les formulaires de connexion et inscription.

? Donner les valeurs du bouton "**`submit`**" dans le cas d'une connexion ou d'une inscription.

? Ecrire les deux conditions permettant de faire la différence entre une connexion et une inscription

☞ Taper les deux conditions dans le fichier "**`compte.php`**"

B. Connexion (`compte.php`)

Le formulaire contient les champs suivants : pseudo et mot de passe. Pour valider un formulaire de connexion, nous devons d'abord rechercher le pseudo et le mot de passe correspondant dans la base de données et si le couple correspond alors la connexion est validée. Le login est enregistré dans un cookie '`login`' pour une durée de 3600 secondes

Pour interroger un base de données, il faut exécuter une requête SQL. Les étapes sont données en Annexe 1

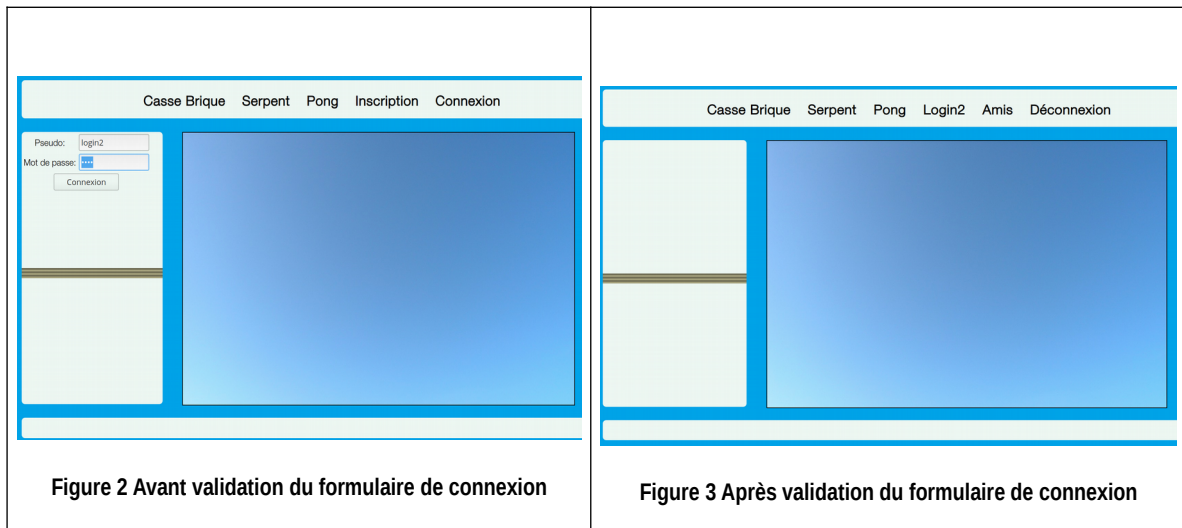


Figure 2 Avant validation du formulaire de connexion

Figure 3 Après validation du formulaire de connexion

? Donner la requête permettant de sélectionner toutes les informations de la table « joueurs » connaissant le pseudo et le mot de passe envoyés par le formulaire.

☞ Tester la requête en utilisant phpmyadmin de WAMP avec les valeurs login2 et mdp2

☞ Ecrire la requête et l'exécuter dans **compte.php**

☞ Si la connexion est réussie, il faut, alors, renvoyer vers le fichier **index.php**. Modifier le fichier **compte.php** pour répondre à la problématique précédente tout en vous aidant de l'Annexe 1

☞ Adapter la requête de l'annexe 1 afin de ne sélectionner un joueur connaissant son login et son mot de passe

☞ Tester la connexion en utilisant le login2 et mdp2

C. Inscription (compte.php)

Il devra contenir : le nom, prénom, pseudo, email et mot de passe. Le mot de passe sera demandé deux fois et on vérifiera (**avec un code fait uniquement en JS**) que les deux mots de passe correspondent avec une taille d'au moins 6 caractères.

? Donner la requête permettant d'insérer un nouveau client en tenant compte des données insérées dans le formulaire

☞ Tester la requête en utilisant phpmyadmin, avec vos données personnelles

☞ Ajouter la requête dans le fichier **compte.php**

D. Cookie

Une fois que l'un des formulaires est complété, une variable cookie est créée afin de conserver l'identité du client et/ou le pseudo du client. Pour créer un cookie, vous utiliserez la fonction php `setcookies(paramètre1,paramètre2,heure de fin)`

Pour déterminer l'heure de fin du cookie, il faut ajouter à l'heure actuelle la durée du cookie.

? En analysant le code suivant `time()+3600`, donner l'heure de suppression du cookie s'il a été créé à 9h00

? Donner le code permettant de créer le cookie pseudo contenant le pseudo d'un joueur valable pendant 10 minutes

🖥️ En complétant la partie connexion dans le fichier `compte.php`, écrire le code permettant de créer un cookie avec le pseudo du joueur et d'une durée de 1 heure lorsque la requête de connexion sera validée

? A l'aide d'une impression d'écran valider la création d'un cookie pseudo

? Donner le code permettant d'accéder à la valeur contenue dans le cookie pseudo

E. Validation de la connexion

🖥️ Modifier le fichier **`compte.php`** afin de rediriger l'utilisateur vers la page **`d'index.php`** lorsque le cookie sera créé.

🖥️ Modifier la barre de navigation dans le fichier **`index.php`** afin d'afficher le pseudo et la déconnexion si le cookie contenant le pseudo du joueur existe bien.

? Montrer que la partie connexion fonctionne correctement

F. Déconnexion

? Que faut-il faire pour supprimer un cookie, en ligne de code ?

🖥️ Modifier le fichier **`compte.php`** afin de supprimer le cookie et de retrouver la page d'accueil d'origine lorsque l'utilisateur cliquera sur le lien de déconnexion dans la barre de navigation.

Partie 3. Profile du joueur

Objectif :

Afficher et modifier le profile du joueur

Résultat :

Procédure

Afin de pouvoir récupérer le choix de l'utilisateur, nous allons faire transiter des informations dans le barre URL. Par exemple, lorsque l'on clique sur l'onglet Amis, nous aurons comme information **index.php?profile**.

Ainsi dans le fichier **index.php**, il suffira de vérifier que **\$_GET['profile']** existe pour afficher le code nécessaire. Pour la suite vous pourrez vous inspirer de l'exemple de l'**Annexe 2**

G. Modifier le profile

Lorsque la personne clique sur l'onglet contenant son pseudo, il pourra modifier les valeurs de son profile.

☞ Modifier le fichier **index.php** pour tenir compte du clique sur l'onglet du profile.

☞ Créer un fichier **profile.php** qui contiendra les formulaires précédents

☞ Tester l'affichage du fichier en l'incluant dans l'**index.php** avec la fonction **include()**

☞ Faire une requête permettant de récupérer les données concernant le joueur

☞ Pré compléter le formulaire avec les données du joueur

? Donner la requête permettant de mettre à jour les données du joueur en utilisant les valeurs du formulaire

☞ Compléter le fichier **profile.php** afin de mettre à jour les données du joueurs.

☐ **Modifier le mot de passe**

- ☐ En utilisant la méthode de votre choix, ajouter une option permettant de modifier le mot de passe en ayant pris soin de vérifier, préalablement, le mot de passe actuel.

Partie 4. Rechercher ses amis

Objectif :

Permettre à l'utilisateur de pouvoir comparer son score avec une liste de joueur qu'il pourra soit rechercher soit consulter dans la liste déjà enregistrée.

Résultat :



Figure 4 Résultat affichage des amis

Procédure :

Pour obtenir le résultat précédent, la section sera divisée en deux parties, la recherche d'ami et la liste d'ami enregistrée



Astuce : vous pouvez utiliser la barre URL pour faire passer des informations nécessaire à votre code.

H. Recherche un pseudo

📁 Créer le fichier **amis.php**

📁 Insère un formulaire qui permet de recherche un joueur connaissant son pseudo

📁 Donner la requête qui permet de sélectionner un pseudo et un score connaissant le pseudo

📁 Afficher en dessous du formulaire le pseudo, le score ainsi qu'une image (un "+" par exemple) permettant d'ajouter le pseudo à sa liste d'amis.


📁 Rendre l'image cliquable. Lorsque l'on clique sur l'image, on ajoute dans la BdD le pseudo rechercher à sa liste d'ami. Vous utiliserez le "id_joueurs" du login recherché afin de l'ajouter dans la table amis de la BdD


? Donner la requête permettant d'insérer un pseudo dans la table amis de la BD

📁 Compléter le fichier amis.php en ajoutant la requête précédente

I. Afficher la liste d'amis

? Donner la requête permettant de sélectionner, à partir de la table « amis », le pseudo des amis, le score connaissant le pseudo du joueur connecté.

 Exécuter la requête et afficher le résultat comme dans l'image précédente

 Ajouter une image avec le symbole "-". Lorsque l'on clique sur l'image, on supprime le joueur de la liste d'amis contenue dans la table "amis" de la BdD

Partie 5. Afficher les " high score "

Objectif :

Afficher le top 5 scores du jeu casse brique. Pour cette partie il faudra afficher les 5 meilleurs scores du jeu casse brique.

Résultat :

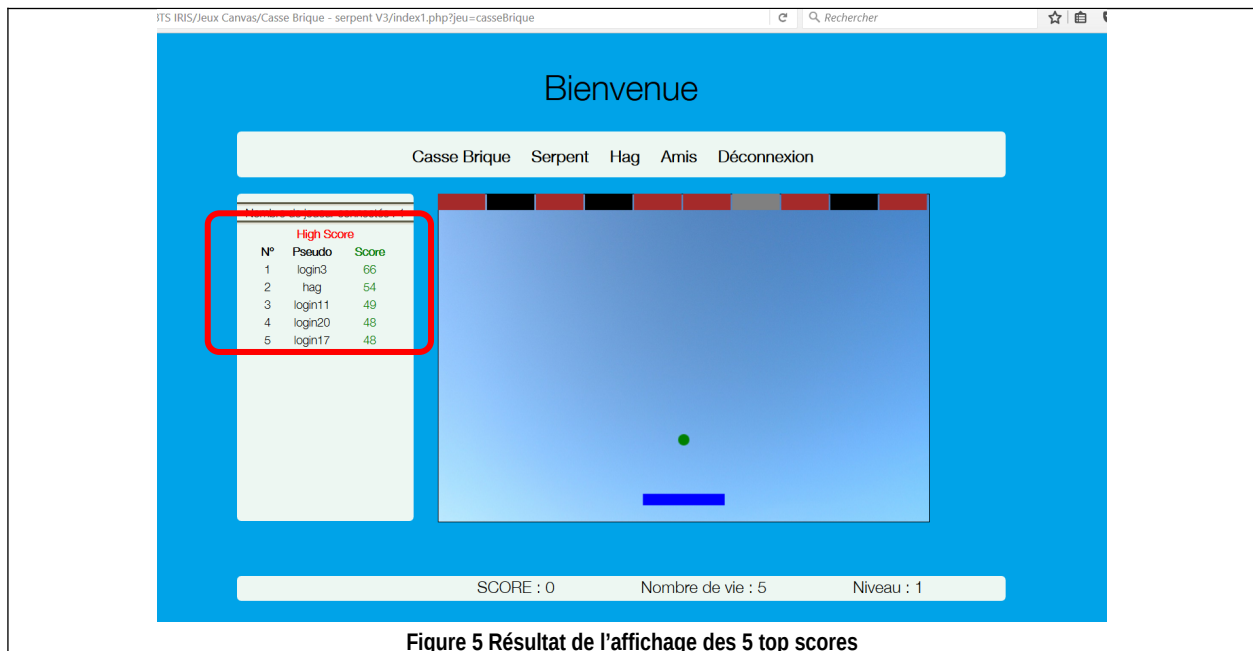


Figure 5 Résultat de l'affichage des 5 top scores

Procédure :

Pour cette partie, vous devrez faire une requête permettant de récupérer les scores des joueurs et n'en afficher que les 5 premiers.

? Donner la requête permettant de sélectionner le pseudo et le score associé par ordre croissant du score

? Exécuter cette requête dans phpmyadmin et valider le résultat obtenu dans le compte-rendu

☞ Modifier le fichier index.php afin d'exécuter la requête précédente lorsque l'on sélectionne le jeu pong

☞ Ajouter une division avec un id= 'nombreDeConnectes' dans l'**index.php**

☞ Exploiter le résultat de la requête afin d'obtenir un affichage des scores semblable à la figure précédente

? Montrer le bon fonctionnement à l'aide d'une impression d'écran

Partie 6. Nombre de connectés

Objectif :

Compter et afficher le nombre de joueurs connectés

Résultat :

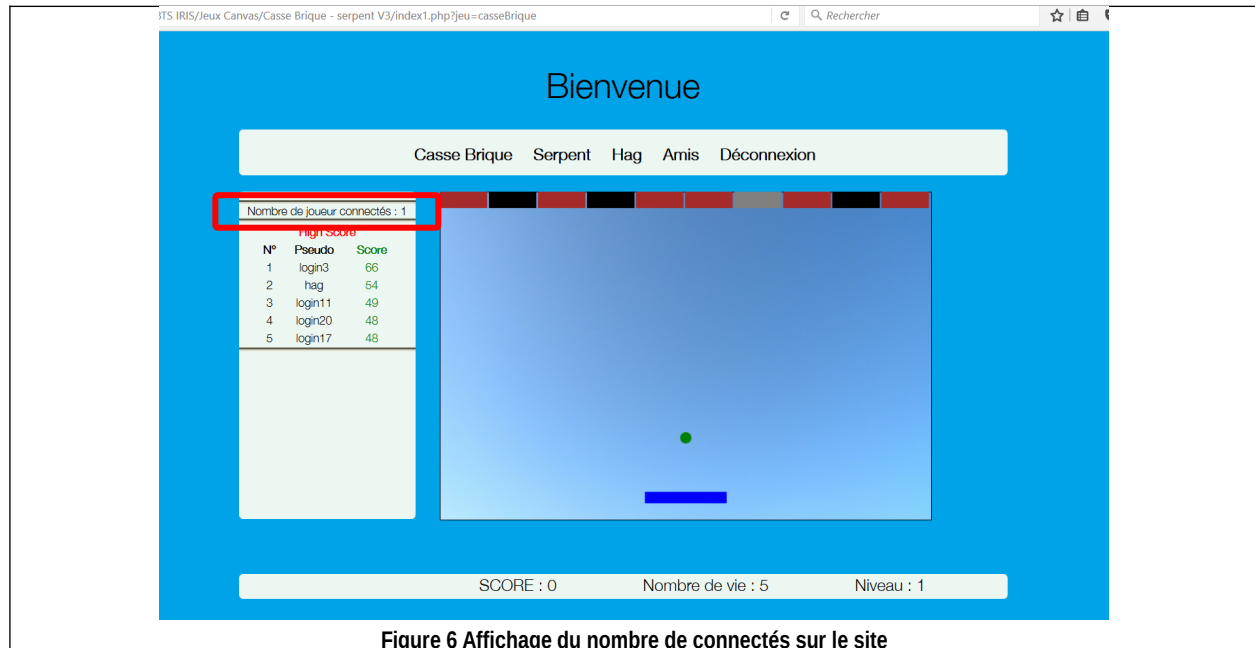


Figure 6 Affichage du nombre de connectés sur le site

Procédure :

Pour cette partie, vous allez devoir modifier la Bdd afin d'enregistrer les adresses IP et l'heure de connexion des joueurs. L'adresse IP sera en « varchar(45) » et l'heure de connexion sera en « int »

J. Requête comptant le nombre de connectés

Pour la requête suivante, vous utiliserez la fonction **count()** du langage SQL.

? En cherchant sur le net, donner un exemple d'utilisation de la fonction count()

🔍 Appliquer la fonction précédente à la nouvelle table « connectes » afin de récupérer le nombre de personnes connectées

🔍 Compléter la table « connectes », en y ajoutant des adresses IP et valider votre requête précédente dans phpmyadmin.

K. Afficher le nombre de connectés

🔍 Modifier le fichier **index.php** afin d'afficher le nombre de connectés dans une division avec une id='nombreDeConnectes'

Annexe 1. Données de la BdD

Cette Annexe vous aidera à exploiter les données récupérées depuis la Base de Données.

Procédure

Pour accéder en PHP aux données d'une base de données MySQL, les étapes sont :

- 1) `$id = mysqli_connect(...)` ; Connexion au serveur qui héberge la base de données et choix de la base de données. L'identifiant de la connexion `$id` vaut 0 en cas d'échec.
- 2) Préparation de la requête
- 3) `$res = mysqli_query ()` ; Envoi de requêtes MySQL. `$res` vaut 0 en cas d'échec ; En cas de succès, `$res` n'est pas directement affichable !! de plus, `$res` pourra contenir plusieurs enregistrements (plusieurs lignes de différentes tables).
- 4) `mysqli_fetch_row`, `mysqli_fetch_array`, `mysqli_fetch_assoc` : Extraction d'une réponse de `$res` ; chaque méthode renvoie un tableau associatif différent.
- 5) Affichage de la réponse en HTML sur le navigateur d'où provient la requête.
- 6) `mysqli_close ($id)` ; Déconnexion de la base de données

Exemple

Supposons que l'on souhaite récupérer et afficher tous les « `login` » et « `mail` » de la table `joueurs` dans la base de données « jeuHTML ». Sachant que le serveur est situé à l'adresse 172.20.183.20

- 1) `$id=mysqli_connect('172.20.183.20','snirlla','snirlla','jeuHTML')` ;
- 2) `$req=' select login, mail from joueurs '` ;
- 3) `$res=mysqli_query($req,$id)` ;
- 4) `$listeDesJoueurs= « »` ;
`while($rep=mysqli_fetch_assoc($res)){`
`$listeDesJoueurs.= « <p>le login est : $rep['login'] et l'adresse mail est:$rep['mail'] </p>» ;`
`}`
- 5) `echo $listeDesJoueurs` ;

Annexe 2. Exploitation de la barre URL
