

Leçon 17

Les boucles for



Objectif

Dans cette leçon, nous allons explorer un concept essentiel en programmation : les boucles. Imaginez que vous avez une liste de tâches à accomplir ou une collection d'éléments à parcourir. Au lieu de répéter le même code encore et encore, vous pouvez utiliser une boucle pour simplifier les choses.

En Kotlin, les boucles `for` et `forEach` sont couramment utilisées pour parcourir des collections comme des listes, des tableaux, ou des maps. Voyons comment cela fonctionne.

La Boucle for

La boucle `for` permet de parcourir une collection d'éléments, un par un. C'est l'une des structures les plus simples et les plus utiles en programmation.

Exemple : Parcourir une Liste d'Animaux

```
val animaux = listOf("Lion", "Tigre", "Ours", "Éléphant")
var resultats = ""

for (animal in animaux) {
    resultats += "Animal rencontré : $animal\n"
}

setContent {
    Text(resultats)
}
```

- `for (animal in animaux)`: Pour chaque élément dans la liste animaux, la variable animal prend successivement la valeur de chaque élément.
- `resultats += "Animal rencontré : $animal\n"` : On ajoute chaque animal rencontré à la chaîne resultats avec un saut de ligne (`\n`) pour chaque nouvel élément.
- `setContent { Text(resultats) }` : Enfin, on affiche tous les résultats en une seule fois.

La méthode `forEach`

La méthode `forEach` est une autre façon de parcourir une collection. Elle fonctionne de manière similaire à la boucle `for`, mais est parfois plus concise.

Exemple : Utilisation de `forEach`

```
val animaux = listOf("Lion", "Tigre", "Ours", "Éléphant")
var resultats = ""

animaux.forEach { animal ->
    resultats += "Animal trouvé : $animal\n"
}

setContent {
    Text(resultats)
}
```

- `animaux.forEach { animal -> ... }`: `forEach` parcourt chaque élément de la liste `animaux`, et pour chaque élément, le code à l'intérieur des accolades est exécuté.
- `resultats += "Animal trouvé : $animal\n"` : On accumule les résultats comme dans l'exemple précédent.
- `setContent { Text(resultats) }` : On affiche ensuite le tout.

Exercices

Exercice 1

1. Créez une variable `val nombre = 5`.
2. Utilisez une boucle `for` pour afficher les 10 premiers multiples de `nombre`.
3. Affichez le résultat sous forme de chaîne de caractères : `"5 x 1 = 5"`, `"5 x 2 = 10"`, etc.

Exercice 2

- Créez une variable `val mot = "Kotlin"` et une variable pour stocker le nombre de voyelles `var nbVoyelles = 0`.
- Parcourez chaque caractère du mot avec une boucle `for` et comptez le nombre de voyelles (a, e, i, o, u).
- Affichez le nombre total de voyelles trouvé dans le mot.

Exercice 3

1. Créez un tableau nombres avec 1, 2, 3, 4, 5.
2. Utilisez une boucle for pour calculer la somme de tous les éléments du tableau.
3. Affichez le résultat : "La somme des éléments est 15".

Exercice 4

1. Créez une liste étudiants avec "Alice", "Bob", "Charlie".
2. Utilisez forEach pour afficher chaque nom en majuscule.
3. Affichez chaque nom avec un message comme : "ALICE est dans la classe".

Exercice 5

1. Créez une liste notes avec 15.5, 18.0, 12.0, 14.5
2. Utilisez une boucle for pour calculer la moyenne des notes.
3. Affichez le résultat : "La moyenne de la classe est de 15.0".

Exercice 6

1. Créez une map produits avec :
 - "Pomme" -> 1.5
 - "Banane" -> 0.8
 - "Orange" -> 1.2
1. Utilisez une boucle for pour afficher chaque produit avec son prix, par exemple : "Pomme coûte 1.5€".
2. Affichez toutes les lignes dans une seule chaîne.

Exercice 7

1. Créez une liste nombres avec 3, 8, 1, 9, 2.
2. Utilisez une boucle for pour trouver le plus grand nombre de la liste.
3. Affichez le résultat : "Le plus grand nombre est 9".

Exercice 8

1. Créez un tableau nombres avec 2, 5, 8, 11, 14.
2. Utilisez une boucle for pour afficher uniquement les nombres pairs.
3. Affichez les nombres pairs dans une chaîne, par exemple : "Les nombres pairs sont : 2, 8, 14".

Exercice 9

1. Créez une map panier avec :
 - "Livre" -> 12.5
 - "Stylo" -> 1.5
 - "Cahier" -> 3.0.
2. Utilisez une boucle for pour calculer le total des prix des articles dans le panier.
3. Affichez le total : "Le total du panier est de 17.0€".

Exercice 10

Contexte :

Vous organisez un festival de musique et devez gérer les genres musicaux, les artistes qui y participent, et leurs horaires de passage. Utilisez des listes pour les genres, des maps pour les artistes et leurs horaires, et manipulez les données à l'aide de boucles.

Étapes :

1. Création des genres musicaux :
 - Créez une liste mutable contenant les genres musicaux suivants : "Rock", "Jazz", "Hip-Hop".
2. Ajout des artistes :
 - Créez une map mutable qui associe chaque genre à une liste d'artistes.
 - Pour "Rock", ajoutez : "The Rolling Stones", "Led Zeppelin".
 - Pour "Jazz", ajoutez : "Miles Davis", "John Coltrane".
 - Pour "Hip-Hop", ajoutez : "Kendrick Lamar", "J. Cole".
3. Gestion des horaires de passage :
 - Créez une autre map mutable associant chaque artiste à son horaire de passage (en heures).
 - "The Rolling Stones" passent à 20h.
 - "Led Zeppelin" passent à 22h.
 - "Miles Davis" passe à 19h.
 - "John Coltrane" passe à 21h.
 - "Kendrick Lamar" passe à 18h.
 - "J. Cole" passe à 23h.
4. Afficher le programme complet :
 - Utilisez une boucle for pour parcourir chaque genre musical.
 - Pour chaque genre, affichez le genre et les artistes associés avec leurs horaires de passage.
 - Créez une seule chaîne de caractères qui contient toutes les informations à afficher précédé du titre : "Programme du Festival:\n"

5. Vérification des horaires :

- Vérifiez si un artiste passe après 21h.
- Utilisez une boucle for pour parcourir tous les artistes et vérifiez leur horaire.
- Ajoutez à votre chaîne de caractères un message pour chaque artiste qui passe après 21h précédé du titre : "Performances en soirée (après 21h) :"

6. Ajout d'un nouveau groupe :

- Un nouveau groupe "The Weeknd" (genre "Hip-Hop") passe à 17h.
- Ajoutez ce groupe à la map des artistes pour le genre "Hip-Hop" et à la map des horaires.

7. Réaffichage du programme mis à jour :

- Ajoutez le nouveau programme pour montrer les changements, précédé du titre : "Programme mis à jour du Festival :"