



Le filtrage de données

Langage PHP avancé

El hadji Mamadou NGUER Enseignant chercheur en Informatique à l'UVS



Séquence 3 : Le filtrage des données dans PHP



Objectifs spécifiques : A la suite de cette séquence, l'étudiant doit être capable de :

1. décrire l'importance du filtrage des données dans le web
2. utiliser la fonction `filter_var` pour filtrer des chaînes de caractères, des email, des URL, etc.

Séquence 3 : Le filtrage des données

≡

Plan de la séquence :

1. Introduction
2. La fonction `filter_var()`
3. Les exemples de filtrage
4. Conclusion

Introduction

Définitions:

- Valider des données = Déterminer si elles sont en bonne forme.
- Assainir des données = Supprimer tout caractère illégal des données.

L'extension Filter de PHP

- Les filtres PHP sont utilisés pour valider et assainir les entrées externes.
- L'extension Filter de PHP comporte plusieurs fonctions nécessaires à la vérification des entrées des utilisateurs et est conçue pour rendre la validation des données plus facile et plus rapide.
- La fonction **filter_list()** peut être utilisée pour indiquer ce que contient l'extension Filter :

```
<?php
foreach (filter_list() as $id =>$filter) {
    echo '<tr><td>' . $filter . '</td><td>' . filter_id($filter) . '</td></tr>';
}
?>
```

Introduction

Pourquoi utiliser des filtres?

De nombreuses applications Web reçoivent des données externes pouvant être:

- Des entrées de formulaire
- Des cookies
- Des données de services Web
- Des variables de serveur
- Des résultats de requête en base de données

Et ces données externes doivent toujours être validées car des données externes non validées peuvent entraîner des problèmes de sécurité et détruire votre site Web!

En utilisant des filtres PHP, vous pouvez être sûr que votre application reçoit les bonnes données!

La fonction `filter_var()`

- Elle permet de valider et de nettoyer des données.
- Elle reçoit en paramètre la variable à filtrer et le type de contrôle à utiliser
- Syntaxe : `filter_var($var, "Type de contrôle");`
- Elle peut être utilisée par exemple pour :
 - Le nettoyage de chaîne de caractères
 - La validation de nombre
 - La validation d'adresse IP
 - Le nettoyage et la validation d'adresse email
 - Le nettoyage et la validation d'URL
 - Etc.

Exemples de filtrage

Nettoyage d'une chaîne

- L'exemple suivant utilise la fonction `filter_var()` pour supprimer toutes les balises HTML dans une chaîne:

- **Exemple**

```
<?php
$str = "<h1>Bonjour!</h1>";
$newstr = filter_var($str, FILTER_SANITIZE_STRING);
echo $newstr;
?>
```

Exemples de filtrage

Validation d'un entier

- L'exemple suivant utilise la fonction `filter_var ()` pour vérifier si la variable `$int` est un nombre entier.

- **Exemple**

```
<?php
```

```
$int = 100;
```

```
if (!filter_var($int, FILTER_VALIDATE_INT) === false) {
```

```
    echo("Entier valide");
```

```
} else {
```

```
    echo("Entier non valide");
```

```
}
```

```
?>
```

Attention : Dans l'exemple ci-dessus, si `$int = 0`, la fonction renverra "Entier pas valide". Pour résoudre ce problème, on utilise le code suivant:

```
if (filter_var($int, FILTER_VALIDATE_INT) === 0 || !filter_var($int,  
FILTER_VALIDATE_INT) === false){
```

```
    echo("Entier valide");
```

```
}
```

Exemples de filtrage

Validation d'une adresse IP

- L'exemple suivant utilise la fonction `filter_var ()` pour vérifier si la variable `$ip` est une adresse IP valide:

- **Exemple**

```
<?php
```

```
$ip = "127.0.0.1";
```

```
if (!filter_var($ip, FILTER_VALIDATE_IP) === false) {  
    echo("$ip est une adresse IP valide");  
} else {  
    echo("$ip n'est pas une adresse IP valide");  
}  
?>
```

Exemples de filtrage

Nettoyage et validation d'une adresse email

- L'exemple suivant utilise la fonction `filter_var()` pour supprimer d'abord tous les caractères illégaux de la variable `$e-mail`, puis vérifier s'il s'agit d'une adresse e-mail valide:

- **Exemple**

```
<?php
```

```
$email = "alaxam.njaay@exemple.com";
```

```
// Supprimer tous les caractères illégaux du courrier électronique
```

```
$email = filter_var($email, FILTER_SANITIZE_EMAIL);
```

```
// Valider l'adresse e-mail
```

```
if (!filter_var($email, FILTER_VALIDATE_EMAIL) === false) {
```

```
    echo("$email est une adresse email valide");
```

```
} else {
```

```
    echo("$email n'est pas une adresse email valide");
```

```
}
```

```
?>
```

Exemples de filtrage

Nettoyage et validation d'une URL

- L'exemple suivant utilise la fonction `filter_var ()` pour supprimer d'abord tous les caractères illégaux d'une URL, puis vérifier si l'URL est valide:

- **Exemple**

```
<?php
```

```
$url = "https://www.ugb.sn";
```

```
// Supprimer tous les caractères illégaux d'une URL
```

```
$url = filter_var($url, FILTER_SANITIZE_URL);
```

```
// Valider une url
```

```
if (!filter_var($url, FILTER_VALIDATE_URL) === false) {
```

```
    echo("$url est une URL valide");
```

```
} else {
```

```
    echo("$url n'est pas une URL valide");
```

```
}
```

```
?>
```

Exemples de filtrage

Validation d'un nombre entier dans une plage

- L'exemple suivant utilise la fonction `filter_var()` pour vérifier si une variable est un entier compris entre 1 et 200:

- **Exemple**

```
<?php  
$int = 122;  
$min = 1;  
$max = 200;
```

```
if (filter_var($int, FILTER_VALIDATE_INT, array("options" =>  
array("min_range"=>$min, "max_range"=>$max))) === false) {  
    echo("La valeur de la variable n'est pas dans l'intervalle");  
} else {  
    echo("La valeur de la variable est dans l'intervalle");  
}  
?>
```

Exemples de filtrage

Validation d'une adresse IPv6

- L'exemple suivant utilise la fonction **filter_var ()** pour vérifier si la variable \$ip a une adresse IPv6 valide:

- **Exemple**

```
<?php
```

```
$ip = "2001:0db8:85a3:08d3:1319:8a2e:0370:7334";
```

```
if (!filter_var($ip, FILTER_VALIDATE_IP, FILTER_FLAG_IPV6) === false) {  
    echo("$ip est une adresse IPv6 valide");  
} else {  
    echo("$ip n'est pas une adresse IPv6 valide ");  
}  
?>
```

Pour une référence complète sur les filtres:

https://www.w3schools.com/php/php_ref_filter.asp

Conclusion

Au cours de cette séquence, nous avons vu :

- l'importance du filtrage des données dans le web
- et comment utiliser la fonction `filter_var` pour effectuer différents filtres (filtre des chaînes de caractères, d'email, d'URL, etc.).

Il vous reste maintenant effectuer les tests de connaissance et les travaux pratiques de la séquence pour consolider vos acquis.