RER - Initiation au Python - Les Tableaux et les Dictionnaires

# Contexte :

Dans cette étude, nous allons apprendre à nous familiariser avec les différentes structures de données telles que les tableaux (listes), les tuples et dictionnaires.

# Mots clés :

* Objet (rappel) : un conteneur symbolique et autonome qui contient des informations et des mécanismes ou fonctions concernant un sujet, manipulés dans un programme.
* Structures de données (Séquences) : Ensemble fini et ordonné d'éléments indicés de 0 à n-1 (si cette séquence comporte n éléments)
* Dictionnaire : Ensemble non-ordonné d’associations clé-valeur. Comme un dictionnaire classique, on accède à la définition par la clé.
* Tableau (listes) : séquence d’élément ordonnées (chaque élément à un index) modifiable
* Listes imbriquées : séquence de listes ordonnées modifiable
* Tuple : séquence d’élément ordonnées non-modifiable
* Containers : objets qui contient un ensemble d’objets. Ex : tuple, liste
* Sets : ensemble d’objets unique non-ordonnés, non indexés, modifiable mais de type non-mutable
* Indice : entier ou objet permettant d’accéder directement à un élément d’une liste
* Indiçage négatif : compter à partir de la fin ou du dernier élément d’une liste
* Objet ordonné : un objet qui est associé à un indice dans un tableau
* Objet modifiable : un objet qui peut être modifié après sa création
* Objet non-mutable : c’est un objet qui ne peut pas être modifier après sa création en Python
* Les crochets : signe de ponctuation qui sert à délimiter un tableau ou à indiquer une opération sur un tableau. Ex : mon\_tableau = [3, 9, 27] ; print(mon\_tableau[2])
* Accolades : signe de ponctuation qui sert à délimiter un dictionnaire. Ex : mon\_dico = { « banane » : « banana », « tomate »: « tomato »}
* Les clés : un élément (integer, float, string) qui permet d'identifier un objet
* Méthode : C’est une fonction associée à un objet pour le manipuler.
  + .keys() : Renvoie un objet de vue (correspond à une “image” prise à l’instant t de l’objet). L'objet vue contient les clés du dictionnaire, sous forme de liste.
  + .values() : Il renvoie un objet de vue. L'objet de vue contient les valeurs du dictionnaire, sous forme de liste.
  + .items() : Renvoie un objet de vue. L'objet de vue contient les paires clé-valeur du dictionnaire, sous forme de tuples dans une liste.
  + .append() : Ajoute les éléments dans une liste mais conserve la forme d'itérable.
* Hachage : Opération sur une donnée clé qui lui associe un identifiant dans un tableau grâce à une fonction hash() (hache). Elle permet d’avoir de la vitesse en utilisant la clé.

# Problématique(s) :

1. Comment les structures de données dans python permettent de stocker, analyser les données ?
2. Quels sont les différentes structures de données ?

# Hypothèses :

1. La nature de la données structurées ou non va permettre de déterminer la structure de la donnée utilisée pour la traiter (Étienne)

FAUX

1. On peut importer des données non structurées dans une structure que l’on définit dans notre programme. (Solenn)

VRAI

1. Il est primordial de bien connaître les dictionnaires de Python pour savoir manipuler les structures de données. (Adeline)

VRAI

1. Les tuples sont non-mutable. (Aude)

VRAI

1. Il est obligatoire d’utiliser les structures de données pour organiser les données. (Loïc)

VRAI

1. Le langage Python convertis toutes les données en liste avant de les manipuler. (Seydou)
2. Les structures de données en Python facilite leur stockage. (Nicolas)

VRAI

1. En Python le dictionnaire permet de stocker des données non-structurées. (Axel)

VRAI

1. Le dictionnaire ne permet pas l’indiçage des données. (JP)

VRAI

1. Dans un même tableau on peut stocker différents type de données. (Tetyana)

VRAI

1. Le dictionnaire est une variable Globale. (Briand)

FAUX

1. On peut stocker des données dans une liste ou dans un dictionnaire (de la même manière). (Adrienx2)

FAUX

1. Le dictionnaire stock les données ordonnées de manière aléatoire (Étienne x2)

FAUX

1. Les structures de données sur Python permettent de modifier des données non-structurée en données structurée afin de les intégrer facilement en base de données. (Loïc x2).

FAUX, on peut avoir des structures de données pour les données non structurées pour les traiter ensuite.

# Plan d'action :

1. Explorer les ressources
2. Définir les mots clefs
3. Répondre aux hypothèses
4. Réaliser le Workshop
5. Répondre aux problématique
6. Restitution du RER
7. Livraison du livrable