

Session 6 - String construction -2022 - A.U. 23-24

November 29, 2023

Construction des chaines a partir des variable en utilisant:

- print
- concatenation
- .format(...)
- f string

```
[19]: nom='ahmed'
      age=18
      moy=15.5
      notes=[14,13.5,6,17.3]
      print(nom,age,'ans','mooyenne=',moy,'notes=',notes)
```

ahmed 18 ans mooyenne= 15.5 notes= [14, 13.5, 6, 17.3]

```
[10]: # toDisplay=nom+age+'ans moyenne='+moy+'notes='+notes
      toDisplay=nom+' '+str(age)+'ans moyenne='+str(moy)+'notes='+str(notes)
      print(toDisplay)

      listNom=['isra','asser','ahmed','kallel','mokhtar']
      for nom in listNom:
          toDisplay=nom+' '+str(age)+'ans moyenne='+str(moy)+'notes='+str(notes)
          print(toDisplay)
```

mokhtar18ans moyenne=(15, 5)notes=[14, 13.5, 6, 17.3]
isra18ans moyenne=(15, 5)notes=[14, 13.5, 6, 17.3]
asser18ans moyenne=(15, 5)notes=[14, 13.5, 6, 17.3]
ahmed18ans moyenne=(15, 5)notes=[14, 13.5, 6, 17.3]
kallel18ans moyenne=(15, 5)notes=[14, 13.5, 6, 17.3]
mokhtar18ans moyenne=(15, 5)notes=[14, 13.5, 6, 17.3]

```
[16]: '{} est il en relation avec {}'.format(3,4)
      '{} est il en relation avec {}'.format(3,[11,22,33],55)
```

```
[16]: '3 est il en relation avec [11, 22, 33]???'
```

```
[4]: listNom=['isra','asser','ahmed','kallel','mokhtar']
      for nom in listNom:
          print(nom,age,'ans','mooyenne=',moy,'notes=',notes)
```

```

isra 18 ans mooyenne= (15, 5) notes= [14, 13.5, 6, 17.3]
asser 18 ans mooyenne= (15, 5) notes= [14, 13.5, 6, 17.3]
ahmed 18 ans mooyenne= (15, 5) notes= [14, 13.5, 6, 17.3]
kallel 18 ans mooyenne= (15, 5) notes= [14, 13.5, 6, 17.3]
mokhtar 18 ans mooyenne= (15, 5) notes= [14, 13.5, 6, 17.3]

```

```

[34]: '{}-{}-{}-{}'.format(nom,age,moy,notes)
      '{} {}ans moyenne={} notes={} '.format(nom,age,moy,notes)

motif='{} {}ans moyenne={} notes={} '
listNom=['isra','asser','ahmed','kallel','mokhtar']
for nom in listNom:
    toDisp=motif.format(nom,age,moy,notes)
    print(toDisp)
motif.format(1+2,age,moy,notes)

```

```

isra 18ans moyenne=15.5 notes=[14, 13.5, 6, 17.3]
asser 18ans moyenne=15.5 notes=[14, 13.5, 6, 17.3]
ahmed 18ans moyenne=15.5 notes=[14, 13.5, 6, 17.3]
kallel 18ans moyenne=15.5 notes=[14, 13.5, 6, 17.3]
mokhtar 18ans moyenne=15.5 notes=[14, 13.5, 6, 17.3]

```

```

[34]: '3 18ans moyenne=15.5 notes=[14, 13.5, 6, 17.3]'

```

```

[30]: '{nom} {age}ans moyenne={moy} notes={notes}'
      f"{nom} {age}ans moyenne={moy} notes={notes}"
      f"{nom.upper()} {age}ans moyenne={moy} notes={notes}"
      f"{print(99)} {age}ans moyenne={moy} notes={notes}"

```

99

```

[30]: 'None 18ans moyenne=15.5 notes=[14, 13.5, 6, 17.3]'

```

Exercice fontion table_mult qui accepte n et affiche la table de multiplication de n (par plusieurs méthodes)

```

[3]: def table_mult1(n):
      for i in range(1,11):
          print(i,'X',n,'=',i*n)
      # table_mult1(5)

      def table_mult2(n):
          for i in range(1,11):
              line=str(i)+' X '+str(n)+' = '+str(i*n)
              print(line)

```

```

# table_mult2(15)

def table_mult3(n):
    # motif='{ } X { } = { }'
    motif='{ } multiplie par { } est egale a { }'
    for i in range(1,11):
        line=motif.format(i,n,i*n)
        print(line)

# table_mult3(44)

def table_mult4(n):
    for i in range(1,11):
        line=f"{i} multiplie par '{n}' est egale a {i*n}"
        print(line)

table_mult4('*')
# table_mult4(33)

```

```

1 multiplie par '*' est egale a *
2 multiplie par '*' est egale a **
3 multiplie par '*' est egale a ***
4 multiplie par '*' est egale a ****
5 multiplie par '*' est egale a *****
6 multiplie par '*' est egale a ****
7 multiplie par '*' est egale a *****
8 multiplie par '*' est egale a *****
9 multiplie par '*' est egale a *****
10 multiplie par '*' est egale a *****

```

[1]: *###1- Sans utiliser les fonctions pre_definies, écrire un script qui permet de*
↪ saisir ph qui commence par une lettre majuscule et se termine par un
↪ caractère spécial.

```

while True:
    ph=input('donner une chaîne selon nos critères')
    if 'A'<=ph[0]<='Z' and not('a'<=ph[-1]<='z' or 'A'<=ph[-1]<='Z' or
    ↪str(0)<=ph[-1]<=str(9)):
        break

```

###2- saisir une liste de mots tels que:
-chaque mot est de len>=4
-la saisi de mots se termine par l'entrée du mot 'end'.

```

list_mot=[]
i=0
while True:
    print('----mot indice=',len(list_mot)+1,'-----')

```

```

while True:
    mot = input('donner un mot de len >=4')
    if len(mot)>=4 or mot=='end':break

    if mot=='end':break
    else:list_mot.append(mot)

#3- remplacer les mot de list_mot dans ph par des '*' en respectant les longueur
new_ph=ph
for word in list_mot:
    new_ph=new_ph.replace(word,len(word)*'*')

#4- remplacer les sous-mot de list_mot dans ph par des '*' sauf pour le premier
    et dernier caractère, et en respectant les longueur
new_ph=ph
for word in list_mot:
    new_ph=new_ph.replace(word[1:-1],len(word[1:-1])*'*')
print(new_ph)

```

```

----mot indice= 1 -----
----mot indice= 2 -----
----mot indice= 3 -----
----mot indice= 4 -----
----mot indice= 5 -----
----mot indice= 6 -----

```

```

A***b sdfdfdsf a****f fdgdfgfdgdfg s***r fgdsfsd dsfsd fsdf gr*****n sfsdfdsfsd
fsdf.

```