

Les boucles en PHP

Contenu

- Rappels: Les conditions
- Le concept de boucle
- La boucle *While*
- La boucle *Do...While*
- La boucle *For*

Rappels: Les conditions

Les structures conditionnelles

- Au dernier cours, nous avons vu les **structures conditionnelles**

```
if (expression booléenne) {
```

Instructions à exécuter si l'expression est **vraie**

```
} else {
```

Instructions à exécuter si l'expression est **fausse**

```
}
```

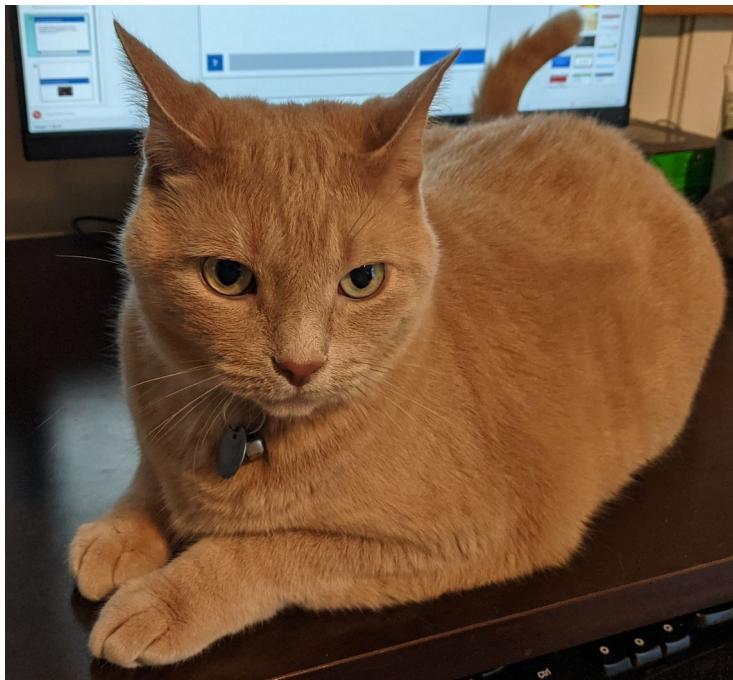
Les conditions

- Nous avons vu qu'une expression booléenne utilisée dans un **if** est appelée une **condition**
- Une condition peut être par exemple:
 - Une variable de type booléen
 - La comparaison de deux variables
 - `==`, `<`, `>`, etc.
 - Une expression plus complexe utilisant des opérateurs logiques
 - `&&`, `||`, `!`

Le concept de boucle

Le concept de boucle

- En programmation, une boucle permet d'exécuter une instruction ou un bloc d'instructions à répétition, **tant qu'** une condition est vraie



Tant qu'il reste
des objets sur
la table, j'en fais
tomber un par terre.

Le concept de boucle

- **Les boucles sont omniprésentes dans les programmes informatiques**
 - Ex: chaque fois qu'on peut identifier quelque chose de semblable qui se répète dans un affichage de données, on est certainement en présence d'une boucle

Exemple 1

- **Colonne d'articles sur le site de Radio-Canada**
 - Chaque carte possède les mêmes caractéristiques, elles sont donc probablement générées par une boucle



POLLUTION

Ottawa ajoute les produits de plastique à une liste de substances toxiques

Les articles manufacturés de plastique sont considérés comme pouvant avoir un effet nocif sur l'environnement.

CORONAVIRUS : LA SITUATION EN ONTARIO

La vérificatrice générale de l'Ontario déplore un manque de suivi des programmes sur ...

POLITIQUE FÉDÉRALE

C-10, censure et culture : deux solitudes?

TRANSPORTS

Le prolongement de l'autoroute 20 sera réintégré au Plan québécois des infrastructures

Exemple 2

- Affichage des titres d'une liste de lecture sur Spotify:
 - Tant qu'il y a des titres à afficher, afficher le prochain titre

The screenshot shows a Spotify Premium interface with a dark theme. At the top, it says "Spotify Premium". Below that is a navigation bar with arrows and a play button, followed by the title "Titres likés". To the right is a user profile for "Pier-Luc Brault". The main area is a table listing seven songs:

#	TITRE	ALBUM	DURÉE
3855	System Of A Down	Hypnotize	2:00
3856	U-Fig	Hypnotize	2:55
3857	Uku	Fakek' choz	1:51
3858	ULTIMATE	Super Taranta	3:29
3859	Uma Menina	Trans-Continental Hu...	4:34
3860	Umami	Notre-Dame-des-Sep...	4:28
3861	Unacceptable	Against The Grain	1:45

Itération

- On appelle « **itération** » chaque exécution d'une boucle
 - Du verbe « **itérer** »

La boucle *While*

La boucle *While*

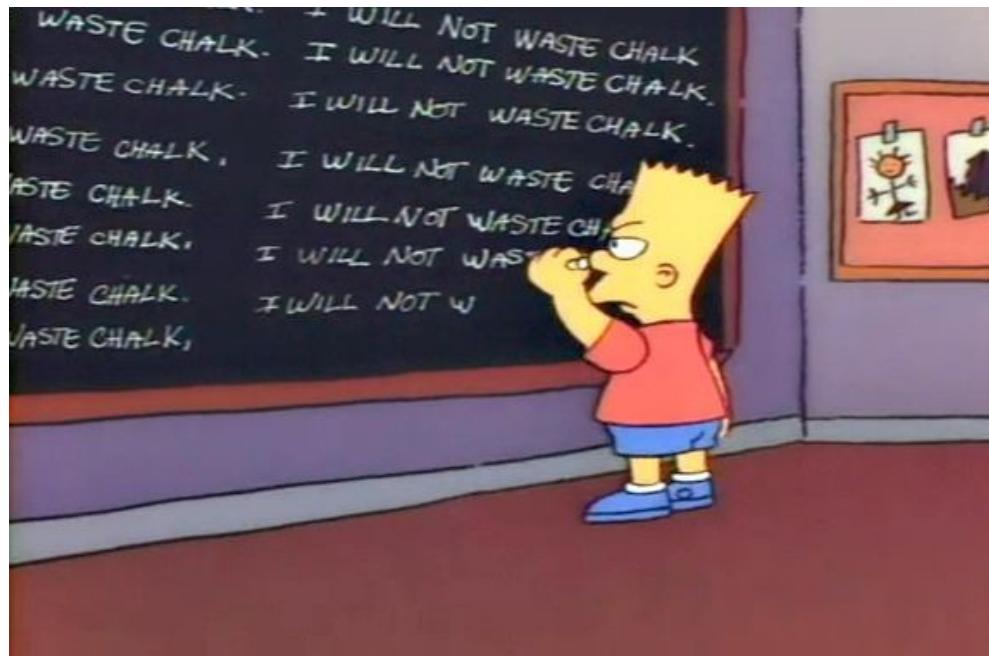
- La boucle **While** (ou **Tant que**) peut être vue comme un **if** qui se répète **tant que la condition est vraie**
- **Syntaxe:**

```
while (expression booléenne) {
```

Instructions à exécuter si l'expression est **vraie**

```
}
```

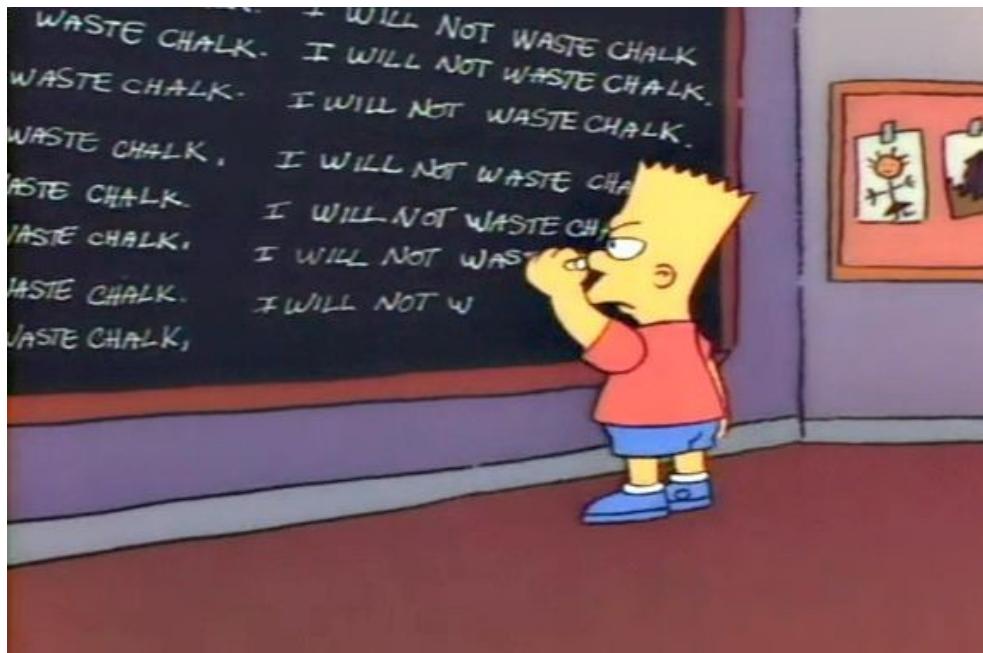
Exemple 1



```
// Affiche 20 fois « I  
will not waste chalk. »  
$compteur = 0;  
while ($compteur < 20) {  
    Echo "<p>I will not  
waste chalk.</p>";  
    $compteur++;  
}
```

Boucle infinie

Attention aux **boucles infinies!**



```
// Affiche 20 fois « I  
will not waste chalk. »  
$compteur = 0;  
while ($compteur < 20) {  
    Echo "<p>I will not  
waste chalk.</p>";  
    // $compteur++;  
}
```

Exemple 2

```
// Simuler "$a % $b" avec une boucle while
$modulo = $a;
while ($modulo >= $b) {
    $modulo -= $b;
}
echo "Le résultat de $a modulo $b est: $modulo";
```

- Exemple avec \$a = 10 et \$b = 3:
 - \$modulo = 10
 - Après la première itération: \$modulo = 7
 - Après la deuxième itération: \$modulo = 4
 - Après la troisième itération: \$modulo = 1
 - $1 < 3$, donc le programme sort de la boucle
 - Affiche "Le résultat de 10 modulo 3 est: 1"

La boucle *Do...While*

La boucle *Do...While*

- La boucle **Do...While** (ou **Faire...Tant que**) est semblable à la boucle While, mais la première itération se fait avant toute vérification de la condition
- **Syntaxe:**

```
do {
```

Instructions à exécuter une première fois, puis à répéter si l'expression est **vraie**

```
} while (expression booléenne);
```

Exemple

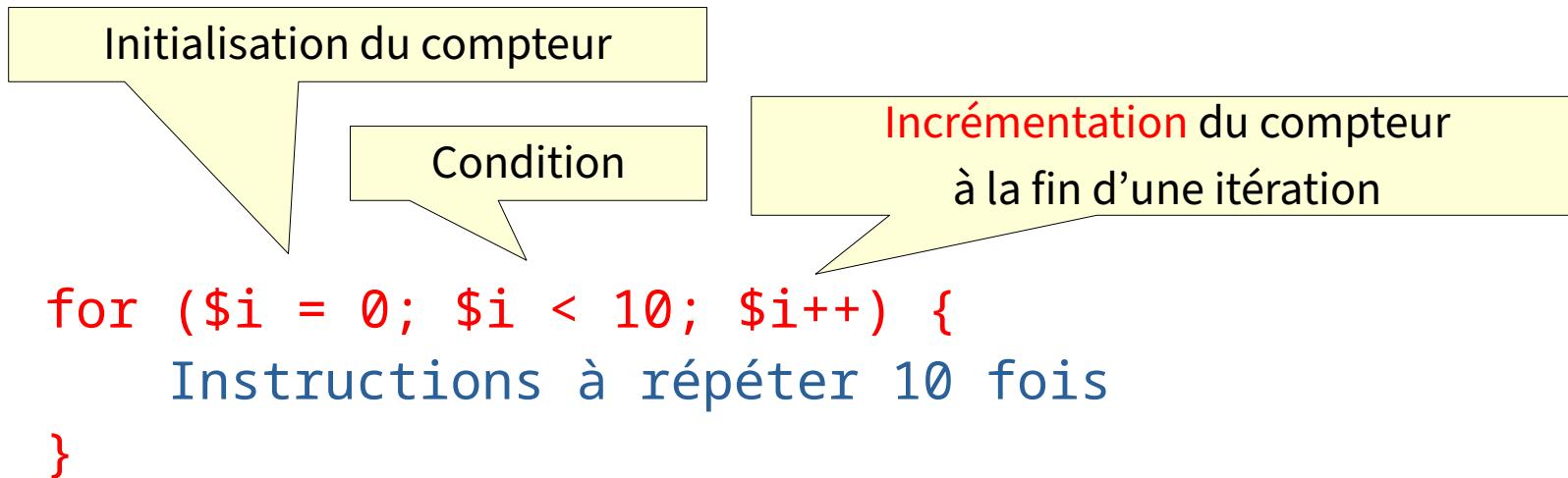


```
// Affiche 20 fois « I  
will not waste chalk. »  
$compteur = 0;  
do {  
    echo "<p>I will not  
waste chalk.</p>";  
    $compteur++;  
} while ($compteur < 20);
```

La boucle *For*

La boucle *For*

- La boucle **For** offre une syntaxe abrégée pour les boucles qui utilisent un compteur
- Exemple: Répéter les mêmes instructions 10 fois



Exemple 1



```
// Affiche 20 fois « I will not  
waste chalk. »
```

```
for ($i = 0; $i < 20; $i++) {  
    echo "<p>I will not waste  
    chalk.</p>";  
}
```

Exemple 2

```
// Affiche les nombres de 1 à 100
for ($i = 1; $i <= 100; $i++) {
    echo "$i <br />";
}
```

Exemple 3

```
// Affiche les nombres de 1 à 100 en ordre décroissant
for ($i = 100; $i > 0; $i--) {
    echo "$i <br />";
}
```

Fin de la présentation

Des questions?

