Laravel Blade les structures de contrôle

1. Objectifs

- Connaitre les structures de contrôle
- Etre capable de travailler avec les structures de contrôle

2. Présentation

- Les boucles sont l'un des composants les plus importants de PHP et de nombreux autres langages de programmation . Ils vous aident à effectuer des tâches répétitives assez facilement sans écrire de morceau de code supplémentaire.
- Les modèles de Blade Laravel permettent de travailler particulièrement facilement avec ce type même de structure de contrôle en PHP. De même, les boucles de Laravel fonctionnent de la même manière que celles trouvées en PHP.
- Il existe en fait quatre principaux types de **boucles** dans Laravel qui sont la boucle `for`, la boucle `foreach`, la boucle `forelse` et la boucle `while`.

1. La boucle for

- **Syntaxe**

```
@for ($i = 0; $i < 10; $i++)
    La valeur actuelle est {{ $i }}
@endfor
```

2. La boucle foreach

- Le Blade Laravel a une directive **foreach** que nous pouvons utiliser de la même manière que nous utilisons la boucle foreach en PHP.
- La directive `@foreach` est plus puissante qu'une boucle **foreach** normale en raison de la variable $loop disponible dans chaque boucle **$foreach**.

- **Syntaxe**

apcpedagogie.com

```
@foreach ($users as $user)

This is user {{ $user->id }}

@endforeach
```

3. **Sauter l'itération en cours**

   - Il est également possible de sauter l'itération en cours ou simplement de terminer la boucle lorsque vous faites une boucle dans Laravel.

```
@foreach ($users as $user)
    @if ($user->type == 1)
        @continue
    @endif

    <li>{{ $user->name }}</li>

    @if ($user->number == 5)
        @break
    @endif
@endforeach
```

4. **$loop dans @foreach**

   - La variable `$loop` est un objet **stdClass** et elle donne accès à des méta-informations utiles sur votre boucle actuelle.

   - **Syntaxe**

```
@foreach ($users as $user)
    @if ($loop->first)
        Ceci est la première itération.
    @endif

    @if ($loop->last)
        Ceci est la dernière itération.
    @endif
```

apcpedagogie.com

```
    <p>C'est user {{ $user->id }}</p>
@endforeach
```

5. La variable `$loop` expose huit propriétés utiles.
    1. `$loop->index`: Renvoie une itération de boucle actuelle basée sur 0; 0 signifierait la première itération
    2. `$loop->iteration`: Renvoie une itération de boucle actuelle basée sur 1; 1 signifierait la première itération
    3. `$loop->remaining`: Nombre d'itérations restantes dans la boucle; s'il y a un total de 10 itérations et que l'itération actuelle est de 3, il renvoie 7
    4. `$loop->count`: Renvoie le nombre total d'itérations ou le nombre total d'éléments dans le tableau
    5. `$loop->first`: Renvoie **true** s'il s'agit de la première itération ou élément de la boucle, sinon renvoie false.
    6. `$loop->last`: Renvoie **true** s'il s'agit de la dernière itération ou élément de la boucle, sinon renvoie false.
    7. `$loop->depth`: Renvoie la profondeur ou le niveau d'imbrication de la boucle actuelle; renvoie 2 s'il s'agit d'une boucle dans une boucle et 3 s'il est imbriqué d'un niveau de plus
    8. `$loop->parent`: Si cette boucle est imbriquée dans une autre boucle `@foreach`, **parent** renvoie la variable de boucle du parent; S'il n'est pas testé, renvoie **null**

6. La boucle forelse

- **Syntaxe**

```
@forelse ($users as $user)
    <li>{{ $user->name }}</li>
@empty
    <p>Aucun utilisateur</p>
@endforelse
```

7. La boucle while

- **Syntaxe**

apcpedagogie.com

```
@while (true)
    <p>I'm looping forever.</p>
@endwhile
```

Sommaire du cours Laravel

Source : https://apcpedagogie.com/laravel-blade-les-structures-de-controle · téléchargé le 08/11/2026 · © APC Pédagogie · apcpedagogie.com

apcpedagogie.com