

Quel paradigme mettre en place pour programmer Minecraft ?

1 Principes du jeu

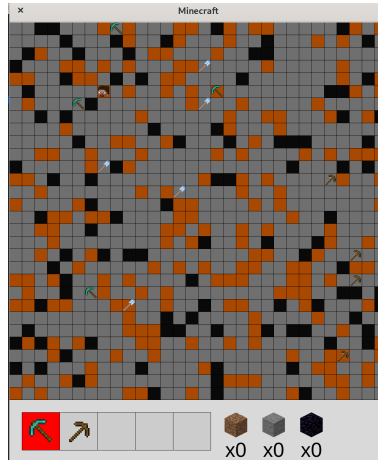


FIGURE 1 – Une version simplifiée

2 Programmation orienté objet

2.1 Définition

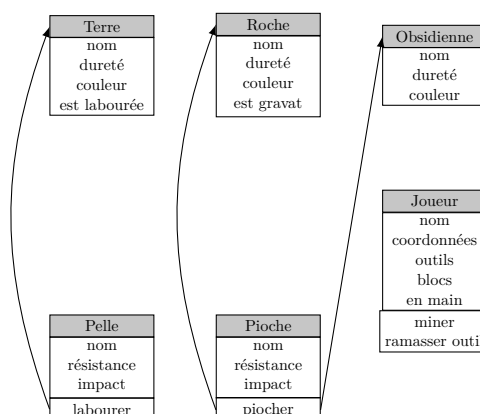
Le **paradigme objet** consiste à construire des *objets* et les faire interagir entre eux. Un objet représente une entité physique, un concept...

À retenir

Un objet possède :

- des caractéristiques : les **attributs**,
- des capacités : les **méthodes**.

2.2 Modélisation



2.3 Instanciation

- Chaque objet est un modèle qui peut être vu comme un squelette.

Terre
nom
dureté
couleur
est labourée

— On crée une **instance** de l'objet. C'est cette instance qui interagit dans le programme.

Terre	Terre	Terre
nom = terre	nom = terre	nom = terre
dureté = 10	dureté = 10	dureté = 10
couleur = marron	couleur = marron	couleur = marron
est labourée = faux	est labourée = faux	est labourée = faux

— Chaque instance possède ses propres *attributs* et *méthodes*.

3 Implémentation

3.1 Créer un objet

```
1 class Terre
```

Code 1 – Créer une classe

3.2 Initialiser les attributs

La méthode `__init__` est appelée automatiquement quand nousinstancions un objet.

```
1 class Pioche:
2     def __init__(self, nom: str):
3         self.nom = nom
4         if nom == "wood_pickaxe":
5             self.resistance = 30
6             self.impact = 5
7         elif nom == "diamond_pickaxe":
8             self.resistance = 100
9             self.impact = 100
```

Code 2 – Les attributs sont initialisés dans le *constructeur*.

3.3 Définir les méthodes

À retenir

On appelle **méthode** une fonction interne à la classe de l'objet.

En Python, le premier paramètre est **toujours self**. C'est un attribut **interne** à la classe.

```
1 def piocher(self, bloc: object) -> bool:
2     bloc.durete -= self.impact
3     self.resistance -= USURE
4     if self.resistance <= 0:
5         return False
6     return True
```