

WEB DYNAMIQUE

INTRODUCTION AU LANGAGE PHP



PHP

Les formulaires

Qu'est-ce qu'un formulaire ?

Un formulaire est un élément **HTML** qui permet aux utilisateurs de saisir des données et de les envoyer à un serveur pour traitement.

Les formulaires sont utilisés pour diverses tâches, telles que la connexion, **l'inscription**, **la soumission** de commentaires et la recherche.



PHP

Les formulaires

Création d'un formulaire HTML

Pour créer un formulaire, vous devez utiliser la balise **<form>**. Cette balise doit contenir au moins un attribut action qui spécifie **l'URL** du script **PHP** qui traitera les données du formulaire. Vous pouvez également utiliser l'attribut **method** pour spécifier la méthode d'envoi des données (**GET** ou **POST**).



PHP

Les formulaires

Création d'un formulaire HTML

À l'intérieur de la balise **<form>**, vous pouvez ajouter différents éléments de formulaire, tels que des champs de texte (**<input type="text">**), des zones de texte (**<textarea>**), des cases à cocher (**<input type="checkbox">**), des boutons radio (**<input type="radio">**) et des listes déroulantes (**<select>**).

Chaque élément de formulaire doit avoir un attribut **name** qui identifie le champ dans le script **PHP**.



PHP

Les formulaires

Création d'un formulaire HTML

Voici un exemple de formulaire:

```
<form action="traitement.php" method="post">  
  <label for="nom">Nom :</label>  
  <input type="text" name="nom" id="nom"><br>  
  <label for="email"> Email :</label>  
  <input type="email" name="email" id="email"><br>  
  
  <input type="submit" value="Envoyer">  
</form>
```



PHP

Les formulaires

Traitement des données du formulaire avec PHP :

Une fois que l'utilisateur soumet **le formulaire**, les données sont envoyées au script **PHP** spécifié dans l'attribut **action**.

Vous pouvez ensuite utiliser les variables superglobales **\$_POST** (pour la méthode **POST**) ou **\$_GET** (pour la méthode **GET**) pour récupérer les valeurs des champs du **formulaire**.



PHP

Les formulaires

Traitement des données du formulaire avec PHP :

Voici un exemple de script PHP qui traite les données du formulaire ci-dessus :

```
<?php
```

```
$nom = $_POST['nom'];
```

```
$email = $_POST['email'];
```

```
echo "Bonjour " . $nom . ", votre email est " . $email ;
```

```
?>
```



PHP

Les formulaires

Sécurité des formulaires:

Il est important de prendre en compte la sécurité lors de la création de formulaires. Voici quelques mesures à prendre :

Validation des données :

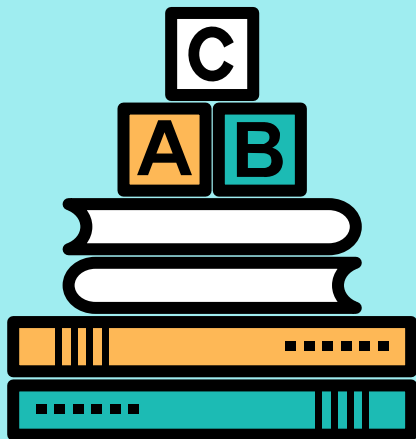
Assurez-vous de valider les données saisies par l'utilisateur avant de les traiter. Cela permet d'éviter les erreurs et les attaques malveillantes.

Protection contre les injections SQL :

Utilisez des requêtes préparées ou des fonctions d'échappement pour vous protéger contre les injections SQL.

Protection contre les attaques XSS :

Utilisez la fonction `htmlspecialchars()` pour échapper les caractères spéciaux dans les données affichées.



GET & POST

DU THEORIE AU PRATIQUE : EXEMPLES



PHP

Les formulaires

La méthode GET :

- ❖ Les données sont envoyées via l'URL
- ❖ Visible dans la barre d'adresse
- ❖ Limitée en taille (environ 2048 caractères)
- ❖ Utilisée pour des requêtes non sensibles



PHP

Les formulaires

La méthode GET :

```
<html lang="fr"><head>
<title>Exemple de Formulaire GET</title></head>
<body>
  <h2>Formulaire d'envoi avec méthode GET</h2>
  <form action="" method="GET">
    <label for="name">Nom :</label>
    <input type="text" name="name" required><br>
    <label for="age">Âge :</label>
    <input type="number" name="age" min="1" required><br>
    <button type="submit">Envoyer</button>
  </form>
```



PHP

Les formulaires

La méthode GET :

```
<?php
```

```
if (isset($_GET['name']) && isset($_GET['age'])) {  
    $name = htmlspecialchars($_GET['name']);  
    $age = intval($_GET['age']);  
    echo "<p>Bonjour, " . $name . "!<br>";  
    echo "Vous avez " . $age . " ans.</p>"; }  

```

```
?>
```

```
</body>
```

```
</html>
```



PHP

Les formulaires

Comment cela fonctionne :

Formulaire HTML :

Le formulaire utilise la méthode GET pour envoyer les données. Les champs name et age sont obligatoires (required), ce qui force l'utilisateur à remplir ces champs avant de soumettre le formulaire.

Traitement des données avec PHP :

Une fois que le formulaire est soumis, les données sont ajoutées à l'URL sous forme de paramètres query string (par exemple : ?name=fatima &age=25).

Le script PHP récupère ces paramètres via \$_GET, les traite pour garantir leur sécurité (avec htmlspecialchars et intval), puis affiche les informations.



PHP

Les formulaires

La méthode POST :

- ❖ Les données sont envoyées dans le corps de la requête HTTP.
- ❖ Non visible dans l'URL.
- ❖ Pas de limite de taille pratique.
- ❖ Recommandée pour les données sensibles (mots de passe, etc.).



PHP

Les formulaires

La méthode POST :

```
<!DOCTYPE html><html lang="fr"><head>
    <title>Exemple de Formulaire POST</title></head><body>
    <h2>Formulaire d'envoi avec méthode POST</h2>
    <form action="" method="POST">
        <label for="name">Nom :</label>
        <input type="text" id="name" name="name" required><br><br>
        <label for="email">Email :</label>
        <input type="email" id="email" name="email" required><br><br>
        <label for="password">Mot de passe :</label>
        <input type="password" id="password" name="password" required>
        <button type="submit">Envoyer</button>
    </form>
```



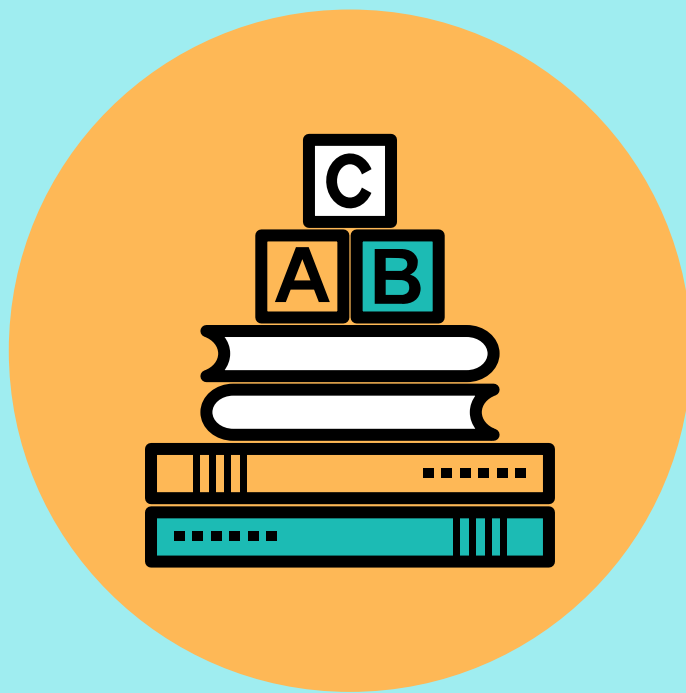
PHP

Les formulaires

La méthode POST :

```
<?php
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $name = htmlspecialchars($_POST['name']);
    $email = filter_var($_POST['email'], FILTER_SANITIZE_EMAIL);
    $password = $_POST['password'];
    echo "<p>Bonjour, " . $name . "!<br>";
    echo "Votre email est : " . $email . "<br>";
}
?>

</body></html>
```

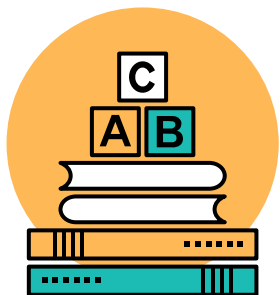
EXERCICES DE SYNTHÈSE

LES FORMULAIRES EN PHP

EXERCICE 1

Énoncé :

Créer un formulaire HTML avec un champ pour le nom et afficher le nom soumis en PHP.

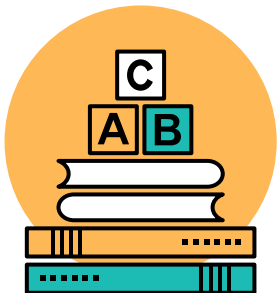


EXERCICE 1

Solution :

```
<form method="get" action="">
  <input type="text" name="nom">
  <button type="submit">Envoyer</button>
</form>
```

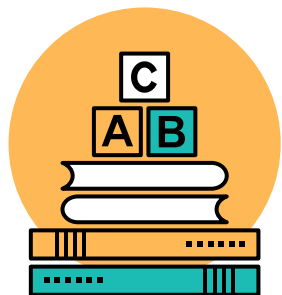
```
<?php
if (isset($_GET["nom"])) {
  echo "Bonjour, " . $_GET["nom"] . " !";
}
?>
```



EXERCICE 2

Énoncé :

Créer un formulaire HTML avec un champ pour l'âge et afficher un message si l'utilisateur est majeur.



EXERCICE 2

```
<form method="post" action="">
  <input type="number" name="age">
  <button type="submit">Envoyer</button>
</form>
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $age = $_POST["age"];
    if ($age >= 18) {
        echo "Vous êtes majeur.";
    } else {
        echo "Vous êtes mineur.";
    } }
?>
```

