# Level 2: Control Flow

## 1. **if / elif / else**

We use conditions to make decisions.

### Example:

marks = 75

if marks >= 90:

print("Grade: A")

elif marks >= 75:

print("Grade: B")

elif marks >= 50:

print("Grade: C")

else:

print("Fail")

👉 Output:

Grade: B

## 2. **for loop**

Used when you know how many times to repeat.

### Example:

for i in range(5):

print("Iteration:", i)

👉 Output:

Iteration: 0

Iteration: 1

Iteration: 2

Iteration: 3

Iteration: 4

## 3. **while loop**

Used when you don’t know how many times, but repeat until a condition is true.

### Example:

count = 1

while count <= 5:

print("Count is:", count)

count += 1

👉 Output:

Count is: 1

Count is: 2

Count is: 3

Count is: 4

Count is: 5

## 4. **break, continue, pass**

* break → stop the loop immediately
* continue → skip current iteration
* pass → do nothing (placeholder)

### Example:

for i in range(1, 6):

if i == 3:

continue # skip 3

if i == 5:

break # stop at 5

print(i)

Output:

1

2

4

Exercises for You

1. Write a program that prints numbers from 1 to 20.
   * Skip multiples of 3.
   * Stop if the number reaches 15.
2. Write a while loop that asks for a password until the user enters "python123".
   * If correct → print "Access Granted".
   * Otherwise → ask again.

# 📝 Problem Statement: Simple ATM System

Design a Python program that simulates a basic ATM machine.

### Requirements:

1. PIN Verification
   * The system should ask the user to enter their ATM PIN.
   * If the PIN is correct (e.g., "1234"), allow access.
   * If the PIN is incorrect, display "Wrong PIN. Exiting..." and stop the program.
2. Menu Options (should repeat until the user chooses Exit):
   * 1 → Check Balance → Display the current balance.
   * 2 → Deposit Money → Ask for deposit amount, add it to balance, and show new balance.
   * 3 → Withdraw Money → Ask for withdrawal amount:
     + If amount > balance → show "Insufficient balance!"
     + Else deduct amount and show new balance.
   * 4 → Exit → End the program with "Thank you for using Python ATM. Goodbye!"
3. Initial Balance → Start with ₹1000 by default.
4. Loop Behavior → The menu should keep showing until the user selects Exit.

👉 This mini project helps practice loops, conditions, user input, arithmetic operations, and break/continue usage.

# 🏧 Simple ATM System

# -----------------------------

# Simple ATM System

# -----------------------------

balance = 1000 # initial balance

pin = "1234"

print("Welcome to Python ATM!")

# Step 1: PIN verification

entered\_pin = input("Enter your ATM PIN: ")

if entered\_pin != pin:

print("Wrong PIN. Exiting...")

else:

# Step 2: Menu with loop

while True:

print("\n----- Menu -----")

print("1. Check Balance")

print("2. Deposit Money")

print("3. Withdraw Money")

print("4. Exit")

choice = input("Enter your choice: ")

if choice == "1":

print(f"Your balance is: ₹{balance}")

elif choice == "2":

amount = int(input("Enter deposit amount: "))

balance += amount

print(f"₹{amount} deposited. New balance: ₹{balance}")

elif choice == "3":

amount = int(input("Enter withdrawal amount: "))

if amount > balance:

print("Insufficient balance!")

else:

balance -= amount

print(f"₹{amount} withdrawn. New balance: ₹{balance}")

elif choice == "4":

print("Thank you for using Python ATM. Goodbye!")

break

else:

print("Invalid choice. Try again.")

### Sample Output

Welcome to Python ATM!

Enter your ATM PIN: 1234

----- Menu -----

1. Check Balance

2. Deposit Money

3. Withdraw Money

4. Exit

Enter your choice: 1

Your balance is: ₹1000

----- Menu -----

1. Check Balance

2. Deposit Money

3. Withdraw Money

4. Exit

Enter your choice: 2

Enter deposit amount: 500

₹500 deposited. New balance: ₹1500

----- Menu -----

1. Check Balance

2. Deposit Money

3. Withdraw Money

4. Exit

Enter your choice: 3

Enter withdrawal amount: 700

₹700 withdrawn. New balance: ₹800

----- Menu -----

1. Check Balance

2. Deposit Money

3. Withdraw Money

4. Exit

Enter your choice: 4

Thank you for using Python ATM. Goodbye!