





**MODÉLER UNE POTENTIELLE**

EWAWAW

MAGNET 2-DUALS - THOUGHTS

N'existe pas dans la lib standard

Peut être encodé avec les génériques (vous avez vu comment) ... c'est pas compliqué mais il manque quelques fonctions pour être réellement utilisable (pure, map, flatmap, fold)

Si vous voulez faire du Java « PRO » en 2022 utilisez au moins VAVR

[https://www.vavr.io/vavr-docs/#\\_either](https://www.vavr.io/vavr-docs/#_either) ...

Ou attendez Java18 pour faire une lib propre sans hacks (il y a du boulot!)

Ou passez à Scala ou Kotlin + Arrow-Kt

ou abandonnez la JVM



# MODÉLISER UNE ERREUR POTENTIELLE

EN JAVA 17

N'existe pas dans la lib standard

Peut être encodé avec les génériques (vous avez vu comment) ... c'est pas compliqué mais il manque quelques fonctions pour être réellement utilisable (pure, map, flatmap, fold)

Si vous voulez faire du Java « PRO » en 2022 utilisez au moins VAVR

[https://www.vavr.io/vavr-docs/#\\_either](https://www.vavr.io/vavr-docs/#_either) ...

Ou attendez Java18 pour faire une lib propre sans hacks (il y a du boulot!)

Ou passez à Scala ou Kotlin + Arrow-Kt

Ou abandonnez la JVM 🐈



# TAKE AWAY

## LES RESULTS

A utiliser pour modéliser les cas d'erreur

Sécurisant, surtout quand on dispose d'ADT

Facile à manipuler avec une syntaxe spécifique

Parfois encodé avec un seul paramètre

```
type 'a result =  
  | Ok of 'a  
  | Error of exn
```

Dans certains langages Result est remplacé par Either (Scala, Haskell), c'est un type plus générique qui par « convention » représente les erreurs dans la valeur Left

```
type ('a, 'b) either =  
  | Left of 'a  
  | Right of 'b
```

