

Python Séance 3 DEV102

December 27, 2023

- Variables simples (int, float) - Variables Collections "séquences" (Listes, String, Tuples, set, dictionnaires)

```
[2]: x = 1256
     y = 1000
     z = x ** y
     print(z)
```

```
97642614840045931933566539510234405727780914277077907000105989060303225733621027
15646611256594293242077193276184749507312304064425744471555317350326782247207469
29670507084189077348037600683030231192272974091433867707063297192284678855846845
60878585373819672285344476181550434885089028380540828881710762655455265346664449
00548352124591678762786378824690230448881032006647920818431785861969774665628058
21570368924182285398027684028156407463344780937028405943956089806960897148418158
29866655356481508514270682119388671982139870122963012504055928744890207161943881
42553374190278861288645540060259281291468788444071694648986343178476198561736542
53119725731335473484561997308489877654272909893158922260187942488346764853077308
83409593068421424666288773524031055026835866890842286204834958065313996647102788
66487536462024506631269449354754730175618824237457928106466444425681078794117972
37013624852543108381551641628019306346975996875181735817292644994549202387842538
33932179098151795384350598501006895787636051367137675089100743773346260121615817
01419998288548340503487299215789777143191600559088729858115130573692958749988414
75799267596934530001675816061477996465197981911744289262199451741156379038395439
9605212844502255887425314595069928319236736428050921747211352914041799295786432
06910873365888389349605707733094740746624126789261063666794615788292670101116449
98172002082608203751340024852276333842927966107779614498913567716140853182510788
18084374334520738458114133524077357601815591009244523817605355618162941687007690
87108521323145308513478035092894168595875119385558814972477510082832328739913514
89148113767815810058112421961725070593054389874528316764428205772906021614494419
44007751993699249150021312260660256269520700259866599015443315818715262751452464
97027491662370222252857218711507608667854477026359580940654579771054096518285551
78030487317352436812251666279161800534859794066578725603198011941054974552689314
16724484220814162030838892360255512113043061864862668814977920337664174531034393
92127065020674943366464754986534800620281947542642500252965278033695779249198952
77364540454038938275250551963346235766959948453420706605554613319264839318107519
42000356446249738947439081372225526634397462057925694940558455787825210613645002
30471825055583978578995207107322073719046378263813156951595722230443513931351020
38597844964277205682872158551338089600668511391547034513209216115806592323196121
52392317403474174260265247905175230479200255804955454632760626640744289023356166
```

```
97337962163915613657970093093808683666746863530320943746028360381469060657174917
41348798996920587424444737249108607559897661403425709255945162335570313066044770
74021293367407785198750524593210865702098775181290456088980742073711235572189815
15996213536608274806172106412639289077459456134470931409271414751154077615669471
61524901736984664080972612755846307862461091045519425284544537912582525291437947
05968535235466208305757573997789459862048848673384897064925200956225703355641042
90681859093484050192373385028788222579934854511878932146590138502065236053712596
75289629256225477443989734280215941108226200562030530789376
```

1 Tuples, Set.

A- Les tuples sont des listes immutables (non modifiables) sont délimités par (). Exemple : T1 = (1,2,6,9,11,"a") print(T1) T1 = (6,9,10) print(T1)

cette opération est possible car elle concerne une affectation et non une modification (mutabilité), la mutabilité est la modification d'une partie de la valeur.

les chaînes de caractères sont aussi immutables: S = "Ceci est un essai" S = "Bonjour" print(S[0]) S[0] = "a"

```
[3]: T1 = (1,2,6,9,11,"a")
      print(T1)
      T1 = (6,9,10)
      print(T1)
```

```
(1, 2, 6, 9, 11, 'a')
(6, 9, 10)
```

Exemple : Créer un tuple qui contient les valeurs de 1000 à 10000 avec un saut de 1000 et le trier.

```
[8]: T1 = tuple(range(10000, 999, -1000))
      print(T1)
      L1 = list(T1)
      L1.sort()
      T1 = tuple(L1)
      print(T1)
```

```
(10000, 9000, 8000, 7000, 6000, 5000, 4000, 3000, 2000, 1000)
(1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000)
```

```
[11]: T1 = tuple(range(10000, 999, -1000))
       print(T1)
       T1 = tuple(sorted(T1))
       print(T1)
```

```
(10000, 9000, 8000, 7000, 6000, 5000, 4000, 3000, 2000, 1000)
(1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000)
```

Destruction (destructuring)

```
[16]: T1 = (16,8,9)
      a,b = T1
      print(b)
```

8

B - Set : une série de valeurs non dupliquées, délimités par

Exemple : S1 = 1,2,1,2,3,16,-7,3 print(S1)

```
[20]: S1 = {1,2,1,2,3,16,-7,3}
      print(S1)
```

{16, 1, 2, 3, -7}

Exemple 2 : considérant la liste suivante : L1 = list(range(1,11)) L2 = list(range(5,20)) L3 = L1 + L2 le L3 contiendrait les valeurs de L1 et de L2, du coup, plusieurs valeurs seront dupliquées, utiliser set pour nettoyer les valeurs doubles.

```
[23]: L1 = list(range(1,11))
      L2 = list(range(5,20))
      L3 = list(set(L1 + L2))
      print(L3)
```

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]

2 Travail à faire :

Développer les différentes méthodes des SETs.

```
[24]: S1 = set(range(1,10))
      S1.
```

3 Dictionnaires

un dictionnaire estv un ensemble d'information concernant une occurrence (stagiaire, article, ...), correspond aux données structures. Exemple : nous désirons gérer les données d'un stagiaire (matricule, nom, prénom, age), les valeurs sont accédées par nom du champs et non l'indice

```
stagiaire = {"matricule" : 1, "nom" : "BOUJAOUANE", "pren" : "AZIZ", "age" : 18}
print(stagiaire["age"])
```

```
[28]: stagiaire = {"matricule" : 1, "nom" : "BOUJAOUANE", "pren" : "AZIZ", "age" : 18}
      print(stagiaire["age"])
      stagiaire["age"] = 17
      print(stagiaire["age"])
      stagiaire["fil"] = "DEV102"
      print(stagiaire)
```

18

17

```
{'matricule': 1, 'nom': 'BOUJAOUANE', 'pren': 'AZIZ', 'age': 17, 'fil':  
'DEV102'}
```

Exercice : Ecrire un programme qui permet de saisir les informations pour plusieurs stagiaires.

```
[32]: stagiaire = []  
  
while True :  
    mat = int(input("Saisir le matricule :"))  
    nom = input("Saisir le nom :")  
    prenom = input("Saisir le prénom :")  
    age = int(input("Saisir l'age :"))  
    st = {"matricule" : mat, "nom" : nom, "pren" : prenom, "age" : age}  
    stagiaire.append(st)  
    choix = input("Voulez vous continuer (o/n) :")  
    if choix == "n" :  
        break  
  
print(stagiaire)
```

```
Saisir le matricule :1  
Saisir le nom :ALOUANE  
Saisir le prénom :AMINE  
Saisir l'age :18  
Voulez vous continuer (o/n) :o  
Saisir le matricule :2  
Saisir le nom :MAARASS  
Saisir le prénom :SAYF  
Saisir l'age :23  
Voulez vous continuer (o/n) :o  
Saisir le matricule :3  
Saisir le nom :SALK  
Saisir le prénom :REDA  
Saisir l'age :18  
Voulez vous continuer (o/n) :0  
Saisir le matricule :4  
Saisir le nom :KARCHAOU  
Saisir le prénom :MED  
Saisir l'age :19  
Voulez vous continuer (o/n) :n  
[{'matricule': 1, 'nom': 'ALOUANE', 'pren': 'AMINE', 'age': 18}, {'matricule':  
2, 'nom': 'MAARASS', 'pren': 'SAYF', 'age': 23}, {'matricule': 3, 'nom': 'SALK',  
'pren': 'REDA', 'age': 18}, {'matricule': 4, 'nom': 'KARCHAOU', 'pren': 'MED',  
'age': 19}]
```

```
[33]: L1 = [1,1,9,-6, 90]  
  
for i in range(len(L1)) :
```

```
print(L1[i])
```

```
1
1
9
-6
90
```

```
[34]: for x in L1 :
      print(x)
```

```
1
1
9
-6
90
```

```
[27]: stagiaire = {"matricule" : 1, "nom" : "BOUJAOUANE", "pren" : "AZIZ", "age" : 18}
      stagiaire.keys()

      for element in stagiaire :
          print(element)
      print("\n"*3)
      for element in stagiaire.keys() :
          print(element)

      print("\n"*3)

      print(stagiaire.values())

      for valeur in stagiaire.values() :
          print(valeur)

      print("\n"*3)

      for elem in stagiaire.keys() :
          print(elem , ": ", stagiaire[elem])
```

```
matricule
nom
pren
age
```

```
matricule
nom
pren
```

age

```
dict_values([1, 'BOUJAOUANE', 'AZIZ', 18])
1
BOUJAOUANE
AZIZ
18
```

```
matricule : 1
nom : BOUJAOUANE
pren : AZIZ
age : 18
```

```
[35]: chaine1 = "ceci est un essai"
print(len(chaine1))
print(chaine1.capitalize())
L10 = chaine1.split()
print(len(L10))
L10 = chaine1.split("e")
print(L10)
```

```
17
Ceci est un essai
4
['c', 'ci ', 'st un ', 'ssai']
```

```
[47]: chaine1 = "Bonjour tout le monde"
print(chaine1)
print(chaine1[1])

# Slicing :
print(chaine1[3:7])
print(chaine1[3:10:2])
print(chaine1[-1])
print(chaine1[2:])
print(chaine1[0:])
print(chaine1[-1::-1])
```

```
Bonjour tout le monde
o
jour
ju o
e
```

njour tout le monde
Bonjour tout le monde
ednom el tuot ruojnoB

```
[52]: nom_testé = input("Saisir une chaine :")

if nom_testé.upper() == nom_testé[-1::-1].upper() :
    print("Palindrôme")
else :
    print("Non Palindrôme")
```

Saisir une chaine :Radar
Palindrôme

[]: