

DATA CLASS

&

ITERATORS

ROUSVAL
Benjamin

Blanc
Arthur

INTRODUCTION



01

ÉMINCÉ DE RECHERCHES

420g de recherches émincées

02

COMPRÉHENSION DE LA RECETTE

Mélanger ce qui se trouve
dans le cerveau

03

CUISSON EN COMMUN

69g de réflexion et de sel

04

DRESSAGE DES SLIDES

Choisir un bon thème et des
bonnes idées pour une
composition parfaite

05

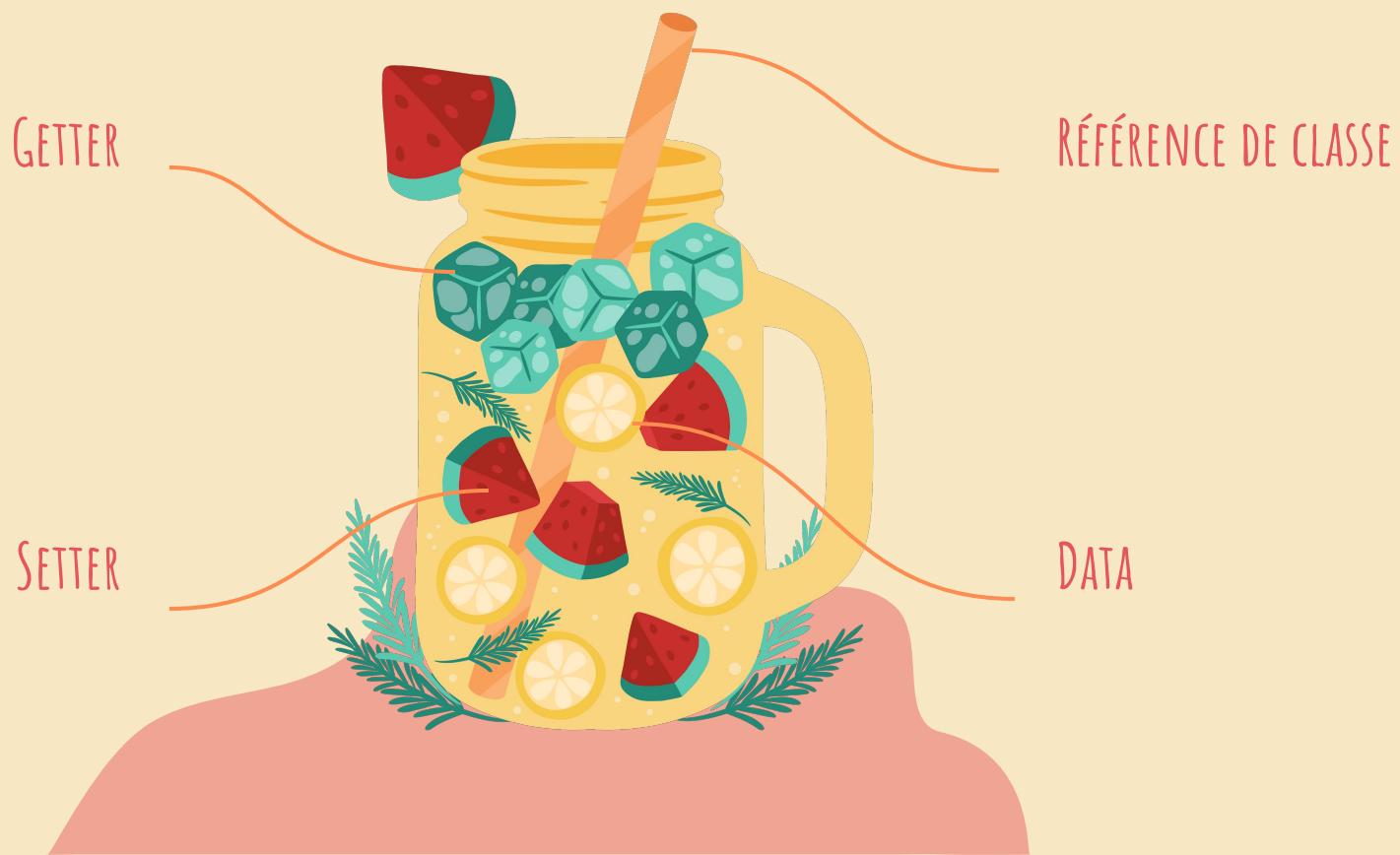
TADOUM

Le début du bonheur, de la
dégustation et de la
présentation au talent



CODE SMELL : DATA CLASS

DATA CLASS



BONNES PRATIQUES

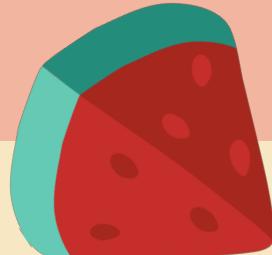
ENCAPSULATION



Champs publics



Encapsuler les champs



GETTER / SETTER



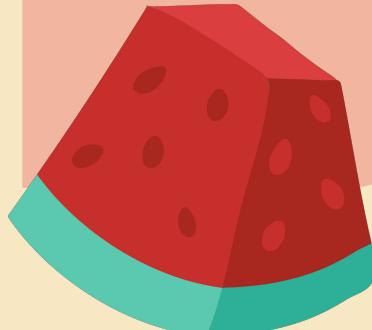
class.name



class.setName()



class.getName()



MANGER DES FRUITS



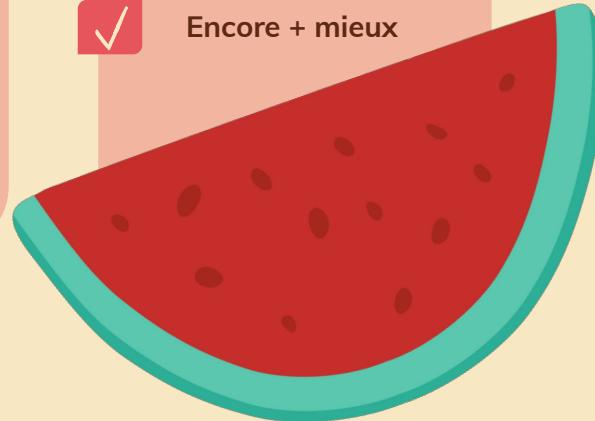
Mieux



Encore mieux



Encore + mieux





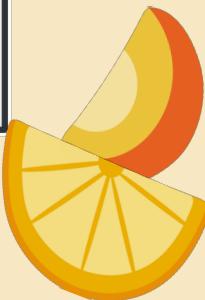
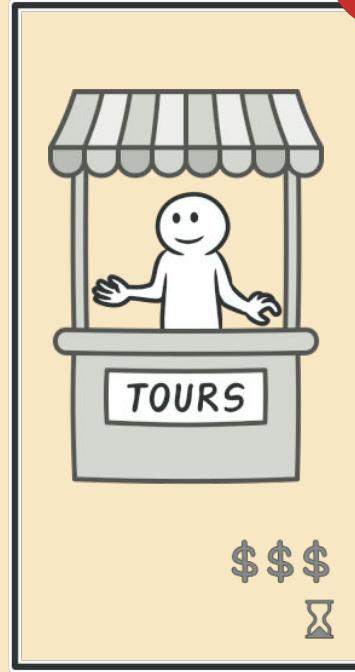
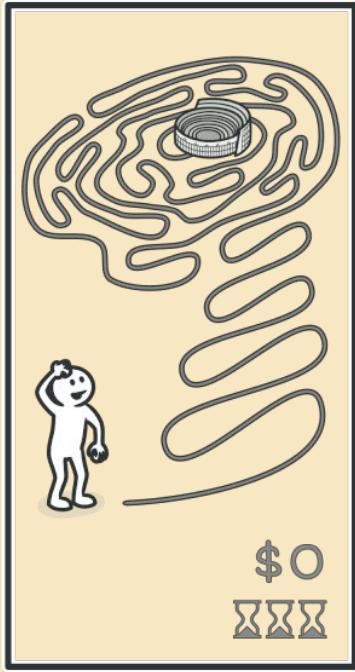
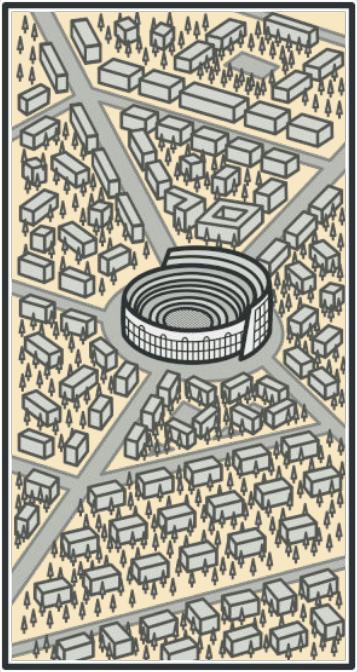
DESIGN PATTERN : ITERATOR

QU'EST CE QU'UN ITÉRATEUR ?

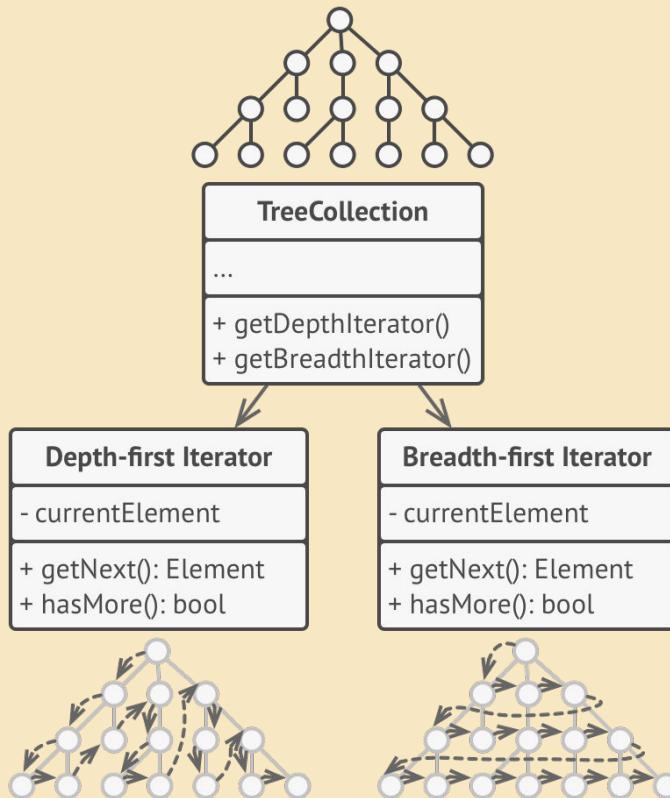
L'itérateur est un design pattern comportemental.
Il permet de parcourir une collection sans exposer
son arborescence sous-adjacente.



SON INTENTION



SON INTENTION



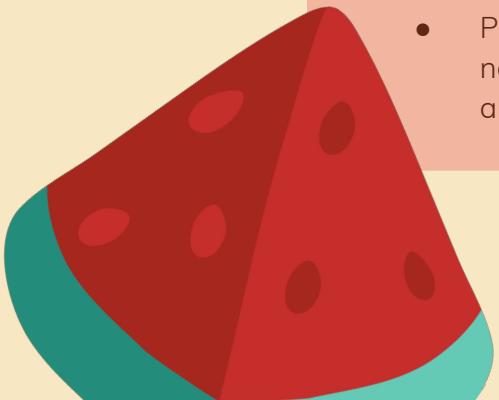
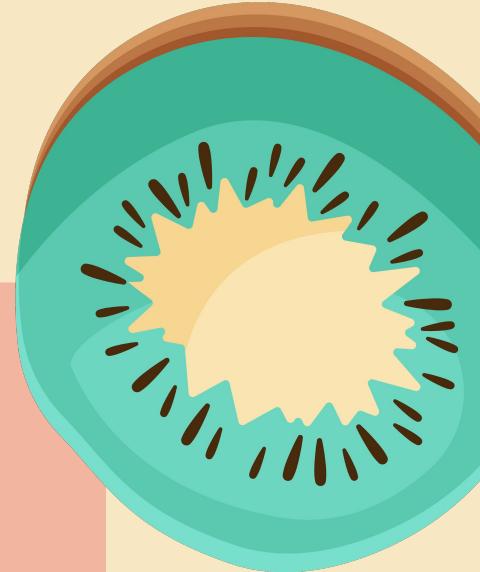
CONSÉQUENCES

BONNES

- Principe de responsabilité unique
- Nettoie le code source
- Évite la duplication
- Permet l'implémentation de nouvelles itérations sans altérer le code source

MAUVAISES

- Pattern Overkill sur des collections simples
- Temps de mise en place parfois longs



EXEMPLE DE CODE

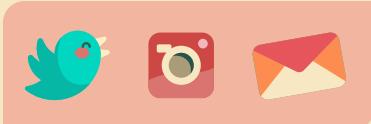


```
while (iterator.hasMore())
profile = iterator.getNext()
System.sendEmail(profile.getEmail(), message)
```



MERCI

Avez-vous des questions ?



CREDITS

Liens :

- <https://refactoring.guru/>
- Presentation template by Slidesgo

