Python (1)

Variables, conditions et expressions

Exercice 0 *QCM – vérification des prérequis* Pour chaque question, une seule réponse est correcte.

1. Après l'exécution de `x = 3`, que contient la variable `x` ?

— A. La chaîne `"3"`

— B. L'entier `3`

— C. Le flottant `3.0`

— D. Rien, car `x` n'a pas été déclarée

2. Quelle est la valeur de `y` après l'exécution du code suivant ?

```
x = 5
y = x + 1
x = 10
```

— A. `11`

— B. `6`

— C. `10`

— D. `5`

3. Que produit `print(type(3 / 1))` ?

— A. `<class 'int'>`

— B. `<class 'float'>`

— C. `<class 'str'>`

— D. Une erreur

4. Quel est l'affichage produit par le code suivant ?

```
a = 2
b = 3
a = b
b = a
print(f"{a} {b}")
```

— A. `2 3`

— B. `3 2`

— C. `3 3`

— D. `2 2`

Exercice 1 *Trace d'exécution* On exécute le programme suivant :

```
prix = 12
quantite = 3
total = prix * quantite
prix = 15
quantite = quantite + 1
print(f"{total} {quantite}")
```

1. Recopier et compléter un tableau qui donne, après chaque ligne, la valeur de `prix`, de `quantite` et de `total`.
2. Qu'affiche le programme ? La valeur de `total` tient-elle compte du nouveau prix ?

Exercice 2 *Évaluation d'expressions* Après les deux affectations ci-dessous, donner la valeur et le type de chacune des expressions qui suivent, sans exécuter le code, puis vérifier.

```
a = 7
b = 2
a + b
a * b - 1
a // b
a % b
a / b
a > b
a % 2 == 1
a > 10 or b < 5
a > 10 and b < 5
not a == b
```

Exercice 3 *Instructions et expressions* Pour chaque ligne, dire s'il s'agit d'une **instruction** ou d'une **expression**, et justifier.

```
n
n = 4
n = n * 2
n * 2
print(n)
n > 3
n > 3 and n < 10
"bon" + "jour"
```

Exercice 4 *Indentation et conditions* Un cinéma fait payer 6 € aux moins de 18 ans et 10 € aux autres ; les adultes reçoivent en plus un bon de réduction de 2 € pour leur prochaine séance. Voici deux programmes qui ne diffèrent que par l'indentation d'une ligne.

Script 1

```
age = 15
tarif = 10
reduction = 0
if age < 18:
    tarif = 6
```

```
else:
    tarif = 10
    reduction = 2
print(f"{tarif} {reduction}")
```

Script 2

```
age = 15
tarif = 10
reduction = 0
if age < 18:
    tarif = 6
else:
    tarif = 10
reduction = 2
print(f"{tarif} {reduction}")
```

1. Qu'affiche chaque script ?
2. Mêmes questions en remplaçant `age = 15` par `age = 20`.
3. Lequel des deux scripts respecte la règle du cinéma ?

Exercice 5 *Répartition en groupes* Un professeur veut répartir sa classe en groupes de même taille. Écrire un programme qui demande le nombre d'élèves de la classe, puis le nombre d'élèves par groupe, et qui affiche :

— soit que tous les groupes sont complets, avec le nombre de groupes ;

— soit le nombre de groupes complets et le nombre d'élèves qui restent.

Par exemple, 30 élèves par groupes de 4 donnent 7 groupes complets et 2 élèves restants.

Exercice 6 *Noms de variables*

1. Que calcule le programme suivant ? Les noms des variables aident-ils à le comprendre ?

```
p = float(input())
q = int(input())
r = p * q
if q >= 10:
    r = r * 0.9
print(r)
```

2. Réécrire ce programme avec des noms de variables explicites, en indiquant à l'utilisateur ce qu'il doit saisir et en affichant un message clair.
3. Ajouter un commentaire (ligne commençant par `#`) qui explique le rôle de la condition.

Exercice 7 *Initialisation d'une variable*

1. Une élève note dans un programme ce qu'elle met de côté chaque semaine :

```
epargne = epargne + 20
epargne = epargne + 35
epargne = epargne + 15
print(epargne)
```

Exécuter ce programme. Quelle erreur Python signale-t-il ? La corriger, puis donner le montant affiché.

2. On veut maintenant calculer 2^5, c'est-à-dire $2 \times 2 \times 2 \times 2 \times 2$, en multipliant cinq fois une variable `puissance` par 2. Par quelle valeur faut-il initialiser `puissance` ? Écrire le programme.

Exercice 8 *Tirages au hasard* Le module `random` fournit des nombres tirés au hasard. Après `import random` :

— `random.randint(1, 6)` renvoie un entier au hasard entre 1 et 6, bornes comprises ;

— `random.random()` renvoie un flottant au hasard entre 0 (compris) et 1 (exclu).

1. Écrire un programme qui simule le lancer d'un dé à six faces, affiche la valeur obtenue, puis affiche `Gagné !` si c'est un 6 et `Perdu` sinon.
2. Avec `random.random()`, écrire un programme qui affiche `Bonus` avec une chance sur quatre, et rien sinon.

Exercice 9 *Plusieurs cas avec elif* Écrire un programme qui demande une température en degrés Celsius (elle peut être décimale) et affiche :

— `gel` si elle est strictement inférieure à 0 ;

— `froid` si elle est comprise entre 0 (inclus) et 15 (exclu) ;

— `doux` si elle est comprise entre 15 (inclus) et 25 (exclu) ;

— `chaud` si elle vaut 25 ou plus.

Exercice 10 *Appartenance à un intervalle* Une boulangerie est ouverte de 7 h à 19 h : elle ouvre à 7 h et ferme à 19 h. Écrire un programme qui demande une heure, éventuellement décimale (`13.5` pour 13 h 30), et affiche si la boulangerie est ouverte à cette heure-là.

Écrire trois versions du programme :

— avec deux `if` imbriqués ;

— avec le mot-clé `and` dans la condition ;

— avec le mot-clé `or` dans la condition.

Exercice 11 *Équation du second degré* On veut résoudre une équation de la forme $ax^2 + bx + c = 0$, où a, b et c sont des nombres saisis par l'utilisateur. On rappelle que, lorsque $a \neq 0$, on calcule le discriminant $\Delta = b^2 - 4ac$:

— si $\Delta > 0$, l'équation a deux solutions, $x_1 = \dfrac{-b - \sqrt{\Delta}}{2a}$ et $x_2 = \dfrac{-b + \sqrt{\Delta}}{2a}$;

— si $\Delta = 0$, elle a une seule solution, $x_0 = \dfrac{-b}{2a}$;

— si $\Delta < 0$, elle n'a pas de solution réelle.

1. Écrire un programme qui demande a, b et c, affiche la valeur du discriminant, puis le nombre de solutions et leurs valeurs. La racine carrée s'obtient avec `math.sqrt`, après `import math`.

2. Que se passe-t-il si l'utilisateur saisit $a = 0$? Modifier le programme pour traiter ce cas : l'équation devient alors $bx + c = 0$.
3. Proposer un jeu de tests qui passe par tous les cas du programme.

Exercice 12 ***Date du lendemain (pour aller plus loin)*** Une année est **bissextile** (février a alors 29 jours) si elle est divisible par 4 sans être divisible par 100, ou si elle est divisible par 400. Ainsi 2024 et 2000 sont bissextiles, mais 1900 et 2026 ne le sont pas.

1. Écrire un programme qui demande une année et affiche si elle est bissextile.
2. Écrire un programme, à partir de celui de la question 1, qui demande aussi un numéro de mois (de 1 à 12) et affiche le nombre de jours de ce mois. Avril, juin, septembre et novembre ont 30 jours ; février en a 28 ou 29 ; les autres mois en ont 31.
3. Écrire un programme qui demande une date (jour, mois, année) et affiche la date du **lendemain**. Par exemple, le lendemain du 28/2/2024 est le 29/2/2024, et celui du 31/12/2026 est le 1/1/2027.