


ement sites web 1€/mois - 100% illimité,  
e Gratuit

le données, SSL, Bande passante illimitée, Baie de stockage SSD hosteur.com  
ites



## Les finalistes compétissent pour une part des 1.5 million dollars en subventions

Africa's Business Heroes

Ou

## Présentation des conditions et des opérateurs de comparaison

Télécharger le PDF  
du cours



Cours complet PHP  
et MySQL

INTRODUCTION AU COURS PHP  
ET MYSQL

1. Introduction au cours :  
définitions et rôles du PHP et du  
MySQL
2. Client et serveur : définitions et  
interactions
3. Mise en place de notre  
environnement de travail
4. Créer, enregistrer et exécuter un  
script PHP

Dans cette nouvelle partie, nous allons étudier et comprendre l'intérêt des structures de contrôle en PHP. Une structure de contrôle est un ensemble d'instructions qui permet de contrôler l'exécution du code.

Il existe différents types de structures de contrôle. Les deux types les plus connus et les plus utilisés sont les structures de contrôle conditionnelles qui permettent d'exécuter un bloc de code si une certaine condition est vérifiée et les structures de contrôle de boucle qui permettent d'exécuter un bloc de code en boucle tant qu'une condition est vérifiée.

Nous allons déjà commencer avec l'étude des structures de contrôle conditionnelles.

### Présentation des conditions en PHP

Les structures de contrôle conditionnelles (ou plus simplement conditions) vont nous permettre d'exécuter différents blocs de code selon qu'une condition spécifique soit vérifiée ou pas.

Par exemple, on va pouvoir utiliser les conditions pour afficher un message de bienvenue différent en PHP sur notre site selon que l'utilisateur soit connu ou un simple visiteur qui ne s'est jamais inscrit sur notre site.



## Les finalistes compétissent pour une part des 1.5 million dollars en subventions

Africa's Business Heroes

Ou

# ement sites web 1€/mois - 100% illimité, e Gratuit

le données, SSL, Bande passante illimitée, Baie de stockage SSD hôteur.com  
ites

8. Opérateurs et concaténation en PHP

[LES STRUCTURES DE CONTRÔLE EN PHP](#)

## 9. Présentation des conditions et des opérateurs de comparaison

10. Les conditions if, if...else et if...elseif...else

11. Créer des conditions robustes avec les opérateurs logiques

12. Ecrire des conditions condensées avec les opérateurs ternaire et fusion null

13. L'instruction switch en PHP

14. Les boucles PHP et les opérateurs d'incrément et de décrément

15. Inclure des fichiers dans un autre en PHP avec include et require

[DÉCOUVERTE DES FONCTIONS EN PHP](#)

16. Introduction aux fonctions PHP

17. Contrôler le passage des arguments

18. Contrôler les valeurs de retour d'une fonction

19. La portée des variables en PHP

20. Constantes et constantes magiques en PHP

[LES VARIABLES TABLEAUX EN PHP](#)

21. Présentation des tableaux et tableaux numérotés en PHP

22. Les tableaux associatifs en PHP

23. Les tableaux multidimensionnels en PHP

[MANIPULER DES DATES EN PHP](#)

24. Le timestamp UNIX et la date

- La condition **if** (si) ;
- La condition **if... else** (si... sinon) ;
- La condition **if... elseif... else** (si... sinon si... sinon).

Nous allons étudier chacune de ces conditions dans la suite de cette partie.

## Présentation des opérateurs de comparaison

Comme je l'ai précisé plus haut, nous allons souvent construire nos conditions autour de variables : selon la valeur d'une variable, nous allons exécuter tel bloc de code ou pas.

En pratique, nous allons donc comparer la valeur d'une variable à une certaine autre valeur donnée et selon le résultat de la comparaison exécuter un bloc de code ou pas. Pour comparer des valeurs, nous allons devoir utiliser des opérateurs de comparaison.



Voici ci-dessous les différents opérateurs de comparaison disponibles en PHP ainsi que leur signification :

Opérateur	Définition
==	Permet de tester l'égalité sur les valeurs
===	Permet de tester l'égalité en termes de valeurs et de types
!=	Permet de tester la différence en valeurs
<>	Permet également de tester la différence en valeurs
!==	Permet de tester la différence en valeurs ou en types
<	Permet de tester si une valeur est strictement inférieure à une autre
>	Permet de tester si une valeur est strictement supérieure à une autre
<=	Permet de tester si une valeur est inférieure ou égale à une autre
>=	Permet de tester si une valeur est supérieure ou égale à une autre

Certain de ces opérateurs nécessitent certainement une précision de ma part. Avant tout, vous devez bien comprendre que lorsqu'on utilise un opérateur de comparaison en PHP, on n'indique pas au PHP que telle valeur est supérieure, inférieure, égale ou différente de telle autre.

ement sites web 1€/mois - 100% illimité,  
e Gratuit

le données, SSL, Bande passante illimitée, Baie de stockage SSD hôteur.com  
ites

<div>27. Les variables superglobales PHP</div> <div>28. Création et gestion des cookies en PHP</div> <div>29. Définir et utiliser les sessions en PHP</div> <div>MANIPULER DES FICHIERS EN PHP</div> <div>30. Introduction à la manipulation de fichiers en PHP</div> <div>31. Ouvrir, lire et fermer un fichier en PHP</div> <div>32. Créer et écrire dans un fichier en PHP</div> <div>33. Autres opérations sur les fichiers en PHP</div> <div>UTILISER LES EXPRESSIONS RÉGULIÈRES OU RATIONNELLES EN PHP</div> <div>34. Introduction aux expressions rationnelles ou expressions régulières</div> <div>35. Les fonctions PCRE PHP</div> <div>36. Les classes de caractères des regex</div> <div>37. Les métacaractères des regex PHP</div> <div>38. Les options des expressions régulières disponibles en PHP</div> <div>PROGRAMMATION ORIENTÉE OBJET (POO) PHP : CONCEPTS DE BASE</div> <div>39. Introduction à la programmation orientée objet PHP : classes, instances et objets</div> <div>40. Propriétés et méthodes en PHP orienté objet</div> <div>41. Les méthodes PHP constructeur et destructeur</div> <div>42. Encapsulation et visibilité des propriétés et méthodes PHP</div> <div>43. Classes étendues et héritage en PHP orienté objet</div>	<div><div>Les finalistes compétissent pour une part des 1.5 million dollars en subventions</div><div>Où</div><div>Africa's Business Heroes</div></div> <div>Revenons à nos opérateurs. Tout d'abord, notez que notre « égal » mathématique (l'égalité en termes de valeurs) se traduit en PHP par le double signe égal <code>==</code>.</div> <div>Ensuite, certains d'entre vous doivent certainement se demander ce que signifie le triple égal. Lorsqu'on utilise un triple égal <code>===</code>, on cherche à effectuer une comparaison non seulement sur la valeur mais également sur le type des deux opérandes.</div> <div>Prenons un exemple simple pour illustrer cela. Imaginons que l'on possède une variable <code>\$x</code> dans laquelle on stocke le chiffre 4. On veut ensuite comparer la valeur stockée dans notre variable à la chaîne de caractères « 4 ».</div> <div>Si on utilise le double signe égal pour effectuer la comparaison, l'égalité va être validée par le PHP car celui-ci ne va tester que les valeurs, et 4 est bien égal à « 4 » en termes de valeurs.</div> <div>En revanche, si on utilise le triple signe égal, alors l'égalité ne va pas être validée car nous comparons un nombre à une chaîne de caractères (donc des types différents de valeurs).</div> <div>On va suivre exactement le même raisonnement pour les deux opérateurs <code>!=</code> et <code>!==</code> qui vont nous permettre de tester respectivement la différence en termes de valeurs simplement et la différence en termes de valeurs ou de type.</div> <div>Utiliser les opérateurs de comparaison</div> <div>Il y a différentes choses que vous devez savoir et comprendre pour bien utiliser les opérateurs de comparaison.</div> <div>La première chose à savoir est que lorsqu'on utilise un opérateur de comparaison, le PHP va comparer la valeur à gauche de l'opérateur à celle à droite. On dit également qu'il évalue la comparaison.</div> <div>Si la comparaison est vérifiée ou validée, alors le PHP renvoie la valeur booléenne <code>true</code>. Si le test de comparaison échoue, alors PHP renvoie la valeur booléenne <code>false</code>. Cela est très important à comprendre car nos conditions vont s'appuyer sur cette valeur booléenne pour décider du code à exécuter ou pas.</div> <div><pre>1. &lt;!DOCTYPE html&gt; 2. &lt;html&gt; 3.   &lt;head&gt; 4.     &lt;title&gt;Cours PHP &amp; MySQL&lt;/title&gt; 5.     &lt;meta charset="utf-8"&gt; 6.     &lt;link rel="stylesheet" href="cours.css"&gt; 7.   &lt;/head&gt; 8.</pre></div>
---	---

ement sites web 1€/mois - 100% illimité,

e Gratuit

Le données, SSL, Bande passante illimitée, Baie de stockage SSD hôteur.com  
ites

47. Les méthodes et les classes abstraites en PHP objet

48. Les interfaces en PHP orienté objet

49. Les méthodes magiques en orienté objet PHP

PROGRAMMATION ORIENTÉE  
OBJET PHP : NOTIONS  
AVANCÉES

50. Le chainage de méthodes en PHP

51. Les closures et les classes anonymes en PHP objet

52. L'auto chargement des classes en PHP

53. Le mot clef final en PHP objet

54. La résolution statique à la volée ou late static bindings en PHP

55. Utiliser les traits en orienté objet PHP

56. L'interface Iterator et le parcours d'objets en PHP

57. Le passage d'objets en PHP : identifiants et références

58. Le clonage d'objets et la méthode magique PHP \_\_clone()

59. La comparaison d'objets PHP

ESPACES DE NOMS, FILTRES ET  
GESTION DES ERREURS EN PHP

60. Les espaces de noms PHP

61. Présentation des filtres PHP

62. Filtres de validation, de nettoyage et drapeaux de l'extension PHP Filter

63. Utilisation pratique des filtres en PHP

64. Définition et gestion des erreurs en PHP

65. Déclenchement, capture et gestion des exceptions PHP : try, throw, catch

19. var\_dump(\$x == "4");  
20. echo '<br>';  
21.  
22. /\*On compare la valeur de \$x à la chaine de caractères "4" en  
23. \*valeur simplement\*/  
24. var\_dump(\$x == "4");  
25. echo '<br>';  
26.  
27. /\*On compare la valeur de \$x à la chaine de caractères "4" en  
28. \*termes de valeur et de type\*/  
29. var\_dump(\$x === "4");  
30. echo '<br>';  
31.  
32. var\_dump(\$x != "4");  
33. echo '<br>';  
34.  
35. var\_dump(\$x !== "4");  
36. echo '<br>';  
37. ?>  
38. <p>Un paragraphe</p>  
39. </body>  
40. </html>

Les finalistes  
compétissent  
pour une part des  
1.5 million dollars  
en subventions

Où

Africa's Business Heroes

Comme je voulais absolument vous montrer que le PHP renvoyait bien une valeur booléenne

Ce site utilise des cookies pour vous fournir la meilleure expérience de navigation possible. En continuant sur ce site, vous acceptez l'utilisation des cookies.

Réglages

ACCEPTER

https://www.pierre-giraud.com/php-mysql-apprendre-coder-cours/operateur-comparaison/

4/6

# ement sites web 1€/mois - 100% illimité, e Gratuit

le données, SSL, Bande passante illimitée, Baie de stockage SSD hôteur.com  
ites

PHPMyAdmin	résultats en soi.
68. Se connecter à une base de données MySQL en PHP	Pour commencer, on compare la valeur contenue dans \$x au chiffre 4. Si le PHP considère l'égalité vérifiée, il renvoie le booléen true. Dans le cas contraire, il renverra code>>false.
69. Créer une base de données MySQL et une table dans la base	Comme vous pouvez le voir, c'est la valeur true qui est renvoyée. En effet, la valeur de notre variable est bien égale en valeur au chiffre 4.
MANIPULER DES DONNÉES DANS DES BASES MYSQL AVEC PDO	On compare ensuite la valeur contenue dans \$x au chiffre 7 avec l'opérateur de supériorité absolue. Si le PHP détermine que la valeur contenue dans \$x est strictement supérieure au chiffre 7, il renvoie true. Dans le cas contraire, il renvoie code>>false. Dans le cas présent, \$x contient le chiffre 4, qui n'est pas strictement supérieur au chiffre 7. C'est donc la valeur code>>false qui est renvoyée.
70. Insérer des données dans une table MySQL	
71. Les requêtes MySQL préparées avec PDO PHP	Pour notre troisième comparaison, on décide de comparer la valeur de \$x à la chaîne de caractères « 4 ». On utilise le double signe égal pour cela, on ne va donc comparer que les valeurs. Comme la valeur 4 est bien égale en valeur à la chaîne de caractères « 4 », PHP renvoie true.
72. Modifier les données d'une table MySQL ou sa structure	Ensuite, on compare à nouveau la valeur contenue dans notre variable \$x à la chaîne de caractères « 4 ». Cependant, cette fois-ci, on utilise le triple signe égal.
73. Supprimer des données, une table ou une base de données MySQL	En faisant cela, on signifie que l'on veut comparer les valeurs mais également les types de chacune des deux valeurs. Comme un nombre et une chaîne de caractères n'ont pas le même type, PHP renvoie cette fois-ci false.
74. Sélection simple de données dans une table MySQL en PHP	Dans notre cinquième opération, on demande à PHP de déterminer si la valeur contenue dans \$x différente (en valeur) de la chaîne de caractères « 4 ». Comme le chiffre 4 n'est pas différent en valeur de la chaîne de caractères « 4 », le PHP renvoie false (car rappelez-vous qu'on teste ici la différence).
75. Utiliser des critères de sélection pour sélectionner des données dans une table MySQL	Enfin, on demande à PHP de déterminer si la valeur contenue dans \$x est différente en valeur ou en type de la chaîne de caractères « 4 ». Le chiffre 4 est bien d'un type différent de la chaîne de caractères « 4 », et donc le PHP renvoie true.
76. Utiliser les fonctions d'agrégation et les fonctions scalaires SQL	
JOINTURES, UNION ET SOUS REQUÊTES	
77. Présentation des jointures SQL	
78. Création de jointures SQL	
79. L'opérateur SQL UNION	
80. Les opérateurs de sous requête SQL	La dernière version majeure du PHP, le PHP 7 a introduit deux nouveaux opérateurs de comparaison qui se comportent différemment des précédents puisque le PHP ne va pas renvoyer true ou false à l'issue de la comparaison.
GESTION DES FORMULAIRES HTML AVEC PHP	
81. Rappels sur les formulaires HTML	Le premier de ces opérateurs est l'opérateur « spaceship » <=>. A l'issue de la comparaison des deux opérandes, le PHP va ici renvoyer :
82. Récupérer et manipuler les données des formulaires HTML en PHP	<ul style="list-style-type: none"><li>• 0 dans le cas où les deux opérandes sont égaux ;</li><li>• -1 si l'opérande à gauche de l'opérateur est plus petit que celui de droite ;</li><li>• 1 si l'opérande à gauche de l'opérateur est plus grand que celui de droite.</li></ul>
83. Sécurisation et validation des formulaires en PHP	Le deuxième opérateur est l'opérateur « fusion null » ?. Cet opérateur va principalement nous permettre de comparer le contenu de deux variables. Le PHP va ici renvoyer :
CONCLUSION DU COURS PHP ET MYSQL	<ul style="list-style-type: none"><li>• La valeur de l'opérande à droite de l'opérateur si la valeur de l'opérande à gauche est</li></ul>

# ement sites web 1€/mois - 100% illimité, e Gratuit

le données, SSL, Bande passante illimitée, Baie de stockage SSD hosteur.com  
ites



## Laisser un commentaire

Vous devez vous connecter pour publier un commentaire.

[Connexion](#) [Confidentialité](#) [CGV](#) [Sitemap](#)

© Pierre Giraud - Toute reproduction interdite - Mentions légales