

# Des statistiques pour votre site !

Par Serialtueur



[www.openclassrooms.com](http://www.openclassrooms.com)

*Licence Creative Commons 6 2.0  
Dernière mise à jour le 9/03/2010*

## Sommaire

Sommaire .....	2
Des statistiques pour votre site ! .....	3
TP n°1 : combien ai-je de visites ? .....	3
Nombre de visites .....	3
Nombre de visites par jour .....	4
TP n°2 : qui est connecté sur mon site ? .....	6
Affichage du nombre de visiteurs connectés .....	6
Affichage du record de connectés .....	8
Affichage de la liste des connectés .....	9
TP n°3 : mais d'où viennent mes visiteurs ? .....	10
3, 2, 1... Correction ! .....	10
Affichage : .....	11
TP n°4 : affluences et heures de pointe .....	12
Correction : Enregistrement .....	13
Correction : Affichage .....	13
Affichage graphique .....	14
Partager .....	15



# Des statistiques pour votre site !

Par Serialtueur

Mise à jour : 09/03/2010

Difficulté : Facile



Référencement, publicité, vous devez les maîtriser pour pouvoir mettre votre site en valeur. Mais, afin d'optimiser un maximum vos efforts, pourquoi ne pas faire des statistiques ?

Par statistiques, j'entends à la fois vos performances (visiteurs connectés, nombre maximum de visiteurs connectés, pages visitées) mais aussi l'impact qu'a votre publicité sur les utilisateurs (provenance, intérêt). Celles-ci vont vous permettre de pouvoir gérer vous-mêmes vos campagnes afin de faire connaître le mieux possible votre site !

Ce tutoriel est une sorte de cours / TP guidé, garni d'infos qui vont vous permettre de mettre pleins de statistiques partout sur votre site. 😊

Sommaire du tutoriel :



- TP n°1 : combien ai-je de visites ?
- TP n°2 : qui est connecté sur mon site ?
- TP n°3 : mais d'où viennent mes visiteurs ?
- TP n°4 : affluences et heures de pointe

## TP n°1 : combien ai-je de visites ?

Ce script devra se diviser en plusieurs parties :

- nombre de visites ;
- nombre de visites par jour (nombre moyen, nombre maximum).



Certaines parties du script ressemblent fortement à celles du [tutoriel officiel de M@teo21](#).

J'en tiens compte mais sachez que ma correction sera différente. Vous pouvez néanmoins vous en inspirer si vous bloquez.

De plus les scripts seront en général plus complets car nous allons beaucoup travailler dessus.

## Nombre de visites

Nous avons juste à stocker un nombre, celui du nombre de visites. En effet, ce script s'exécute sur toutes les pages de votre site. À chaque visite on devra incrémenter ce nombre.

Il est évident que nous n'allons pas stocker ce nombre dans une base de données SQL, la connexion est bien trop lente et si on devait se reconnecter à chaque ouverture de page pour afficher UN NOMBRE, le serveur ralentirait inutilement.

Un fichier contenant ce nombre semble donc parfaitement indiqué.

On rangera tous les scripts de ce tutoriel dans un dossier "stats".

Ce fichier s'appellera visiteurs.txt et il contiendra une seule ligne, le nombre de visiteurs. Pour faire marcher votre script à la première utilisation, on l'initialise en mettant comme valeur 0.



N'oubliez pas de régler le CHMOD à 777 !

Voici comment se divisera votre script :

- ouverture en lecture et écriture du fichier visiteurs.txt ;

- lecture de l'unique ligne du fichier contenant le nombre de visiteurs ;
- incrémentation de ce nombre ;
- écriture du nombre incrémenté dans le fichier ;
- affichage du nombre incrémenté.

Nous utiliserons l'extension `mysqli` dans ce tutorial.

Avant tout le script du tutorial, on se connecte à notre base de données comme ceci :

**Code : PHP**

```
<?php
$connexion = mysqli_connect('host', 'username', 'password');
$db = mysqli_select_db($connexion, 'stats');
?>
```

### Correction !

Avouez que ce n'était pas bien dur !

Comment ça vous n'avez pas réussi ? 😬

Si c'est le cas, il serait préférable de revoir vos bases, non ?

**Code : PHP**

```
<?php
$nb_visites = file_get_contents('data/pagesvues.txt');
$nb_visites++;
file_put_contents('data/pagesvues.txt', $nb_visites);
echo 'Nombre de pages vues : <strong>' . $nb_visites .
'</strong><br/>';
?>
```

Comme vous pouvez le voir, ce n'était pas compliqué. 😊

On attaque le...

## Nombre de visites par jour

Vous l'aurez compris, ce script est applicable pour les mois, les années, bref ici on prendra les jours car c'est le plus utilisé. Votre mission (si vous l'acceptez 😊) sera d'obtenir les statistiques suivantes pour les afficher sur votre site :

- nombre de visites aujourd'hui ;
- garder en mémoire les nombres de visites de chaque jour ;
- afficher le nombre maximum de visites en un jour suivi de la mention « Établi le ... » ;
- faire la moyenne du nombre de visites par jour.

Eh oui, ça se complexifie. 😬



Il est tout à fait possible d'utiliser un fichier pour ce script, mais par mesure de confort et de facilité, nous allons faire ça avec une table SQL. C'est plus simple pour débiter.

Allez, cette fois je donne pas plus d'indications. 😬

Bonne chance !



Si vous avez du mal avec les données temporelles de MySQL, je vous conseille ce tutorial : [MySQL et les données temporelles](#).

### 1, 2, 3... Correction !

Alors, c'est plus compliqué que ça en a l'air hein ?



Ne vous inquiétez pas si vous n'y êtes pas arrivé directement, servez-vous de la correction pour compléter votre code.

On commence par la structure de la table SQL que je vais utiliser.

**Code : SQL**

```
--  
-- Structure de la table `visites_jour`  
--  
  
CREATE TABLE `visites_jour` (  
  `visites` mediumint(9) NOT NULL,  
  `date` date NOT NULL  
);
```

On voit donc que c'est une table très basique, mais nous n'aurons pas besoin de plus.

Passons au code PHP.

**Code : PHP**

```
<?php  
//ETAPE 1 - Affichage du nombre de visites d'aujourd'hui  
$retour_count = mysqli_query($connexion, 'SELECT COUNT(*) AS  
nbre_entrees FROM visites_jour WHERE date=CURRENT_DATE() '); //On  
compte le nombre d'entrées pour aujourd'hui  
$donnees_count = mysqli_fetch_assoc($retour_count); //Fetch-array  
echo 'Pages vues aujourd'hui : <strong>'; // On affiche tout de  
suite pour pas le retaper 2 fois après  
  
    if ($donnees_count['nbre_entrees'] == 0) //Si la date  
d'aujourd'hui n'a pas encore été enregistrée (première visite de la  
journée)  
    {  
        mysqli_query($connexion, 'INSERT INTO visites_jour(visites, date)  
VALUES (1, CURRENT_DATE());'); //On rentre la date d'aujourd'hui et  
on marque 1 comme nombre de visites.  
        echo '1'; //On affiche une visite car c'est la première visite de  
la journée  
    } else { //Si la date a déjà été enregistrée  
        $retour = mysqli_query($connexion, 'SELECT visites FROM  
visites_jour WHERE date=CURRENT_DATE() '); //On sélectionne l'entrée  
qui correspond à notre date  
        $donnees = mysqli_fetch_assoc($retour);  
        $visites = $donnees['visites'] + 1; //Incrémentement du nombre de  
visites  
        mysqli_query($connexion, 'UPDATE visites_jour SET visites =  
visites + 1 WHERE date=CURRENT_DATE() '); //Update dans la base de  
données  
        echo $visites; //Enfin, on affiche le nombre de visites  
d'aujourd'hui !  
    }  
    echo '</strong></br>';  
  
//ETAPE 2 - Record des connectés par jour  
  
$retour_max = mysqli_query($connexion, 'SELECT visites, date FROM  
visites_jour ORDER BY visites DESC LIMIT 0, 1'); //On sélectionne
```

```

l'entrée qui a le nombre visite le plus important
$donnees_max = mysqli_fetch_assoc($retour_max);
echo 'Record : <strong>' . $donnees_max['visites'] . '</strong>
établi le <strong>' . $donnees_max['date'] . '</strong><br/>'; //On
l'affiche ainsi que la date à laquelle le record a été établi

//ETAPE 3 - Moyenne du nombre de visites par jour
$total_visites = 0; //Nombre de visites
/*(pour éviter les bugs on ne prendra pas le nombre du premier
exercice,
mais celui-ci reste utile pour être affiché sur toutes les pages
car il est plus rapide,
contrairement à $total_visites dont on ne se servira que pour la
page de stats)*/

$total_jours = 0; //Nombre de jours enregistrés dans la base

$total_visites = mysqli_fetch_assoc(mysqli_query($connexion,
'SELECT SUM(visites) FROM visites_jour AS total_visites'));
$total_visites = $total_visites['total_visites'];

$total_jours = mysqli_fetch_assoc(mysqli_query($connexion, 'SELECT
COUNT(*) FROM visites_jour AS total_jours'));
$total_jours = $total_jours['total_jours'];

$moyenne = $total_visites/$total_jours; //on fait la moyenne
echo 'Moyenne : <strong>' . $moyenne . '</strong> visiteurs par
jour<br/>'; // On affiche ! Terminé !!!

?>

```



Vous pouvez ajouter un champ "id\_page" dans la table qui vous permettra d'enregistrer le nom de la page qui a été vue et ainsi voir les pages qui intéressent le plus vos internautes. 😊

## TP n°2 : qui est connecté sur mon site ?

Ce script se déroulera en trois parties :

- affichage du nombre de visiteurs connectés ;
- affichage du record du nombre de connectés suivi de la date où il a été établi ;
- affichage de la liste des connectés avec leurs IP et la page qu'ils visitent (il est déconseillé d'afficher les IP des visiteurs publiquement, on préférera séparer cette partie dans le panel admin).

### Affichage du nombre de visiteurs connectés

Mettons-nous d'accord sur un point : il est impossible de réellement savoir qui lit nos pages. En effet, même si l'internaute reste connecté pour lire un article de notre site pendant 1 heure, on ne le saura pas, on va donc faire des **approximations**.

Nous allons donc considérer que :

- lorsqu'un internaute affiche notre page, il reste connecté dessus 5 minutes ;
- si au bout des 5 minutes il n'a pas changé de page (donc s'il n'en a pas chargé une autre) on le supprime, il n'est plus connecté ;
- nous allons identifier les connectés grâce à leur IP ;
- quand l'internaute change de page, on recommence le décompte de 5 minutes.



Nous allons faire un truc intéressant : on va aussi mettre dans un champ la page où se trouve le visiteur, ce qui permettra de faire une liste des visiteurs connectés plus complète !



Je vous rappelle que l'IP d'un visiteur se récupère grâce à la variable Superglobale `$_SERVER['REMOTE_ADDR']` . Pour la convertir en nombre, utilisez la fonction `ip2long()`. Pour convertir ce nombre en IP, utilisez `long2ip()`. Cela facilitera l'enregistrement dans la base de données.

Si vous bloquez, vous pouvez aller [voir ce TP](#) qui est (très) similaire et qui vous donnera quelques indications en plus. 😊

### 3, 2, 1, correction !

Structure de la table 'connectes' :

Code : SQL

```
CREATE TABLE IF NOT EXISTS `connectes` (
  `ip` int(12) NOT NULL,
  `timestamp` int(11) NOT NULL,
  `page` varchar(50) NOT NULL
);
```

Script PHP :

Code : PHP

```
<?php
//On prend l'adresse de la page à laquelle on enlève le \ du début
(1er caractère) :
$page = substr($_SERVER['PHP_SELF'], 1);
// On stocke dans une variable le timestamp qu'il était il y a 5
minutes :
$timestamp_5min = time() - (60 * 5); // 60 * 5 = nombre de secondes
écoulées en 5 minutes
//On commence par virer les entrées trop vieilles (+ de 5 minutes)
mysqli_query($connexion, 'DELETE FROM connectes WHERE timestamp < '
. $timestamp_5min);

$retour = mysqli_query($connexion, 'SELECT COUNT(*) AS nb_connectes
FROM connectes WHERE ip=\'' . ip2long($_SERVER['REMOTE_ADDR']) .
'\'' );
$donnees = mysqli_fetch_assoc($retour); //On regarde si le visiteur
est déjà dans la table

if ($donnees['nb_connectes'] == 0) // Si il n'y est pas, on l'ajoute
{
    mysqli_query($connexion, 'INSERT INTO connectes(ip, timestamp,
page) VALUES(\'' . ip2long($_SERVER['REMOTE_ADDR']) . '\', NOW()),
\'' . $page . '\')');
}
else // Sinon, on remet le décompte de 5 minutes à 0
{
    mysqli_query($connexion, 'UPDATE connectes SET timestamp=NOW(),
page=\'' . $page . '\'' WHERE ip=\'' .
ip2long($_SERVER['REMOTE_ADDR']) . '\')');
}

//Enfin, on calcule le nombre total d'entrées puis on l'affiche !
$retour = mysqli_query($connexion, 'SELECT COUNT(*) AS nb_connectes
FROM connectes');
$donnees = mysqli_fetch_assoc($retour);

$visiteurs_connectes = $donnees['nb_connectes'];
// Affichage
echo 'Visiteurs connectés : <strong>' . $donnees['nb_connectes'] .
'</strong><br/>';
```

&gt;

## Affichage du record de connectés

Ce script va directement à la suite du premier vu qu'on aura besoin de la variable \$visiteurs\_connectes.

Ce script est très simple, on doit stocker deux nombres :

- le record ;
- la date à laquelle il a été établi.

Inutile de faire une table rien que pour ça, on va donc utiliser...

**Secret (cliquez pour afficher)**

... les fichiers !



Comment allons-nous faire ?

On va organiser les données du fichier en une ligne, contenant le record suivi de la date.

Le fichier se situe dans le répertoire de nos statistiques : stats/records.txt.

On l'initialise en mettant "0 0/0/0" dedans, puis on met son CHMOD à 777.

Pour séparer les données de vos chaînes de caractères, il existe la fonction `explode()` qui va vous permettre de couper en plusieurs morceaux le contenu de votre fichier.

### Correction ?

#### Code : PHP

```
<?php

$f_records = fopen('records.txt', 'r+'); //On ouvre le fichier
$dernierRecord = fgets($f_records); //On prend sa première ligne
$dernierRecord = explode(' ', $dernierRecord); //Je vous avais dit
de regarder la fonction explode !
//Elle va permettre de séparer notre fichier en 2 parties :
//Le record (0) dans $dernierRecord[0] et la date (0/0/0) dans
$dernierRecord[1]
echo 'Record du nombre de connectés : <strong>'; //On le marque tout
de suite
//Ici on va avoir besoin de la variable $visiteurs_connectes de
l'exercice précédent
if ($visiteurs_connectes > $dernierRecord[0]) //Si le nombre de
connecté est plus important que le record actuel
{
    rewind($f_records); //On "rebobine " le fichier
    $ligne = $visiteurs_connectes . ' ' . date('d/m/Y');
    fwrite($f_records, $ligne); //On écrit la ligne sous la forme
fixée au départ
    echo $visiteurs_connectes . '</strong> établi le <strong>' .
date('d/m/Y');
} else { //sinon, on affiche le record du fichier.
    echo $dernierRecord[0] . '</strong> établi le <strong>' .
$dernierRecord[1];
}

// On ferme la balise puis le fichier
echo '</strong><br>';
fclose($f_records);
?>
```



## Affichage de la liste des connectés

Script archi-simple, on va juste afficher :

- l'IP du visiteur ;
- la page qu'il visite.

*Euh... Hum... Correction* 🤪

Code : PHP

```
<?php
echo '<strong>Liste des connectés : </strong>(' .
$visiteurs_connectes . ')<br/>';
$retour = mysqli_query($connexion, 'SELECT ip, page FROM
connectes');
while ($donnees = mysqli_fetch_assoc($retour))
{
    echo long2ip($donnees['ip']) . ' : ' . $donnees['page'] . '<br />';
}
?>
```

Pas trop dur ? 😊



Eh bah dis donc ! Ça c'est du beau boulot !

Eh oui, vous avez vu ce qu'on est capable de faire ? Ces parties 1 et 2 n'étaient pas si faciles que ça finalement. 🤪

Allez, contemplez ce magnifique résultat (le mien est un peu plus sophistiqué, mais bon, c'est du pareil au même 😊) :



## Statistiques

Nombre de pages totales visitées : **22629**

Nombre de pages visitées aujourd'hui : **475**

Record du nombre de visites par jour : **740** établi le **2010-01-01**

Moyenne du nombre de pages visitées : **408** visiteurs par jour

Visiteurs actuellement connectés : **3**

Record du nombre de connectés : **4** établi le **01/01/2010**

**Liste des connectés : (3)**

██████████ : Index

██████████ : Index

██████████ : Administration

La partie suivante est encore plus compliquée à réaliser (quoique moins longue). Mais elle est super utile et quand vous l'aurez

finie, je peux vous garantir que vous serez fier de vous !

Prenez un peu de repos et... À vos claviers !!!

### TP n°3 : mais d'où viennent mes visiteurs ?

Allez, ne me mentez pas, je suis sûr que vous vous êtes déjà posé cette question ! Si si ! 😊

Sérieusement, ça ne vous arrive jamais de vous poser la question « Mais au fait, comment je fais pour avoir autant de visiteurs avec mon site tout pourri ? »

Eh bien dans cette partie nous allons pouvoir y répondre ! Et ceci grâce à un merveilleux instrument : la variable Superglobale `$_SERVER['HTTP_REFERER']` .



Si vous ne vous rappelez pas ce que veut dire "Variable Superglobale", je vous conseille d'aller voir par là : [Les variables Superglobales](#).

D'ailleurs je vais citer l'auteur brillantissime de ce chapitre :

**Citation : M@teo21**

`$_SERVER['HTTP_REFERER']` : C'est l'URL de la page qui a amené le visiteur sur la page courante. Cela peut être utile notamment pour faire des statistiques : vous saurez par exemple que le site "supersite.com" a fait un lien vers votre site et vous amène des visiteurs. 😊

Si vous allez voir ce chapitre, vous êtes clairement averti par M@teo :



La documentation de PHP nous avertit clairement que vous ne pouvez pas vous fier à 100% à `$_SERVER['HTTP_REFERER']` car le client peut très facilement refuser d'envoyer cette information ou même la modifier. Bref, n'ayez pas trop confiance en elle.

Bon, alors, passons aux choses sérieuses. 😊

Nous ne nous occupons QUE de l'enregistrement dans ce tutoriel. À vous de gérer l'affichage au mieux pour vos statistiques.

Voici ce que votre script devra faire :

- vérifier que le HTTP REFERER ne vient pas d'une de vos pages pour éviter que quelqu'un arrive sur votre index alors qu'il était déjà sur le site ;
- vérifier que l'utilisateur n'a pas ouvert deux fois un lien par inadvertance en mettant un timeout de 5 minutes ;
- sinon, enregistrer le HTTP REFERER et l'IP du visiteur dans une base de données ;
- ce n'est pas obligatoire, mais un champ ID ne serait pas de trop pour gérer vos stats ensuite ;
- vous aurez besoin d'un autre champ, lisez bien l'énoncé. 😊

...

Eh oui c'est dur hein !

Tu-tut ! on ne regarde pas la correction avant d'avoir fini !

C'est bon ?

**3, 2, 1... Correction !**

Structure de ma table "provenance" :

**Code : SQL**

```
CREATE TABLE IF NOT EXISTS `provenance` (  
  `id` int(11) NOT NULL auto increment,
```

```

`ip` int(12) NOT NULL,
`http_referer` text NOT NULL,
`heure` timestamp NOT NULL default CURRENT_TIMESTAMP,
PRIMARY KEY (`id`)
);

```

Et oui, comment auriez-vous fait si vous n'aviez pas enregistré le timestamp dans un champ "heure" ? Impossible de faire le timeout !

On passe au PHP ?

**Code : PHP**

```

<?php
if (strpos(strtolower($_SERVER['HTTP_REFERER']), 'monsite.com') ===
0) // Si le visiteur provient d'un autre site.
{
    $heureAffichage = time()-60*5; // Le temps qu'il était il y a 5
    minutes
    echo $heureAffichage;

    // On sélectionne toutes les entrées ayant l'IP du visiteur pour
    lesqueltes l'heure enregistrée est plus grande que l'heure qu'il
    était il y a 5 minutes.
    $retour = mysqli_query($connexion, 'SELECT COUNT(*) AS nbre_entrees
FROM provenance WHERE ip=\'' . ip2long($_SERVER['REMOTE_ADDR']) .
'\'' AND UNIX_TIMESTAMP(heure) > \'' . $heureAffichage);
    $donnees = mysqli_fetch_assoc($retour); // Fetch-array
    if ($donnees['nbre_entrees'] == 0) // S'il n'y a aucune entrée qui
    a notre IP et qui a été enregistrée il y a 5 minutes
    {
        mysqli_query($connexion, "INSERT INTO provenance(id, ip,
http_referer, heure) VALUES ('', '' .
ip2long($_SERVER['REMOTE_ADDR']) . ', '' .
mysqli_real_escape_string($_SERVER['HTTP_REFERER']) . ',
CURRENT_TIMESTAMP());"); // Insérer une nouvelle entrée
    }
}
?>

```

Alors, heureux ? 🤖 C'est dingue ce qu'on sait faire !

### Affichage :



Si ça vous intéresse, voilà un bout de code que j'ai écrit. Il va vous permettre d'afficher tout d'abord les sites d'où proviennent vos visiteurs, et ces sites seront sous forme de liens qui montreront les pages des sites.

Exemple :

**Citation**

```

www.siteduzero.com
www.machinchose.truc

```

Et quand on clique sur "www.siteduzero.com" :

**Citation**

```

http://www.siteduzero.com/tutoriel-10-202092.html
http://www.siteduzero.com/news.html

```

## Code : PHP

```

<?
if (!empty($_GET['url']))
{
    $url = mysqli_real_escape_string($_GET['url']);
    echo '<a href="admin.php">Retour à la liste complète</a><br />';
    $retour = mysqli_query($connexion, 'SELECT http_referer,
COUNT(http_referer) AS nb_vues_tot FROM provenance WHERE
http_referer LIKE \'http_7/\' . $url . \'%\'' GROUP BY http_referer');
    while ($donnees = mysqli_fetch_assoc($retour))
        echo $donnees['nb_vues_tot'] . ' <a href="' .
    $donnees['http_referer'] . '">' . $donnees['http_referer'] .
    '</a><br/>';
    } else {
        $retour = mysqli_query($connexion, 'SELECT http_referer,
COUNT(http_referer) AS nb_vues_tot FROM provenance GROUP BY
http_referer');
        $url = array();
        while ($donnees = mysqli_fetch_assoc($retour))
        {
            if ($donnees['http_referer'] == '')
            {
                $donnees['http_referer'] = 'URL directe';
                echo $donnees['http_referer'] . $donnees['nb_vues_tot'] .
            '<br/>';
            } else {
                $referer = explode('/', $donnees['http_referer']);
                if (array_key_exists($referer[2], $url))
                    $url[$referer[2]] = $url[$referer[2]] + $donnees['nb_vues_tot'];
                else
                    $url[$referer[2]] = $donnees['nb_vues_tot'];
            }
        }

        foreach($url as $key => $value)
            echo '<a href="admin.php?url=' . $key . '">' . $key . '</a>' .
    $value . '<br/>';
    }
?>

```

PS : ne me remerciez pas. 😊

## TP n°4 : affluences et heures de pointe

Voici le but de ce TP : visualiser l'affluence de votre site en fonction des heures de visite des internautes.  
Pour chaque heure, on verra donc le nombre de visites total que cette heure a recensé depuis que ce module est en place.



Le script sera facilement adaptable à la fin pour visualiser l'affluence de votre site à un jour précis.



T'es sûr qu'on va y arriver ?

Mais oui. 😊 Par contre ça ne va pas être de tout repos.

Là je sens que vous allez avoir besoin d'infos parce que moi-même j'ai mis du temps à élaborer la technique. 😊

Nous allons avoir une table, nommons-la "affluence".  
Elle comportera trois champs : la date, l'heure et le nombre de visites.



Ne faites pas un champ DATETIME pour les deux premiers éléments que j'ai cités. Ce serait possible de le faire comme ça, mais extrêmement plus complexe.

Notre champ date sera simplement de type DATE, notre champ heure sera TINYINT(2). En effet, à moins que vous ayez besoin de visualiser le nombre de visites chaque minute, on va se contenter de voir le nombre de visites pour chaque heure de la journée. 😊

Voilà la technique que nous allons utiliser.

#### Enregistrement :

- un visiteur va sur une page ;
- si l'heure (ex : 12) et la date (ex : 2010-01-04) de l'instant où il va sur la page ne sont pas dans une entrée présente dans la table, on la crée et on initialise son "Nombre de visites" à 1 ;
- si au contraire l'entrée (date+heure du moment où il se connecte) est présente, on incrémente le nombre de visites.

#### Affichage :



On ne doit pas afficher les données d'aujourd'hui, car si un visiteur se connecte à midi, il verra une affluence plus élevée pour le matin que pour l'après-midi. On n'affichera donc que les données de la veille et antérieures.

On va donc, pour chaque heure, en utilisant les boucles (ah, un indice 🤖 \*note note note\*), compter le nombre de visites totales de tous les jours précédents à l'heure choisie et l'afficher sous forme de tableau.

#### Correction : Enregistrement

C'est pas ça le plus compliqué, si vous suivez mes indications vous tombez dessus sans problème. 😊

##### Code : PHP

```
<?php
$heure = date ('H');
$retour_afflu = mysqli_query($connexion, 'SELECT COUNT(*) AS
maintenant FROM affluence WHERE date = CURRENT_DATE() AND heure = '
. $heure);
$donnees_afflu = mysqli_fetch_assoc($retour_afflu);
if ($donnees_afflu['maintenant'] == 0) //Si l'heure n'a pas encore
été enregistrée
{
    mysqli_query($connexion, 'INSERT INTO affluence(heure, date,
visites) VALUES (' . $heure . ', CURRENT_DATE(), 1);'); //On rentre
la date et l'heure et on marque 1 comme nombre de visites.
} else {
    mysqli_query($connexion, 'UPDATE affluence SET visites = visites +
1 WHERE date = CURRENT_DATE() AND heure = ' . $heure); //On rentre
l'heure et la date d'aujourd'hui et on marque 1 comme nombre de
visites.
}
?>
```

#### Correction : Affichage

##### Code : PHP

```
<?php

echo '<table>';

echo '<tr>';
for ($i = 0; $i < 23; $i++)
{
```

```
echo '<td><strong>' . $i . 'h</strong></td>';
}
echo '</tr>';

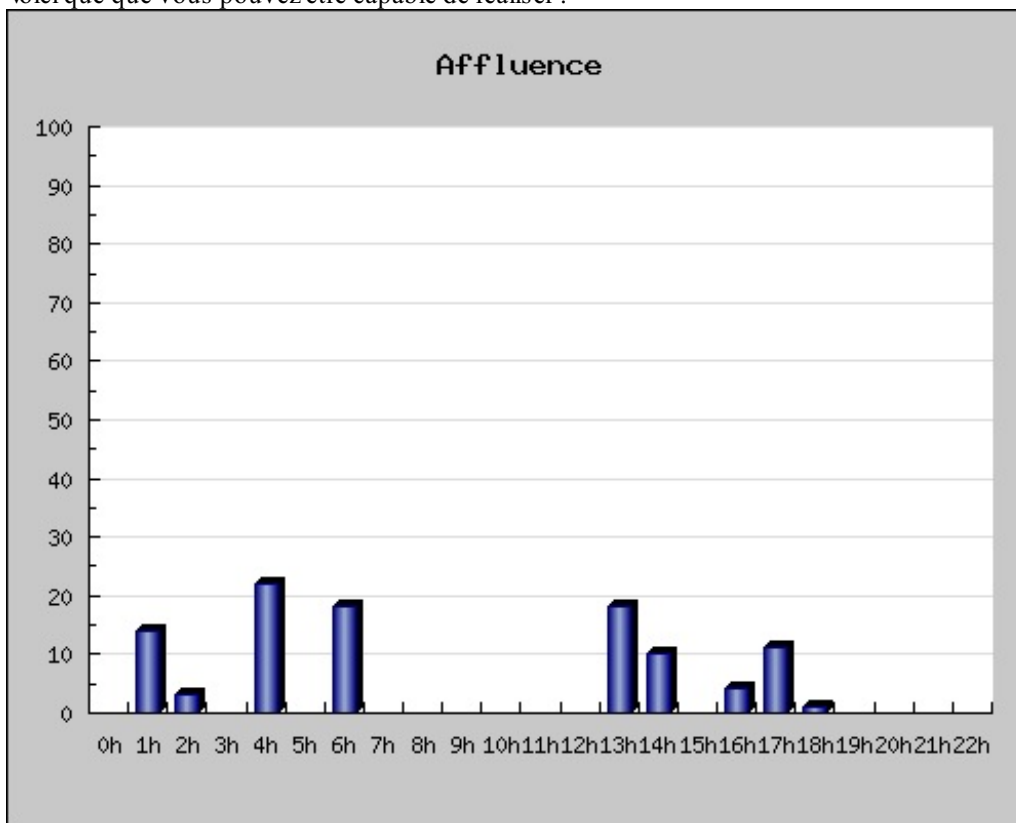
echo '<tr>';
$req = mysqli_query($connexion, "SELECT SUM(visites) AS nb_visites
FROM affluence WHERE date != CURRENT_DATE() GROUP BY heure ORDER BY
heure ASC");
while($dat = mysqli_fetch_assoc($req))
    echo '<td>' . $dat['nb_visites'] . '</td>';

echo '</tr>';
echo '</table>';
?>
```

### Affichage graphique

Si vous voulez afficher un graphique, vous pouvez apprendre à vous servir d'une librairie graphique comme JpGraph, Artichow... Vous pouvez également vous servir d'un site sympa qui vous permettra d'afficher un graphique sans installation : [Jphistopct](#).

Le détail du fonctionnement est expliqué sur le site. Vous devez juste envoyer un lien contenant les valeurs que vous obtenez. Voici que que vous pouvez être capable de réaliser :



Bonne chance ! 😊

Ce tutoriel s'arrête là.

J'espère qu'il vous aura aidé mais surtout fait découvrir de nouvelles choses. 😊

Et n'oubliez jamais, c'est en phpant que l'on devient phperon. 🤖

Bonne chance pour la suite !

## Partager



Ce tutoriel a été corrigé par les [zCorrecteurs](#).