Huawei HCCDA-AI certification

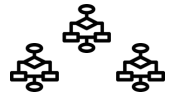
Trainer: Fawad Bahadur Marwat

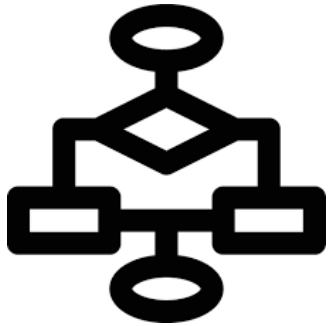
3

Conditional Statements

Definition

Conditional statements enable programs to execute different actions based on whether a condition evaluates to True or False.



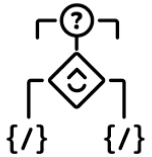


Condition

A logical expression resulting in a boolean (True or False).

Purpose

Controls program flow by enabling decision-making.

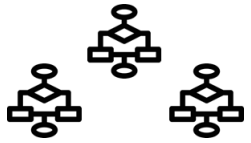


Real-Life Analogy

- **Traffic light:**
  Green → Cars go.
  Red → Cars stop.

Why Use Them?

- Execute actions when a condition is True.
- Skip actions when a condition is False.



CORVIT
Reaching end to end

Trainer: Fawad Bahadur Marwat

Conditions in Python

Definition

Expressions evaluating to True or False

Usage

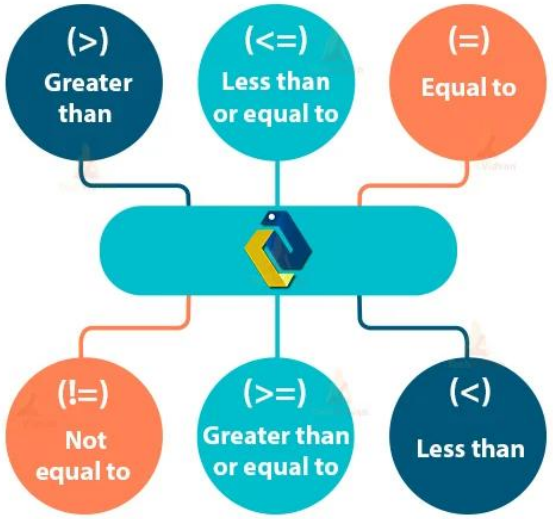
Control flow in if, if-else, if-elif-else statements

Example

How It Works: x > 5 evaluates to True, triggering the print statement.

```
x = 5
if x > 5:
    print("x is greater than 5")
```

Comparison Operators



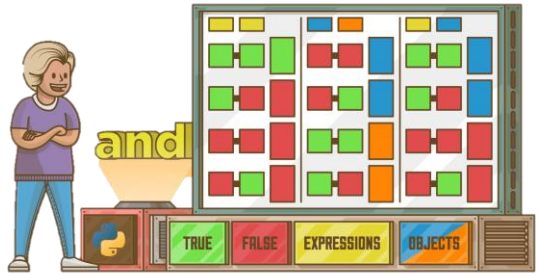
CORVIT
Reaching end to end

Trainer: Fawad Bahadur Marwat

Logical Operators in Python

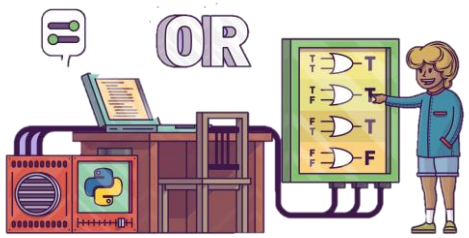
Purpose

Combine multiple conditions to return True or False



Types

and: True if both conditions are True
or: True if at least one condition is True
not: Reverses a condition
(True → False, False → True)



Example

and: Both age > 18 and age < 30 must be True to print.

or: Either grade == 'A' or grade == 'B' being True triggers the print.

```
# and operator
age = 20
if age > 18 and age < 30:
    print("You're a young adult")

# or operator
grade = 'B'
if grade == 'A' or grade == 'B':
    print("You passed")
```

CORVIT
Reaching end to end

Trainer: Fawad Bahadur Marwat

If Statements in Python

Purpose

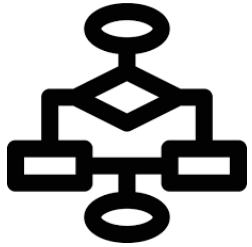
Execute a code block only if a condition is True

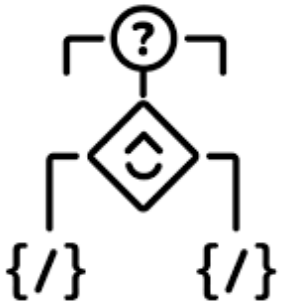
Syntax

if keyword, followed by a condition, then a colon :

Behavior

True condition: Code block executes
False condition: Code block is skipped

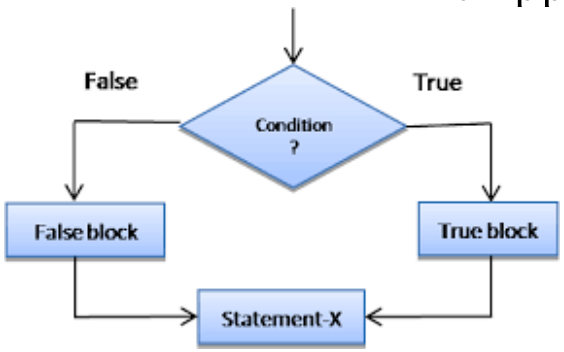




Example

```
num = 10
if num > 5:
    print("The number is greater than 5")
```

num > 5 evaluates to True, so the message is printed.
If num <= 5, the print statement is skipped.



Trainer: Fawad Bahadur Marwat

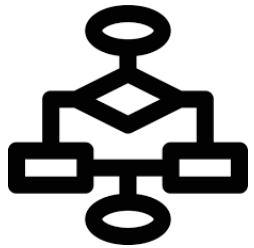
If-else Statements in Python

Purpose

Execute one code block if a condition is True, otherwise execute a different code block.

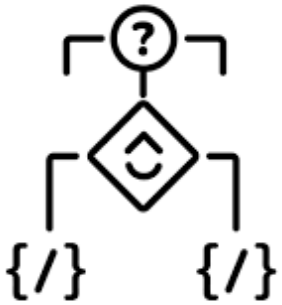
Syntax

If a condition is True, then execute a code block; else, execute a different code block.



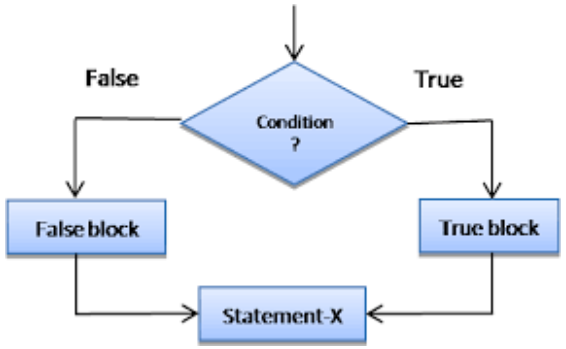
Behavior

True condition: Code block 1 executes
False condition: Code block 2 executes



Example

```
num = 10
if num > 5:
    print("The number is greater than 5")
else:
    print("The number is not greater than 5")
```



Trainer: Fawad Bahadur Marwat

If-Elif-Else Statements in Python

Purpose

Evaluate multiple conditions sequentially

Syntax

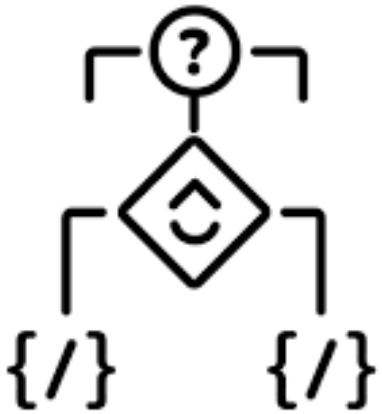
if: Checks the first condition
elif: Checks additional conditions if previous ones are **False**
else: Runs if no conditions are **True** (optional fallback)

Behavior

Executes the first True condition's block or the else block if none are True

Example

```
score = 75
if score >= 90:
    print("Grade: A")
elif score >= 80:
    print("Grade: B")
elif score >= 70:
    print("Grade: C")
else:
    print("Grade: D")
```



CORVIT
Reaching end to end

Trainer: Fawad Bahadur Marwat