

# TP : PHP

Thibaud Bardin

# TP : PHP

- Bases
- Boucles
- Formulaires
- Structure du Projet
- Connexion BDD
- Recommandations
  
- Pour ceux qui ont de l'avance

# TP : PHP - Bases

- PHP est un langage « serveur »
  - Utilisation obligatoire d'un serveur (WAMP, MAMP, ...)
- On ouvre les balises php comme suit :
  - `<?php`
- Il n'est pas obligatoire de fermer la balise PHP

# TP : PHP - Bases

- Une variable PHP commence toujours par \$
  - Ex : \$maVariable
- En PHP, le signe
  - = sert à affecter une valeur
  - == sert à comparer les valeurs
  - === sert à comparer les valeurs et les type (integer, boolean, string, ...)

# TD : PHP - Boucles

- Boucle foreach :

// La boucle foreach s'effectue tant que le tableau \$array contient des valeurs

```
foreach($array as $key => $value) {
```

```
    // $key est la clé
```

```
    // $value est la valeur associée à la clé
```

```
}
```

# TD : PHP - Boucles

- Boucle while :

// La boucle while s'effectue tant que la condition entre parenthèses n'est pas atteinte

```
$i = 0;
```

```
while($i >= 5) {
```

```
    $i++; // On incrémente (+1) la valeur de $i
```

```
}
```

# TD : PHP - Formulaires

- Exemple formulaire HTML5

```
<form method=«post»>
```

```
    <label for=«champ1»>Champ 1 :</label>
```

```
    <input type=«text» name=«champ1»>
```

```
        <input type=«submit» name=«form_submit»  
value=«Envoyer !»>
```

```
</form>
```

# TD : PHP - Formulaires

- On traite l'envoi du formulaire ci-avant de la manière suivante :

```
If (isset($_POST['form_submit'])) {  
    // Ici, on traite les champs du formulaire  
    envoyés en POST  
    $champ1 = $_POST['champ1'];  
}
```



# TD : PHP - Structure du Projet

- admin/
- class/
- config/
  - config.php
- functions/
  - database.fn.php
- js/
  - main.js
- styles/
  - style.css
- templates/
  - \_footer.phtml / \_footer.html
  - \_header.phtml / \_header.html
  - \_navbar.phtml / \_navbar.html
- \_footer.php
- \_header.php
- index.php

# TD : PHP – Connexion BDD

- Rappel « CRUD »

- Create

- « INSERT INTO table (`id`, `champ1`, `champ2`) VALUES (null, 'contenu1', 'contenu2') »;

- Read

- « SELECT \* FROM table WHERE id = 1 »;

- Update

- « UPDATE table SET `champ1` = 'nouveauContenu1' »;

- Delete

- « DELETE FROM table WHERE id = 1 »;

# TD : PHP – Connexion BDD

RAPPEL : « mysqli » obligatoire

```
$link = mysqli_connect('host', 'username',  
'password', 'databaseName');
```

```
$sql = « MA REQUETE SQL »;
```

```
$result = mysqli_query($link, $sql);
```

# TD : PHP – Connexion BDD

RAPPEL : Si on veut afficher le résultat d'une requête de type SELECT, il faut faire un « FETCH »

```
$articles = mysqli_query($link, $sql);  
  
while($article = mysqli_fetch_assoc($articles))  
{  
    echo $article['id'];  
    echo $article['champ1'];  
}
```

# TD : PHP – Connexion BDD

## RAPPEL : PROTECTION SQL

Il faut obligatoirement « échapper » tout ce qui est envoyé par l'utilisateur pour éviter les attaques de type « Injection SQL » : [http://fr.wikipedia.org/wiki/Injection\\_SQL](http://fr.wikipedia.org/wiki/Injection_SQL)

```
$username = mysqli_real_escape_string($link,  
$_POST['username']);
```

```
$password = mysqli_real_escape_string($link,  
$_POST['password']);
```

# TD : PHP - Recommendations

- Version de PHP > 5.3.2
- IDE : JetBrains PhpStorm 6

# TD : PHP - Pour ceux qui ont de l'avance

- Découverte de la POO (Programmation Orientée Objet)
  - <http://www.siteduzero.com/informatique/tutoriels/programmez-en-orientee-objet-en-php>
- Découverte de l'extension PDO (Connexion BDD en objet)
  - <http://www.siteduzero.com/informatique/tutoriels/pdo-interface-d-acces-aux-bdd>