PHP

CM2-2 : PHP et MySQL

Mickaël Martin Nevot

V4.4.0

Cette œuvre de Mickaël Martin Nevot est mise à disposition selon les termes de la licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Partage à l'Identique 3.0 non transposé.

BY-NC-SA

PHP

BY-NC-SA

PHP et MySQL

BD : base de données

- **Connexion** à un serveur MySQL :

```
mysqli_connect($host = …, $username = …, $passwd = …, …)
```

- Sélectionne une **BD** :

```
mysqli_select_db($link, $dbname)
```

- Ferme la connexion MySQL :

```
mysqli_close($link)
```

Une connexion est fermée par défaut à la fin du script l'ayant ouverte

- Envoie un message en cas d'erreur : `die($str)`

```
$link = mysqli_connect($host, $username, $passwd) or die('Erreur : ' . $host);
```

Important : l'activation de l'extension MySQLi est obligatoire (par défaut à partir de PHP 5.3.0) !

BY-NC-SA

PHP et MySQL

- Envoie une **requête** au serveur :
  `mysqli_query($link, $query)`
- Retourne un **enregistrement** :
  - Tableau indexé :
    `mysqli_fetch_row($result)`
  - Tableau associatif :
    `mysqli_fetch_assoc($result)`
- Nombre d'enregistrements d'un résultat :
  `mysqli_num_rows($result)`
- Libère la mémoire du résultat :
  `mysqli_free_result($result)`

BY-NC-SA

Exemple PHP et MySQL

```
$link = mysqli_connect('localhost', 'mysql_username', 'mysql_passwd')
    or die('Pb de connexion au serveur: ' . mysqli_connect_error());
mysqli_select_db($link, 'my_dbname') or die ('Pb de sélection BD : ' . mysqli_error($link));

$query = 'SELECT name AS username FROM table WHERE id = 1';
$result = mysqli_query($link, $query);
if (!$result)
{
    echo 'Impossible d\'exécuter la requête ', $query, ' : ', mysqli_error($link);
}
else
{
    if (mysqli_num_rows($result) != 0)
    {
        while ($row = mysqli_fetch_assoc($result))
        {
            echo $row['username'];
        }
    }
}
```

BY-NC-SA

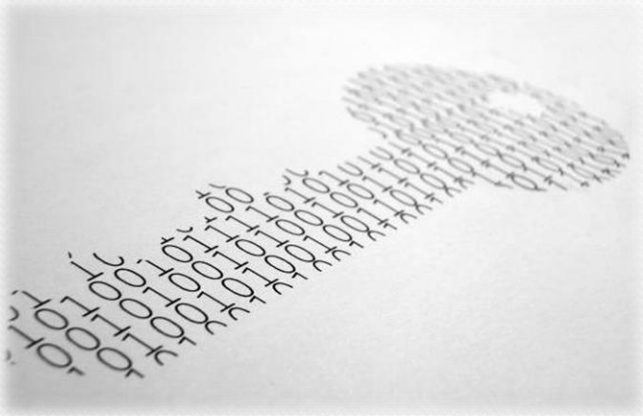
Utilité des pages sécurisées

- Restreindre une partie d'un site Web
- **Protéger des données**
- Utilisations courantes :
  - Partie d'un site accessible qu'à certains utilisateurs
  - Partie d'un site qui comporte des **données personnelles** (compte bancaire, adresse postale, etc.)
  - Compte ***e-mail***
  - Compte pour un site *d'**e-commerce***

BY-NC-SA

Moyens de sécurisation

- Niveau **réseau** : (par exemple HTTPS)
  - On utilise SSL (*secure socket layer*) pour chiffrer le canal de communication :
    - Utilise un certificat pour l'identification
    - Certificat hébergé chez un tiers
- Niveau **applicatif** :
  - Authentification
  - Pages sécurisées



BY-NC-SA

Authentification

- Table **utilisateur** dans la base de données :
  - Un identifiant (*e-mail*, pseudonyme, etc.)
  - Un mot de passe encodé
- Lors de l'**authentification** :
  - On vérifie le couple identifiant / mot de passe encodé avec les informations en base de données
  - On met les informations dans la variable `$_SESSION`
- **Vérification de l'authentification à chaque page** :
  - L'utilisateur ne retape pas ses identifiants
  - Vérification faite grâce à la variable `$_SESSION`

BY-NC-SA

Exemple d'authentification

```php
// Page d'authentification.
if (filter_input(INPUT_POST, 'login') && filter_input(INPUT_POST, 'pwd'))
{
    // Connexion au serveur de la base de données, à la base de données et
    // récupération dans la table utilisateur du login dans la variable $login
    // et du mot de passe dans la variable $pwd.
    // …
    if ($login == $_POST['login'] && password_verify($_POST['pwd'], $pwd))
    {
        session_start();
        $_SESSION['suid'] = session_id();
        header('location: page.php');
    }
    else
    {
        header('location: index.html');
    }
}
```

BY-NC-SA

Exemple d'authentification

```php
// page.php.
session_start();
if (isset($_SESSION['suid']))
{
    // L'authentification est validée.
}
```

BY-NC-SA

Post-Redirect-Get (PRG)

- Page Web du formulaire :

```php
<?php
require 'path_to_file/validate.php';
validate();
?>
<html>…<form action="validate.php" method="post">…</form>…</html>
```

- Traitement :

```php
<?php
function validate(): void {
    // Aucun affichage dans cette fonction.
    if ($_POST) {
        // Valider les données d'entrée.
        if (/* Données d'entrée ok */) {
            // Traitement (requêtes à la base de données, données en session, etc.).
            // Redirection.
            header('Location: ' . $_SERVER['REQUEST_URI'], true, 303);
            exit();
        }
    }
}
```

BY-NC-SA

Index

- Structures de données, physiquement et logiquement indépendantes des données stockées dans la base
- Permet un accès direct (**rapide**) aux enregistrements
- Permet l'**optimisation de requêtes**
- Peut être composite (multi-champs)
- Bonne utilisation :
  - Trouver le meilleur compromis entre :
    - Efficacité des requêtes
    - Coût d'exécution des mises à jour
    - Espace de stockage nécessaire

BY-NC-SA

Cas d'utilisation d'index

- Champs utilisés dans des **conditions de sélections** simples (c'est-à-dire sans `!=`, `IS NULL`, `NOT IN`, `LIKE`, `||`, une fonction de calcul horizontal ou vertical)
- Champs utilisés pour des **jointures** (**clefs étrangères**, etc.)
- **Index composites** plutôt que plusieurs index simples
- Tous les champs ayant une **forte cardinalité**
- Pas pour un champ **volatile**
  (avec une fréquence de mise à jour des données élevée)
- Taille des données indexées importante :
  privilégier les champs de type entier
- Attention à l'ordre de spécification des champs indexés !

BY-NC-SA

Création d'index

- Types d'index :
  - `PRIMARY KEY` : **index à valeur unique non nulle**
  - `KEY` : synonyme de `INDEX`
  - `INDEX` (simple ou composite) :
    - `INDEX index (name, firstname)`
  - `UNIQUE` : **index à valeur unique**

Un seul par table

Pouvant être nulle

- Création/destruction :
  - `CREATE INDEX` :

```
mysql> CREATE INDEX index ON table (col1, col2);
```

  - `DROP INDEX` :

```
mysql> DROP INDEX index ON table;
mysql> ALTER TABLE table DROP INDEX index;
```

BY-NC-SA

Moteur/type de tables MySQL

MyISAM

Le plus utilisé

- Application :
  - Table en **lecture seule**, « log »
  - Recherche **plein texte**
- Avantages :
  - Moteur **rapide**
  - Gain de place sur disque
- Inconvénients :
  - Pas de gestion des contraintes de clefs étrangères
  - Pas de gestion de transactions (pas de `COMMIT/ROLLBACK` possibles)

InnoDB

Par défaut à partir de MySQL 5.5

- Application :
  - Gestion des **transactions**
  - Fiabilité de l'information
- Avantages :
  - Gestion des **clefs étrangères**
  - Gère les gros volumes de données
- Inconvénients :
  - Lenteur de certaines opérations telles que `SELECT COUNT(*) FROM myTable`
  - `TRUNCATE` est synonyme de `DELETE`

BY-NC-SA

Aller plus loin

- Pièces jointes dans un *e-mail*
- Flux d'entrée/sortie
- Utiliser la librairie GD (traitement des images)

BY-NC-SA

Liens

- Documents électroniques :
  - Manuels :
    - http://php.net/manual
    - http://fr.php.net/manual/fr/ref.mysql.php
  - *Framework* :
    - http://framework.zend.com
    - http://www.symfony-project.org
  - Génération de données :
    - http://www.generatedata.com/#generator
  - Fonctions images :
    - http://www.manuelphp.com/php/ref.image.php

Crédits

Auteur

Mickaël Martin Nevot
mmartin.nevot@gmail.com

Carte de visite électronique

Relecteur

- Christophe Delagarde (christophe.delagarde@univ-amu.fr)
- Pierre-Alexis de Solminihac (pa@solminihac.fr)

Cours en ligne sur : www.mickael-martin-nevot.com

CC BY NC SA