

Leçon 9

var ou val



Objectif

Nous allons explorer comment choisir entre var et val pour vos variables en Kotlin. C'est comme choisir entre une boîte avec un couvercle amovible et une boîte scellée. Vous apprendrez comment et quand utiliser chaque type pour rendre votre code aussi clair que possible.

Qu'est-ce que var et val ?

var (Variable Mutable) :

- Imaginez une boîte dans laquelle vous pouvez ajouter ou retirer des objets quand vous le souhaitez. var fonctionne comme cette boîte : vous pouvez changer ce qu'elle contient à tout moment.
- Exemple : Si vous avez un compteur pour compter les points dans un jeu, vous aurez besoin de var car le nombre de points change constamment.
- Signification : **var** vient de "variable", ce qui signifie que la valeur peut varier, c'est-à-dire changer.

val (Variable Immutable) :

- Maintenant, pensez à une boîte scellée avec un objet à l'intérieur. Une fois scellée, vous ne pouvez pas changer son contenu. val fonctionne comme cette boîte scellée : la valeur reste fixe une fois définie.
- Exemple : Si vous stockez la date de naissance d'un utilisateur, vous utilisez val car la date ne change pas.
- Signification : **val** vient de "value" (valeur en anglais), ce qui signifie que la valeur est constante et ne peut pas être modifiée après sa définition.

Quand utiliser var et quand utiliser val ?

Utilisez **var** quand la valeur doit pouvoir changer.

- Exemple : Si vous suivez le nombre de vies dans un jeu vidéo, utilisez **var** car ce nombre change au fur et à mesure que vous perdez ou gagnez des vies.

Utilisez **val** quand la valeur doit rester constante.

- Exemple : Utilisez **val** pour la valeur de PI dans les calculs mathématiques, car PI ne change jamais.

Manipulation des Variables avec var et val

Voyons quelques exemples concrets pour utiliser **var** et **val** dans des situations amusantes.

Déclaration et Utilisation de **var** :

```
var score = 0  
score = score + 10
```

Vous pouvez modifier le score du jeu, donc on utilise **var**.

Déclaration et Utilisation de **val** :

```
val maximumScore = 100
```

Si nous essayons de modifier la valeur par la suite, ceci provoquerait une erreur car maximumScore est immuable donc non modifiable.

Exercices

Exercice 1

- Créez une variable nombreVisites avec var et initialisez-la à 0.
- Imaginez que chaque fois que quelqu'un visite votre site, le nombre de visites augmente. Simulez cela en ajoutant 1 à nombreVisites.
- Affichez le message : "Vous avez maintenant [nombreVisites] visites !"

Exercice 2

- Créez une variable prixMagie avec val et initialisez-la à 19.99f.
- Imaginez que vous achetez un sort magique et que le prix est fixe. Affichez le prix magique avec le message : "Le prix du sort magique est de [prixMagie] euros."

Exercice 3

- Créez une variable trésorInitial avec val et initialisez-la à 1000.
- Créez une variable trésorTrouvé avec var et initialisez-la à 250.
- Simulez la découverte d'un nouveau trésor en ajoutant 500 à trésorTrouvé.
- Affichez les trésors avec le message : "Le trésor initial était de [trésorInitial] pièces, et maintenant vous avez [trésorTrouvé] pièces."

Exercice 4

- Créez une variable points avec var et initialisez-la à 30.
- Ajoutez 20 points à points pour simuler une victoire dans un jeu.
- Affichez les points actuels avec le message : "Vous avez [points] points après votre victoire !"

Exercice 5

- Créez une variable facteurMagie avec val et initialisez-la à 1.5.
- Créez une variable forceSort avec var et initialisez-la à 10.
- Multipliez forceSort par facteurMagie pour obtenir la puissance magique.
- Affichez la puissance magique avec le message : "La puissance magique est maintenant [forceSort] grâce au facteur de magie [facteurMagie]."