



## Bootstrap my\_meetic

Mon premier système de connexion inscription

## Contents

Ma base de données .....	3
Mon model User.....	3
Inscription.....	4
Connexion.....	4
Le hash.....	4
Protéger votre site.....	5

## Ma base de données

Créez une base de données my\_meetic.

Cette base de données doit contenir une table users.

Cette table doit contenir les champs suivants :

- id
- nom
- prenom
- date\_de\_naissance
- sexe
- email
- mot\_de\_passe



**Checkpoint : Attention, il est inutile de poursuivre si vous n'êtes pas arrivé correctement à ce stade. En cas de doute, vérifiez à nouveau votre travail.**

## Mon model User

Créez un fichier **User.php**.

Ce fichier doit contenir une classe User avec un attribut privé bdd qui contiendra la connexion PDO à votre base de données, et les méthodes suivantes :

- object getUsers()  
Renvoie tous les utilisateurs de la base de données  
**A compléter par vous-même à la suite du bootstrap**
- object getUsersByFilter(array \$filters)  
Renvoie tous les utilisateurs correspondant aux filtres passés en paramètre  
**A compléter par vous-même à la suite du bootstrap**
- object userExist(string \$email, string \$password)  
Renvoie true si l'utilisateur correspondants aux paramètres existe en base de données, sinon renvoie false
- boolean addUser(string \$nom, string \$prenom, string \$date\_de\_naissance, string \$sexe, string \$email, string \$password)  
Ajouter un enregistrement à la table users et renvoyer true en cas de succès ou false en cas d'échec



**Checkpoint : Attention, il est inutile de poursuivre si vous n'êtes pas arrivé correctement à ce stade. En cas de doute, vérifiez à nouveau votre travail.**

## Inscription

Créez un fichier **subscribe.html** contenant un formulaire de connexion dont la cible est un fichier **controller\_subscribe.php**. Le fichier **controller\_subscribe.php** doit faire appel à la méthode `addUser` et rediriger vers :

- **controller\_subscribe.html** en cas de succès
- **index.html** en cas d'échec

## Connexion

Créez un fichier **connect.html** contenant un formulaire de connexion dont la cible est un fichier **process\_connect.php**. Le fichier **process\_connect.php** doit faire appel à la méthode `userExist` et rediriger vers :

- **connect.html** en cas de succès
- **index.html** en cas d'échec

En cas de succès, votre fichier devra créer un jeton de connexion dans la session.



**Pour information, une implémentation simple d'un jeton de connexion pourrait être de mettre dans la session un hash md5 du nom d'utilisateur concaténé à son mot de passe.**

## Le hash

Assurez-vous de hasher le mot de passe lors son enregistrement dans la base de données



**Attention !  
Une nouvelle loi européenne sur la protection des données oblige à présent les développeurs à hasher tous leurs mots de passe.**



Pour information, les méthodes md5 et sha1 ne sont pas sécurisées, il faut utiliser la fonction `password_hash` en conjonction avec la fonction `password_verify` pour protéger efficacement vos mots de passe.

## Protéger votre site

Afin que vous n'ayez pas fait tout ce travail inutilement, empêcher les utilisateurs non connectés d'accéder à votre `index.php`.



Pour information, vous devez implémenter un système de session avec un identifiant unique (genre l'email et/ou l'id) (`session_start` et `$_SESSION`).