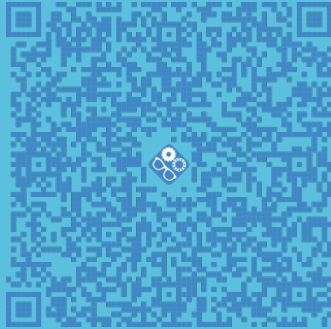
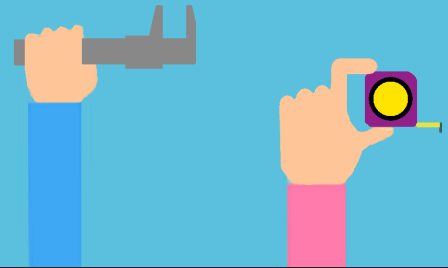




Traitement de listes et import-export de fichiers



Renaud Costadoat
Lycée Dorian



Traitement de listes

Import/export de fichiers

La manipulation de listes

Les listes (ou list / array) en python sont une variable dans laquelle on peut mettre plusieurs variables.

Créer une liste vide

```
>>> liste = []
```

Ajouter une valeur à une liste

```
>>> liste = [1,2,3]
```

Voir le contenu de la liste

```
>>> liste  
[1, 2, 3]
```

Ajouter du contenu à la liste

```
>>> liste.append(4)  
>>> liste  
[1, 2, 3, 4]
```

La manipulation de listes

Afficher un item d'une liste (le premier item commence toujours avec l'index 0)

```
>>> liste = ["a","b","c","d"]
>>> liste[0]
'a'
>>> liste[2]
'c'
```

Modifier une valeur de une liste

```
>>> liste[1] = "c"
>>> liste
['a', 'c', 'c', 'd']
```

Supprimer une entrée avec un index

```
>>> liste = ["a","b","c","d"]
>>> del liste[1]
>>> liste
['a', 'c', 'd']
```

La manipulation de listes

Supprimer une entrée avec sa valeur

```
>>> liste = ["a","b","c","c"]
>>> liste.remove("b")
>>> liste.remove("c")
>>> liste
['a', 'c']
```

Inverser les valeurs d'une liste

```
>>> liste = ["a", "b", "c"]
>>> liste.reverse()
>>> liste
['c', 'b', 'a']
```

Compter le nombre d'items d'une liste

```
>>> liste = ["a", "b", "c", "d"]
>>> len(liste)
4
```

La manipulation de listes

Compter le nombre d'occurrences d'une valeur

```
>>> liste = ["a", "c", "c", "d"]
>>> liste.count("a")
1
>>> liste.count("c")
2
```

Trouver l'index d'une valeur

```
>>> liste = ["a", "c", "c", "d"]
>>> liste.index("a")
0
>>> liste.index("c")
1
```

La manipulation de listes

Manipuler une liste

```
>>> liste = [1, 2, 3, 4, 5]
>>> liste[-1] # Cherche la dernière occurrence
5
>>> liste[-4:] # Affiche les 4 dernières occurrences
[2, 3, 4, 5]
>>> liste[:] # Affiche toutes les occurrences
[1, 2, 3, 4, 5]
>>> liste[2:4] = [6, 7]
[1, 2, 6, 7, 5]
>>> liste[:] = [] # vide la liste
```

Boucler une liste

Afficher les valeurs d'une liste

```
>>> liste = ["a","b","c"]
>>> for lettre in liste:
...     print lettre
...
a
b
c
```

Afficher les valeurs d'une liste et récupérer l'index

```
>>> for lettre in enumerate(liste):
...     print lettre
...
(0, 'a')
(1, 'b')
(2, 'c')
```

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ↺ 🔍 ↻

Copier une liste

```
>>> x = [1,2,3]
>>> y = x
>>> y[0] = 4
>>> x
[4, 2, 3]
```

```
>>> x = [1,2,3]
>>> y = x[:]
>>> y[0] = 4
>>> x
[1, 2, 3]
>>> y
[4, 2, 3]
```

Utilisation du module `copy`.

```
>>> import copy
>>> x = [[1,2], 2]
>>> y = copy.deepcopy(x)
>>> y[1] = [1,2,3]
>>> y
[[1, 2], [1, 2, 3]]
```

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ↺ 🔍 ↻

Liste et string

Transformer une string en liste

```
>>> ma_chaine = "Lycee:Dorian:Paris"
>>> ma_chaine.split(":")
['Lycee', 'Dorian', 'Paris']
```

Transformer une liste en string

```
>>> liste = ["Lycee", "Dorian", "Paris"]
>>> ":".join(liste)
'Lycee:Dorian:Paris'
```

Trouver un item dans une liste

```
>>> liste = ["Lycee", "Dorian", "Paris"]
>>> 'Paris' in liste
True
>>> 'Lille' in liste
False
```

Manipulation de liste

La fonction range

```
>>> range(10)
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

Combiner deux listes

```
>>> x = [1, 2, 3]
>>> y = [4, 5, 6]
>>> x + y
[1, 2, 3, 4, 5, 6]
```

Multiplier une liste

```
>>> x = [1, 2]
>>> x*5
[1, 2, 1, 2, 1, 2, 1, 2, 1, 2]
```

Créer une liste de zéros

```
>>> [0] * 5
[0, 0, 0, 0, 0]
```

Importer un fichier

Il peut être très intéressant de manipuler des fichiers avec python afin de récupérer (ou de pouvoir les transmettre) des données d'un autre logiciel. Cela peut être aussi très intéressant de sauvegarder ces données.

Changer le répertoire de travail courant

```
>>> import os
>>> os.chdir("C:/fichiers python")
```

Ouvrir un fichier, il faut indiquer le chemin et le mode d'ouverture (droits):

- **r** : ouverture en lecture (Read),
- **w** : ouverture en écriture (Write). Le contenu du fichier est écrasé. Si le fichier n'existe pas, il est créé,
- **a** : ouverture en écriture en mode ajout (Append). On écrit à la fin du fichier sans écraser l'ancien contenu du fichier. Si le fichier n'existe pas, il est créé.

```
>>> fichier = open("fichier.txt", "r")
```



Lire et écrire dans un fichier

Lire l'intégralité du fichier

```
>>> fichier = open("fichier.txt", "r")
>>> contenu = fichier.read()
>>> print(contenu)
Texte contenu dans le fichier.
```

Fermer le fichier

```
>>> fichier.close()
```

Écrire dans un fichier

```
>>> fichier = open("fichier.txt", "w")
>>> fichier.write("Texte a ecrire \n Seconde ligne")
>>> fichier.close()
```



Application: Comptes informatiques

Il nous ait demandé par le service informatique du lycée de créer à partir d'une liste de classe (au format csv) de générer un fichier (au format csv) intégrant les **login** et **mot de passe** des comptes informatiques.

La liste est fournie au format suivant: `NOM;Prenom\n`.

Le résultat doit être mis sous la forme suivante: `NOM;Prenom;login;passwd\n`, avec:

- `login`: première lettre du prénom et 7 premières lettres du nom en minuscule,
- `passwd`: trois premières lettres du prénom converties en chiffres ('a=0', 'b=1', ..., 'z=25')

Question: Proposer un script python permettant de répondre à ce cahier des charges.