

Algo et prog Q2

🕒 Date de création	@9 février 2024 08:09
☑ Relue ?	<input type="checkbox"/>

question

30 min pour se préparer + 5 minute de conversation(aide)
si bien compris la question code pendant 2h
si pas compris la question c ciao

Paradigme : représentation du monde dans un sens large (le fait de changer sa vision d'une chose)

l'implémentation c'est l'action qui va suivre une réflexion pour la concrétiser

Paradigme de programmation → manière de concevoir le code et un moyen pour formuler le problème dans un code informatique/imaginer sa solution et l'implémenter

Programmation procédurale : programmation impérative (ordre) et structuré (l'exécution dans un ordre précis)

- Suite d'instructions avec structures de contrôle
- appel de procédures réutilisable
- s'inspire de l'architecture des ordinateurs

objets 1

- Attributs
- méthodes

ageUtilisateur = 22 est un objet de type integer par exemple

Par exemple l'ordinateur est un objet, il a des propriétés, peut réaliser des actions, on peut avoir plusieurs ordinateurs qui auront des caractéristique différente

un objet sera divisé en 3

- Objet (un nom)
- se trouve dans un état (propriétés)
- répond à des messages (comportements)

Attribut → variable de classe (nom, prénom, age, taille, ...)

Propriété → manière de manipuler des attributs (lecture seule, accès non autorisée,...)

avantage de la POO

- Lisibilité et maintenance du code
- code source réduit, modulaire, évolutif

Inconvénient

- L'apprentissage est plus complexe que la procédural car beaucoup de concepts

une classe → un peu comme un plan de conception, genre (humain)

objet → instance de classe (ex Johan, Erwin, Joakim,...)

Points a retenir

- Un objet est une entité qui possède un état (défini par ses **attributs**) et des comportements (définis par ses **méthodes**)

- une classe est une **description** d'un ensemble d'objets qui partagent les **mêmes attributs** et **méthodes**
- un objet est une **instance** de sa classe
- une classe peut admettre plusieurs **conceptions**

Un point désigne un emplacement et sa seule caractéristique c'est sa position

Méthode → fonction d'une classe (procédure aussi)

- c'est une action
- ex : manger(TypeNourriture, quantité)

Méthode qui ne renvoie rien → procédure

Méthode de classe → fonction d'une classe avec une forte appartenance à une classe

Méthode statique → fonction d'une classe mais indépendante de la classe dans laquelle elle est appelée (on n'utilisera pas l'objet sur laquelle la méthode est appelée)

▼ Encapsulation

Encapsuler nos données dans notre objet et il ne sera donc plus visible de l'extérieur (on protège l'attribut et on ne pourra plus y accéder normalement)

Obtenir la valeur d'un attribut → Méthode d'accès "Getter"

Modifier la valeur d'un/plusieurs attributs → "Setter"

L'encapsulation va permettre de définir le niveau de visibilité d'éléments de la classe. On définit le droit d'accès aux données.

Les attributs d'une classe sont privés et ses méthodes publiques

3 niveau de protection

- Public → accès par tout
- Protected → la classe a accès et tout ses "enfants" aussi (faire ca par défaut)
- Private → uniquement depuis la classe elle même

choisir le bon niveau

- Protection maximum du code depuis l'extérieur
- Sensibilité de chaque élément
- Impacts des niveaux d'accès à un élément sur le reste de la classe
- Identification des éléments qui ont besoin d'accès pour fonctionner

▼ Interface

C'est le contrat d'une classe et la définition de ses méthodes

(une classe avec que des méthodes)

ex : classe humain, avec nom, prénom, age → on est tous humain ici et on a une interface professeur avec une méthode enseigner et une étudiant avec méthode étudier et dès que l'un des humains prend la classe professeur acquerra la méthode enseigner et si classe étudiant acquerra méthode étudier, mais ce n'est pas parce qu'il l'a reçu que son utilisation est obligatoire.

Constructeur est la première méthode de votre classe

- Il fonctionne comme une fonction
- Il peut prendre des paramètres

Un constructeur est une méthode particulière de la classe, on utilisera le nom de la classe comme nom du constructeur. Il est appelé au moment de la création de l'objet

Il peut être de 3 types différents :

- Constructeur par défaut → pas de paramètre (obligatoire, doit toujours être là)
- constructeur par recopie/de copie → qu'1 paramètre, en gros une photocopie des attributs
- constructeur paramétrique → on peut avoir un deuxième constructeur avec 1 paramètre qui va affecter tout les attributs