

CM1: Ap.

Clé moodle : a4c926

Structure itérable \rightarrow on peut la parcourir.

On peut utiliser une boucle for pour les parcourir.

séquentielle \Rightarrow itérable, mais il existe des itérables non séquentielles.

Par exemple, les ensembles (set) et les dictionnaires (dict).

- * Ensemble : type de donnée qui regroupe une collection de valeurs sous une \hat{m} référence avec deux propriétés:
 - \rightarrow aucun doublon
 - \rightarrow pas ordonné.

Comment créer ensemble ? :

- On écrit la séquence d'éléments séparés par une virgule, entre accolades. (Construct' littérale)

`{ "Timoléon", "Raymond", "Carmen" }`

- Le constructeur set

`set("Hello, world!")`

`set([2, 1, 3, 13])`

\rightarrow L'ensemble vide : `set()`

l'ensemble des carrés des éléments compris entre -2 et 5

$\{k*k \text{ for } k \text{ in Range}(-2,6)\}$

Parcél mais impair

$\{k*k \text{ for } k \text{ in Range}(-2,6) \text{ if } k \% 2 == 1\}$

On peut créer un ensemble de tuple.

⚠ On ne peut pas créer un ensemble d'ensembles ni de listes.

⇒ Le type doit être non-mutable pour aller dans un ensemble.

• taille d'un ensemble:

$\text{len}(\{1, 3, 2, 1, 1, 2\}) == 3 \Rightarrow \text{True}$.

• itérer sur un ensemble

for element in ensemble seulement! ✓

• Test d'appartenance

$3 \text{ in } \{1, 2, 4\} \Rightarrow \text{False}$.

↖ ça ne coûte pas cher.

• Méthodes

→ add : ajoute un élément à l'ensemble : $\{1, 2, 3\}$ - add(4)

ne pas écrire `ens = ens.add()`,

→ Remove : enlève un élément : $\{1, 2, 3, 4\}$ - Remove(4)

⚠ $\{1, 2, 3\}$. Remove(4) ⇒ KeyError: 4.

set.union, set.intersection, set.issubset, set.isdisjoint, set.copy

* des dictionnaires: Association clé-valeur.

Valeurs pas uniques, mais les clés, si.

fruits = {'pomme': 1.95, 'banane': 2.59, 'orange': 1.95}

Construction avec dict :

dict(in: list[tuple[Any, Any]]) → dict

Toute valeur ne peut pas servir de clé, seules les valeurs non mutables peuvent servir de clé.

• dictionnaire vide

{ } ou dict()

- Compréhension:

$$d = \{k: k \times k \text{ for } k \text{ in range}(10)\}$$

$$d[5] = 25.$$

- Taille: nombre d'association = nombre de clés.

len(dict),

Pas d'indices, on accède de aux éléments par les clés.

Si la clé n'existe pas \Rightarrow erreur.

Accès sécurisé: dict.get(key, default)

fruits.get("fraises", -1) \rightarrow -1.

'banana' in fruits \rightarrow True

'Fraises' in fruits \rightarrow False.

- Mutabilité:

fruits['banana'] = 2.10. \rightarrow modifie

fruits['fraises'] = 3.1 \rightarrow ajoute.

\rightarrow On supprime avec del

del fruits['fraises']

• Itération : boucle for, itère sur les clés.

itération sur les valeurs: `for prix in fruits.values()`

itérer sur les couples: `for k, v in set.items()`.

