Bing

Bien sûr, voici un autre QCM de 10 questions sur les opérations en Python, avec des réponses et des feedbacks. Bonne chance!

- Quel est le résultat de l'expression suivante?

[python]

p = 10

q = 3

r = p / q

print(r)

A) 3

B) 3.0

C) 3.3333333333333335

D) 3.33

Réponse: C) 3.3333333333333335

Feedback: L'opérateur / signifie la division réelle, qui renvoie un nombre à virgule flottante même si les deux opérandes sont des entiers. Donc, p / q vaut 10 / 3, qui est environ 3.3333333333333335 en Python. Si on voulait obtenir une valeur arrondie à deux décimales, on pourrait utiliser la fonction round(), comme round(p / q, 2).

- Quel est le résultat de l'expression suivante?

[python]

x = True

y = False

z = x and y or not x and not y

print(z)

A) True

B) False

C) None

D) SyntaxError

Réponse: B) False

Feedback: Les opérateurs and, or et not sont des opérateurs logiques, qui s'appliquent à des valeurs booléennes (True ou False). L'opérateur and renvoie True si les deux opérandes sont True, sinon il renvoie False. L'opérateur or renvoie True si au moins un des opérandes est True, sinon il renvoie False. L'opérateur not renvoie l'inverse de son opérande. L'ordre de priorité de ces opérateurs est not > and > or, donc on commence par évaluer les expressions avec not, puis celles avec and, puis celle avec or. Donc, z vaut (x and y) or (not x and not y), qui vaut (True and False) or (False and True), qui vaut False or False, qui vaut False.

- Quel est le résultat de l'expression suivante?

[python]

s = "Hello"

t = s[1:4:2]

print(t)

A) He

B) el

C) Hl

D) eo

Réponse: D) eo

Feedback: L'opérateur [::] signifie la découpe d'une séquence par un début, une fin et un pas. Donc, s[1:4:2] renvoie les éléments de s de l'indice 1 inclus à l'indice 4 exclu, en sautant un élément sur deux. Les indices commencent à 0, donc s[1] vaut "e" et s[4] vaut "o". En sautant un élément sur deux, on obtient "eo".