**Python Operators –Questions**

**1. Arithmetic Operators – Apple Store**

A store is selling apples for ₹25 each. A customer buys 7 apples.

* Calculate the total cost.
* Find the remaining money if the customer gives ₹200.

**2. Relational Operators – Cricket Score**

A batsman scores 105 runs. The target was 100.  
Check if the batsman scored more than the target.

**3. Logical Operators – Movie Time**

You can watch a movie if:

* You have at least ₹200 **and**
* The movie is available in the theatre.  
  Check if you can watch the movie. (True or False)

**4. Assignment Operators – Chocolates**

You start with 10 chocolates. Each day you eat 2 chocolates.  
Update the count using assignment operators and print the chocolates left.

**5. Ternary – Free Delivery**

If you shop for ₹500 or more, you get free delivery, otherwise ₹50 delivery charge.  
You shopped for ₹650.  
Use a ternary operator to show the delivery message.

**6. Ternary – Discount Eligibility**

If you are a **premium member**, you get a 20% discount. Else, no discount.  
Use a ternary operator to show the discount message.

**Python-based questions related to ternary operator:**

1. **Comparison Operators:  
   Write a Python program that declares two variables a and b and compares them using the operators ==, !=, >, <, >=, and <=, then prints the results.**
2. **Even or Odd (Ternary):  
   Write a Python program that uses the ternary operator to determine if a given number is even or odd and prints the result.**
3. **Grade Classification (Nested Ternary):  
   Declare a variable score (0–100). Use nested ternary operators to classify the score into:**
   * **A (>= 90)**
   * **B (>= 75)**
   * **C (>= 60)**
   * **D (>= 45)**
   * **F (below 45)  
     Print the grade.**
4. **Temperature Check (Nested Ternary):  
   Declare a variable temperature. Use nested ternary operators to categorize it as:**
   * **"Hot" (above 30)**
   * **"Warm" (20–30)**
   * **"Cool" (10–19)**
   * **"Cold" (below 10)  
     Print the result.**
5. **Age Group Classification (Ternary):  
   Declare a variable age. Use the ternary operator to classify the age into:**
   * **"Child" (0–12)**
   * **"Teen" (13–19)**
   * **"Adult" (20–64)**
   * **"Senior" (65 and above)  
     Print the result.**

**Python Operators — Student Worksheet**

# 1. Arithmetic Operators – Apple Store

price\_per\_apple = 25

quantity = 7

total\_cost = price\_per\_apple \* quantity

remaining\_money = 200 - total\_cost

print("Total Cost:", total\_cost) # 175

print("Remaining Money:", remaining\_money) # 25

# 2. Relational Operators – Cricket Score

runs\_scored = 105

target = 100

print(runs\_scored > target) # True

# 3. Logical Operators – Movie Time

money = 250

movie\_available = True

can\_watch = money >= 200 and movie\_available

print(can\_watch) # True

# 4. Assignment Operators – Chocolates

chocolates = 10

chocolates -= 2

print("Chocolates left:", chocolates) # 8

# 5. Ternary – Free Delivery

amount = 650

message = "Free Delivery" if amount >= 500 else "Delivery charge ₹50"

print(message) # Free Delivery

# 6. Ternary – Discount Eligibility

premium\_member = True

discount\_msg = "20% Discount" if premium\_member else "No Discount"

print(discount\_msg) # 20% Discount

**📌 Real-Time Analogy Tasks for Conditional Statements + in / not in**

**1. Cafe Menu Checker**

* **Task:** Write a program that checks if a customer’s requested item is available in the cafe menu.
* **Menu:** ["coffee", "tea", "sandwich", "juice"]
* If available → print "Yes, we have {item}".
* If not → print "Sorry, {item} is not available today."

**2. Airport Baggage Security**

* **Task:** Check if any forbidden item is inside a passenger’s baggage.
* **Forbidden items:** ["knife", "gun", "lighter"]
* If baggage contains any forbidden item → print "Security Alert! Remove {item}"
* Otherwise → print "Baggage cleared for boarding."

**3. Online Shopping Discount Eligibility**

* **Task:** Check if the entered coupon code is in the list of valid discount codes.
* **Valid codes:** ["SAVE10", "WELCOME", "FREESHIP"]
* If coupon is valid → print "Coupon applied successfully!"
* Else → print "Invalid coupon code."

**4. Movie Ticket Age Restriction**

* **Task:** Some movies are age-restricted. If the movie is in the list of "restricted\_movies", check if the user is 18+.
* **Restricted movies:** ["Deadpool", "Joker", "John Wick"]
* If not restricted → print "Enjoy the movie!"
* If restricted but user under 18 → print "You are not allowed to watch this movie."

**5. Password Security Check**

* **Task:** Check if a user’s password contains any **banned words** ("password", "1234", "admin") using in.
* If found → "Weak password! Please choose another."
* Else → "Password accepted."

**Solutions**

**1. Cafe Menu Checker**

menu = ["coffee", "tea", "sandwich", "juice"]

item = input("Enter the item you want: ").lower()

if item in menu:

print(f"Yes, we have {item}")

else:

print(f"Sorry, {item} is not available today.")

**2. Airport Baggage Security**

forbidden\_items = ["knife", "gun", "lighter"]

baggage = input("Enter baggage items separated by commas: ").lower().split(",")

for item in baggage:

if item.strip() in forbidden\_items:

print(f"Security Alert! Remove {item.strip()}")

break

else:

print("Baggage cleared for boarding.")

**3. Online Shopping Discount Eligibility**

valid\_codes = ["SAVE10", "WELCOME", "FREESHIP"]

code = input("Enter coupon code: ").upper()

if code in valid\_codes:

print("Coupon applied successfully!")

else:

print("Invalid coupon code.")

**4. Movie Ticket Age Restriction**

restricted\_movies = ["Deadpool", "Joker", "John Wick"]

movie = input("Enter movie name: ")

age = int(input("Enter your age: "))

if movie in restricted\_movies and age < 18:

print("You are not allowed to watch this movie.")

else:

print("Enjoy the movie!")

**5. Password Security Check**

banned\_words = ["password", "1234", "admin"]

password = input("Enter your password: ")

if any(bad in password.lower() for bad in banned\_words):

print("Weak password! Please choose another.")

else:

print("Password accepted.")

**Python Tasks on loops**

**Task 1: Print Numbers from 1 to 10**

**Instructions:**

* Use a for loop to print numbers from 1 to 10.
* Display the numbers in the console.

**Task 2: Print Even Numbers from 1 to 20**

**Scenario:** Write a Python program that prints all even numbers from 1 to 20.  
**Instructions:**

* Use a for loop to iterate from 1 to 20.
* Use an if statement to check if a number is even (number % 2 == 0) and print it.

**Task 3: Print Multiples of 5 up to 50**

**Scenario:** Write a Python program that prints all multiples of 5 up to 50.  
**Instructions:**

* Use a for loop to iterate from 1 to 50.
* Use an if statement to check if a number is a multiple of 5 (number % 5 == 0) and print it.

**Task 4: Sum of Numbers from 1 to 100**

**Scenario:** Write a Python program that calculates the sum of numbers from 1 to 100.  
**Instructions:**

* Use a for loop to iterate from 1 to 100.
* Add each number to a variable total\_sum.
* Print the total sum.

**Task 5: Factorial of a Number**

**Scenario:** Write a Python program that calculates the factorial of a given number using a for loop.  
**Instructions:**

* Prompt the user to enter a number.
* Initialize factorial = 1.
* Use a for loop to multiply numbers from 1 to the given number.
* Print the factorial.

**Task 6: Print Numbers in Reverse Order**

**Scenario:** Write a Python program that prints numbers from 10 to 1 in reverse order.  
**Instructions:**

* Use a for loop to count down from 10 to 1.
* Display the numbers in the console.

**Task 7: Print the Alphabet (A to Z) *(Optional)***

**Scenario:** Write a Python program that prints the alphabet from A to Z.  
**Instructions:**

* Use a for loop to iterate through ASCII values of uppercase letters (65 to 90).
* Convert ASCII values to characters using chr() and print them.

**Task 8: Multiplication Table (1 to 5)**

**Scenario:** Write a Python program that uses nested loops to print a multiplication table for numbers 1 through 5.  
**Instructions:**

* Use an outer for loop for numbers 1 to 5.
* Use an inner for loop for multiplying each number from 1 to 5.
* Print the results in a formatted way.

**Break and Continue – Real-Time Analogy Tasks**

**1. Bus Stop Passenger Search (break)**

**Given a list of passengers, find your friend in the list. Stop searching when found.**

**2. Skipping Expired Milk Packets (continue)**

**Given a list of milk packets with expiry status, skip expired packets and display only fresh packets.**

**3. First Defective Phone Found (break)**

**Given a list of phone statuses, find the first defective phone and stop checking.**

**4. Skipping Students Without Fees (continue)**

**Given a list of students with payment status, print only those who have paid.**

**5. ATM PIN Attempts (break)**

**Allow maximum 3 attempts for entering the correct PIN. Stop when correct PIN is entered.**

**Break and Continue – Solutions**

**1. Bus Stop Passenger Search (break)**

passengers = ["Ravi", "Anita", "Suresh", "Teja", "Meena", "Kiran"]

friend\_name = "Teja"

for person in passengers:

if person == friend\_name:

print("Found!")

break

**2. Skipping Expired Milk Packets (continue)**

milk\_packets = [

{"id": 101, "expiry": False},

{"id": 102, "expiry": True},

{"id": 103, "expiry": False},

{"id": 104, "expiry": True},

]

for packet in milk\_packets:

if packet["expiry"]:

continue

print(f"Packet {packet['id']} is fresh")

**3. First Defective Phone Found (break)**

phones = ["Working", "Working", "Defective", "Working", "Working"]

for index, status in enumerate(phones, start=1):

if status == "Defective":

print(f"First defective phone found at position {index}")

break

**4. Skipping Students Without Fees (continue)**

students = [

{"name": "Amit", "paid": True},

{"name": "Ravi", "paid": False},

{"name": "Teja", "paid": True},

{"name": "Kiran", "paid": False},

]

for student in students:

if not student["paid"]:

continue

print(student["name"])

**5. ATM Pin Attempts (break)**

correct\_pin = "1234"

for attempt in range(3):

pin = input("Enter PIN: ")

if pin == correct\_pin:

print("Access Granted")

break

else:

print("Wrong PIN")

**Conditional Statements – Real Life Analogies**

**Task 1 – Movie Ticket Discount**

A cinema offers:

* 50% discount for kids below 12 years
* 30% discount for senior citizens (60+ years)
* No discount for others

Write a program to take the **age** of the person and print the discount they get.

**Task 2 – Library Late Fee**

A library charges a late fee for book returns:

* 1–5 days late → ₹5/day
* 6–10 days late → ₹10/day
* More than 10 days late → ₹15/day

Take **days late** as input and calculate the total fine.

**Task 3 – Coffee Shop Loyalty Card**

At a coffee shop:

* If you have bought **10 or more coffees**, you get a free one.
* If you have bought less than 10, display how many more to get the free one.

Take **coffees purchased** as input and display the result.

**Task 4 – Weather Decision**

Write a program that:

* Prints **"Carry an umbrella"** if it's raining.
* Prints **"Wear sunscreen"** if it's sunny.
* Prints **"Wear a jacket"** if it's cold.

**Task 5 – ATM Withdrawal**

Check if the user has enough balance to withdraw the entered amount.  
If yes, deduct it; if not, print **"Insufficient balance"**.

**2. Loops – Real Life Analogies**

**Task 6 – Bus Passenger Counter**

A bus conductor notes the number of passengers entering at each stop (until the bus reaches its destination).  
Take the passenger counts for each stop and calculate the total passengers.

**Task 7 – Restaurant Order Summary**

Ask the customer to enter the price of each food item until they type **"done"**.  
Then print the total bill.

**Task 8 – Elevator Capacity Check**

An elevator can carry **500 kg max**.  
Keep taking passenger weights until the limit is reached or exceeded.  
Display how many passengers entered and the total weight.

**Task 9 – Multiplication Practice Game**

Ask the student to solve multiplication problems (random numbers 1–10) until they give **3 wrong answers**.  
At the end, show their score.

**Task 10 – Grocery Store Inventory**

A store starts the day with **100 apples**.  
Customers buy apples (you take input for each).  
Stop when the apples are finished and print how many customers were served.

**3. Combined Conditional + Loops**

**Task 11 – Fuel Station**

* A car starts with a given amount of fuel.
* For each kilometer driven, fuel decreases by 1 liter.
* If fuel goes below 5 liters, print **"Low Fuel – Please refill"** and stop.

**Task 12 – Gym Membership**

Take monthly attendance of a gym member for 12 months.  
If attendance in any month is less than 10 days, print **"Membership at risk"**.

**Task 13 – Bank Loan Approval**

Ask for a customer’s credit score and salary.

* Loan approved if credit score > 700 **and** salary > ₹30,000.
* Otherwise, rejected.  
  Repeat for 5 customers.

**Task 14 – School Attendance Bonus**

Loop through 30 days of a month.  
Count days present (P) and absent (A).  
If attendance is 90% or more, print **"Bonus Marks Awarded"**, else **"No Bonus"**.

**solutions**

**1. Conditional Statements – Solutions**

**Task 1 – Movie Ticket Discount**

age = int(input("Enter age: "))

if age < 12:

print("50% discount")

elif age >= 60:

print("30% discount")

else:

print("No discount")

**Task 2 – Library Late Fee**

days = int(input("Days late: "))

if 1 <= days <= 5:

fine = days \* 5

elif 6 <= days <= 10:

fine = days \* 10

elif days > 10:

fine = days \* 15

else:

fine = 0

print("Total fine: ₹", fine)

**Task 3 – Coffee Shop Loyalty Card**

coffees = int(input("Coffees purchased: "))

if coffees >= 10:

print("You get a free coffee!")

else:

print(f"Buy {10 - coffees} more to get a free one.")

**Task 4 – Weather Decision**

weather = input("Enter weather (rainy/sunny/cold): ").lower()

if weather == "rainy":

print("Carry an umbrella")

elif weather == "sunny":

print("Wear sunscreen")

elif weather == "cold":

print("Wear a jacket")

else:

print("Invalid input")

**Task 5 – ATM Withdrawal**

balance = 5000

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

if amount <= balance:

balance -= amount

print("Withdrawal successful. Remaining balance:", balance)

else:

print("Insufficient balance")

**2. Loops – Solutions**

**Task 6 – Bus Passenger Counter**

total\_passengers = 0

stops = int(input("Number of stops: "))

for i in range(stops):

count = int(input(f"Passengers at stop {i+1}: "))

total\_passengers += count

print("Total passengers:", total\_passengers)

**Task 7 – Restaurant Order Summary**

total = 0

while True:

item = input("Enter item price or 'done': ")

if item.lower() == "done":

break

total += float(item)

print("Total bill: ₹", total)

**Task 8 – Elevator Capacity Check**

capacity = 500

total\_weight = 0

passengers = 0

while total\_weight < capacity:

weight = int(input("Enter passenger weight: "))

if total\_weight + weight > capacity:

break

total\_weight += weight

passengers += 1

print("Passengers entered:", passengers)

print("Total weight:", total\_weight)

**Task 9 – Multiplication Practice Game**

import random

score = 0

wrong = 0

while wrong < 3:

a = random.randint(1, 10)

b = random.randint(1, 10)

ans = int(input(f"{a} x {b} = "))

if ans == a \* b:

score += 1

else:

wrong += 1

print("Game over! Your score:", score)

**Task 10 – Grocery Store Inventory**

apples = 100

customers = 0

while apples > 0:

bought = int(input("Apples bought by customer: "))

if bought <= apples:

apples -= bought

customers += 1

else:

print("Not enough apples available")

print("Customers served:", customers)

**3. Combined Conditional + Loops – Solutions**

**Task 11 – Fuel Station**

fuel = int(input("Enter fuel in liters: "))

for km in range(1, fuel + 1):

fuel -= 1

print(f"Driven {km} km, Fuel left: {fuel} liters")

if fuel < 5:

print("Low Fuel – Please refill")

break

**Task 12 – Gym Membership**

for month in range(1, 13):

days = int(input(f"Enter attendance days for month {month}: "))

if days < 10:

print("Membership at risk")

break

**Task 13 – Bank Loan Approval**

for i in range(5):

score = int(input("Enter credit score: "))

salary = int(input("Enter salary: "))

if score > 700 and salary > 30000:

print("Loan approved")

else:

print("Loan rejected")

**Task 14 – School Attendance Bonus**

present\_days = 0

for day in range(1, 31):

status = input(f"Day {day} (P/A): ").upper()

if status == "P":

present\_days += 1

attendance\_percent = (present\_days / 30) \* 100

if attendance\_percent >= 90:

print("Bonus Marks Awarded")

else:

print("No Bonus")

**🐍 Day 8 – Python Tasks on functions**

**Task 1: Greeting Function**

📌 **Scenario**: Create a function that takes a name as an argument and returns a greeting message.  
**Question:**

* Write a function greet(name) that takes one parameter name.
* The function should return "Hello, [name]!".
* Call the function with at least **three different names** and print the results.

**Task 2: Sum Function**

📌 **Scenario**: Create a function that takes two numbers as arguments and returns their sum.  
**Question:**

* Write a function sum(a, b) that takes two parameters a and b.
* The function should return the sum of a and b.
* Test the function with at least **three different pairs of numbers** and print the results.

**Task 3: Square Function**

📌 **Scenario**: Create a function that takes a number as an argument and returns its square.  
**Question:**

* Write a function square(num) that takes one parameter num.
* The function should return the **square of num**.
* Call the function with at least **three different numbers** and print the results.

**Task 4: Average Function**

📌 **Scenario**: Create a function that takes a list of numbers as an argument and returns the average.  
**Question:**

* Write a function average(arr) that takes a list arr.
* The function should return the **average of the numbers**.
* Example: [2, 5, 2] → 9/3 = 3.
* Test the function with at least **three different lists**.

**Task 5: Vowels Function**

📌 **Scenario**: Create a function that checks if a string contains vowels.  
**Question:**

* Write a function contains\_vowels(text) that takes one string text.
* The function should return whether the string **contains vowels (a, e, i, o, u)** or not.
* Use **loops and if conditions**.
* Example:
  + "hello" → contains vowels
  + "hll" → does not contain vowels

**Task 6: Temperature Converter**

📌 **Scenario**: Create a function that converts temperatures between Celsius and Fahrenheit.  
**Question:**

* Write a function convertTemperature(temp, scale) that takes two parameters:
  + temp → temperature value
  + scale → the scale to convert to ("C" for Celsius or "F" for Fahrenheit)
* If scale = "C", convert Fahrenheit to Celsius using:  
  (temp−32)×5/9(temp - 32) × 5/9
* If scale = "F", convert Celsius to Fahrenheit using:  
  (temp×9/5)+32(temp × 9/5) + 32
* Test the function with at least **three different inputs**.

**🔹 Basic Level**

**Task 1:**  
Take input from the user and convert it to **uppercase**.

**Task 2:**  
Take input from the user and convert it to **lowercase**.

**Task 3:**  
Take input from the user and **reverse the string**.  
👉 *Hint:* Use split(), reverse(), and join() methods.

**🔹 Advanced Level**

**Task 1: Username Validation**  
Prompt the user to enter a username.  
Check if the username length is between **5 to 15 characters**.  
Display whether it is valid or not.

**Task 2: Email Formatter**  
Prompt the user to enter their **first name** and **last name**.  
Convert both to lowercase and join them with a dot (.).  
Add @gmail.com at the end.  
📌 Example:  
Input → jenny, joy  
Output → jenny.joy@gmail.com

**Task 3: Word Counter**  
Prompt the user to enter a **sentence**.  
Split the sentence into words and count them.  
👉 *Hint:* Use split() and len().

**Task 4: Palindrome Checker**  
Prompt the user to enter a string.  
Check if the string reads the same forward and backward.  
📌 Example:  
Input → john → Output: Not a palindrome  
Input → dad → Output: It is a palindrome  
👉 *Hint:* Use split(), join(), and reverse().

**Python Questions**

1. **Add elements to a list and iterate**
   * Create a Python list of your favorite movies.
   * Add a few more movies to the list using append() or extend().
   * Iterate over the list using:
     + a normal for loop (with index),
     + a for-in loop (direct iteration).
   * Print each movie to the console.
2. **Remove elements from a list**
   * Create a list of items.
   * Remove the **first element** using pop(0) or del.
   * Remove the **last element** using pop().
   * Print the updated list.
3. **Reverse a list using a for loop**
   * Write a Python program to reverse a list **without using built-in reverse() or slicing**.
   * Use a loop to append elements to a new list in reverse order.
   * Print the reversed list.
4. **Find even and odd numbers in a list**
   * Given: arr = [12, 3, 5, 6, 22, 56, 29].
   * Separate even and odd numbers.
   * Print the **sum of even numbers** and the **sum of odd numbers**.
5. **Separate data types from a heterogeneous list**
   * Input:
   * arr = ["apple", "banana", "mango", "banana", 3, 4, 5, 6, True, {"name": "object"}]
   * Create 4 new lists:
     + num = [3,4,5,6]
     + str = ["apple","banana","mango","banana"]
     + bool = [True]
     + obj = [{"name": "object"}]
   * Use type() inside a loop to check each element and store it in the correct list.

**Python Tasks**

**Task 1: Using list concatenation**

* **Objective:** Merge two or more lists.
* **Task:** Create two lists, one with your favorite sports and one with your favorite hobbies. Use + operator (list concatenation) to merge them into a single list.
* **Expected Output:** Display the merged list.

**Task 2: Using list slicing (like splice in JS)**

* **Objective:** Modify a list by adding, removing, or replacing elements.
* **Task:** Create a list of numbers from 1 to 10. Use slicing to remove the numbers 4, 5, 6 and replace them with 40, 50, 60.
* **Expected Output:** Display the list before and after the operation.

**Task 3: Using slicing (like slice in JS)**

* **Objective:** Extract a portion of a list without modifying the original list.
* **Task:** Create a list of the days of the week. Use slicing to create a new list that contains only the weekdays (Monday to Friday).
* **Expected Output:** Display the original list and the new list.

**Task 4: Using join**

* **Objective:** Convert a list to a string.
* **Task:** Create a list of words that form a sentence. Use " ".join(list) to combine them into a single string with spaces between each word.
* **Expected Output:** Display the resulting sentence.

**Task 5: Using sort**

* **Objective:** Sort the elements of a list.
* **Task:** Create a list of random numbers. Use .sort() method to sort the numbers in ascending order.
* **Expected Output:** Display the sorted list.