Informatique

Introduction aux concepts orientés objets

Patrick Sentinel - 2025

Introduction aux concepts objets

Source: https://www.roberthalf.com/blog/salaries-and-skills/4-advantages-of-object-orien ed

Présentation rapides des termes

Présentation rapide des termes et ensuite effectuez une lecture du document

Design de classes

Ce que nous allons parcourir pour le cours d'aujourd'hui

- Création de classes et d'objets en C#
- Constructeurs
- Encapsulation et protection des données
- Méthodes de classes et statiques
- Le principe des relations entre les classes

Utilisation des classes et relations

Dans les postes pour lesquels vous allez travailler, vous ne devriez pas effectuer de la conception. Cependant, il faut comprendre le fonctionnement pour pouvoir les utiliser correctement. Donc, nous allons survoler le design de classe mais l'objectif n'est pas de vous faire des experts de design de classes.

Fonctionnement du marché et la POO

Fonctionnement du marché du travail versus la théorie, ce qu'il faut savoir:

- ***** Apprendre à se débrouiller *****
- ***** Google is your best friend *****
- Comprendre les concepts généraux
- La qualité prime sur la rapidité
- Prendre le temps de penser avant d'exécuter

Où est la complexité des projets ?

La complexité des projets n’est pas dans les langages de programmation ou dans les algorithmes. La complexité des projets dépend en général de deux aspects:

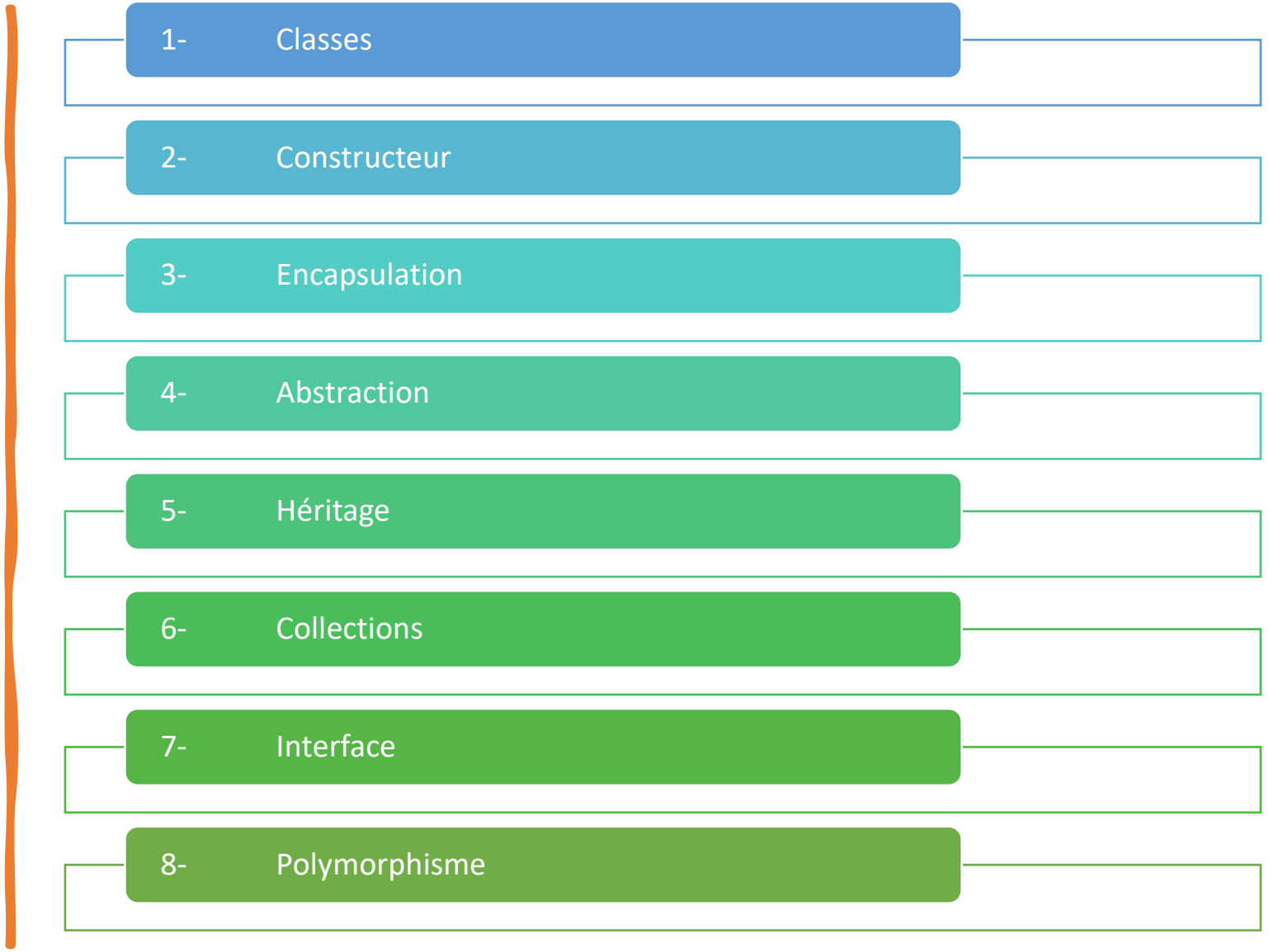
1- Le temps

Les projets doivent parfois être réalisés rapidement

2- Les *frameworks*

Il y a beaucoup de *frameworks* et il faut apprendre à se débrouiller avec eux.

Huit grands concepts de la POO à maitriser



Les classes

Pour cette session de travail nous allons aborder les classes

Les classes

Qu’est-ce qu’une classe ?

Une classe n’est rien de plus qu’un fichier de type *.cs* en C#.

À quoi sert la classe ?

Elle sert de *blue print*

La classe sert de fondation à la construction

Exemple de classes

```
class MyFirstClass
{
    0 references
    static void Main(string[] args)
    {
        Console.WriteLine("Hello World!");
    }
}
```

Le mot réservé *class* indique que le fichier est de type *class* et le nom MyFirstClass est le nom de la classe.

Membres de classes

Pour les besoins de notre formation, les classes ont trois types membres:

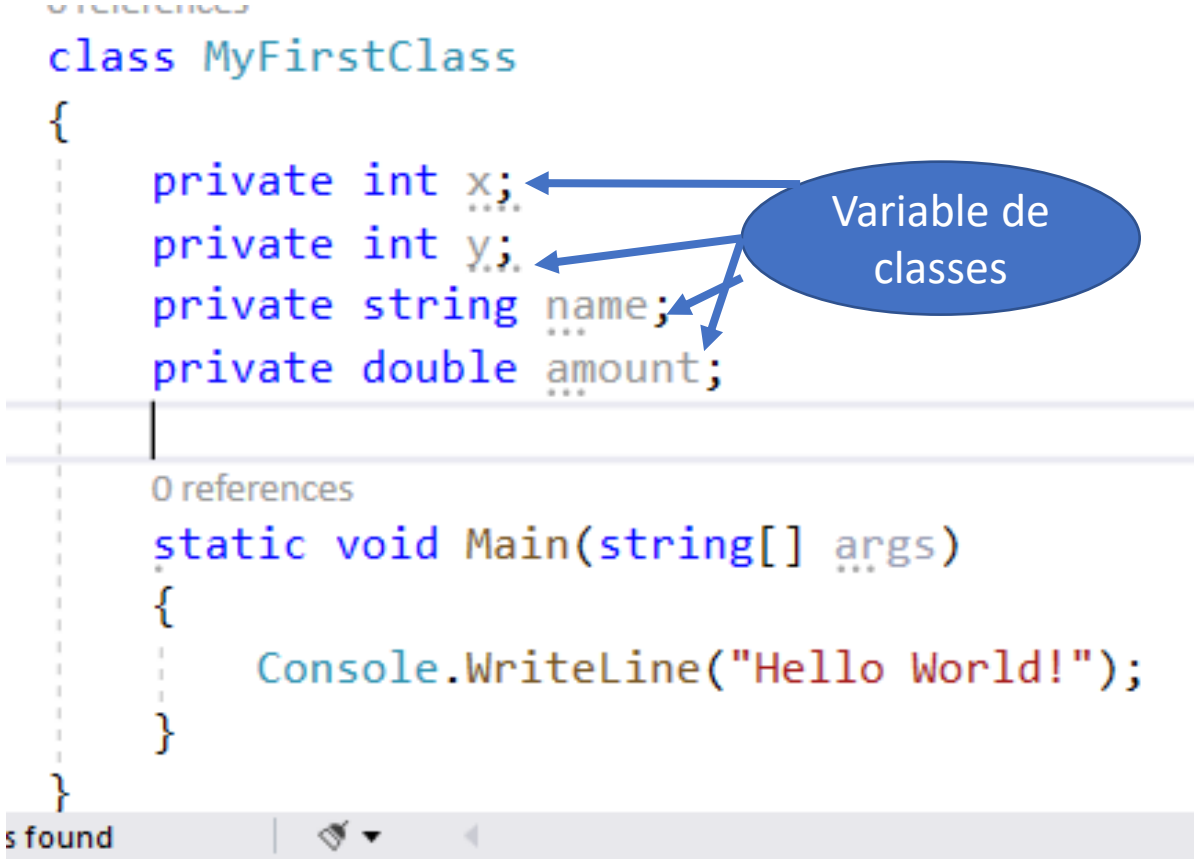
1- Les variables de classes ;

2- Les propriétés ;

2- Les fonctions ;

Les variables de classes

Les variables de classes sont les variables qui appartiennent à la classe. Pour le moment, utilisons uniquement des variables *private*.

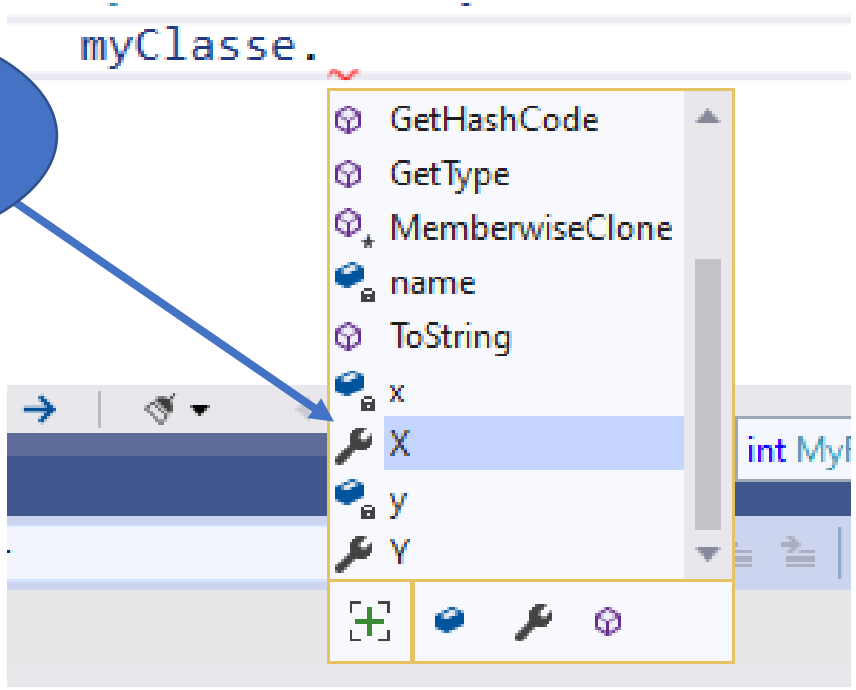


Les propriétés des classes

Les propriétés des classes permettent d'avoir accès aux membres de la classes. Les propriétés servent uniquement (il y a des exceptions) à mettre une valeur dans la variable et à obtenir la valeur de la variable. On ne doit **JAMAIS** accéder à une variable en dehors des propriétés si l'on est à l'extérieurs de la classe.

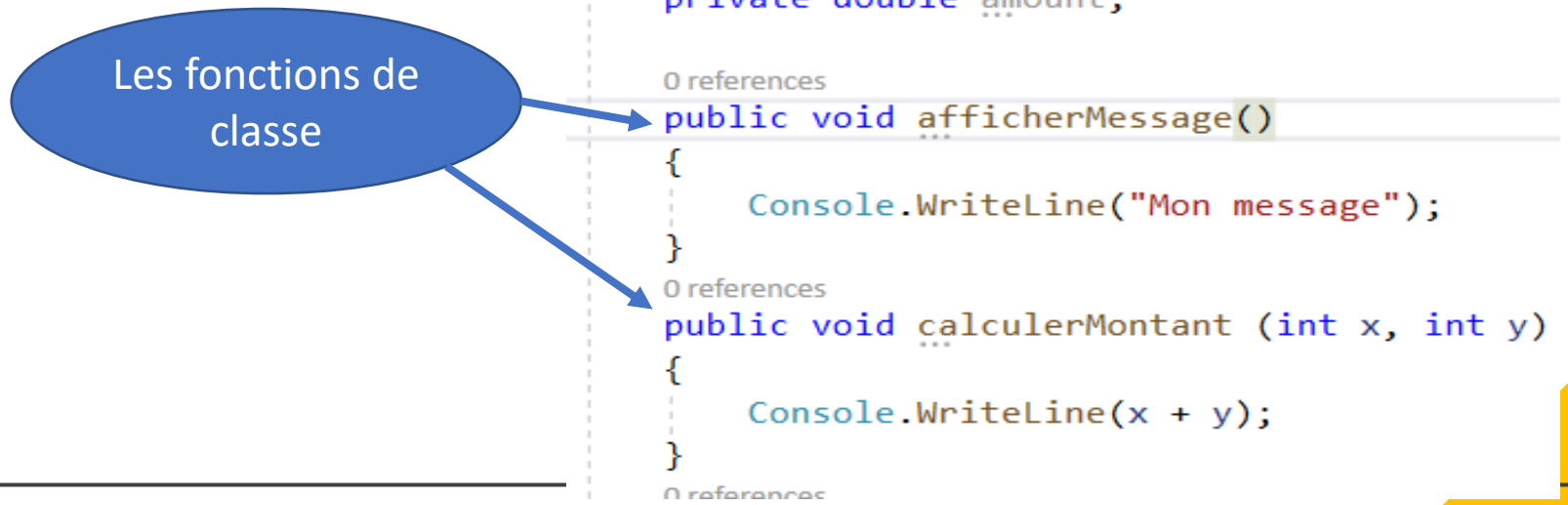
```
private int x;
private int y;
private string name;
private double amount;

0 references
public int X { get => x; set => x = value; }
0 references
public int Y { get => y; set => y = value; }
```

Accéder aux propriétés par le clé à mollette



Les fonctions de classes

Les fonctions de classes sont des fonctions qui sont accessibles uniquement lorsque l'objet a été instancié (**<u>à moins d'être *static*</u>**). Ces fonctions appartiennent à la classe et elles sont membres de la classe.



Les fonctions de classes

Lorsqu'il faut appeler une fonction d'une classe en dehors de la classe, il faut utiliser une instance de l'objet.

Exemple:

```
0 references
static void Main(string[] args)
{
    MyFirstClass myClasse = new MyFirstClass();
    myClasse.
}
```

```
calculerMontant
Equals
GetHashCode
GetType
MemberwiseClone
name
ToString
x
y
```

void MyFirstClass.calculerMontant(int x, int y)

Servez-vous de la .Dot notation pour avoir accès aux membres

Il nécessite de passer par l'instance de la classe pour avoir accès aux méthodes

Discussion

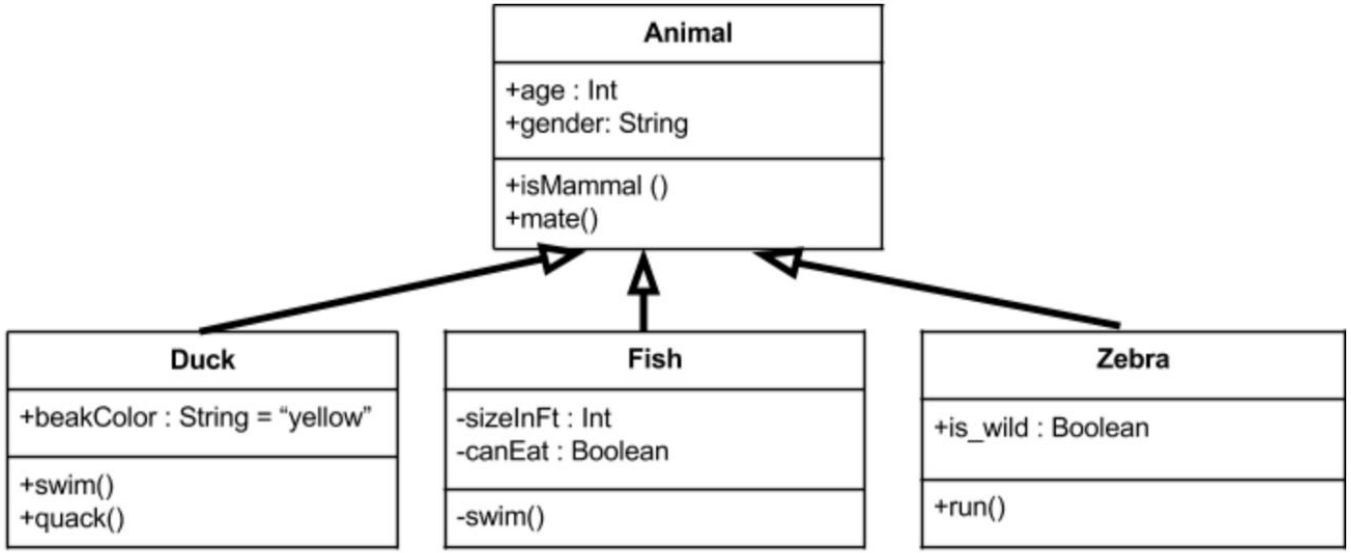
Qu’est-ce qu’une propriété ?

Comment accède-t-on à une fonction de classe depuis l’extérieur de l’objet?

Quels sont les membres d’une classe (3) ?

La POO et la vie réelle

La POO tente de simuler le monde réel par des objets



- Source: https://medium.com/@smagid_allThings/uml-class-diagrams-tutorial-step-by-step-520fd83b300b

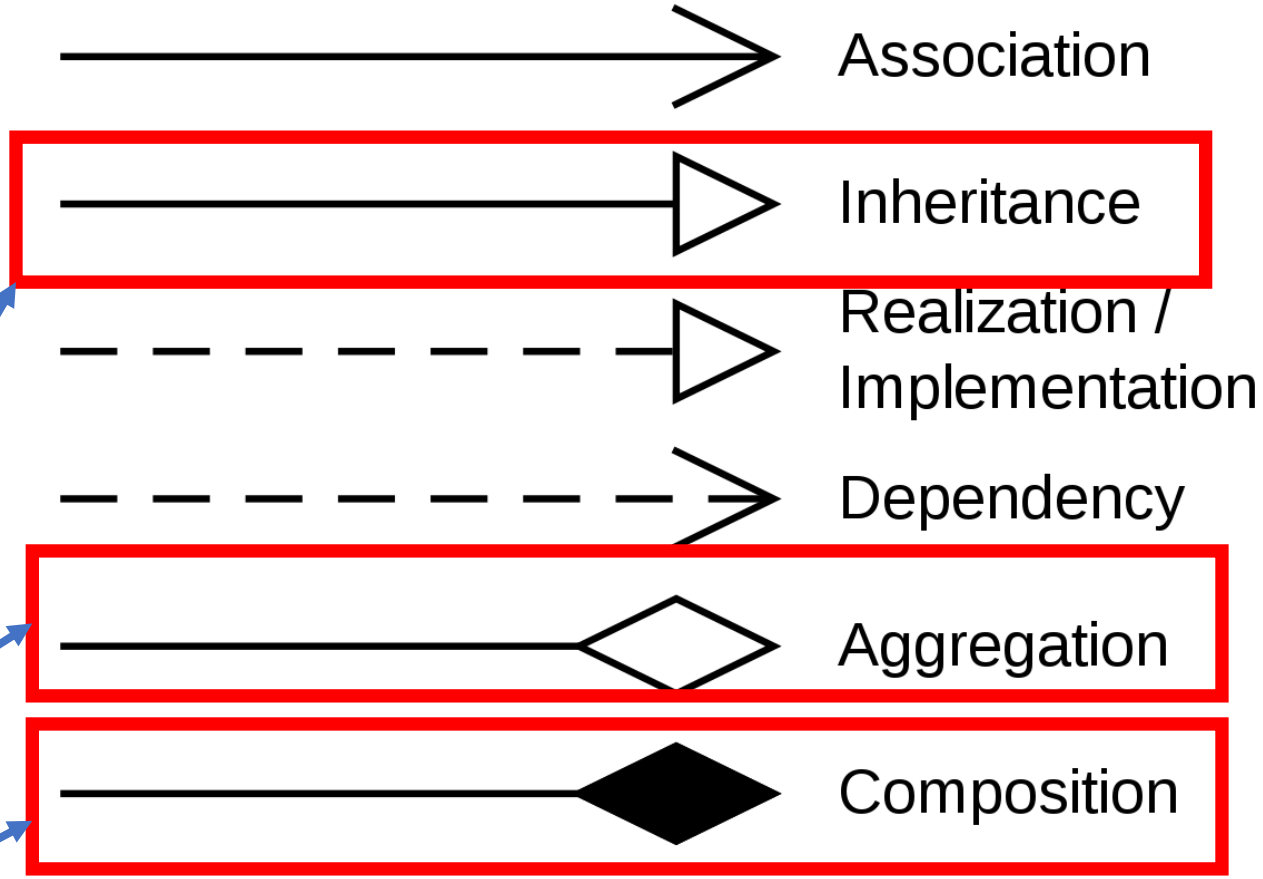
Le principe des relations entre les classes

L'objectif de l'orienté objet est d'effectuer des relations entre les classes.

Une classe peut être reliée à une ou plusieurs autres classes.

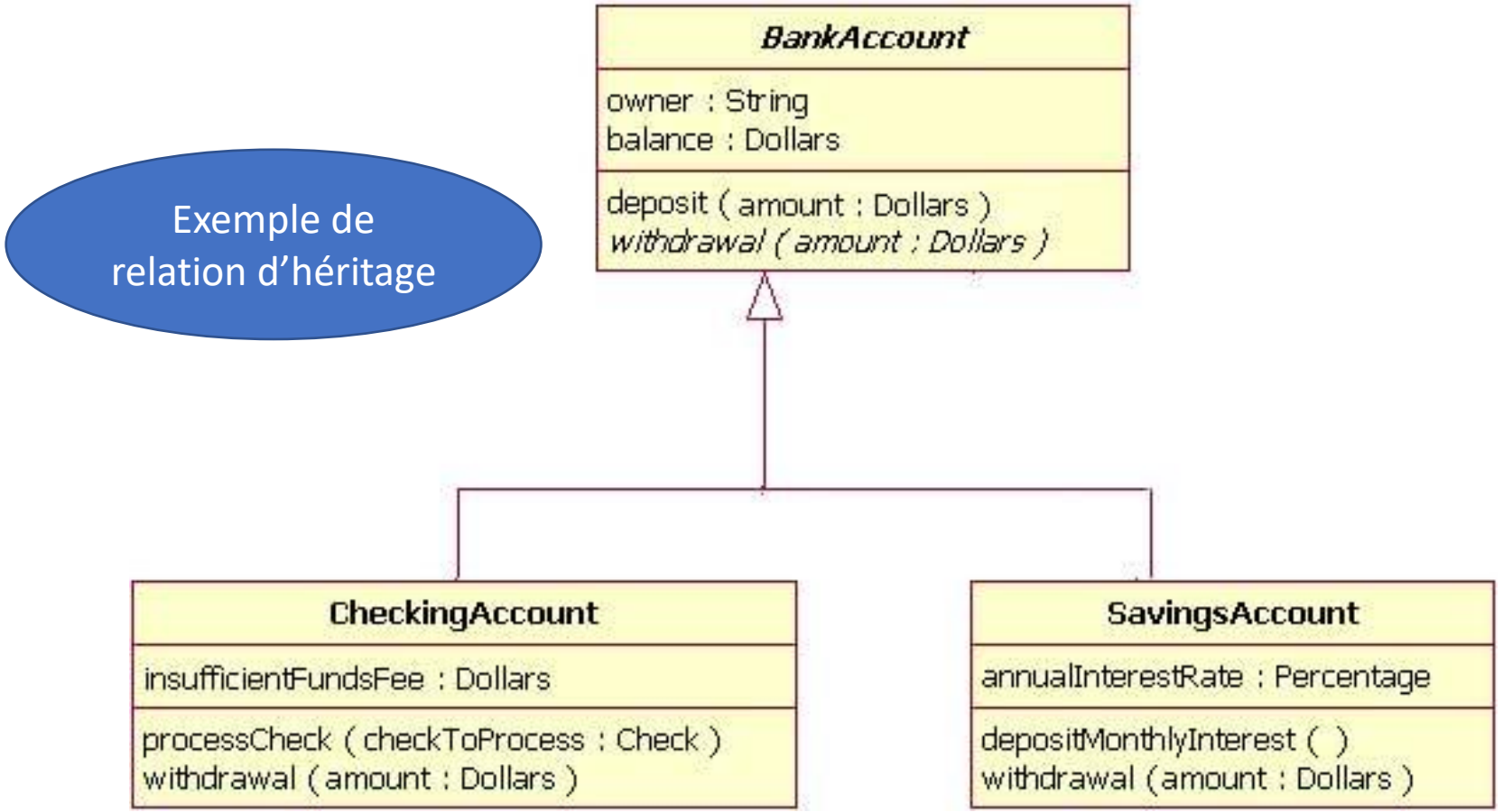
Un bon design contient de **bonnes** relations entre les classes.

Notations

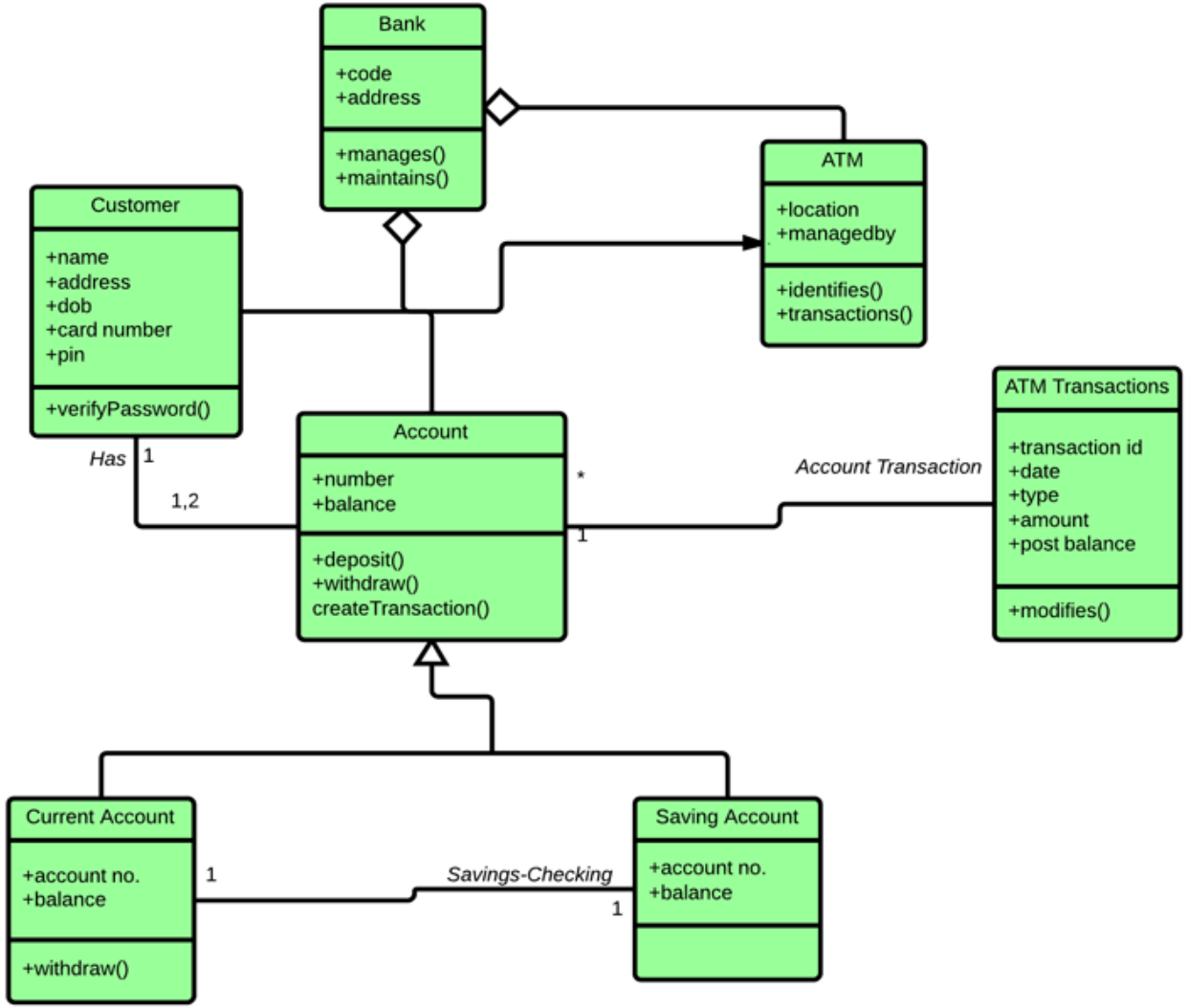


L'héritage, l'aggrégation et la composition sont les notions qu'il faut comprendre

Exemple de relations entre les classes



Exemple diagramme de classes



Source: https://www.lucidchart.com/pages/fr/diagramme-de-classes-uml

INFORMATION IMPORTANTES

L’objectif du laboratoire suivant n’est pas de vous faire expert en design mais de vous donner une compréhension globale du fonctionnement des classes.

L’héritage et les classes collatérales sont les notions les plus importantes du design de classe (dans le contexte de la formation Rebon numérique)

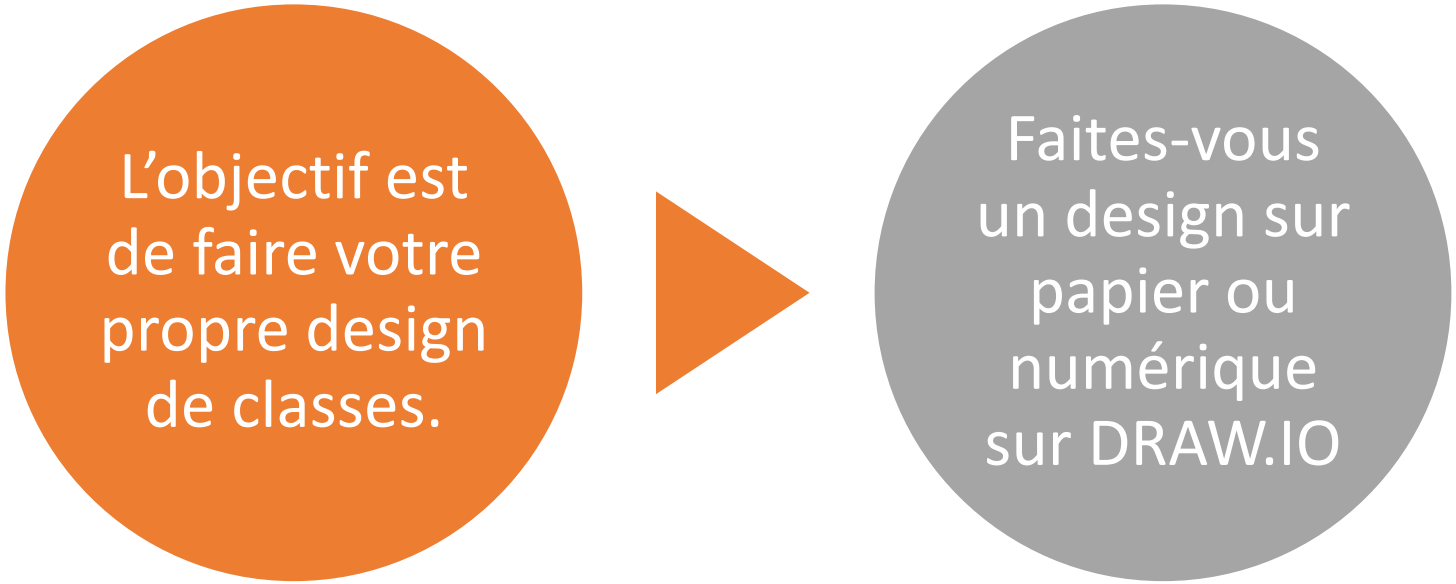
Discussions

Qu’avez-vous compris des classes et diagramme de classes ?

Pourriez-vous identifier l’utilité des classes et diagrammes de classes ?

Quelle est l’importance de faire un bon découpage de classes ?

Labs


L’objectif est de faire votre propre design de classes.
Faites-vous un design sur papier ou numérique sur DRAW.IO

Lab 2
1 heure

Vous devez créer une architecture de classes ayant ces spécifications:

1- Animal

2- Félin

3- Canin

4- Chat

5- Chien

6- Carnivore

7- Chiot

8- Lapin

9- Herbivore

10- Girafe

11- Mouton

Lab 3
1 heure

Vous devez créer une architecture de classe représentant la description suivante:

1- Patient (nom, prénom, numéro de téléphone, id, allergies)

2- Rendez-vous (date, heure, endroit)

3- Prescription (date, nom du produit)

4- Clinique (numéro, adresse, numéro de téléphone)

5- Médecin (nom, prénom)

Lab 3 Suite…

Présentez-nous votre découpage! Nous avons besoin de volontaires!

Comment avez-vous fait votre design? Présentez-le nous!

Lab 4
1 heure

Vous devez créer une architecture de classe représentant la description suivante:

1- Véhicule

2- Voiture

3- Motocyclette

4- Scooter

5- Vélo

6- Planche à roulette

7- Bateau à moteur

8- Avion

9- Hélicoptère

10- Véhicule à essence

11- Canot

Il peut y avoir des classes sous-entendus que vous pouvez ajouter. Exemple: Véhicule terrain, véhicule aquatique, etc.

Questions ?