Sesión_1

April 7, 2025

[ ]:
```
x=int(input("Ingrese el valor de x "))
y=int(input("Ingrese el valor de y "))
if x>0:
    z=y*2
    print("Positivo")
elif x==0:
    z=y*3
    print("Cero")
else:
    z=y*4
    print("Negativo")
print(z)
```

```
Ingrese el valor de x 5
Ingrese el valor de y 6
Positivo
12
```

[ ]:
```
for i in range(10):
    print(i)
```

```
0
1
2
3
4
5
6
7
8
9
```

[ ]:
```
for i in [0,1,2,3,4,5,6,7,8,9]:
    print(i)
```

```
0
1
2
3
```

```
4
5
6
7
8
9
```

[ ]:
```
list(range(10))
```

[ ]:
```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

[ ]:
```
for i in range(1,10):
    print(i)
```

```
1
2
3
4
5
6
7
8
9
```

[ ]:
```
for i in range(1,10,2):
    print(i)
```

```
1
3
5
7
9
```

[ ]:
```
for x in range(1,11):
    if x<=5:
        print(f'{x} -> {x*2}')
    else:
        print(f'{x} -> {x*3}')
```

```
1 -> 2
2 -> 4
3 -> 6
4 -> 8
5 -> 10
6 -> 18
7 -> 21
8 -> 24
9 -> 27
10 -> 30
```

[ ]:
```
i = 1
while i <= 10:
    print(i)
    i=i+1
```

```
1
2
3
4
5
6
7
8
9
10
```

[ ]:
```
lista = ["manzana", "plátano", "naranja", "mandarina", "maracuyá"]
print(lista)
```

```
['manzana', 'plátano', 'naranja', 'mandarina', 'maracuyá']
```

[ ]:
```
print(lista[0])
```

```
manzana
```

[ ]:
```
print(lista[2])
```

```
naranja
```

[ ]:
```
print(lista[-1])
```

```
maracuyá
```

[ ]:
```
print(lista[-2])
```

```
mandarina
```

[ ]:
```
lista = ["manzana", "plátano", "naranja", "mandarina", "maracuyá", "toronja",␣
  ↪"mango", "guayaba"]
print(lista[2:6])
```

```
['naranja', 'mandarina', 'maracuyá', 'toronja']
```

[ ]:
```
print(lista[:6])
```

```
['manzana', 'plátano', 'naranja', 'mandarina', 'maracuyá', 'toronja']
```

[ ]:
```
print(lista[2:])
```

```
['naranja', 'mandarina', 'maracuyá', 'toronja', 'mango', 'guayaba']
```

[ ]:
```
print(lista[-1:-3])
```

```
[]
```

[ ]:
```
print(lista[-3:-1])
```

```
['toronja', 'mango']
```

[ ]:
```
print(lista[-1])
print(lista[len(lista)-1])
```

```
guayaba
guayaba
```

1 Factorial de un número

$$n! = 1 * 2 * 3 * 4 * ... * n$$

$$0! = 1$$

[ ]:
```
def factorialC(n):
    if n>=0:
        f = 1   # Acumulador
        for i in range(1,n+1):
            f = f*i
        return f
    else:
        print("Número con capacidades diferentes.")
        return None
```

[ ]:
```
x = 50
print(f'{x}! = {factorialC(x)}')
```

```
50! = 30414093201713378043612608166064768844377641568960512000000000000
```

2 Factorial en forma recursiva

$$n! = n * (n-1)!$$

$$5! = 5 * 4!$$

$$4! = 4 * 3!$$

$$3! = 3 * 2!$$

$$2! = 2 * 1!$$

$$1! = 1 * 0!$$

[ ]:
```
def factorialR(n):
    if n>0:
        f = n*factorialR(n-1)
        return f
    elif n==0:
```

```
        return 1
    else:
        return None
```

[ ]:
```
x = 959
print(f'{x}! = {factorialR(x)}')
```

```
---------------------------------------------------------------------------
RecursionError                            Traceback (most recent call last)
<ipython-input-3-4d27d0de8375> in <cell line: 2>()
      1 x = 959
----> 2 print(f'{x}! = {factorialR(x)}')

<ipython-input-1-5936c919726c> in factorialR(n)
      1 def factorialR(n):
      2     if n>0:
----> 3         f = n*factorialR(n-1)
      4         return f
      5     elif n==0:

… last 1 frames repeated, from the frame below …

<ipython-input-1-5936c919726c> in factorialR(n)
      1 def factorialR(n):
      2     if n>0:
----> 3         f = n*factorialR(n-1)
      4         return f
      5     elif n==0:

RecursionError: maximum recursion depth exceeded in comparison
```

3 Numpy

[ ]:
```
import numpy as np
arreglo = np.array([1,2,3,4,5])
print(arreglo)
```

```
[1 2 3 4 5]
```

[ ]:
```
lista = [1,2,3,4,5]
print(lista)
```

```
[1, 2, 3, 4, 5]
```

[ ]:
```
arreglo2 = np.array([[1,2,3],[4,5,6]])
print(arreglo2)
```

```
[[1 2 3]
 [4 5 6]]
```

[ ]:
```
print(arreglo2.ndim)
```

```
2
```

[ ]:
```
arreglo = np.array([[1,2,3,4,5], [6,7,8,9,10]])
print(arreglo[1, -1])
```

```
10
```

[ ]:
```
print(arreglo[-1,-1])
```

```
10
```

[ ]:
```
import numpy as np
arreglo = np.array([1, 2, 3, 4, 5])
arreglo2 = arreglo[1:4]
print(arreglo2)
```

```
[2 3 4]
```

[ ]:
```
print(arreglo2[-3:-1])
```

```
[2 3]
```

[ ]:
```
print(arreglo2[-1:-3])
```

```
[]
```

[ ]:
```
import numpy as np
arreglo = np.array([1, 2, 3, 4, 5, 6, 7])
print(arreglo[1:5:2])
```

```
[2 4]
```

[ ]:
```
import numpy as np
arreglo = np.array([[1, 2, 3, 4, 5], [6, 7, 8, 9, 10]])
print(arreglo[1, 1:4])
```

```
[7 8 9]
```

[ ]:
```
print(arreglo[0:2, 2])
```

```
[3 8]
```

[ ]:
```
print(arreglo[0:2, 1:4])
```

```
[[2 3 4]
 [7 8 9]]
```

[ ]:
```
import numpy as np
arreglo = np.random.rand(20)
print(arreglo)
```

```
[0.59793361 0.29295938 0.35648778 0.82583401 0.78469675 0.77503944
 0.71534854 0.29386111 0.36856962 0.04229477 0.65596469 0.78744518
 0.35859627 0.18873221 0.78112778 0.79282017 0.89929627 0.29884512
 0.93715576 0.77075094]
```

[ ]:
```
import numpy as np
arreglo = np.random.rand(5, 3)
print(arreglo)
```

```
[[0.50192316 0.25062811 0.44988926]
 [0.90652375 0.3850137  0.30640688]
 [0.94742379 0.27972452 0.58919275]
 [0.37998488 0.82158526 0.72533803]
 [0.30242836 0.02173225 0.52773112]]
```

[ ]:
```
import numpy as np
arreglo = np.random.rand(5, 3, 2)
print(arreglo)
```

```
[[[0.82325199 0.7921431 ]
  [0.39365942 0.62783579]
  [0.61361481 0.37772566]]

 [[0.42484743 0.49698194]
  [0.09515008 0.64945657]
  [0.09585235 0.03699993]]

 [[0.69569629 0.45804238]
  [0.57085378 0.9454701 ]
  [0.98855115 0.82840778]]

 [[0.67209852 0.28690747]
  [0.96603543 0.07726727]
  [0.93845423 0.98226327]]

 [[0.98816228 0.6032631 ]
  [0.24484934 0.8747269 ]
  [0.76031158 0.91379241]]]
```

[ ]:
```
import numpy as np
arreglo = np.append([[1, 2], [3, 4]], [[10, 20], [30, 40]], axis=0)
print(arreglo)
```

```
[[ 1  2]
 [ 3  4]
```

```
 [10 20]
 [30 40]]
```

[ ]:
```
import numpy as np
arreglo = np.append([[1, 2], [3, 4]], [[10, 20], [30, 40]], axis=1)
print(arreglo)
```

```
[[ 1  2 10 20]
 [ 3  4 30 40]]
```

[ ]:
```
import numpy as np
x = np.random.rand(10,5)
with np.printoptions(precision=4, suppress=True):
    np.set_printoptions(formatter={'float': '{: 0.4f}'.format})
    print(x)
```

```
[[ 0.4960  0.9387  0.0350  0.9758  0.9267]
 [ 0.8688  0.8008  0.6789  0.5612  0.0800]
 [ 0.3515  0.9020  0.6711  0.7909  0.4025]
 [ 0.7650  0.7511  0.2193  0.4735  0.3603]
 [ 0.0186  0.2610  0.0011  0.7906  0.4687]
 [ 0.5963  0.0567  0.9673  0.2778  0.9401]
 [ 0.8704  0.9889  0.5275  0.8812  0.3850]
 [ 0.9818  0.2240  0.3783  0.6337  0.5070]
 [ 0.4583  0.4164  0.8477  0.6289  0.7336]
 [ 0.7316  0.2147  0.3381  0.6614  0.0801]]
```

[ ]:
```
import numpy as np
arreglo = np.random.rand(5, 3, 2)

for x in arreglo:
    for y in x:
        for z in y:
            print(z)
```

```
0.8464264999580259
0.5396161732232413
0.365906351214646
0.008993690107141017
0.6330264971907361
0.9768938230773706
0.4910709345522991
0.36692047962927965
0.5654468195284397
0.5366801744120243
0.856898982840132
0.6836519323083912
0.7089953055403663
0.7735367533845893
```

```
0.16600400073162125
0.9673297992138653
0.5871654462537841
0.6204051975522569
0.11681287200327539
0.7704761055026805
0.007820549921977338
0.8770650422212877
0.5845047095639851
0.23689121480259667
0.32915005091904315
0.23666702264245765
0.7858862284321182
0.13310912686904663
0.5755517718561979
0.8747424571846256
```

4 Listas

[ ]:
```
lista = ["fresa", "maracuyá", "mango", "pera"]
print(lista)
```

```
['fresa', 'maracuyá', 'mango', 'pera']
```

[ ]:
```
lista.append(3.5)
print(lista)
```

```
['fresa', 'maracuyá', 'mango', 'pera', 3.5]
```

[ ]:
```
lista2 = [1,2,3,4,5]
lista3 = ['a', 'b', 'c']
```

[ ]:
```
lista.append(lista2)
lista.append(lista3)
```

[ ]:
```
print(lista)
```

```
['fresa', 'maracuyá', 'mango', 'pera', 3.5, [1, 2, 3, 4, 5], ['a', 'b', 'c']]
```

[ ]:
```
megaLista = [lista, lista2, lista3]
```

[ ]:
```
print(megaLista)
```

```
[['fresa', 'maracuyá', 'mango', 'pera', 3.5, [1, 2, 3, 4, 5], ['a', 'b', 'c']],
[1, 2, 3, 4, 5], ['a', 'b', 'c']]
```

[ ]:
```
for x in megaLista:
    for y in x:
        print(y)
```

```
fresa
maracuyá
mango
pera
3.5
[1, 2, 3, 4, 5]
['a', 'b', 'c']
1
2
3
4
5
a
b
c
```

[ ]: