



Topic: Conditional Statements (if, elif, else)

What Are Conditional Statements?

They help your program **make decisions** based on certain **conditions**.

Like real life:

"If it's raining, take an umbrella. Else, wear sunglasses."

Syntax in Python:

```
if condition:  
    # code if condition is True  
  
elif another_condition:  
    # code if elif is True  
  
else:  
    # code if none of the above is True
```

◊ 1. Basic **if** Statement

```
age = 20
```

```
if age >= 18:  
    print("You are eligible to vote")
```

- ✓ it checks the condition and runs the code inside only if it's **True**.
-

- ◊ **2. if with else**

```
age = 16
```

```
if age >= 18:  
    print("You are eligible to vote")  
  
else:  
    print("You are not eligible to vote")
```

- ✓ **else** runs when the condition is **False**.
-

- ◊ **3. if, elif, else (Multiple Conditions)**

```
marks = 75
```

```
if marks >= 90:  
    print("Grade A")  
  
elif marks >= 70:  
    print("Grade B")  
  
elif marks >= 50:  
    print("Grade C")  
  
else:  
    print("Fail")
```

Explain:

- Only **one block** is executed (the first **True** one).
- Once a condition matches, it **skips the rest**.

◊ 4. Real-Time Use Case: Login System

```
username = input("Enter username: ")  
password = input("Enter password: ")  
  
if username == "admin" and password == "1234":  
    print("Login successful!")  
  
else:
```

```
print("Invalid credentials")
```

- ✓ Combine **comparison** and **logical** operators here.

◊ 5. Nested Conditions

```
age = 25
has_license = True

if age >= 18:
    if has_license:
        print("You can drive")
    else:
        print("You need a license to drive")
else:
    print("You are too young to drive")
```

- ✓ Mention: Good for real-world validations, but **avoid deep nesting**.

✓ 1. Exam Eligibility Example (using and)

▀ Logic:

If **marks >= 50** and **attendance >= 75**, student is eligible for the exam.

▀ Code:

```
marks = 55
attendance = 80
```

```
if marks >= 50 and attendance >= 75:  
    print("Eligible for exam")  
else:  
    print("Not eligible")
```

✓ 2. Mobile Recharge Offer (using or)

💻 Logic:

If recharge amount ≥ 399 or data balance $\geq 1\text{GB}$, user gets bonus.

💻 Code:

```
recharge_amount = 200  
data_balance = 1.5 # in GB  
  
if recharge_amount >= 399 or data_balance >= 1:  
    print("You are eligible for bonus data")  
else:  
    print("No bonus data")
```

✓ 3. Restaurant Offer (using and + or)

💻 Logic:

If the **bill is above ₹1000** and it's a **weekend**,
or the user is a **Gold Member**, they get 20% off.

💻 Code:

```
bill = 1200
```

```
day = "Saturday"
membership = "Gold"

if (bill > 1000 and day in ["Saturday", "Sunday"]) or membership
== "Gold":
    print("You get 20% discount")
else:
    print("No discount")
```



Best Practices:

- Use **elif** instead of multiple **if** checks.
- Keep conditions **readable** and **logical**.
- Indentation matters! (Each block inside **if/else** should be indented)



Quick Recap Table:

Statement	Description
if	Checks a condition
elif	Checks another condition if the first is False
else	Runs if no condition is True

Gowtham SB

www.linkedin.com/in/sbgowtham/

Instagram - @dataengineeringtamil

About the Author

Gowtham SB is a **Data Engineering expert, educator, and content creator** with a passion for **big data technologies, as well as cloud and Gen AI**. With years of experience in the field, he has worked extensively with **cloud platforms, distributed systems, and data pipelines**, helping professionals and aspiring engineers master the art of data engineering.

Beyond his technical expertise, Gowtham is a **renowned mentor and speaker**, sharing his insights through engaging content on **YouTube and LinkedIn**. He has built one of the **largest Tamil Data Engineering communities**, guiding thousands of learners to excel in their careers.

Through his deep industry knowledge and hands-on approach, Gowtham continues to **bridge the gap between learning and real-world implementation**, empowering individuals to build **scalable, high-performance data solutions**.

Socials

 **YouTube** - <https://www.youtube.com/@dataengineeringvideos>

 **Instagram** - <https://instagram.com/dataengineeringtamil>

 **Instagram** - <https://instagram.com/thedatatech.in>

 **Connect for 1:1** - <https://topmate.io/dataengineering/>

 **LinkedIn** - <https://www.linkedin.com/in/sbgowtham/>

 **Website** - <https://codewithgowtham.blogspot.com>

 **GitHub** - <http://github.com/Gowthamdataengineer>

 **WhatsApp** - <https://lnkd.in/g5JrHw8q>

Gowtham SB

www.linkedin.com/in/sbgowtham/

Instagram - @dataengineeringtamil

✉ Email - atozknowledge.com@gmail.com

📱 All My Socials - <https://lnkd.in/gf8k3aCH>