

TD 4 : Dictionnaires et Traitement de textes

1 Manipulation de dictionnaires

1. Construire un dictionnaire appelé `infos` contenant les clés : `nom`, `prenom`, `age`, `ville`. Ajouter les valeurs correspondantes (`ville` = ville où vous vivez).
2. Afficher les clés et les valeurs du dictionnaire en une commande.
3. Ajouter une clé `loisirs` ayant pour valeur une liste de loisirs de votre choix et afficher le dictionnaire avec le nouvel élément.
4. Retirer l'élément `age`.
5. Afficher la phrase : "Je m'appelle {nom} {prenom} et j'habite à {ville}."

2 Traitement de textes

2.1 Partie 1

```
1 texte = "Si tous les autres acceptaient le mensonge imposé par le Parti - si tous les rapports  
    racontaient la même chose -, le mensonge passait dans l'histoire et devenait vérité.  
    Celui qui a le contrôle du passé, disait le slogan du Parti, a le contrôle du futur. Celui  
    qui a le contrôle du présent a le contrôle du passé."
```

1. Mettre le texte en minuscules.
2. Retirer les points, les virgules et les tirets.

Note : Il se peut qu'il y ait des doubles espaces à la suite de ce traitement. Vous pouvez les retirer en utilisant un `replace()`.

```
1 texte = texte.replace("  ", " ")  
2 print(texte)
```

3. Afficher la taille du texte en nombre de caractères.
4. Couper le texte en mots avec la fonction `split()`. Stocker le résultat dans une variable appelée `mots`.
5. Afficher le nombre de mots du texte.
6. Compter les occurrences de chaque mot et les stocker dans un dictionnaire.

2.2 Partie 2

Dans cette partie, vous allez manipuler les trois phrases du texte extraites après les traitements faits dans la première partie.

```
1 phrases = ["si tous les autres acceptaient le mensonge imposé par le parti si tous les  
    rapports racontaient la même chose le mensonge passait dans l'histoire et devenait vérité"  
    , "celui qui a le contrôle du passé disait le slogan du parti a le contrôle du futur", "  
    celui qui a le contrôle du présent a le contrôle du passé"]
```

1. Afficher chaque phrase avec une boucle.
2. Afficher le nombre de caractères par phrase.
3. Afficher le nombre de mots par phrase.
4. Répondre aux trois questions précédentes en utilisant une seule boucle.
5. Stocker les informations des questions 1, 2 et 3 dans un dictionnaire pour chaque phrase. Les clés sont les suivantes : `énoncé`, `nb_caractères`, `nb_mots`. (Voir exemple ci-dessous).

```

1 # Dictionnaire pour la phrase 1
2
3 {
4 'énoncé': "si tous les autres acceptaient le mensonge imposé par le parti si tous les
   rapports racontaient la même chose le mensonge passait dans l'histoire et devenait vé
   rité",
5
6 'nb_caractères': 164,
7
8 'nb_mots': 27
9 }

```

3 Chiffrement par substitution

On se propose de réaliser une fonction de chiffrement et déchiffrement par substitution, c'est-à-dire que chaque lettre de l'alphabet est remplacée par une autre.

La clé de chiffrement sera représentée par un dictionnaire. Chaque *clé* sera une lettre de l'alphabet et la valeur associée sera la lettre par laquelle on la substitue.

Dans l'exemple suivant, le dictionnaire `cle_chiffrement` contient les substitutions à réaliser et `chiffrement` est la fonction qui réalise la substitution.

```

1 cle_chiffrement = { 'n': 'o', 'o': 'p' }
2 chiffrement("non", cle_chiffrement)
3
4 # résultat : opo

```

De la même façon, la fonction `dechiffrement` permet de réaliser la substitution inverse.

1. Écrire un dictionnaire (manuellement ou à l'aide d'une boucle) qui associe chaque lettre de l'alphabet à la lettre qui la suit (le z sera remplacé par a).
2. Écrire une fonction qui prend en paramètre une chaîne de caractères et un dictionnaire représentant une clé de chiffrement et qui réalise la substitution de chaque lettre de la chaîne de caractères en suivant le dictionnaire.
3. Écrire la fonction de déchiffrement.