

3. Conditions.

1. Définition des conditions.

Les **conditions** permettent à un programme de prendre des décisions en fonction de certains critères. Elles exécutent un bloc de code uniquement si une condition est vraie (True).

En Python, on utilise des mots-clés comme `if`, `else`, et `elif` pour écrire des conditions.

2. Utilisation des conditions.

Structure de base :

```
if condition: # Bloc de code exécuté si la condition est vraie
```

Exemple :

```
age = 20 if age >= 18: print("Vous êtes majeur.")
```

Sortie :

```
Vous êtes majeur.
```

3. Coder les instructions `else`.

L'instruction `else` permet de spécifier un bloc de code à exécuter si la condition du `if` est fausse.

Syntaxe :

```
if condition: # Bloc exécuté si la condition est vraie else: # Bloc exécuté si la condition est fausse
```

Exemple :

```
age = 16 if age >= 18: print("Vous êtes majeur.") else: print("Vous êtes mineur.")
```

Sortie :

```
Vous êtes mineur.
```

4. Incorporer `elif`.

L'instruction `elif` (abréviation de "else if") permet de tester plusieurs conditions. Si la première condition est fausse, Python teste la suivante.

Syntaxe :

```
if condition1: # Bloc exécuté si condition1 est vraie elif condition2: # Bloc exécuté si condition1 est fausse et condition2 est vraie else: # Bloc exécuté si toutes les conditions sont fausses
```

Exemple :

```
age = 65 if age < 18: print("Vous êtes mineur.") elif age < 65: print("Vous êtes adulte.") else: print("Vous êtes senior.")
```

Sortie :

```
Vous êtes senior.
```

5. Conditions multiples et imbriquées.

Conditions multiples :

On peut combiner plusieurs conditions à l'aide des opérateurs logiques `and`, `or`, et `not` :

- `and` : Toutes les conditions doivent être vraies.
- `or` : Au moins une condition doit être vraie.
- `not` : Inverse une condition.

```
temperature = 25 humidite = 60 if temperature > 20 and humidite < 70: print("Il fait beau !")
```

Conditions imbriquées :

Les conditions peuvent être imbriquées dans d'autres :

```
age = 20 if age >= 18: if age < 65: print("Vous êtes adulte.")
```

Exercices pratiques.

1. Exercice 1 :

Écrivez un programme qui demande à l'utilisateur une note (entre 0 et 20) et affiche :

- "Insuffisant" si la note est inférieure à 10,
- "Passable" si elle est entre 10 et 12,
- "Bien" si elle est entre 13 et 16,
- "Très bien" si elle est supérieure à 16.

```
note = 9 if note < 10: print("Insuffisant") elif note >= 10 and note <= 12:
print("Passable") elif note >= 13 and note <= 16: print("Bien") else: print("Très
bien")
```

2. Exercice 2 :

Demandez à l'utilisateur d'entrer un nombre. Affichez s'il est **positif**, **négatif**, ou **nul**.

```
n = int(input("Entrer un nombre : ")) if n > 0: print("Positif") elif n < 0:
print("Négatif") else : print("Nul")
```

3. Exercice 3 :

Créez un programme qui vérifie si une année donnée est bissextile. Une année est bissextile si :

- Elle est divisible par 4,
- Mais pas divisible par 100, sauf si elle est aussi divisible par 400.

```
annee_bissextile = 2016 if annee_bissextile % 4 == 0: print("{annee_bissextile}
est une année bissextile.") else : print("{annee_bissextile} n'est pas une année
bissextile.")
```

Résumé.

Concept	Description	Exemple
Condition simple	Exécute un bloc de code si une condition est vraie.	<pre>if age >= 18: print("Vous êtes majeur.")</pre>
Condition avec <code>else</code>	Exécute un autre bloc si la condition est fausse.	<pre>if age >= 18: print("Majeur") else: print("Mineur")</pre>
Condition avec <code>elif</code>	Teste plusieurs conditions successives.	<pre>if age < 18: print("Mineur") elif age < 65: print("Adulte") else: print("Senior")</pre>
Conditions multiples	Combine des conditions avec <code>and</code> , <code>or</code> , <code>not</code> .	<pre>if age > 18 and age < 65: print("Adulte actif")</pre>
Conditions imbriquées	Place une condition à l'intérieur d'une autre.	<pre>if age >= 18: if age < 65: print("Adulte actif")</pre>
Opérateurs relationnels	Utilisés pour comparer des valeurs : <code><</code> , <code>></code> , <code><=</code> , <code>>=</code> , <code>==</code> , <code>!=</code> .	<pre>if a != b: print("Différent")</pre>
Opérateurs logiques	<code>and</code> , <code>or</code> , <code>not</code> permettent de combiner ou inverser des conditions.	<pre>if temperature > 20 and humidite < 70: print("Beau temps")</pre>