

Types natifs en Python - Exercices

1.1 Questions à reponses courtes

- Quels sont les 2 types numeriques en Python?
- Quels sont les types séquentiels natifs en Python?
- Comment fait-on l'affectation multiple de la valeur 3 à la variable a et 55 à la variable b?
- Quels sont les types de chacune des expressions suivantes?

- [10,20,30]
- ['a': 10, 'b': 20, 'c': 30]
- (10,20,30)

- Quelles méthodes s'appliquent aux Listes Python?

append, insert, items, keys, sort, values

- Quelles méthodes s'appliquent au Dictionnaires Python?

append, insert, items, keys, sort, values

- Quels types sont immuables en Python?

Chaine de caractères, Dictionnaire, Listes, Tuples

1.2 Listes et Tuples

- Quelle expression Python a pour valeur la liste [2, 5, 8, 11]?

- [3 * i + 2 for i in range(3)]
- [3 * i + 2 for i in range(4)]
- [3 * i + 1 for i in range(3)]
- [3 * i - 1 for i in range(3)]

- On définit la liste L suivante :
L = [('Theo', 12), ('Lidia', 15), ('Emilie', 15), ('Vincent', 17)]

Quelle est la valeur de l'expression [element[0] for element in L if element[1] >=15]

- ['Theo']
- ['Lidia', 'Emilie', 'Vincent']
- ['Vincent']
- ('Lidia', 'Emilie', 'Vincent')
- Ecrire l'instruction qui permettra d'ajouter ('Richard', 3) à la fin de la liste L
- Ecrire l'instruction qui permettra d'insérer ('Camille', 16) entre ('Theo', 12) et ('Lidia', 15)

- Après ces modifications, on écrit : `L.pop()`. Qu'est ce qui est retourné dans la console Python? Quel est alors le contenu de la liste L?
- c. Quel est le type de l'expression `f(4)` si la fonction `f` est définie par :

```
1 def f(x) :
2     return (x, x**2)
```

Partie 2

Echiquier et compréhension de Liste

Chaque case d'un échiquier peut être représentée par le tuple (colonne, ligne). Par exemple, la première case noire en bas à gauche est ['A', '1']. Chaque rangée horizontale de l'échiquier est elle-même liste de tuples (un tuple par case). Par exemple, la ligne 1 est représentée par :

```
[('A', '1'), ('B', '1'), ...]
```

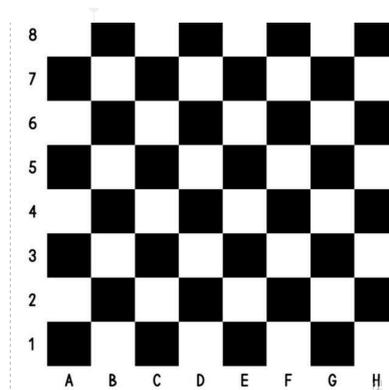


FIGURE 1 – Echiquier 64 cases

- a. Ecrire une instruction Python permettant de générer en compréhension la ligne 1 " d'un échiquier (8 cases)

Aide : Utiliser la chaîne de caractères `lettres = 'ABCDEFGH'`

- b. Ecrire une instruction Python permettant de générer en compréhension la matrice `M` d'un échiquier de 64 cases, rangées dans l'ordre de l'image, sous forme d'un Tuple de listes :

```
1 [[('A', '8'), ('B', '8'), ...] ...]
```

Aide : Utiliser la chaîne de chiffres `chiffres = '12345678'`

Découpage d'une Liste (slicing)

Le découpage de liste (*slicing*) permet d'extraire une séquence d'une liste. Soit L une liste. La syntaxe $L[i:j]$ permet d'extraire tous les éléments consécutifs compris entre l'élément de rang i inclus et l'élément de rang j exclus (jusqu'à $j-1$).

On donne :

```
1 mois = ["Janvier", "Fevrier", "Mars", "Avril",  
2       "Mai", "Juin", "Juillet", "Aout", "Septembre"  
3       "Octobre", "Novembre", "Decembre"]
```

a. En utilisant la technique du *slicing* écrire une instruction permettant :

- d'extraire de la liste `mois` uniquement les 6 premiers mois.
- d'extraire les 2 mois d'été
- extraire la période de mars à octobre inclus

b. Les objets créés, sont ils copiés de la liste d'origine par *valeur* ou bien par *référence*?

c. On fait une copie de `mois` de la manière suivante :

```
1 months_in_french = mois
```

On modifie `months_in_french` avec l'instruction suivante :

```
1 months_in_french[11] = "December"]
```

d. Que vaut la liste `months_in_french`?

- ["Janvier", .., , "Novembre", "Decembre"]
- ["Janvier", .., "Novembre", "December"]

e. Que vaut la liste `mois`? (Justifier)

- ["Janvier", .., , "Novembre", "Decembre"]
- ["Janvier", .., "Novembre", "December"]

Parcours d'un tableau

On dispose d'une table `tab` constituée d'une liste de trois sous-listes (un tableau). Chacune des sous-listes contient quatre caractères.

```
1 tab=[['A', 'B', 'C', 'D'],  
2      ['E', 'F', 'G', 'H'],  
3      ['I', 'J', 'K', 'L'] ]
```

- a. Parmi les propositions suivantes, laquelle permet de convertir cette table en une liste L contenant dans l'ordre, ligne par ligne, les 12 caractères de tab ?

script A :

```
1 L = []
2 for i in range(3):
3     for j in range(4):
4         L.append(tab[i][j])
```

script B :

```
1 L = []
2 for i in range(4):
3     for j in range(3):
4         L.append(tab[j][i])
```

Partie 5

Copie de Listes

- a. Que vaut L1 après le script suivant ?

```
1 L1 = [1, 2, 3, 4]
2 L2 = L1
3 L2.pop()
```

- b. Que vaut L1 après le script suivant ?

```
1 L1 = [1, 2, 3, 4]
2 L2 = L1
3 L2.append(5)
```

Partie 6

Dictionnaires

On définit :

```
1 contacts = {'Toto': 'toto@nsi.fr', 'Chloé': 'chloe@nsi.com', 'Paul': 'paul@nsi.net', 'Clémence': 'clemence@nsi.org' }
```

- a. Parmi les propositions suivantes, laquelle est exacte ?

- 'Chloé' est une valeur de la variable contacts
- 'Chloé' est une clé de la variable contacts
- 'Chloé' est un attribut de la variable contacts
- 'Chloé' est un champ de la variable contacts

b. pour le dictionnaire `contacts` de la question précédente :

- Comment accède t-on à la valeur `'clemence@nsi.org'` ?
- Comment ajoute t-on l'adresse mail `lea19@nsi.fr` au contact Léa dans ce dictionnaire ?
- Comment obtenir à l'aide d'une instruction, la liste de tous les contact (et pas leur mail) ?
- Comment obtenir la liste de tous les mails à l'aide d'une instruction ?

c. On définit le dictionnaire :

```
1 repertoire = [{'nom': 'Francette', 'poste': 412}, {'nom': 'Jeanne', 'poste':  
2 222}, {'nom': 'Eric', 'poste': 231}]
```

Quelle expression permet d'accéder au poste Eric ?

d. Quel est l'affichage dans la console au fur et à mesure des instructions suivantes ?

- script 1 :

```
1 capitales = {'France': 'Paris',  
2             'Italie': 'Rome',  
3             'Angleterre': 'Londres'}  
4 list(capitales.keys())  
5 # Affiche ....
```

- script 2 :

```
1 capitales['France']  
2 # Affiche ...
```

- script 3 :

```
1 list(capitales.values())  
2 # Affiche ...
```

- script 4 :

```
1 capitales['Espagne'] = 'Madrid'  
2 list(capitales.items())  
3 # Affiche ...
```

Problèmes

7.1 Des mathématiciens célèbres

Chaque lettre des noms des six mathématiciens a été remplacée par un nombre. Cette substitution suit une règle très spéciale. Pouvez-vous la percer ?

- 10 8 3 4 4 3 - 13 3 - 14 3 4 20 1 5
- 8 12 1 1 23 - 26 3 21 5 9 26
- 3 7 23 19 8 13 3
- 19 3 9 26 16 1 4 13 - 3 7 19 3 4
- 12 9 10 16 8 3 - 13 3 4 20 1 8 26
- 10 6 5 16 1 15 9 4 3

7.2 Lettres manquantes

a. Quelle séquence suivent les lettres qui ont été supprimées dans les mots suivants ?

- `_LM_N_CH`
- `_AO_A_`
- `_ON_O_TER`
- `_IVI_EN_E`
- `_L_M_NT`

b. Peut-on reconstruire ces chaînes grâce à un algorithme ?