

TYPES CONSTRUITS – EXERCICES CORRIGES

1. TUPLES ET LISTES (QCM):

a. Parmi les affirmations suivantes, lesquelles sont vraies ?

- Une liste peut contenir plusieurs éléments.
- On peut ajouter des éléments à un tuple.
- On peut modifier les éléments d'une liste.
- Un tuple peut contenir à la fois des nombres et des chaînes de caractères.

b. Si *liste* est la liste [1, 3, 5], quelles sont les opérations valides ?

- `liste.append(4)`
- `liste[0]`
- `liste[0] = 4`
- `liste[4] = 7`
- `liste = [1, 3, 10, 7, 3]`

c. Si *triplet* est le tuple (1, 3, 5), quelles sont les opérations valides ?

- `triplet.append(4)`
 - `triplet[0]`
 - `triplet[0] = 4`
- on ne peut pas modifier un tuple (non mutable)

2. CALCUL D'UN MOYENNE:

fichier 5-moyenne.py

À la fin du semestre, il est temps de calculer les moyennes de français avant l'édition des bulletins. Pronote exporte les notes de Marta et Tiago sous la forme d'une liste:

```
marta= [10, 15, 11, 18, 7, 12, 13, 16, 18]
tiago= [8, 11, 12, 14, 10, 9, 15, 12, 11,13,18]
```

Téléchargez et ouvrez le fichier python 5-moyenne.py. Le professeur a coché la case "arrondir la moyenne au dixième".

Le fichier 5-moyenne-correc.py est téléchargeable sur le site

- a. Compléter la fonction `moyenne(liste)` qui permet de calculer la moyenne d'un élève arrondie au dixième.

```
def moyenne(liste):
    somme=0 #initialisation de la somme
    for i in liste:
        somme+=i #calcul de la somme (équivalent à somme=somme+i)
    nb=len(liste) #nombre de notes
    moyenne=round(somme/nb,1) #moyenne arrondie au dixième
    return moyenne #retour de la valeur calculée
```

Appel des fonctions:

```
print("Moyenne de Marta:",moyenne(marta))
print("Moyenne de Tiago:",moyenne(tiago))
```

- b. Le professeur décide de cocher la case "enlever la plus mauvaise note". À l'aide de la fonction `indexMini(liste)`, affichez la moyenne corrigée de Marta et Tiago.

```
def indexMini(liste):
    """détermination de l'index
    de la plus mauvaise note"""
    return liste.index(min(liste))

#affichage des index de la plus mauvaise note, appel de la fonction
indexMini(liste)
print("index de la plus mauvaise note de Marta:",indexMini(marta))
print("index de la plus mauvaise note de Tiago:",indexMini(tiago))

#on la supprime par la fonction pop()
marta.pop(indexMini(marta))
tiago.pop(indexMini(tiago))

#affichage des moyennes corrigées par rappel de la fonction
moyenne()
print("Moyenne corrigée de Marta:",moyenne(marta))
print("Moyenne corrigée de Tiago:",moyenne(tiago))
```

3. DICTIONNAIRES (QCM):

```
dico = {"a": True, "b": False, "c": False}
```

a. Quelle est la valeur de `dico[1]` ?

- "a"
- True
- "b"
- False
- Rien car l'expression n'est pas valide.

b. Quelle est la valeur de `dico["a"]` ?

- True
- False
- Rien car l'expression n'est pas valide.

c. Quelle instruction permet de modifier le dictionnaire de façon à ce que sa nouvelle valeur soit {"a": True, "b": False, "c": False, "e": True}?

- `dico["e"] = True`
- `dico.append("e")`
- `dico.append("e", True)`

- Ce n'est pas possible car un dictionnaire n'est pas modifiable.

d. Quels sont les affichages possibles lors de l'exécution du code suivant ?

```
for i in dico.keys():  
    print(i, end=" ")
```

- a b c (a, True) (b, False) (c, False) True False False

La boucle « for » itère sur les clés du dictionnaire.

e. Quels sont les affichages possibles lors de l'exécution du code suivant ?

```
for i in dico.items():  
    print(i, end=" ")
```

- a b c (a, True) (b, False) (c, False) True False False

La boucle « for » itère sur les couples clés/valeurs du dictionnaire.

4. COMPREHENSIONS ET STRUCTURES IMBRIQUEES (QCM):

a. Si liste désigne la liste [1, [2, 3], [4, 5], 6, 7], que vaut len(liste) ?

- 1 3 5 7

b. Que vaut [2 * n for n in range(5)] ?

- [0, 2, 4, 6, 8] [0, 2, 4, 6, 8, 10] [0, 2, 4] Autre chose

c. Supposons que liste = [-5, 2, 3, -7, 42, 7]. Que vaut [n for n in liste if n > 0] ?

- [-5, 2, 3, -7, 42, 7] [2, 3, 42, 7] [False, True, True, False, True, True] Autre chose

Les éléments de la liste sont filtrés : seuls les éléments positifs seront présents dans la liste en compréhension.

5. LE CARRE MAGIQUE

Voir la correction sur le notebook ou le fichier Python

6. LA RECHERCHE DU MINIMUM

Voir l'animation sur le site: <http://www.fredpeuriere.com/NSI/algos-1/algorithmes-illustres/min.html>

```
def minimum (liste):  
    min=liste[0]  
    for i in range(len(liste)):  
        if liste[i]<min:  
            min=liste[i]  
    return (min,liste.index(min))
```

La fonction retourne le minimum de la liste et son indice (sous forme de tuple)