Bing

Je suis heureux que vous ayez adopté le mode d'apprentissage par concept. Le concept de classe en python est très important pour la programmation orientée objet. Une classe est comme un modèle ou un plan qui définit les attributs et les méthodes communs à un groupe d'objets. Par exemple, si vous voulez créer des objets qui représentent des vélos, vous pouvez définir une classe Bike qui a des attributs comme le nom, le nombre de vitesses, la couleur, etc. et des méthodes comme démarrer, freiner, changer de vitesse, etc.

Pour créer une classe en python, vous utilisez le mot-clé class suivi du nom de la classe et d'un bloc indenté qui contient la définition de la classe. Par exemple :

[python]

class Bike:

# attributs de la classe

name = ""

gear = 0

# méthode spéciale pour initialiser un objet de la classe

def \_\_init\_\_(self, name, gear):

self.name = name

self.gear = gear

# méthode pour afficher le nom du vélo

def show\_name(self):

print("Le nom du vélo est " + self.name)

# méthode pour changer de vitesse

def change\_gear(self, new\_gear):

self.gear = new\_gear

print("La vitesse du vélo est maintenant " + str(self.gear))

Ici, nous avons défini une classe Bike qui a deux attributs (name et gear) et trois méthodes (init, show\_name et change\_gear). La méthode init est spéciale car elle est appelée automatiquement lorsque vous créez un objet de la classe. Elle prend comme premier paramètre self, qui représente l'objet lui-même, et d'autres paramètres qui sont utilisés pour initialiser les attributs de l'objet. Les autres méthodes prennent également self comme premier paramètre et peuvent accéder ou modifier les attributs de l'objet en utilisant le point (.) .

Pour créer un objet de la classe Bike, vous utilisez le nom de la classe suivi de parenthèses contenant les arguments nécessaires à l'initialisation. Par exemple:

[python]

# créer un objet de la classe Bike

b1 = Bike("VTT", 21)

# appeler les méthodes de l'objet

b1.show\_name() # affiche "Le nom du vélo est VTT"

b1.change\_gear(15) # affiche "La vitesse du vélo est maintenant 15"

J'espère que ces exemples vous ont aidé à comprendre le concept de classe en python. Si vous voulez en savoir plus, vous pouvez consulter les liens suivants:

- Classes in Python with Examples - Python Geeks

- Python Classes - W3Schools

- 9. Classes — Python 3.11.4 documentation

- Python Classes and Objects (With Examples) - Programiz