📓 Lecture 7: Conditional Statements in Python

🔑 1. Why Do We Use Conditional Statements?

Definition: Conditional statements allow your program to **make decisions** based on whether a condition is **True** or **False**.

🔑 Real-Life Examples:

- If it rains → take an umbrella
- If light is green → drive
- If password is correct → login

Insight: They control **which block of code runs** and **change program flow.**

□ 2. Basic Conditional Keywords in Python

Statement	Purpose
`if`	Executes code **only if** condition is True
`else`	Executes **if condition is False**
`elif`	Checks **multiple conditions**

✅ `if` Statement

```
python
CopyEdit
traffic_light = "green"
if traffic_light == "green":
    print("You can go")
```

Output:

```
go
CopyEdit
You can go
```

❌ `else` Statement

```
python
CopyEdit
traffic_light = "red"
if traffic_light == "green":
    print("You can go")
else:
    print("You must stop")
```

Output:

```
arduino
CopyEdit
You must stop
```

🔂 `elif` Statement

```
python
CopyEdit
traffic_light = "yellow"
if traffic_light == "green":
    print("You can go")
elif traffic_light == "yellow":
    print("Slow down and prepare to stop")
else:
    print("You must stop")
```

Output:

```
arduino
CopyEdit
Slow down and prepare to stop
```

□ 3. Nested Conditions

Check **a condition inside another condition**.

```
python
CopyEdit
car_speed = 60
seat_belt = True

if car_speed <= 60:
    if seat_belt:
        print("You are driving safely")
    else:
        print("Please wear your seat belt")
else:
```

```
    print("Slow down, you are exceeding the speed limit")
```

Output:

```
sql
CopyEdit
You are driving safely
```

🔐 4. Logical Operators in Conditions

Operator	Description
`and`	All conditions must be True
`or`	At least one condition is True
`not`	Reverses the condition

`and` Example:

```
python
CopyEdit
ped_signal = "walk"
is_flashing = False

if ped_signal == "walk" and not is_flashing:
    print("Pedestrian can cross safely")
else:
    print("Pedestrian should wait")
```

Output:

```
sql
CopyEdit
Pedestrian can cross safely
```

`or` Example:

```
python
CopyEdit
traffic_light = "red"
vehicle_type = "ambulance"

if traffic_light == "green" or vehicle_type == "ambulance":
    print("You can go")
else:
```

```
    print("You must stop")
```

Output:

```
go
CopyEdit
You can go
```

`not` Example:

```
python
CopyEdit
ped_signal = "don't walk"

if not ped_signal == "walk":
    print("Do not cross")
else:
    print("You can cross")
```

Output:

```
pgsql
CopyEdit
Do not cross
```

⚠ 5. Case Sensitivity in Conditions

Python is **case-sensitive**.

```
python
CopyEdit
traffic_light = "Red"

if traffic_light == "red":
    print("You must stop")
else:
    print("Slow down and prepare to stop")
```

Output:

```
arduino
CopyEdit
Slow down and prepare to stop
```

✅ 6. Best Practices

✔ Use **proper indentation** (4 spaces)
✔ Keep conditions **clear and readable**
✔ Use **logical operators** for combining conditions
✔ Be careful with **case sensitivity** in string comparisons
✔ Use `elif` instead of multiple `if` blocks when conditions are related

---

📌 Key Takeaways

Concept	Summary
`if`	Checks if a condition is True
`else`	Runs when `if` is False
`elif`	Used to check **more than two** conditions
Nested Conditions	For **multi-level decisions**
Logical Operators	`and`, `or`, `not` make complex logic possible
Case Sensitivity	`"Red"` ≠ `"red"`

🧠 Practice Task (for you):

Create a program that:

- Asks for user's age
- If age < 13 → print "You are a child"
- If age between 13-19 → print "You are a teenager"
- If age between 20-59 → print "You are an adult"
- Else → print "You are a senior citizen"
