

# Les Médecins Maîtres-Toile

[Accueil](#) ▶ [Boite à Outils](#) ▶ Réaliser un formulaire de contact (5/4)

**Publié le : 27 avril 2004**

 [Imprimer cet article](#)

**Auteur :**  
**Philippe Auriol**



## Réaliser un formulaire de contact (5/4)

**html, css, js, php et Mysql**

Jusque là nous nous sommes contentés de travailler avec un seul ordinateur : le serveur web. Pourtant, Internet ne serait pas devenu aussi incontournable s'il n'y avait pas eu derrière l'affichage que fait le web des serveurs de données : ordinateurs dédiés au stockage et au traitement des données informatiques. Le standard actuel étant mysql, c'est lui que je vous propose d'utiliser dans cet article après avoir sommairement parlé d'un stockage en texte.

### Sommaire

- [Stockage dans un fichier](#)
- [Stockage dans une base de données](#)

Avoir des données, c'est bien. Les recevoir par mail dès qu'elle sont entrées, c'est mieux. Mais la panacée c'est tout de même la persistance des données.

Il y a deux manières courantes de stocker les données sur Internet : le fichier et la base de donnée.

Nous allons d'abord compléter notre formulaire php pour permettre un stockage dans un fichier texte, puis nous aborderons plus en détail le stockage dans une base de donnée.

### Stockage dans un fichier

Nous avons vu arriver les données dans des variables post dans l' [article précédent](#).

Ajoutons simplement en bas du fichier précédent ces quelques lignes :

```
/* enregistrer les coordonnées de l'envoi dans un fichier texte (chmod 755)
```

```

dans un répertoire admin que l'on protégera avec un fichier .htaccess */

/* on définit la date et l'heure de l'entree */
$datedujour = date('d/m/Y/H:i:s');
/* on définit le nom et le lieu du fichier qui sera créé pour stocker les
infos */
$filename="./admin/monfichier.txt";
/*on écrit dans le fichier les infos, séparées par une pipe (|) et on
termine la ligne par un retour à la ligne */
$f1=fopen($filename,"a+");
fwrite($f1,"'$datedujour'|'$nom'|'$prenom'|'$email'|'$sujet'|'$texte'|\n\r
");
/* on ferme le fichier */
fclose($f1);

```

Simple non ? il vous suffira de lire le fichier dans votre navigateur, c'est un fichier texte. Vous ne savez pas créer un fichier htaccess ? [C'est là](#). ou [encore mieux : là](#).

Le stockage en fichier texte est généralement suffisant pour des données de faible taille (quelques méga octets), au delà il faut un système de gestion de base de données.

## Stockage dans une base de données

Habituellement, votre fournisseur d'accès Internet vous a donné les coordonnées de votre serveur mysql.

Bien souvent, le serveur mysql est hébergé sur le même ordinateur que le serveur web et son nom est alors "localhost". C'est par exemple le cas si vous utilisez [easyphp](#) sur votre windowsette ou si vous avez installé [mysql sur votre mac](#) ou votre station linux préférée.

Votre hébergeur vous aura donné également un nom d'utilisateur et un mot de passe pour accéder à cette base.

Avec easyphp c'est "" et "" : soit rien 😊 ce qui n'est pas forcément la meilleure sécurité sur une machine windows déjà bien défaillante de ce point de vue, si vous avez installé vous même votre serveur c'est vous qui avez donné les noms des bases, d'utilisateur et le mot de passe.

### PhpMyadmin, mon ami

Un serveur mysql, c'est un petit programme qui fonctionne en tâche de fond et que l'on interroge avec des requêtes cabalistiques en ligne de commande. Comme beaucoup d'entre nous ne sont pas doués pour les langues et que nous adorons cliquer partout, mère nature (en l'occurrence rebaptisée Olivier Mueller pour l'occasion) a mis au point le programme [phpmyadmin](#) qui répond justement à ce besoin. La quasi totalité des hébergeurs qui proposent un serveur mysql proposent également d'y accéder via ce programme et c'est tout de suite plus sympathique !

Les utilisateurs mac lui préféreront peut être une application autonome comme [CocoaMysql](#)

Connectez vous donc au serveur en question par le web à l'adresse Internet que vous a donné votre hébergeur.

### Créons

Arrivés à notre base de données si vide nous ne pouvons que nous réjouir de tout ce que nous allons pouvoir lui faire faire.

---

Il existe un champ marqué "créer une nouvelle table sur la base votrenomdebase", ça tombe bien : il nous le faut.

**Mettez :** formulaire\_contact, le nombre de champs dont vous aurez besoin (pour notre exemple : 5) et validez.

Une nouvelle page s'offre à vous avec 5 champs (le nombre dépend de ce que vous avez spécifié) non encore paramétrés.

### Faisons cela tout de suite :

La première ce sera nom, VARCHAR, 255, not null.

La suivante prenom, VARCHAR, 255, not null

Puis email, VARCHAR, 255, not null

Puis sujet, VARCHAR, 100, not null

Puis texte, TEXT, not null

Et validez ! La table est créée dans votre base de données. Toutes les cases sont donc faites et vous leur avez donné des caractéristiques : VARCHAR c'est une variable de caractère et 255 c'est sa taille maximum, TEXT, c'est un texte... le *not null* signifie juste que ces champs DOIVENT être remplis : ça tombe bien **notre script vérifiait justement que tout était rempli.**

### Connectons nous à la base ainsi créée

```
<?php
$dbhost = "localhost";
$dbuser = "votrenomutilisateur";
$dbpass="votremotdepasse";
/* Le '@' indique a l'interpreteur Php de ne pas afficher d'erreur s'il y
a lieu. */ @mysql_connect($dbhost,$dbuser,$dbpass) or die("La Connexion a
échouée"); echo "la connexion a réussie";
mysql_close();
?>
```

Enregistrons tout cela dans un fichier php comme : test.php.

Nous avons indiqué les paramètres de connection à la place de ceux indiqués et en lançant le fichier dans notre navigateur, nous savons si la connection a réussi : si le message de réussite s'affiche c'est ok, on passe à la suite sinon...vérifier les données : elles ne sont pas bonnes.

### Allez hop : on rentre des données dans la base !

Jusque là, nous nous connectons au serveur..mais pas à la base !

Modifions un peu le script précédent :

```
<?php
$dbhost = "localhost";
$dbuser = "votrenomutilisateur";
$dbpass="votremotdepasse";
$dbbase="votrenomdebase";
@mysql_connect($dbhost,$dbuser,$dbpass) or die("La Connexion a échouée");
/* On se connecte à la base ou on envoie un message d'erreur */
@mysql_select_db($dbbase) or die("La Base $dbbase n'existe pas");
/* On sélectionne tout les champs de la table formulaire_contact créée
plus haut */
$requete = mysql_query("SELECT * FROM formulaire_contact");
```

```

/* on récupère TOUT (*) dans la table formulaire_contact et on le met dans
une variable */
$resultat = mysql_numrows($requete);
/* on affiche le résultat dans la page web */
echo "Il y a $resultat enregistrements dans votre table";
/* on referme notre connection mysql */
mysql_close();
?>

```

Facile hein ?

Comment ça ça ne vous suffit pas ? Alors placez ces lignes avant le mysql\_close() ;

```

/* Tant qu'il y a un enregistrement on l'affiche. */
while($resultat != $i) {
/* On récupère la valeur des champs pour la $i ème ligne*/
$nom = mysql_result($requete,$i,"nom");
$prenom = mysql_result($requete,$i,"prenom");
$email = mysql_result($requete,$i,"email");
/* On affiche à l'écran les valeurs */
echo "Nom: $nom ; Prenom: $prenom; e-mail : $email <br/>";
/* On saute une ligne */
$i++;
}

```

Vous voilà avec l'affichage de ce qu'il y a dans la base 😊 mais euh..comment ça il n'y a rien dedans ? Evidemment, vous avez mis la charrue avant les boeufs : vous visitez votre base mais...vous ne l'avez pas remplis ! Comment ça c'est ma faute ? Bon ok...allons y.

### Entrer des données à partir du formulaire !

Bon, alors c'est toujours la même chose : on se connecte, on envoie les données dans la base et on ferme la base.

```

<?php
$dbhost = "localhost";
$dbuser = "votrenomutilisateur";
$dbpass="votremotdepasse";
$dbbase="votrenomdebase";
@mysql_connect($dbhost,$dbuser,$dbpass) or die("La Connexion a
échouée"); @mysql_select_db($dbbase) or die("La Base $dbbase n'existe
pas");
/* On écrit dans la base les variables récupérées du formulaire*/
$enregistrement = "INSERT INTO formulaire_contact
(nom,prenom,email,sujet,texte) VALUES
('$nom','$prenom','$email','$sujet','$texte)";
mysql_query($enregistrement);
mysql_close();
?>

```

Et voilà, tout est inscrit, allez : maintenant refaites votre requête vue au dessus car cette fois la base n'est plus vide 😊

### On peut faire mieux ?

---

Oui, sans problème. D'abord sortons les données de connection des scripts comme ça on va pouvoir les utiliser sans les réécrire à chaque fois.

Remplacez :

```
$dbhost = "localhost";  
$dbuser = "votrenomutilisateur";  
$dbpass="votremotdepasse";  
$dbbase="votrenomdebase";
```

par :

```
include ("maconfig_sql.php");
```

et évidemment vous créez un fichier maconfig\_sql.php où vous mettez \$dbhost etc.

### Quoi de mieux ?

Vous voulez modifier des infos dans votre base peut-être ? Alors ça s'écrit comme ça :

```
mysql_query("UPDATE formulaire_contact SET prenom='John' WHERE  
email='john@lennon.com'");  
print "La Base de données à été mise à jour.";
```

Voilà, vous avez dit à votre serveur mysql de chercher dans la table "formulaire\_contact" l'email john@lennon.com et dans cette ligne de remplacer le prénom par John.

Tout récupérer ? Vous êtes durs avec moi...

```
$requete = mysql_query("SELECT * FROM formulaire_contact");  
/* On entre tout les résultats dans un tableau.*/  
while ($tabl_resultat = mysql_fetch_array($requete))  
{  
/* On affiche à l'écran les valeurs */  
echo "Nom : $tabl_resultat[nom] ; Prenom : $tabl_resultat[prenom] ; e-mail  
: $tabl_resultat[email] ; Sujet : $tabl_resultat[sujet1] ; Texte :  
$tabl_resultat[texte] ; <br/>";  
}
```

Et voilà, c'est fini pour ce petit apprentissage. Pour aller plus loin, il va vous falloir faire briller vos méninges avec la lecture des liens suivants :

[Mysql avec Nexen](#)

[Php avec Nexen](#)

Bonne lecture !



 [Imprimer cet article](#)

---

Copyright Médecins Maîtres-Toile francophones  
Espace membres - Administration - Crédits

---