STAV Python

Données

Type de données

Affectation de variable

nombre = 7
diviseur=5
calc=nombre/diviseur

Saisie utilisateur

input("Texte de la demande")
Attention, input génère par défaut une variable de <class 'str'>
exemple:
a=input("Saisir votre réponse texte:")
b=float(input("Saisir le résultat numérique décimale:"))
c=int(input("Saisir le résultat en valeur entière:"))

Entier

a=23
type(a) => renvoi <class 'int'> pour nombre entier

Nombre à virgule

a=23.542
type(a) => renvoi <class 'float'> pour nombre en virgule flottante 23542x10^-3

Chaine de caractères

a="Ceci est une chaine de caractère"
type(a) => renvoi <class 'str'> pour string = chaine de caractères

Liste de données

a=[2,43,"AZE",56]
type(a) => renvoi <class 'list'>

Liste

Permet la mémorisation des données
symbole [ ]

Création d'une liste vide

maliste=[]
toto=[]

Ajout de données dans une liste

maliste.append(12)
toto.append("ABC")

Appel d'une données d'une liste

toto=[1,3,"ABC"]
toto[0] => 1er élément de la liste toto
donc
print(toto[2]) => affiche le 3eme élément de la liste toto
donc le texte ABC

Calculs

Opérateurs

Addition +

entier, réel chaîne de caractères
6+4 => 10
"a" + "b" => "ab"

Soustraction -

entier, réel
6-4 => 2

Multiplication *

entier, réel, chaîne de caractères
6*4 => 24
1.2 * 1 => 1.2
3 * "s" => "sss"

Puissance **

entier, réel
12**2 => 144

Division /

entier, réel
6/4 => 1.5

Division entière //

Renvoi le quotient entier de la division euclidienne
entier, réel
6//4 => 1

Modulo %

Renvoi le reste entier de la division euclidienne
entier, réel
6%4 => 2

Structure séquentielles

Boucle bornée

Exemple

```
for i in range(10):
    print(i**2)
```

```
for.... :
    indentation
```

Boucle non bornée

Exemple

```
max=10
i=0
while i <= max:
    print(i)
```

```
while... :
    indentation
```

Test

Opérateurs de comparaison

if a>10: => si a sup à 10
if a<=0: => si a inf ou égal à 0
if a!=3: => si a différent de 3
if a=="abc": => si a égal à "abc"

Exemple

```
a=1
if a>=4:
    print("a plus grand ou égal à 4")
elif a>=2:
    print("a est entre 4 et 2 inclu")
else:
    print("a strictement plus petit que 2")
```

```
if .... :
    instruction
elif ....:
    instruction
else:
    instruction
```

Définition de fonction

Exemple

```
def volume_cube(rayon):
    volume=rayon**3
    return(volume)

print(volume_cube(5)) => renvoi 125 = 5*5*5
```

```
def nom(paramètres):
    instruction
    instruction
    return(résultat)
```

Bibliothèque

Appel des fonctions des bibliothèque avec
import BIBLIO? as INITIALES?
ou
from BIBLIO? import FONCTION?

random

Fonctions aléatoires

```
import random as rd
from random import *
from random import randint
```

matplotlib

Création de graphique

```
import matplotlib.pyplot as plt
```

math

Fonctions de maths

```
from math import pi
import math as m
```

Affichage dans la console

Afficher un texte

```
print("Texte")
```

Afficher une variable

```
n=4
print(n)
```

Afficher du texte et une variable

```
n=5
print("ceci est la valeur de n=",n)
```

Affichage graphique

Exemple

```
import numpy as np
import matplotlib.pyplot as plt

x=np.linspace(-2,6,100) #genere un tableau de donnees de 100 valeurs allant de -2 à 6
y=x**2

plt.axis([-3,10,-1,100]) #param axe x (entre -3 et 10) et axe y (entre -1 et 100)
plt.plot(x,y) #trace les points x,y
plt.grid() #affiche la grille
plt.show() #affiche le graphe
```