

Introduction

C'est quoi un objet ?

On prend l'exemple des joueurs sur un terrain.

Comment gérer les différents joueurs ?

Information du joueur : image, pseudo, vie, sa position

Action du joueur: se déplacer, tirer, être touché, mourir...

Solution1(mauvaise) : programmation historique,
créer des variables et modules séparés.

Solution2(bonne) : programmation objet. Regrouper les
variables et modules qui concernent le joueur dans un
même nouveau "type" qui se nommera Joueur.

Ce type joueur en fait va être une classe. Le principe de la

classe Joueur :

propriétés :

String nom
int personnage
int vie
...

méthodes :

déplacer(int sens)
tirer()
mourir()
...

classe Liste :

propriétés :

String[] element
int sélectionné
...

méthodes :

add(String ligne)
del(int index)
clear()
...

classe Curseur :

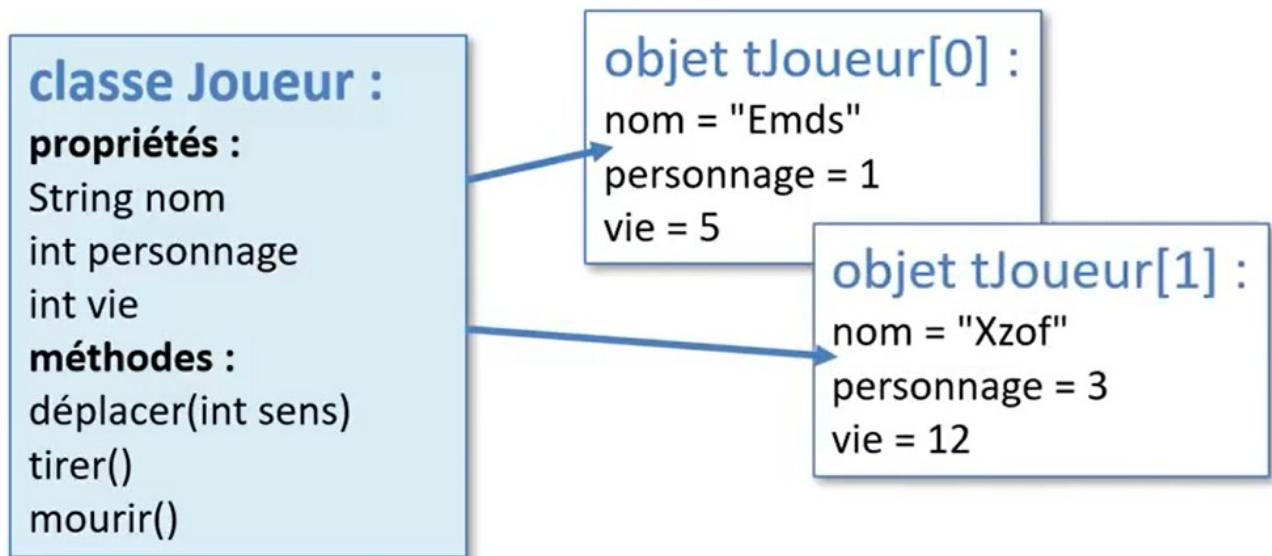
méthodes :

init(String sql)
suivant()
...

classe va être de regrouper tout ce qui concerne le joueur,
propriétés et méthodes appelées membres de la classe dans
une même classe. Du coup il s'en dégage une certaine
cohérence, une certaine logique.

Une classe est un type complexe.

Un objet est une variable dont le type est une classe.

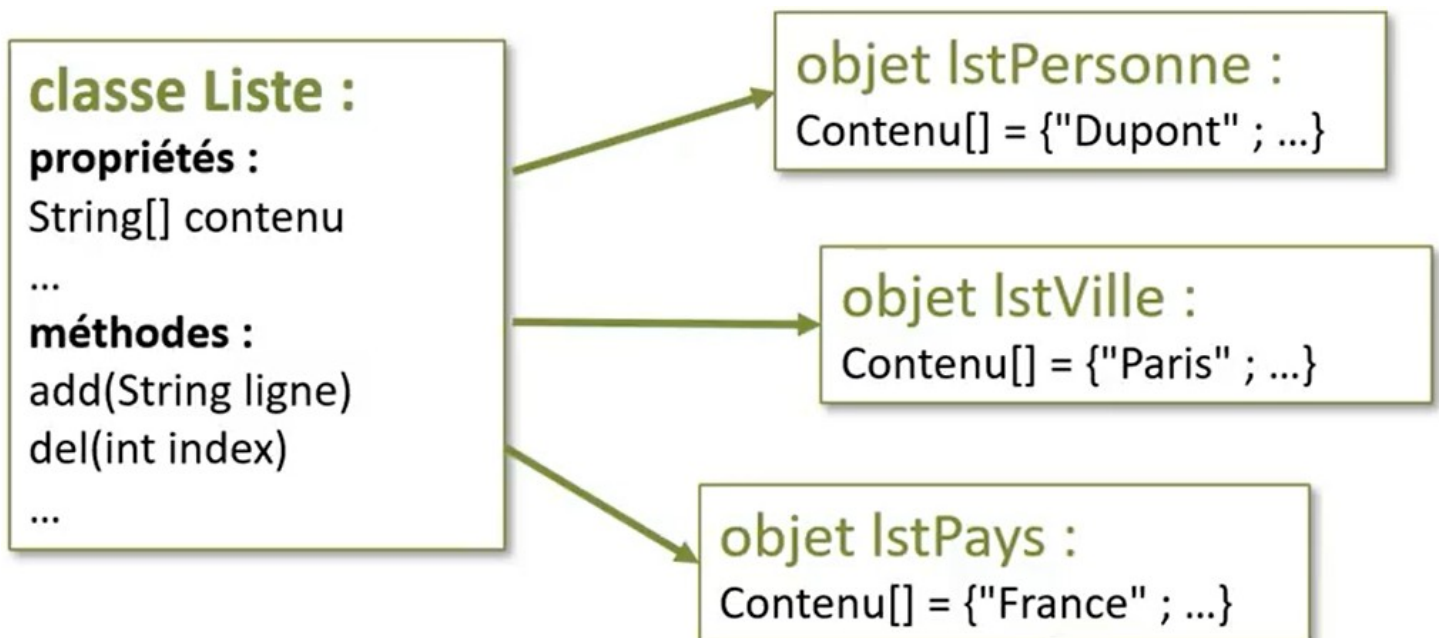


Créer un objet = instancier une classe.

Un objet est une instance de classe.

Pour utiliser un objet, il faut d'abord le déclarer puis le créer (certains langages permettent d'écrire les deux étapes sur une seule ligne).

Mise en pratique:



A partir de la classe Liste et des 3 objets présentés ci-dessus, écrire les instructions qui permettent de répondre aux demandes suivantes :

a- Ajouter "Bernard" à la liste des personnes.

LstPersonne.add("Bernard")

b- Ajouter Nice à la liste des villes.

LstVille.add("Nice")

c- Supprimer le premier élément (d'indice 0) de la liste des pays.

LstPays.del(0)

Utilisation :

classe Joueur :

propriétés :

String nom

int personnage

Int vie

méthodes :

déplacer(int sens)

tirer()

mourir()

```
...  
int k;  
Joueur[] tJoueur;  
...  
tJoueur[0] = new Joueur();  
...  
tJoueur[0].nom = "Emds";  
...  
tJoueur[0].tirer();  
...
```

Déclaration d'un
tableau d'objets

Création de l'objet

Utilisation
d'une
propriété

Utilisation d'une
méthode



Emds : 5



Xzof : 12

Un objet se déclare comme variable classique. Mais attention, il faut ensuite créer l'objet pour pouvoir l'utiliser.

Exercice d'application:

Dans un jeu, une classe Personnage a été créée. Voici en vrac certaines membres de cette classe. Pour chaque

membre vous préciserez si c'est une propriété ou une méthode.

String pseudo // propriété

déplace(int direction) // méthode

Boolean estMort() // méthode

int vie // propriété

Exercice 2

Déclarer et créer un objet de type Personnage et faites sur cet objet, les opérations suivantes :

- initialiser la vie à 10 et le pseudo à "Xzof"
- Le faire déplacer vers la droite (direction 2)
- Mettre sa vie à 0 s'il est mort.

Solution:

```
Personnage unPersonnage ;
```

```
unPersonnage = new Personnage() ;
```

```
unPersonnage.vie = 10;
```

```
unPersonnage.pseudo = "Xzof";
```

```
unPersonnage.deplace(2);
```

```
if(unPersonnage.estMort()) {
```

```
unPersonnage.vie = 0;
```

```
}
```