

PHP et son interface WEB

- ▶ Auteur → Bruno Diémert
- ▶ Scénariste → Lucile Clausel
- ▶ Pilotage → Mamagalubru
- ▶ Licence → Creative Commons



Plan du cours

- Les variables d'environnement
- Les formulaires
- Les contrôles des formulaires
- Conserver l'information sur le Web
- Les cookies
- Les variables globales
- Les variables de session



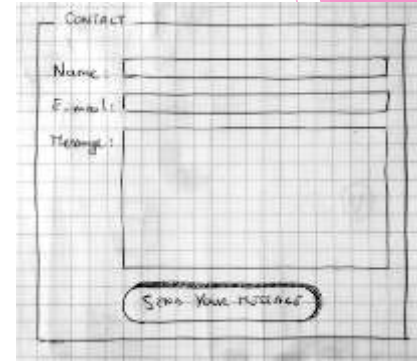
Les variables d'environnement



- Variables shell
 - Le serveur PHP propose des variables créées automatiquement
 - Ces variables sont accessibles à tout programme php
 - **PATH, REMOTE_ADDR, HTTP_USER_AGENT, ...**
- Variables globales
 - Le serveur PHP propose sous forme de tableau des variables créées automatiquement
 - Ces variables sont accessibles à tout programme php
 - **GLOBALS, _REQUEST, _GET, _POST, COOKIE, SERVER, ...**

Comprendre l'utilisateur via les formulaires

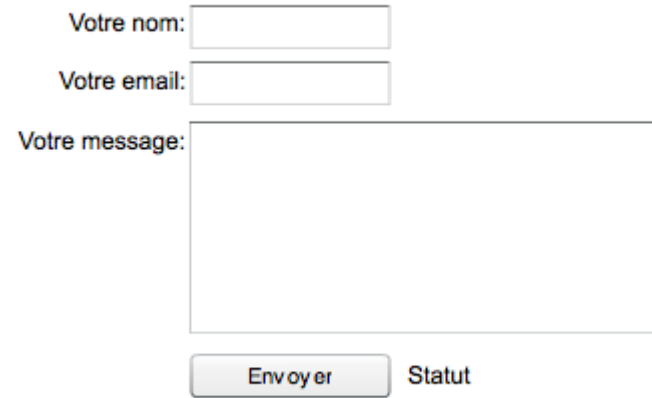
- Que sont les formulaires ?
- Créer des formulaires HTML
 - Balise BODY
 - Balise FORM
- Soumission de formulaires
 - L'attribut ACTION
 - L'attribut METHOD
 - Différence entre METHOD GET et METHOD POST
- Utiliser les contrôles (balises INPUT) de formulaires
 - L'attribut VALUE
 - L'attribut NAME



A hand-drawn sketch of a contact form on graph paper. The form is titled "CONTACT" at the top. It contains three input fields: "Name:", "Email:", and "Message:". Below these fields is a button labeled "Send Your Message".

Chaine de requête transmise

- Un formulaire
- Une balise <A> chargée
 - URL
 - un nom de programme
 - valeurs
- Un programme récupérant les valeurs
 - Soit par l'URL → Méthode GET
 - Soit par un message vers serveur → Méthode POST



Votre nom:

Votre email:

Votre message:

Statut

Utilisation des formulaires

- Afficher des informations HTML depuis une page PHP
 - `echo 'bonjour
';`
 - `print ("<p>".$_SERVER['HTTP_USER_AGENT'].</p>");`
- Lire les entrées des formulaires depuis une page PHP
 - Boîtes de saisie → `echo '<INPUT TYPE="text"...';`
 - Cases à cocher → `echo '<INPUT TYPE="checkbox"...';`
 - Liste de valeurs → `echo '<INPUT TYPE="select"...';`
 - Bouton submit → `echo '<INPUT TYPE="submit"...';`
 - Texte caché → `echo '<INPUT TYPE="hidden"...';`
 - ...

Cochez les aliments que vous aimez manger :

- ☒ Frites
- ☒ Steak haché
- ☐ Epinards
- ☐ Huitres

Définir un formulaire

- Principe
 - faire saisir des valeurs à l'utilisateur
 - transmettre ces valeurs à un programme php
- Exemple de formulaire

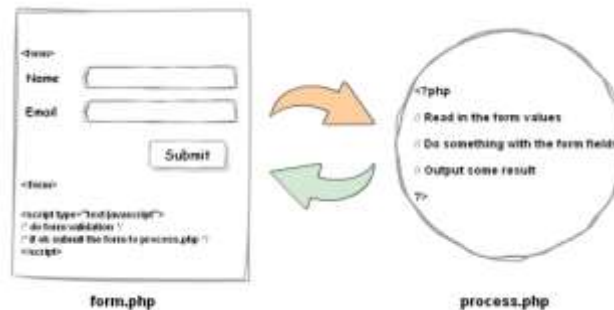
```
<FORM NAME="monFormulaire" ACTION="traiterFormulaire.php">
  <INPUT TYPE="text" NAME="entreprise">
  <SELECT NAME="listeDomaine" SIZE=3 METHOD=GET>
    <OPTION VALUE="info">Informatique</OPTION>
    <OPTION VALUE="electro">Electronique</OPTION>
    <OPTION VALUE="auto">Automatique</OPTION>
  </SELECT>
  <INPUT TYPE="checkbox" NAME="os" VALUE="LINUX">LINUX
  <INPUT TYPE="checkbox" NAME="os" VALUE="WINDOWS">Windows 8
  <INPUT TYPE="checkbox" NAME="os" VALUE="ANDROID">ANDROID
  <INPUT TYPE="radio" NAME="Editeur" VALUE="NotePad">Vous utilisez NotePad
  <INPUT TYPE="radio" NAME="Editeur" VALUE="Komodo">Vous utilisez KOMODO
  <INPUT TYPE="submit" VALUE="Valider">
</FORM>
```

- URL (si méthode GET)

`traiterFormulaire.php?entreprise=APPLE&listeDomaine=info&os=LINUX,ANDROID&Editeur=Komodo`

Recueillir les informations du formulaire

- Utiliser les variables formulaires
 - Lire les valeurs des zones de texte, des listes, des cases à cocher, des boutons radio, ... grâce à la variable globale `$_POST`
 - Ranger ces valeurs dans des variables programmes :
 - `$Societe=$_POST['entreprise'];`
 - `$Systeme=$_POST['os'];`
 - Eventuellement sécuriser leur contenu avec des fonctions standards
 - `$Societe=htmlspecialchars($_POST['entreprise']);`
 - `$Nom=strip_tags($_POST['nom']);`
 - ...



Exemple de Mathieu Nebra (formulaire html)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Page protégée par mot de passe</title>
  </head>
  <body>
    <p>Veuillez entrer le mot de passe pour obtenir
    les codes d'accès au serveur central de la NASA :</p>
    <form action="secret.php" method="post">
      <p>
        <input type="password" name="mot_de_passe" />
        <input type="submit" value="Valider" />
      </p>
    </form>
    <p>Cette page est réservée au personnel de la NASA.
    Si vous ne travaillez pas à la NASA, inutile d'insister
    vous ne trouverez jamais le mot de passe ! ;-)</p>
  </body>
</html>
```

Exemple de Mathieu Nebra (php appelé)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Codes d'accès au serveur central de la NASA</title>
  </head>
  <body>
    <?php
    if (isset($_POST['mot_de_passe']) AND $_POST['mot_de_passe'] == "kangourou")
    { // si bon mot de passe on affiche un message de succès en HTML
    ?>
      <h1>Voici les codes d'accès :</h1>
      <p><strong>CRD5-GTFT-CK65-JOPM-V29N-24G1-HH28-LLFV</strong></p>
      <p>Cette page est réservée au personnel de la NASA. <br />
      N'oubliez pas de la visiter régulièrement. <br />
      La NASA vous remercie de votre visite.</p>
    <?php
    }
    else // Sinon, on affiche un message d'erreur en PHP
    {
      echo '<p>Mot de passe incorrect</p>';
    }
    ?>
  </body>
</html>
```

Conserver l'information sur le WEB

- Principe
- Différentes façon de conserver l'information
 - Les variables de session
 - Les cookies
 - Les fichiers sur le serveur (voir cours correspondant)
 - Les bases de données sur le serveur (voir cours correspondant)

La variable de session

● Principe

- Un nom et un identifiant
- Tout type de données
- Ces variables sont accessibles de manière privée
- Accès uniquement pendant la session (détruite quelques instants après la fermeture voir php.ini)

● Utilisation par des fonctions dédiées ou le tableau \$_SESSION

- `session_start()`, `session_unset()` et `session_destroy()`
- `session_register(var)`
- `session_unregister(var)`
- `session_isregistered(var)`
- `$mavaleursession = $_SESSION['var']`

Exploitation de variables de session

- Initialiser chaque page avec un début de session

- avant tout code HTML

```
session_start();
```

- Récupération des informations de sessions écrites par autres pages

- par le contenu du tableau session
- par le contenu d'informations systèmes

```
$alias = $_SESSION['alias'];
```

```
$codesession = session_id();
```

- Autres actions

- éventuellement destruction de la session
- modifier un contenu dans tableau session
- écrire un nouveau contenu dans tableau session

```
session_destroy();
```

```
$_SESSION['alias'] = "Georgio";
```

```
$_SESSION['connect'] = TRUE
```

Exemple de Mathieu Nebra (initialisation de session)

```
<?php
// On démarre la session AVANT d'écrire du code HTML
session_start();

// On s'amuse à créer quelques variables de session dans $_SESSION
$_SESSION['prenom'] = 'Jean';
$_SESSION['nom'] = 'Dupont';
$_SESSION['age'] = 24;
?>

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Titre de ma page</title>
  </head>
  <body>

    <p>
      Salut <?php echo $_SESSION['prenom']; ?> !<br />
      Tu es à l'accueil de mon site (index.php). Tu veux aller sur une autre page ?
    </p>

    <p>
      <a href="mapage.php">Lien vers mapage.php</a><br />
      <a href="monscript.php">Lien vers monscript.php</a><br />
      <a href="informations.php">Lien vers informations.php</a>
    </p>

  </body>
</html>
```

Exemple de Mathieu Nebra (utilisation de session)

```
<?php
session_start(); // On démarre la session AVANT toute chose
?>

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Titre de ma page</title>
  </head>
  <body>

    <p>Re-bonjour !</p>

    <p>
      Je me souviens de toi ! Tu t'appelles <?php echo $_SESSION['prenom'] . ' ' .
$_SESSION['nom']; ?> !<br />
      Et ton âge hummm... Tu as <?php echo $_SESSION['age']; ?> ans, c'est ça ? :-D
    </p>

  </body>
</html>
```

Le cookie

- Qu'est-ce que c'est ?
 - Petit fichier sur disque client
 - Dossier de stockage décidé par le navigateur
- Accès physique
 - Lecture du fichier cookie
 - Uniquement son créateur
- Sécurité et Confidentialité
 - Option débrayable sur tout navigateur
 - Aucune garantie d'exécution

Exemple de Mathieu Nebra (initialisation de cookie)

```
<?php
setcookie('pseudo', 'M@teo21', time() + 365*24*3600, null, null, false, true); // On
écrit un cookie
setcookie('pays', 'France', time() + 365*24*3600, null, null, false, true); // On écrit
un autre cookie...

// Et SEULEMENT MAINTENANT, on peut commencer à écrire du code html
?>

<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8" />
        <title>Ma super page PHP</title>
    </head>
    <body>

etc.
```

Exemple de Mathieu Nebra (utilisation de cookie)

```
1 <p>
2   Hé ! Je me souviens de toi !<br />
3   Tu t'appelles <?php echo $_COOKIE['pseudo']; ?> et tu viens de <?php echo
   $_COOKIE['pays']; ?> c'est bien ça ?
4 </p>
```

Exploitation de cookies

● Définir le cookie

- Nom
- Contenu
- Date d'expiration

```
setcookie("monCookie", "bonjour", time() + 3600);
```

● Durée de vie du cookie

- Date d'expiration
- Suppression explicite

```
setcookie("monCookie");
```

● Récupération du cookie

- Variable globale `_COOKIE`
- Variable globale `_REQUEST`

```
$_COOKIE("monCookie")
```

Quiz



- ▶ Trier par ordre de rapidité (1, 2, 3, 4, 5, ...), les accès aux informations via les techniques PHP suivantes (cookie, fichier, BD, nuage, session, ...)
- ▶ Le passage de paramètres PHP est plus sécurisé avec la méthode GET que POST (Oui/Non) ?
- ▶ La variable globale contenant les saisies des utilisateurs sur un formulaire est ... ?
- ▶ Marier les supports physique (mémoire, session, cookie, fichier, base de donnée) avec leurs particularité de vie (volatile, non volatile)
- ▶ Le langage PHP utilise les fonctions suivantes pour les sessions :
session_start(), session_unset(), session_destroy(), session_stop(), session_cookie(), session_register()

Exercices



- ✓ Afficher quelques variables d'environnement
- ✓ Afficher en boucle les variables d'environnement
- ✓ Créer un formulaire simple HTML avec méthode GET
- ✓ Créer un formulaire simple HTML avec méthode POST
- ✓ Créer une variable de session simple
- ✓ Créer un cookie simple