

## REVISION

Quel sera l'affichage obtenu après l'exécution des codes suivants ?

```
[1]: x = 2  
y = 3  
x = y  
y = x  
print(x, y)
```

- A) x = 2 et y = 2
- B) x = 3 et y = 2
- C) x = 3 et y = 3
- D) x = 2 et y = 3

```
[2]: x = 2  
y = 3  
z = x  
x = y  
y = z  
print(x, y)
```

- A) x = 2 et y = 2
- B) x = 3 et y = 2
- C) x = 3 et y = 3
- D) x = 2 et y = 3

```
[3]: x = 2  
y = 3  
x, y = y, x  
print(x, y)
```

- A) x = 2 et y = 2
- B) x = 3 et y = 2
- C) x = 3 et y = 3
- D) x = 2 et y = 3

```
[4]: 3 * 0.1 == 0.3
```

- A) True
- B) False

```
[5]: print(3 + 5.0)
```

- A) 8
- B) 8.0
- C) 15
- D) 15.0

```
[6]: a = -0.5  
res = (a <= 0) or (a > 1 and a < 3 and a != 2)  
print(res)
```

- A) True
- B) False

```
[7]: s1 = "hello"
      s2 = 'world'
      print(s1 + s2)
```

- A) "hello world"
- B) hello world
- C) helloworld
- D) "helloworld"

```
[8]: s1 = "hello"
      s2 = 'world'
      print(s1, s2)
```

- A) "hello world"
- B) hello world
- C) helloworld
- D) "helloworld"

```
[9]: s1 = "hello"
      print(len(s1))
```

- A) 5
- B) 6
- C) 7

```
[10]: s1 = "hello"
        s2 = 'world'
        print(s1 > s2)
```

- A) True
- B) False

```
[11]: s1 = "hello"
        s2 = 'world'
        print(s1 in s2)
```

- A) True
- B) False

```
[12]: s1 = "hello"
        print(s1.upper())
```

- A) Hello
- B) HELLO
- C) "Hello"
- D) "HELLO"

```
[13]: print("hello" + 3)
```

- A) Hello3
- B) "Hello3"
- C) TypeError

[14]: `print("hello" * 3)`

- A) Hello333
- B) hellohellohello
- C) TypeError
- D) hello\*3

[15]: `print(type(5 + 2.0))`

- A) 7
- B) 7.0
- C) <class 'float'>
- D) <class 'int'>

[16]: `s = input("entrer un nombre entier\n")  
# s = 5  
print(s - 1)`

- A) 4
- B) TypeError

[17]: `x = 5  
if x > 10:  
 print("x est plus grand que 10")  
else:  
 print("x n'est pas plus grand que 10")`

- A) x est plus grand que 10
- B) x n'est pas plus grand que 10
- C) x est égal à 10
- D) Une erreur sera générée

[18]: `x = "a"  
if x == "a" :  
 print("A")  
 print("B")  
print("C")`

- A) A
- B) A et B
- C) C
- D) B et C
- E) A et B et C
- F) A et C

[19]: `x = "b"  
if x == "a" :  
 print("A")  
 print("B")  
print("C")`

- A) A
- B) A et B
- C) C
- D) B et C
- E) A et B et C
- F) A et C

```
[20]: x = "b"
if x == "a":
    print("A")
    print("B")
else:
    print("C")
    print("D")
print("E")
```

- A) A et B
- B) A et B et E
- C) C et D et E
- D) C et D
- E) E

```
[21]: x = "b"
if x == "a" :
    print("A")
    print("B")
elif x == "b" :
    print("C")
    print("D")
else :
    print("E")
    print("F")
print("Z")
```

- A) A et B
- B) A et B et Z
- C) C et D et Z
- D) E et F
- E) E et F et Z

```
[22]: x = "b"
x = "a"
if x == "b" :
    print("A")
    print("B")
elif x == "a" :
    print("C")
    print("D")
else :
    print("L")
print("M")
```

- A) A et B et M
- B) C et D et M
- C) L
- D) L et M

```
[23]: x = "a"
if x == "b" :
    print("A")
    print("B")
else :
    print("C")
    print("D")

if x == "a" :
    print("X")
    print("Y")
```

- A) A et B
- B) C et D
- C) C et D et X et Y
- D) A et B et X et Y

```
[24]: x = "a"
if x == "b" or x == "a" :
    print("A")
```

- A) A
- B) Aucun affichage

```
[25]: x = 0
if x != 0 and 2 // x == 2 :
    print("Test")
```

- A) Test
- B) Aucun affichage
- C) ZeroDivisionError

```
[26]: x = 0
if 2 // x == 2 and x != 0:
    print("Test")
```

- A) Test
- B) Aucun affichage
- C) ZeroDivisionError

```
[27]: x = 0
if x == 0 or 2 // x == 2:
    print("Test")
```

- A) Test
- B) Aucun affichage
- C) ZeroDivisionError

```
[28]: x = 0
      if 2 // x == 2 or x == 0:
          print("Test")
```

- A) Test
- B) Aucun affichage
- C) ZeroDivisionError

```
[29]: n = int(input("Tapez la valeur de l'entier n "))
      for i in range(1, n+1):
          if n % i == 0:
              print(i)
```

- A) Affiche tous les nombres impairs inférieur à n
- B) Affiche tous les nombres premiers inférieur à n
- C) Affiche tous les multiples de n
- D) Affiche tous les diviseurs de n

```
[30]: ch= 'TESTISN'
      r= ''
      for c in ch:
          r = r + c
      print (r)
```

- A) TESTISN
- B) NSITSET
- C) un message d'erreur
- D) aucune des réponses précédentes

```
[31]: somme = 0
      for n in range(5):
          somme = somme + n**2
      print (somme)
```

- A) 0/1/4/9/16
- B) 55
- C) 0/1/5/14/30
- D) 30

```
[32]: for i in range(1, 11) :
      print(i)
```

- A) Affiche tous les nombres de 0 à 11
- B) Affiche tous les nombres de 1 à 11
- C) Affiche tous les nombres de 0 à 10
- D) Affiche tous les nombres de 1 à 10

```
[33]: for i in range(11) :
      print(i)
```

- A) Affiche tous les nombres de 0 à 11
- B) Affiche tous les nombres de 1 à 11
- C) Affiche tous les nombres de 0 à 10
- D) Affiche tous les nombres de 1 à 10

```
[34]: for i in range(3, 9, 2) :  
      print(i)
```

- A) 0, 3, 5, 7, 9
- B) 1, 3, 5, 7, 9
- C) 3, 5, 7, 9
- D) 3, 5, 7

```
[35]: for i in range(4, 1, -1) :  
      print(i)
```

- A) 4, 3, 2, 1, 0
- B) 4, 3, 2, 1
- C) 4, 3, 2
- D) 4, 3

```
[36]: n = input("Entrez un nombre :")  
      #n = 5  
      result = 0  
      for i in range(1, n+1):  
          result += i  
      print(result)
```

- A) 15
- B) 0
- C) "5"
- D) 5
- E) TypeError

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

- A) Affiche tous les nombres de 1 à 10
- B) Affiche tous les nombres de 1 à 9

```
[38]: for i in range(1, 10) :  
      if i % 2 == 0 :  
          continue  
          print("A")  
      print(i)  
      print("B")
```

- A) Affiche tous les nombres impairs de 1 à 9 compris et B
- B) Affiche tous les nombres impairs de 1 à 10 compris et B
- C) Affiche tous les nombres impairs de 1 à 9 compris et A
- D) Affiche tous les nombres impairs de 1 à 9 compris et A et B
- E) Affiche tous les nombres pairs de 1 à 9 compris et B
- F) Affiche tous les nombres pairs de 1 à 9 compris et A et B

```
[39]: for i in range(1, 10) :  
        if i % 2 == 0 :  
            break  
        print(i)  
print("B")
```

- A) Affiche tous les nombres impairs de 1 à 9 compris et B
- B) Affiche tous les nombres impairs de 1 à 10 compris et B
- C) Affiche 1
- D) Affiche B
- E) Affiche 1 et B

```
[40]: for i in range(100):  
        print(i)  
        if i == 7:  
            break
```

- A) Affiche tous les nombres de 0 à 100 qui sont divisible par 7
- B) Affiche tous les nombres de 0 à 7
- C) Aucun affichage
- D) Affiche 7
- E) Affiche tous les nombres de 0 à 99 qui sont divisible par 7