

20/07/2022

PHP

Exercices de mise en situation



Rudy Lesur

Table des matières

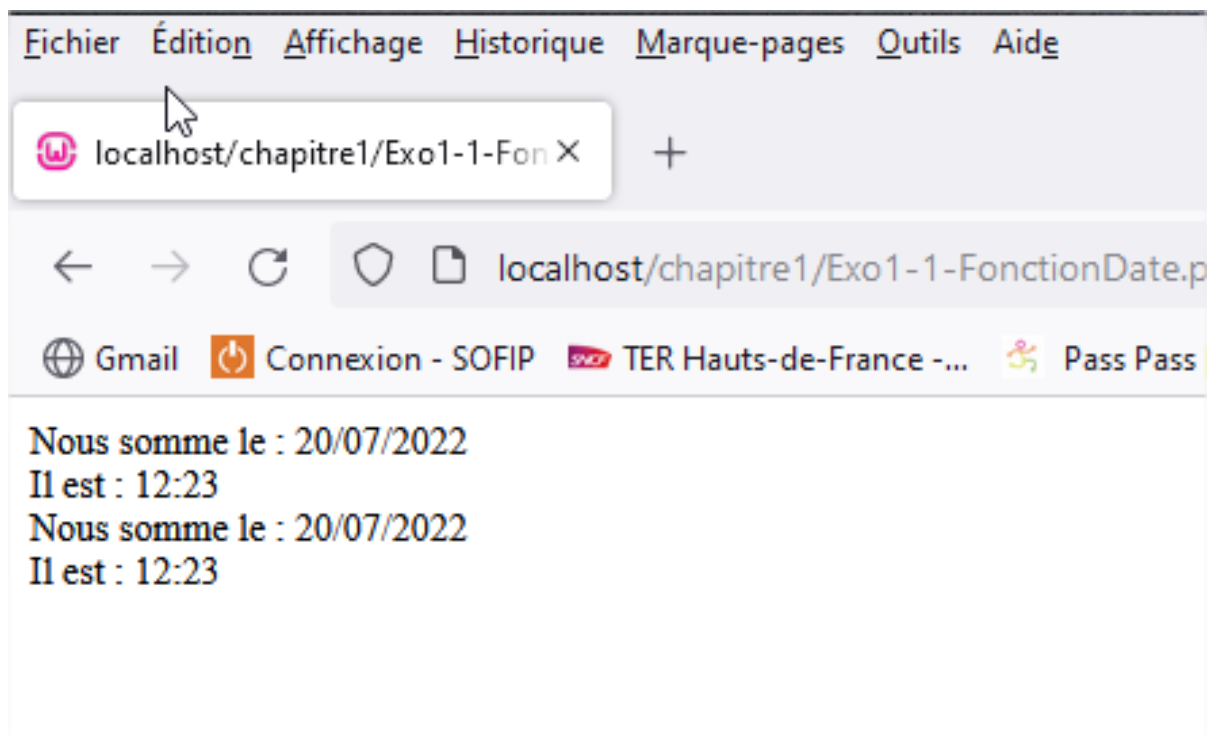
<u>I Prise en main</u>	<u>2</u>
1/ Appel de fonction (date)	2
2/ Variables d'environnement / Explode / Concaténation	3
3/ Ouverture / Lecture de fichiers	4
4/ Lecture / Ecriture dans un fichier	5
<u>II Les formulaires.....</u>	<u>6</u>
1/ Premier formulaire pour un sondage.....	6
2/ Exploitation des résultats du sondage	8
<u>III Accès au système de fichiers</u>	<u>9</u>
1/ Accès contenu d'un répertoire / Utilisation Expressions régulières (facultatif)	9
2/ Upload de fichiers	11
<u>IV Accès aux bases de données.....</u>	<u>13</u>

I Prise en main

Au cours de ce premier exercice, nous allons créer un premier programme contenant des instructions PHP. Nous appellerons notre première fonction (la fonction date) et nous utiliserons des fonctions bien utiles comme explode. Enfin, nous utiliserons également l'accès aux variables d'environnement.

1/ Appel de fonction (date)

On souhaite afficher dans une page Web la date et l'heure du jour. Le format désiré est le suivant :
Nous sommes le : 20/07/2022
Il est 12:00



Points abordés :

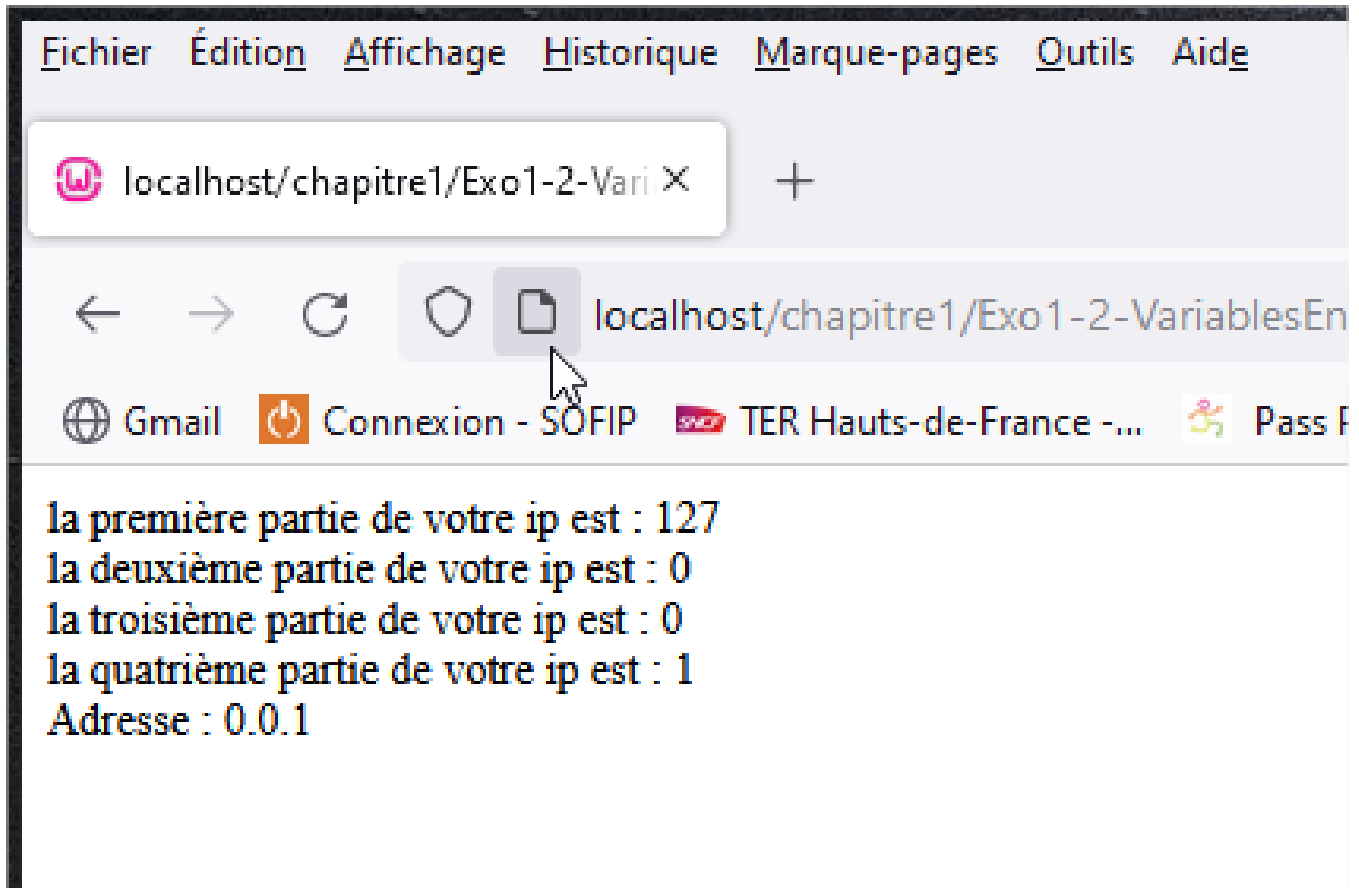
- La fonction date
- Print
Ce n'est pas une fonction mais une structure de langage. Il permet d'afficher une chaîne de caractères.

2/ Variables d'environnement / Explode / Concaténation

On veut récupérer l'adresse IP de la machine cliente. Une fois récupérée, on veut la “ découper ” de manière à savoir si l'adresse IP est située en France (IP : 192) ou un autre pays puis à afficher l'adresse dans le domaine correspondant.

Attention :

L'adresse 127.0.0.1 dénote l'adresse de bouclage (localhost – la machine elle-même).



Points abordés :

- Le tableau associatif \$_SERVER et l'élément [REMOTE_ADDR](#).
- La fonction Explode

3/ Ouverture / Lecture de fichiers

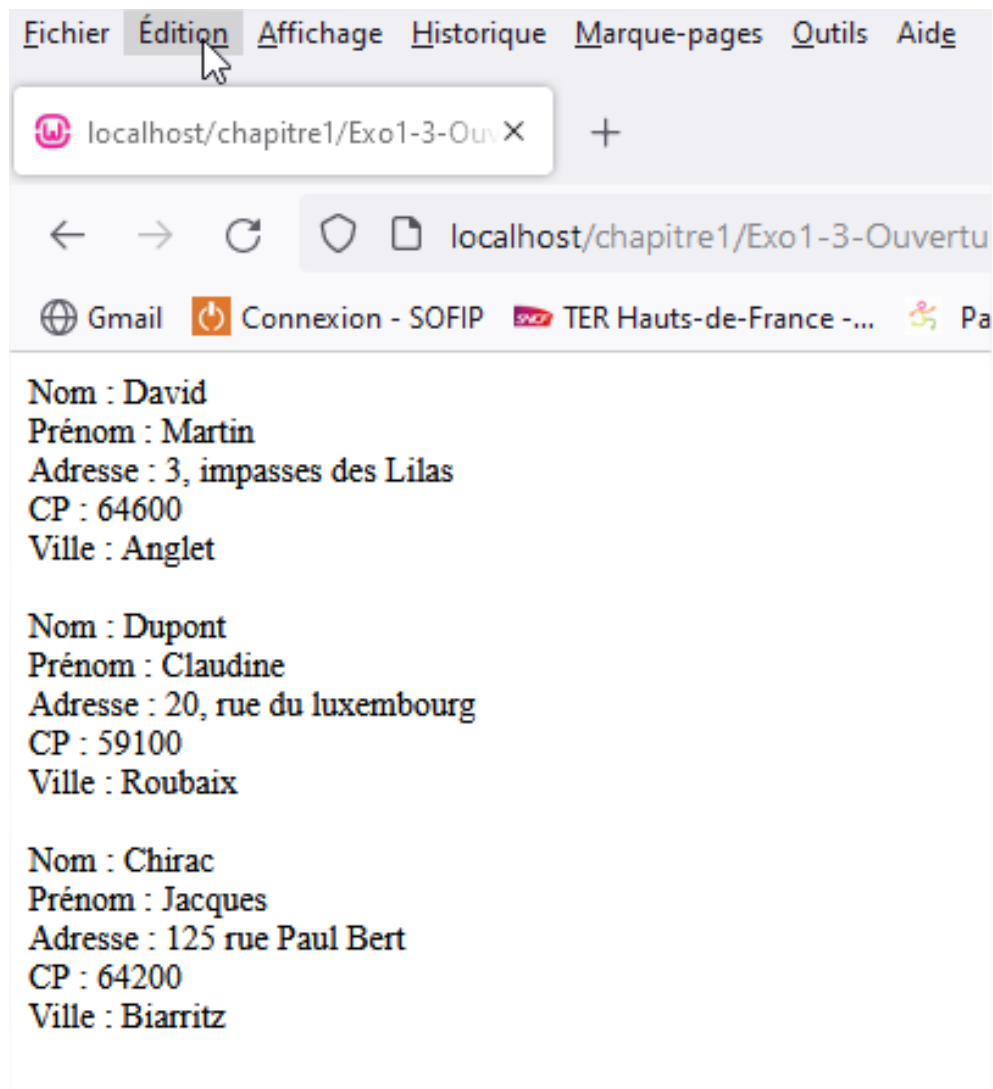
On souhaite créer avec un éditeur de texte le fichier suivant (appelé calepin.txt).

David | Martin | 3, impasse des Lilas | 64600 | Anglet

Dupont | Claudine | 20, rue du Luxembourg | 59100 | Roubaix

Chirac | Jacques | 125 avenue Paul Bert | 64200 | Biarritz

A la création du fichier texte, appuyez sur Entrée après chaque ligne. On va ouvrir ce fichier en PHP et réaliser un affichage comme suit :



Points abordés :

- La gestion des fichiers avec PHP : les fonctions fopen(), fputs(), fgets(), feof(), fclose(),

4/ Lecture / Ecriture dans un fichier

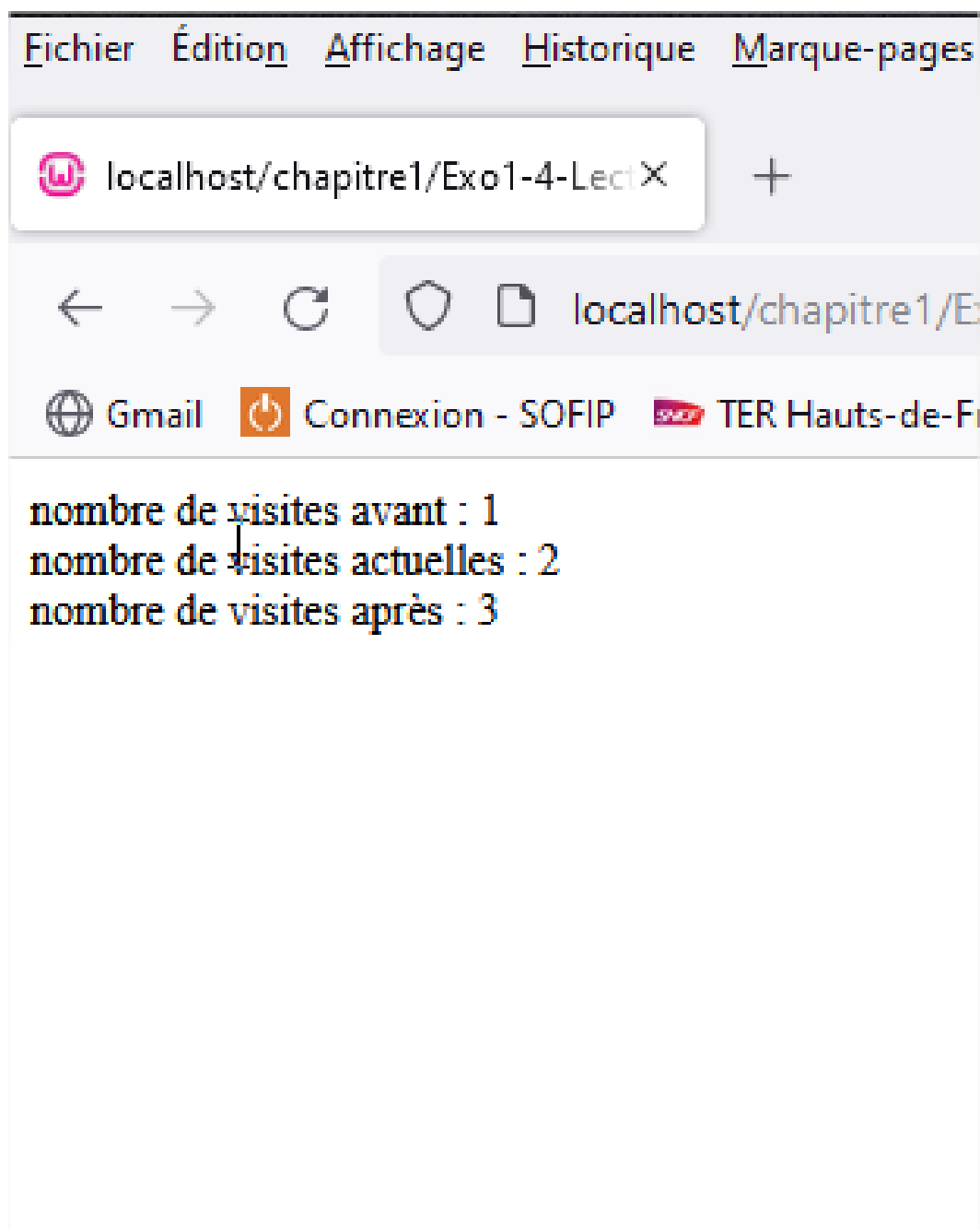
On désire réaliser un compteur de visites dans une page d'accueil. Celui-ci doit se mettre à jour à chaque chargement de la page, même si l'adresse IP de la personne est identique (cas entre autres des personnes passant par un proxy).

Plusieurs solutions sont possibles :

Soit en ouvrant le fichier contenant le compteur en lecture, le fermer puis le rouvrir en mode écriture.

Soit en ouvrant une seule fois le fichier en mode mise-à-jour (qui permet la lecture et la mise à jour).

Le compteur est mémorisé dans un fichier .txt nommé cpt.txt.



II Les formulaires

Objectif : Prendre en main l'utilisation et l'exploitation des formulaires (méthodes GET et POST).
Utiliser les fichiers.

1/ Premier formulaire pour un sondage

On désire créer un formulaire pour un site de dons pour des associations :

- Nom
- Age
- Mail
- Valeur en € du don

A chaque validation de formulaire (appui sur un bouton 'OK'), on récupèrera les informations entrées par le visiteur. Ces informations seront enregistrées dans un fichier appelé '*resultats.txt*' sous le format suivant :

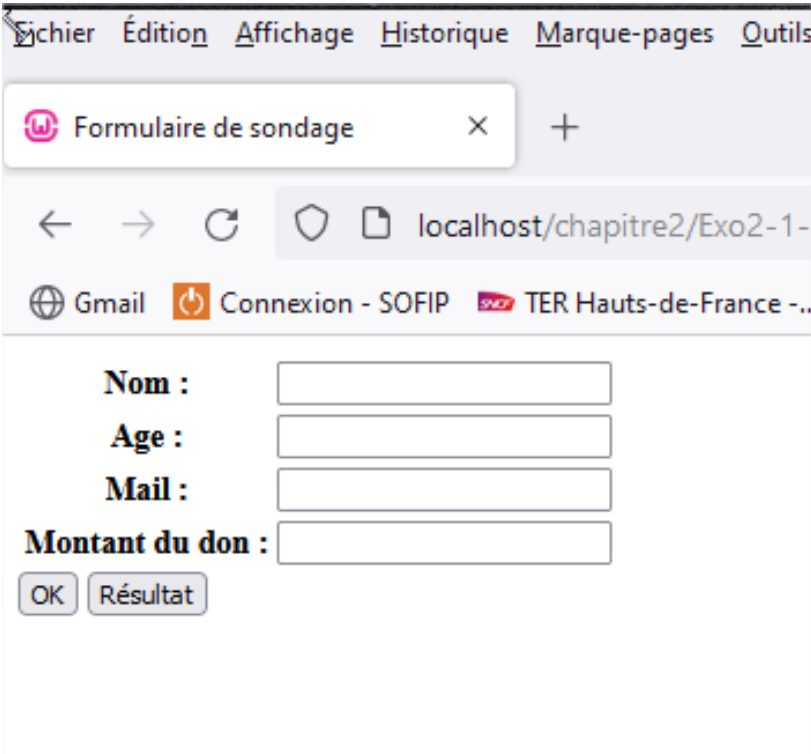
Nom1 | Age1 | Mail1 | Don1 Nom2 | Age2 | Mail2 | Don2

On proposera une trace à l'écran des informations récupérées.

Nous avons donc 3 pages à créer :

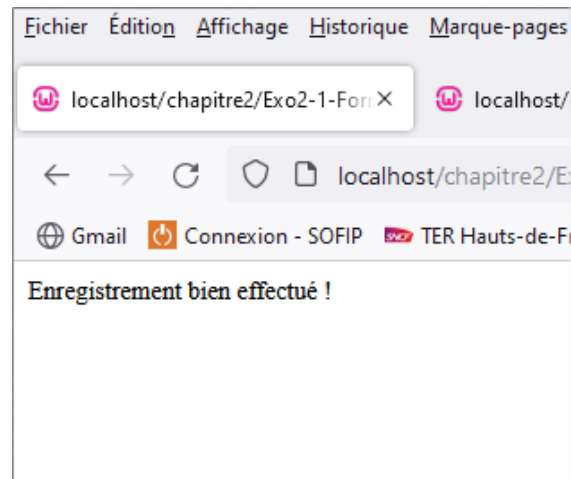
La page d'accueil qui sera une page d'accueil qui nous permet de saisir les informations dans un formulaire.

Une page PHP appelée sur le submit du bouton OK. Dans cette page, nous enregistrerons les informations saisies sur la page d'accueil.



The screenshot shows a web browser window with the title 'Formulaire de sondage'. The address bar displays 'localhost/chapitre2/Exo2-1-'. The form contains four input fields with labels: 'Nom :', 'Age :', 'Mail :', and 'Montant du don :'. Below the input fields are two buttons: 'OK' and 'Résultat'.

Une fois l'écriture dans le fichier resultats.txt effectuée, nous envoyons un message de confirmation.



Une 3^{ème} page PHP appelée sur le submit du bouton Résultats. Dans cette page, nous ouvrons le fichier resultats.txt et affichons les informations qu'il contient.

Tester les méthodes Get et Post.

Nous utiliserons les tableaux associatifs globaux \$_POST ou \$_GET.

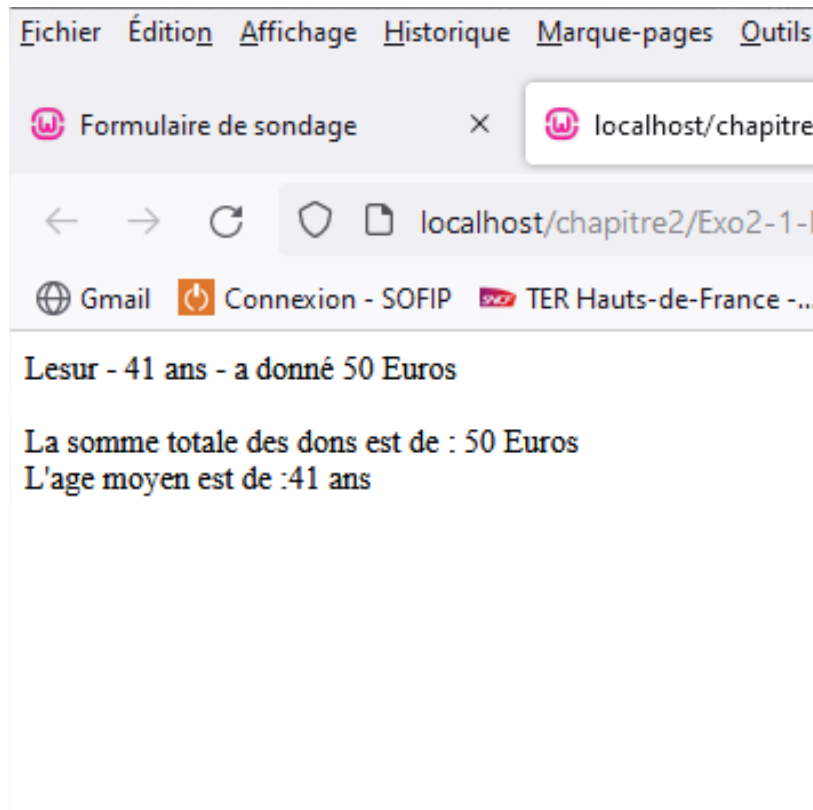
Chaque élément de ces tableaux est accessible par le name donné dans le formulaire. Un élément sans name ne sera pas transmis.

2/ Exploitation des résultats du sondage

On souhaite maintenant exploiter les résultats du sondage (bouton '*resultats*' sur la page d'accueil précédente).

On va envoyer un mail (avec trace à l'écran) à chacun des donneurs. Ce mail contiendra un rappel de son propre don, la somme globale reçue ainsi que la moyenne d'âges des personnes ayant répondu.

Nous modifierons le contenu de la 3^{ème} page précédente.



Points abordés :

- Fonction mail

Il existe de nombreuses bonnes raisons pour être amené à vouloir envoyer un mail via PHP :

- Email de confirmation d'inscription.
- Alerte webmaster lorsqu'un utilisateur propose un lien

Malheureusement assez peu d'hébergeurs (gratuits) proposent cette fonctionnalité et cela se comprend aisément.

En effet, il est si simple avec ce procédé d'envoyer des centaines d'emails à des adresses stockées en base de données que les hébergeurs sont soumis à 2 risques principaux :

- Voir leurs ressources systèmes diminuer fortement (au risque d'écrouler le serveur)
- Etre accusé de faciliter le "spamming".

III Accès au système de fichiers

Objectif :

Accès au contenu des disques serveur (là où sont stockés les fichiers).

Accès au contenu des répertoires, sous répertoires et exploitation des fichiers distants.

Utilisation de la fonction `preg_match` (la fonction `ereg` est obsolète) permettant l'utilisation d'expressions régulières.

Enfin, mise en œuvre de la fonction d'upload permettant de transférer des fichiers du poste local sur le serveur.

1/ Accès contenu d'un répertoire / Utilisation Expressions régulières (facultatif)

On désire écrire un programme qui affichera le contenu du répertoire spécifié dans une variable.

Selon les extensions des fichiers, il proposera une couleur d'affichage différente, sa signification et sa taille en octets.

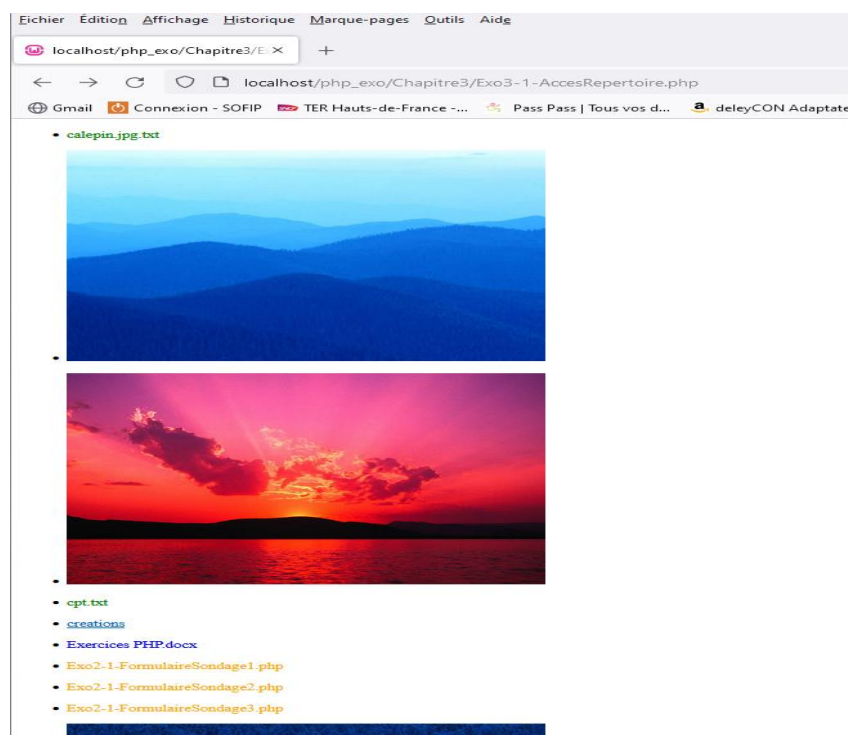
Les extensions reconnues sont :

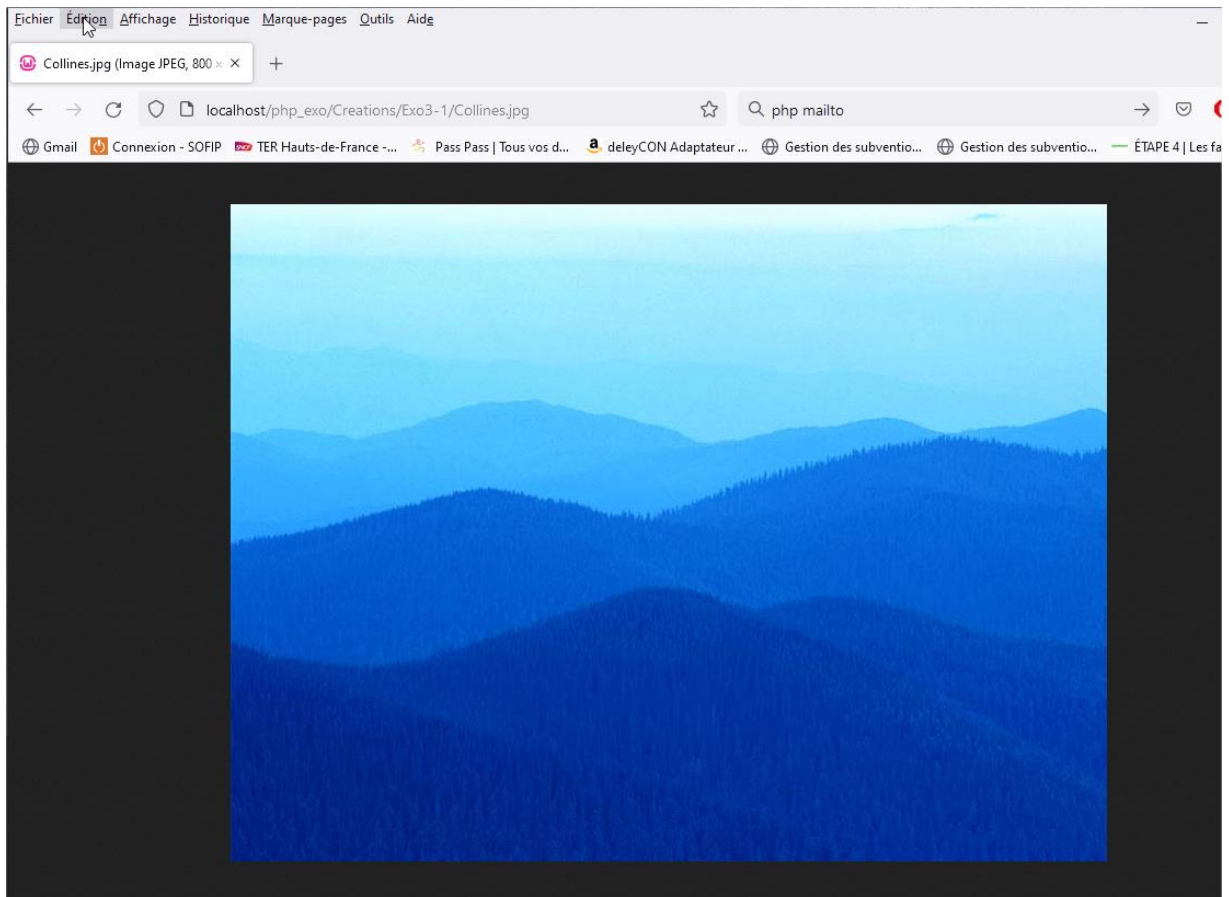
- **.txt** (document texte) : affichage en vert
- **.docx, .xlsx, .pptx**(document office) : affichage en bleu
- **.php, .asp** (document script) : affichage en rouge

Pour les extensions relatives à des images, on affichera directement l'image au ¼ de sa taille normale.

On proposera un lien dessus permettant de la visionner en taille réelle.

Pour cet exercices faite vous un dossier contenant des images des fichiers txt, des fichiers word excel etc..





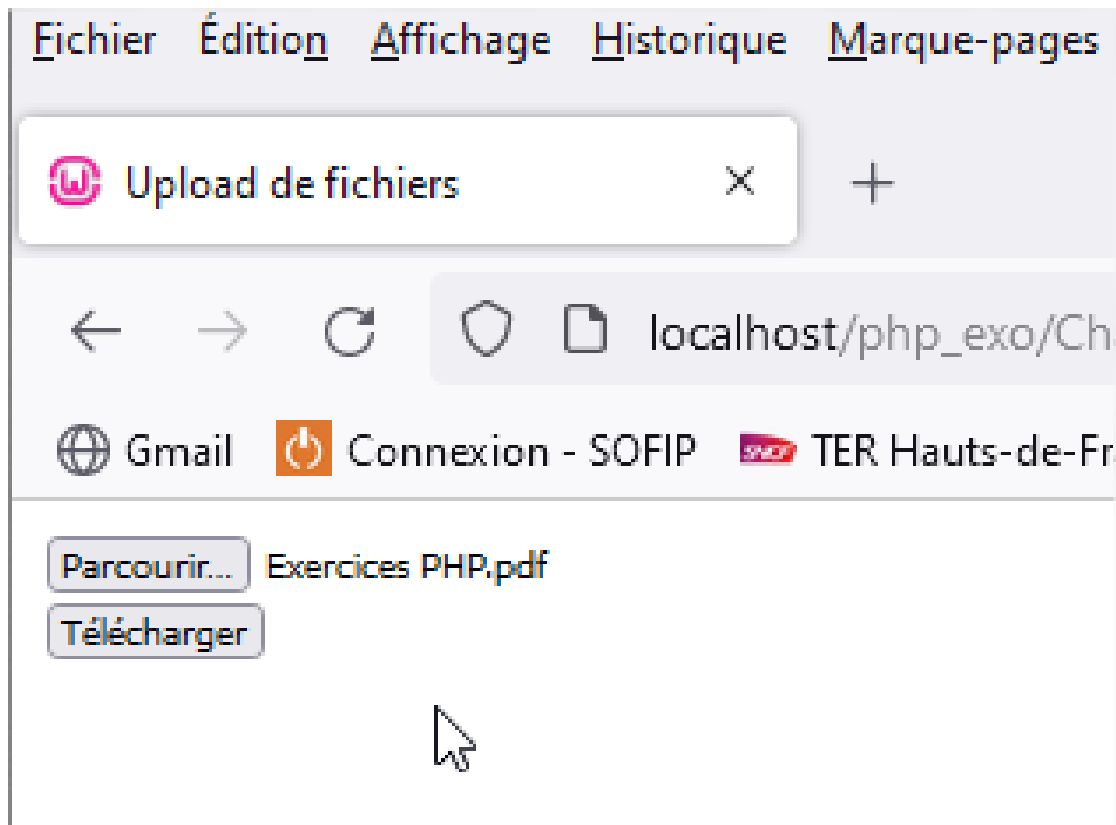
Si je clique sur une image :

Points abordés :

- Preg_match
N'utilisez pas preg_match() si vous voulez uniquement savoir si une chaîne est contenue dans une autre. Utilisez dans ce cas les fonctions strpos() ou strstr(), qui sont beaucoup plus rapides.

2/ Upload de fichiers

On va créer une page HTML qui permettra le téléchargement de fichiers sur le serveur (upload).



Points abordés :

- La balise `<input>` de type `file`
- L'attribut `ENCTYPE` de la balise `<form>`

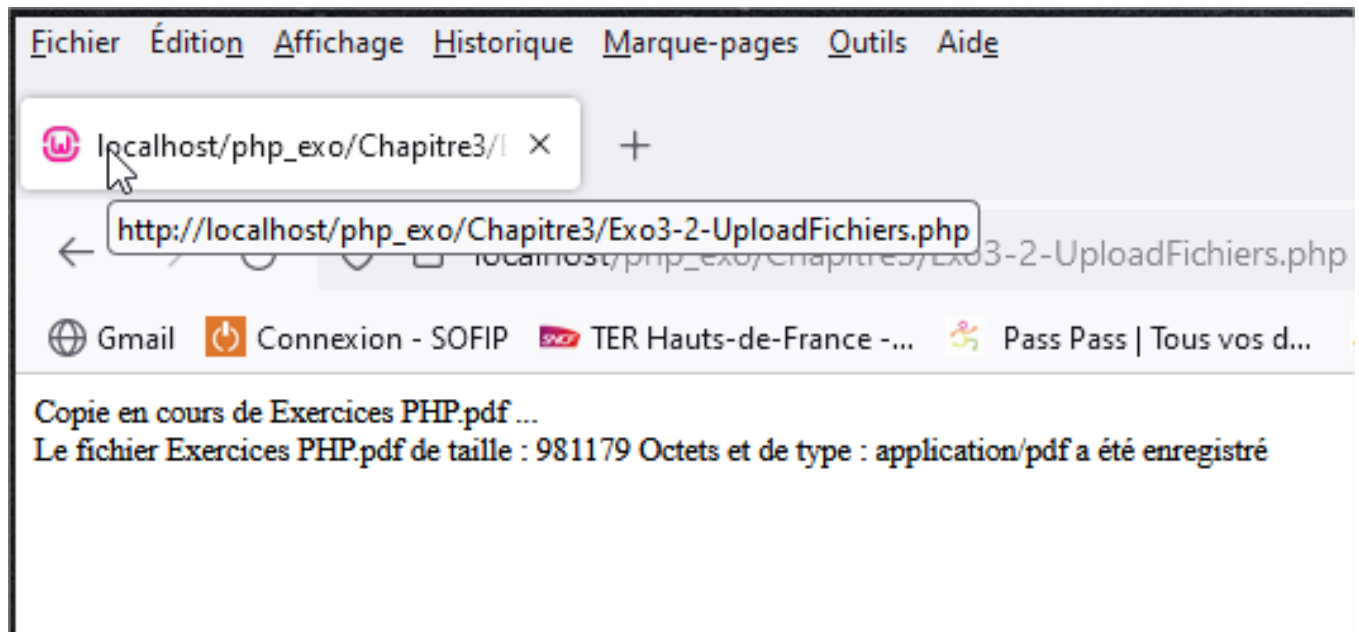
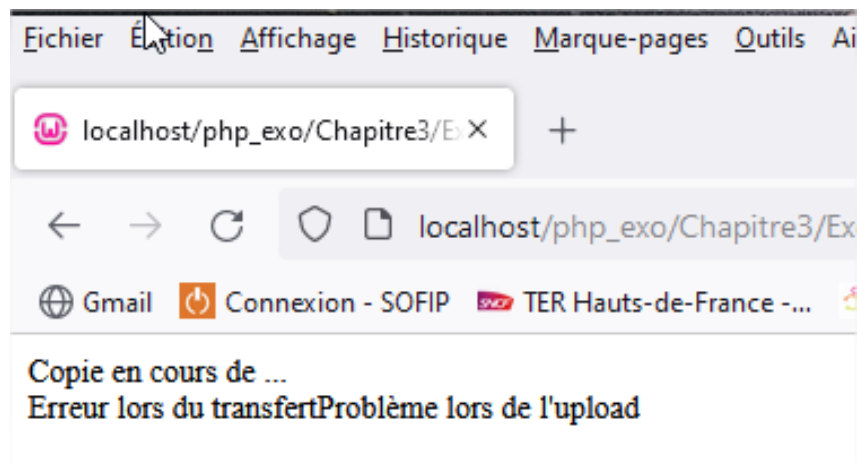
`ENCTYPE` spécifie l'encodage utilisé pour la forme que prendra le contenu du formulaire. Ce paramètre ne peut être utilisé qu'accompagné par la méthode `POST`. `ENCTYPE` peut prendre plusieurs valeurs :

`ENCTYPE="application/x-www-form-urlencoded"` valeur par défaut; encode le contenu du formulaire selon une forme URL qui est difficilement lisible par le destinataire;
`http://Votre.Serveur.com/cgi-bin/VotreScript?ObjetDeFormulaire1=valeur1&...`

`ENCTYPE="text/plain"` le contenu du formulaire sera retourné en format texte lisible par le destinataire ; option accompagnée le plus souvent de `ACTION=mailto:`

`ENCTYPE="multipart/form-data"` permet d'expédier un fichier attaché au message Le clic sur le bouton `Télécharger` appellera une page PHP qui effectuera le téléchargement du fichier. Afficher un message pour indiquer que la copie est en cours.

Afficher un message lorsque l'enregistrement est terminé (ou éventuellement un message d'erreur si problème de chargement).



Points abordés :

Le tableau associatif \$_FILESmove_uploaded_file()

IV Accès aux bases de données

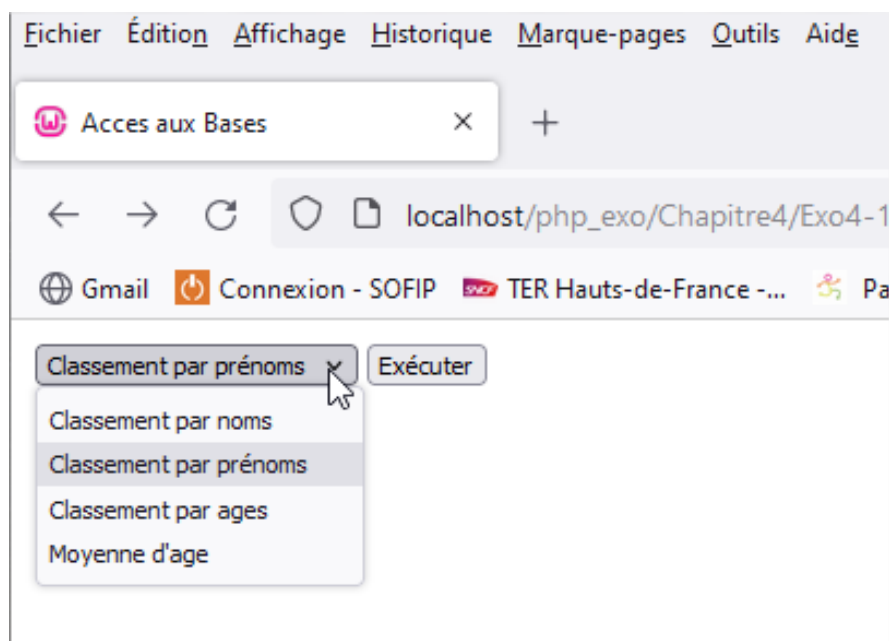
Objectif : cet exercice va permettre de voir les mécanismes nécessaires à la connexion à un SGBD de type MySQL (pour un autre SGBD, il faudra simplement adapter les noms de fonctions).

Vous créerez sur votre compte MySQL une table agenda avec les informations suivantes : *nom* (50 car. Max), *prenom* (50 car. Max), *adresse* (255 car. Max), *age* (entier), *telephone* (10 car. Max).

Vous remplirez cette table avec 4 à 5 enregistrements.

Une fois ceci réalisé, vous proposerez une page permettant d'interroger la base, pour obtenir

- Un classement par nom,
- Un classement par âge,
- La moyenne d'âge des personnes.



Fichier Édition Affichage Historique Marque-page

Accès aux Bases × localhost

localhost/php_exo/C

Gmail Connexion - SOFIP TER Hauts-de-

connexion reussie

Classement par noms

GREY
Jean
Xavier School

35

KILE
Selina
Gotham City

30

LESUR
Rudy
Douai

41

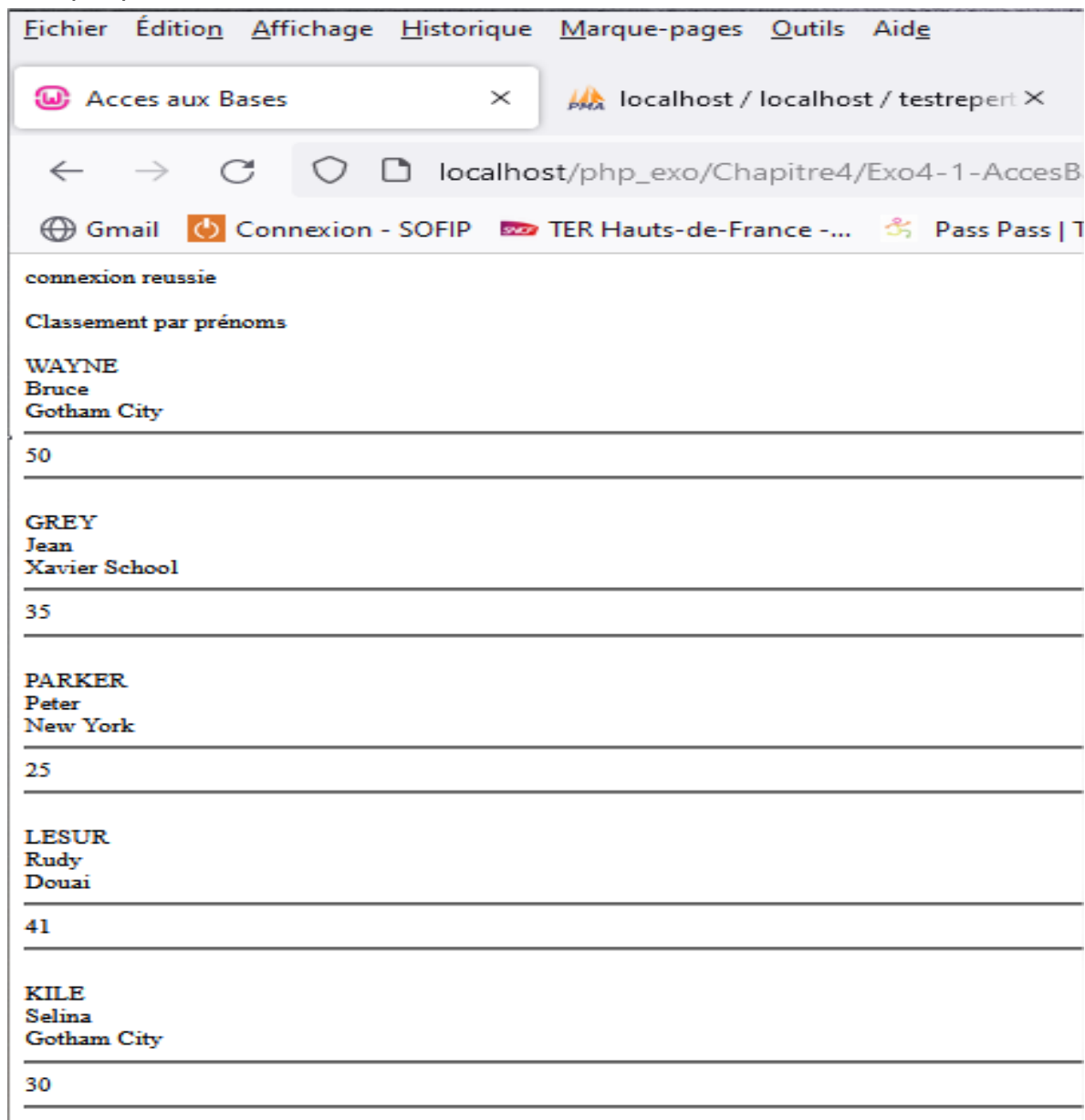
PARKER
Peter
New York

25

WAYNE
Bruce
Gotham City

50

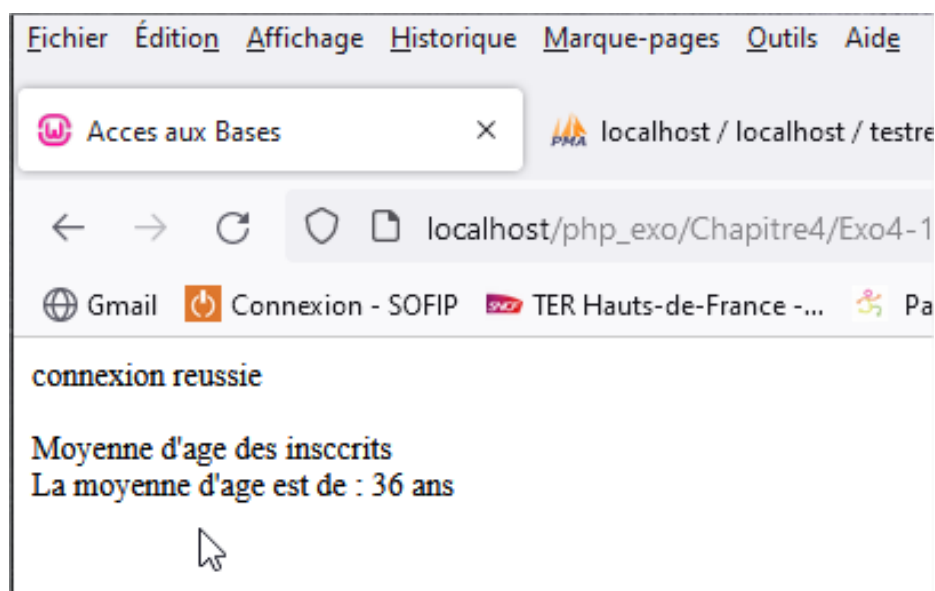
Classement par prénoms :



The screenshot shows a web browser window with the address bar displaying 'localhost/php_exo/Chapitre4/Exo4-1-AccesB'. The page content includes a message 'connexion reussie' and a table titled 'Classement par prénoms'. The table lists five users: WAYNE (Bruce, Gotham City, 50), GREY (Jean, Xavier School, 35), PARKER (Peter, New York, 25), LESUR (Rudy, Douai, 41), and KILE (Selina, Gotham City, 30).

Classement par prénoms			
WAYNE	Bruce	Gotham City	50
GREY	Jean	Xavier School	35
PARKER	Peter	New York	25
LESUR	Rudy	Douai	41
KILE	Selina	Gotham City	30

Moyenne d'âge :



The screenshot shows a web browser window with the address bar displaying 'localhost/php_exo/Chapitre4/Exo4-1'. The page content includes a message 'connexion reussie' and a text block stating 'Moyenne d'age des inscrits' and 'La moyenne d'age est de : 36 ans'.

Moyenne d'age des inscrits	
La moyenne d'age est de : 36 ans	

Points éventuellement abordés :

- Classe PDO
- Méthode query()
- Méthode prepare()
- Méthode execute()
- Méthode exec() die()
- Gestion des erreurs
- close()
- errorInfo()
- print_r()