

```

#Les collections:
#liste l=[ 1,2] g0 d-1 mutables
#tuples t=(1,2) non mutables
#ensembles e={} pas doublons , mutables
#dictionnaires clé valeurs keys values ,mutables , cles immutables
#d={'k1':v1 , 'k2':v2 }
#d=dict([('k1',v1),('k2',v2)])

Joueur = ('A','B')
Score={'A':0,'B':0}
Tour=input('Qui va commencer ? A/B \t')
while(Tour.upper() not in 'AB'):
    Tour=input('Qui va commencer ? A/B \t')
print(Tour)
while(abs(Score['A']-Score['B'])<2):
    C=0
    while(C<20):
        print("C'est le tour du joueur",Tour.upper())
        x=input('Saisir un entier 1 ou 2 ou 3\t')
        while(x not in '123'):
            x=input('Saisir un entier 1 ou 2 ou 3\t')
        C+=int(x)
        print('Compteur =',C)
        if(Tour.upper()=='A'):
            Tour='B'
        else:
            Tour='A'
        Score[Tour.upper()]+=C
        ProchainTour=Joueur-{Tour.upper()}
        Tour=ProchainTour.pop()
    if(Score['A']==max[Score.value()]):
        print('Le gagnant est A')
    else:
        print('Le gagnant est B')

```

```

ch="""Les fonctions nous permettent de regrouper des lignes de code dans un mini-
programme appelé sous-programme.
Ensuite, chaque fois que nous avons besoin de l'utiliser, nous «appelons» simplem
ent cette fonction ; appeler une fonction signifie l'utiliser dans un autre code.
Généralement, une fonction prend une entrée et produit une sortie. Python offre d
eux catégories de fonction : les fonctions ordinaires et les fonctions lambda."""
alpha={'k', 'x', 'c', 'é', 'v', 'è', 'a', 'w', 'à', 'p', 'd', 'ù', 'g', 'm', 'b',
'l', 'h', 'o', 'r', 'q', 'u', 'i', 't', 'f', 'ç', 'n', 'î', 'ê', 'y', 's', 'j',
'z', 'ï', 'e'}
ponct={'', ' ', '?', '.', ':', ';', '!'}

# 1. Ecrivez le script python permettant d'afficher les caractères de ponctuation
utilisés dans **ch**.

e1=(set(ch.lower())-alpha)&ponct
print(e1)

# 2. Ecrivez le script python permettant d'afficher les lettres utilisées dans *
ch*. Ne différenciez pas les lettres majuscules des minuscules.

e2=set(ch.lower())&alpha
print(e2)

# 3. Ecrivez le script python permettant d'afficher les caractères utilisés dans
**ch** qui sont ni de ponctuation ni des lettres .

e3=set(ch.lower())-(alpha|ponct)
print(e3)

# 4. Afficher l'ensemble de mots utilisés dans ch et le nombre d'apparition de ch
aque mot.

l=ch.split()
print(l)
#supprimer la ponctuation à la fin de chaque mot str.strip([chars]) str.join(iterable)
#print (''.join(ponct)) #!?.:.;
#'?fghj;'.strip(''.join(ponct))
ll=[mot.strip(''.join(ponct)) for mot in l]
print(ll)
for i in range(ll.count('')):
    ll.remove('')
print(ll)
# 4.2 le nombre d'apparition de chaque mot.
ensMots=set(ll)
d=dict()
for m in ensMots:
    d[m]=ll.count(m)
print(d)

```

```

#listes
a=[1,2,3,4,5,6,7]
a.append(8) ----> a=[1,2,3,4,5,6,7,8]
a.extend([9,10]) ---->a=[1,2,3,4,5,6,7,8,9,10]
a.insert([0,0])----> a=[0,1,2,3,4,5,6,7,8,9,10]
a.remove(10)----->a=[0,1,2,3,4,5,6,7,8,9]
a.sort()----->a=[0,1,2,3,4,5,6,7,8,9] #trie
a.reverse()-----> #ma9louba
a.count(1) ----->1 #9adeh min marra mawjouda

#dict
monDictionnaire={'Nom' : 'Ben Abdallah', 'Prenom' : 'Abdallah', 'Age' : 20}
dic.items() : #traja3 (clé,valeur).
monDictionnaire.items()-----
>dict_items([('Nom', 'Ben Abdallah'), ('Prenom', 'Abdallah'), ('Age', 20)])
dic.keys() : #Renvoie le cle
monDictionnaire.keys()----> dict_keys(['Nom', 'Prenom', 'Age'])
dic.values() : #Renvoie les valeurs
monDictionnaire.values()---->dict_values(['Ben Abdallah', 'Abdallah', 20])
dic.pop(cle) : #Renvoie la valeur et supprime la paire clé/valeur(KeyError si la
clé est absente).

                #i7otou fi blasa okhra so iwali inexistant
resPop=monDictionnaire.pop('Age')----
>{'Nom': 'Ben Abdallah', 'Prenom': 'Abdallah'}

#ensembles
ens.add(elt) : #Ajoute un élément à l'ensembl.
ens.remove(elt) : #Retire un élément à un ensemble(KeyError si élément absent).
ens.clear() : #supprime tout les élément d'un ensemble.
ens1.difference(ens2) : #Renvoie l'ensemble des éléments de ens1 qui n'appartienn
ent pas à ens2.
ens1.issuperset(ens2) : #Renvoie True si ens1 contient ens2 E appartient
ens1.issubset(ens2) : #Renvoie True si ens1 est inclu dans ens2 C inclu
ens1.union(ens2) : #Renvoie l'union des deux ensembles. U union
ens1.intersection(ens2) : #Renvoie l'intersection des deux ensembles ens1 et ens2
. Intersect

len() #longueur
max() min() sum()
MaListe=[1,3,4,2.3,3.4]
print("Pour la liste {} max={} min={} sum={}",format(MaListe,max(MaListe),min(MaL
iste),sum(MaListe)))

----->Pour la liste [1, 3, 4, 2.3, 3.4] max=4 min=1 sum=13.700000000000001

```

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

```
#tuples
```

```
Days=set(["Mon","Tue","Wed","Thu","Fri","Sat","Sun"])
```

```
Months={"Jan","Feb","Mar"}
```

```
Dates={21,22,17}
```

```
print(Days)
```

```
print(Months)
```

```
print(Dates)
```

```
string.strip([chars]) ----> #supprime chars mil string
```

```
myTuple = ("John", "Peter", "Vicky")
```

```
x = "#".join(myTuple)
```

```
print(x) # equivalent lil extend fil liste
```

```
l.upper()= majuscule
```

```
l.lower()=miniscule
```

```
l.pop()---->next one
```

