



ACCA4020

Python – Decision Making

Dr. Seena Joseph

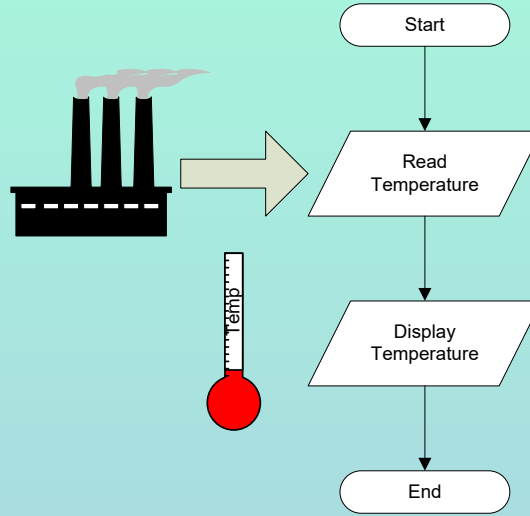
Decision making

- Programming is not just about reading, processing, and displaying data—it's about making intelligent decisions based on that data.
- Decision making – process of making choices based on certain conditions.



Decision making

- Reading the temperature from a nuclear reactor.



-  If the temperature is normal, the reactor is safe.
-  If the temperature goes too high ????

Decision making

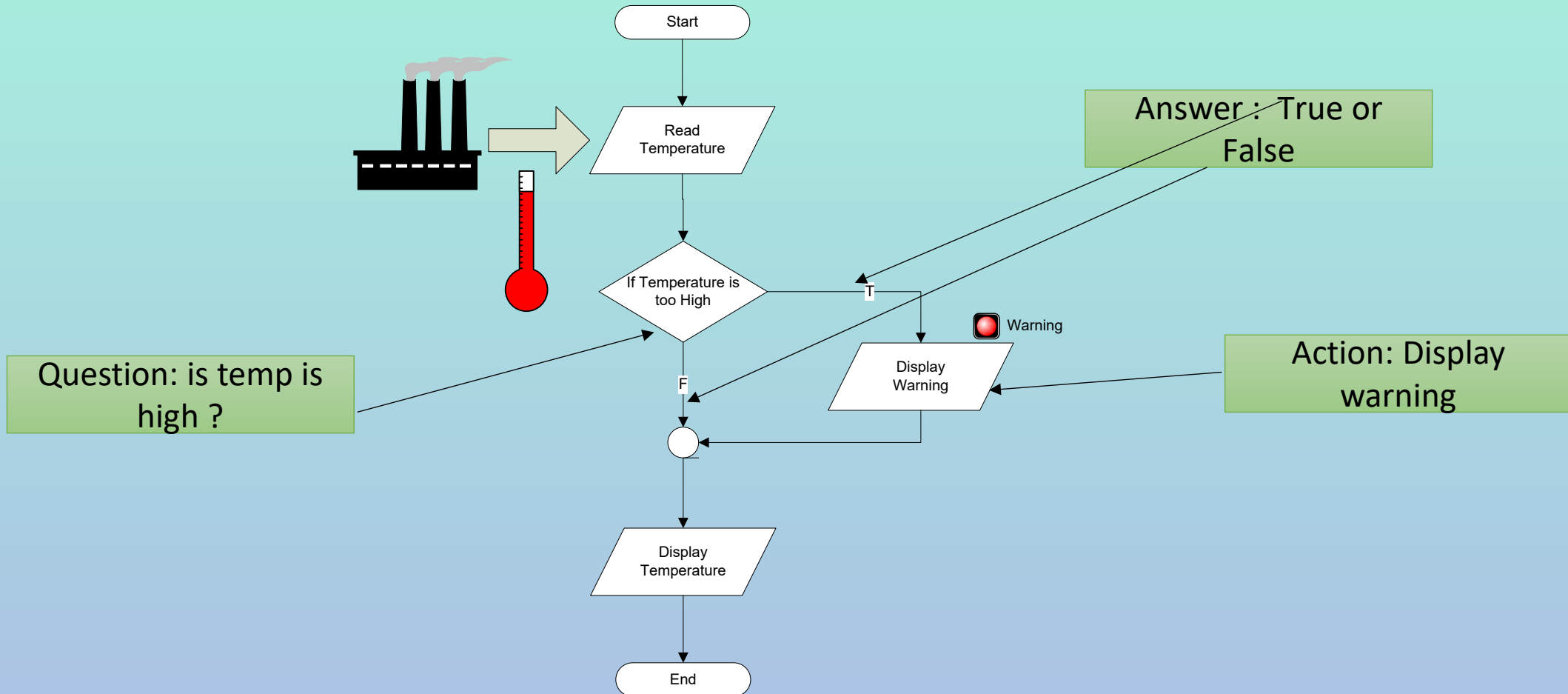
How decision making works ?

- Every decision starts with a **question**.
- The answer must be either **TRUE** or **FALSE**
- Programs take an **action** based on the answer

1. Ask a simple question

Decision making

- If the temperature goes to high (800 degrees) then the reactor explodes, so how we may want to display some alarm when this happens.



Decision making

- Decision making is based on questions that can be answered with TRUE or FALSE.
- To ask these questions in programming, **conditional (comparison) operators** are used.
- Conditional or comparison operator are to compare two values

Operator	Meaning	Example Question	True	False
==	Equal to	Is the mark same ?	45==45 	45 == 50 
!=	Not equal to	Is the time not equal ?	9!=5 	9!= 9 
>	Greater than	Is the temperature too high ?	45 > 40 	35 > 40 
<	Less than	Is the speed below the limit ?	45 < 60 	65< 60 
>=	Greater than or equal to	Is the phone battery charge enough (50 or above ?)	65 >= 50 	45 >= 50 
<=	Less than or equal to	Is the phone battery charge low (20 or below ?)	11 <= 20 	34 <= 20 

Decision making

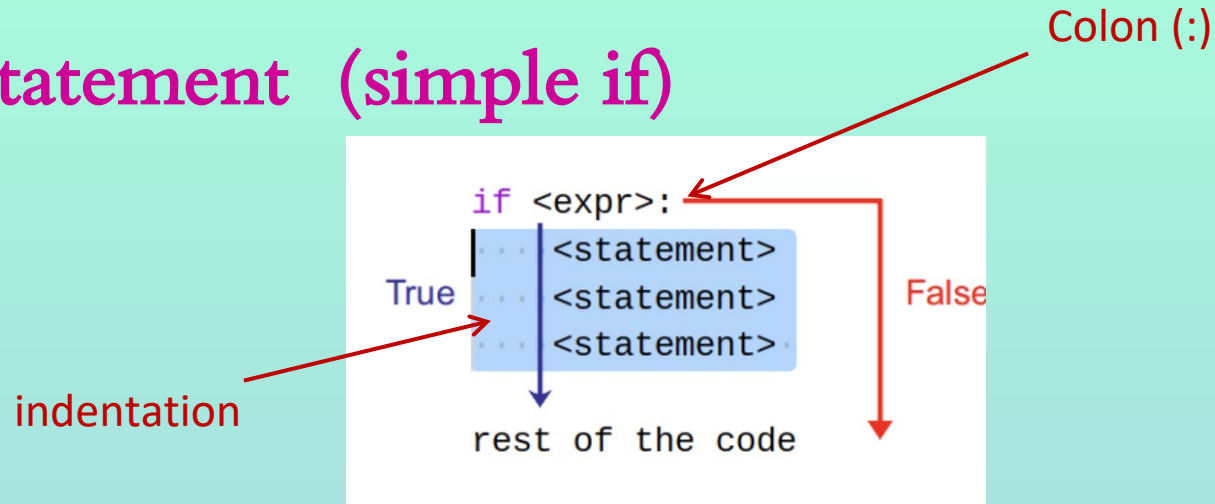
- Python use **if** to make decision

Operator	Meaning	Python
==	Equal to	if (x==y)
!=	Not equal to	if (x!=y)
>	Greater than	if (x>y)
<	Less than	if (x<y)
>=	Greater than or equal to	if (x>=y)
<=	Less than or equal to	if (x<=y)



Decision making

- if statement (simple if)



- Python code Example:

```
num1=int(input('Enter a number'))
if num1 > 0:
    print(f'The input number {num1} is a positive number')
```

```
Enter a number 45
The input number 45 is a positive number
```

Decision making

- if statement (simple if)

```
num1=int(input('Enter a number'))  
if num1 > 0:  
    print(f'The input number {num1} is a positive number')
```

```
Enter a number 45  
The input number 45 is a positive number
```

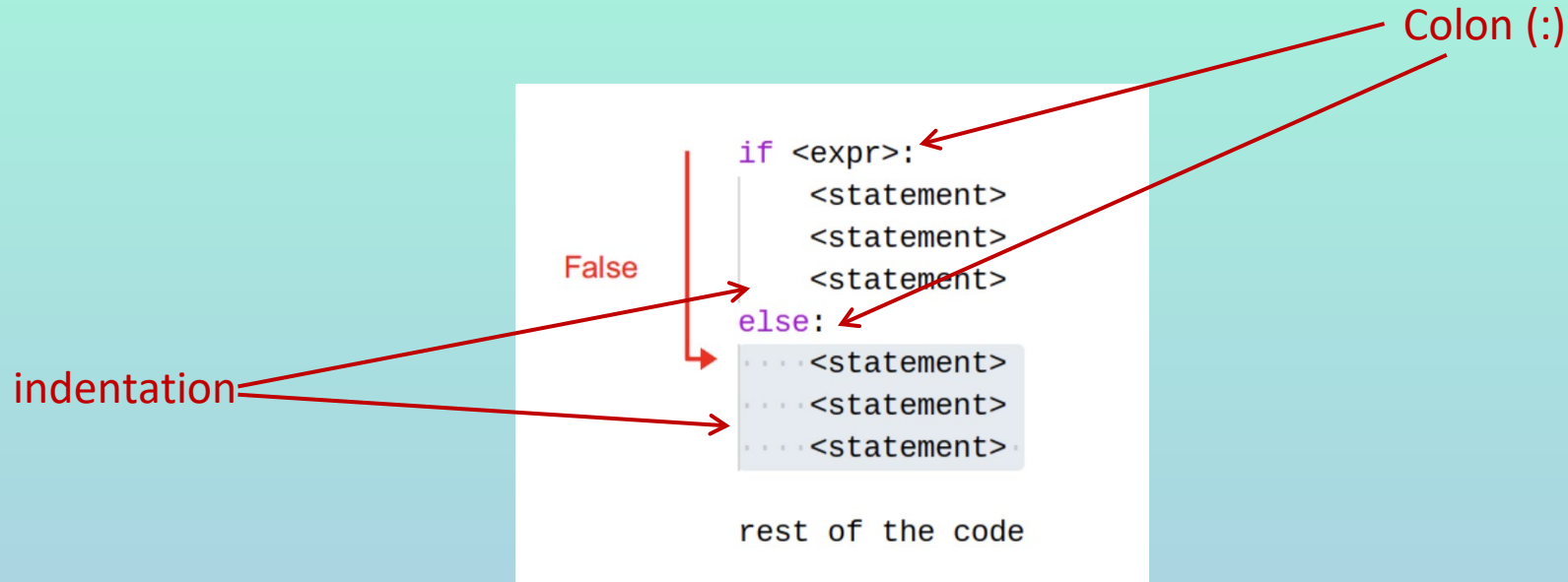
In this example, if the number is greater than 0 is **True** then the program will print 'The input number is a positive number'.

If the condition is **False** nothing happened.

Decision making

- **if else statement**

if else statement provides an alternative block of code to execute if the condition is **false**.



Python code Example:

```
Age=int(input('Enter Age'))  
if Age >=18:  
    print('You are an adult')  
else:  
    print('you are not an adult')
```

Sample output:

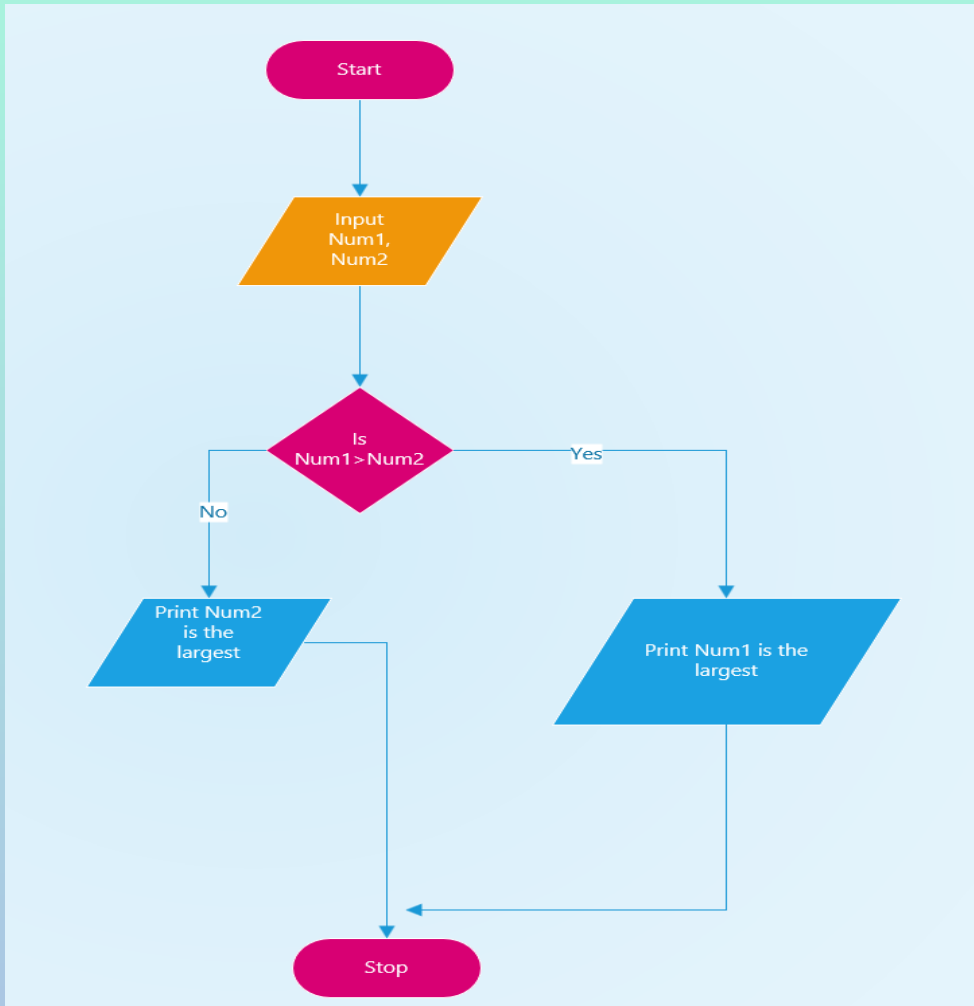
```
Enter Age 15  
you are not an adult
```

```
Enter Age 30  
You are an adult
```

Decision making

- if else statement

Is this an example for if else ?



Decision making

- if else statement

```
Num1=int(input('Enter First Number'))
Num2=int(input('Enter Second Number'))
if Num1 > Num2:
    print(f'First number, {Num1} is the largest')
else:
    print(f'Second number, {Num2} is the largest')
```

Sample output

```
Enter First Number 34
Enter Second Number 45
Second number, 45 is the largest
```

```
Enter First Number 56
Enter Second Number 43
First number, 56 is the largest
```

Exercise 1

Write a Python program that:

- Task 1
 - asks the user to input a number and prints whether it is positive or negative
- Task 2
 - checks if a person is eligible for a senior citizen discount. The person should be 65 or older to be eligible.
- Task 3
 - Check if a Number is even or odd.
- Task 4
 - Check if a Number is Divisible by 5

Exercise 2

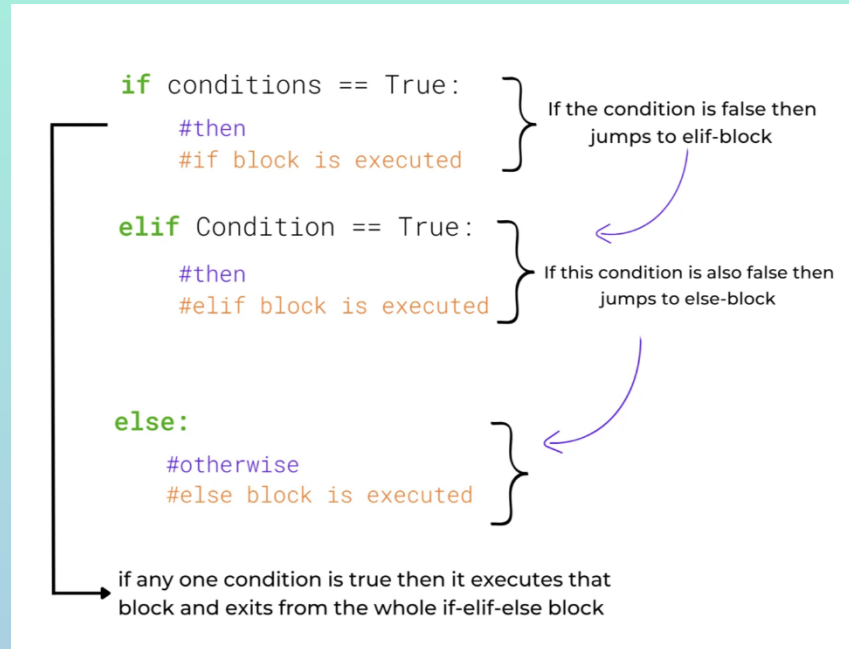
Write a Python program that demonstrates the following ATM withdrawal process.

1. Start
2. Input PIN
3. Is PIN correct?
 1. Yes: Go to step 4
 2. No: Print "Incorrect PIN" and End
4. Input withdrawal amount
5. Is amount $\leq$ balance?
 1. Yes: Print "Withdrawal successful" , update balance and print the current balance
 2. No: Print "Insufficient funds"
6. End

Decision making

- **if elif else statement**

if elif else statement allows to check multiple conditions. It's way to handle more than two possibilities. The *elif* (else if) allows to check for additional conditions if the previous condition is false.



Decision making

- **if elif else statement**

if elif else statement allows to check multiple conditions. It's way to handle more than two possibilities. The *elif* (else if) allows to check for additional conditions if the previous condition is false.

Python code Example:

```
Num1=int(input('Enter First Number'))
Num2=int(input('Enter Second Number'))
if Num1 > Num2:
    print(f'First number, {Num1} is the largest')
elif Num2 > Num1:
    print(f'Second number, {Num2} is the largest')
else:
    print(f' Both input numbers, {Num1} and {Num2} are equal')
```

Sample output:

```
Enter First Number 56
Enter Second Number 43
First number, 56 is the largest
```

```
Enter First Number 34
Enter Second Number 45
Second number, 45 is the largest
```

```
Enter First Number 34
Enter Second Number 34
Both input numbers, 34 and 34 are equal
```

Decision making

- if elif else statement

Python code Example: To check if a number is positive, negative or Zero

```
Num=int(input('Input a Number'))

if Num > 0:
    print(f'The number, {Num} is positive ')
elif Num < 0:
    print(f'The number, {Num} is Negative ')
else:
    print(f'The number is Zero ')
```

Sample output:

```
Input a Number 9
The number, 9 is positive
```

```
Input a Number -78
The number, -78 is Negative
```

```
Input a Number 0
The number is Zero
```

Exercise 3

Write a Python program that:

- Task 1

- Write a program that takes the student's score and prints their grade based on the following criteria:
 - 90 and above: "A"
 - 80 to 89: "A"
 - 70 to 79: "B"
 - 60 to 69: "C"
 - 50 to 59: "D"
 - Below 50: "F"

Decision making

Logical operators

What if we need to check multiple conditions together?”

Example: Consider a scenario where a person can vote only if they are at least 18 years old and a citizen.

Python code Example:

Nested if

```
Age=int(input('Enter your Age :'))
CitizenStatus=input('Are you a citizen ? YES/NO :')

if Age>=18:
    if CitizenStatus=='YES':
        print('You are eligible to Vote')
    else:
        print('You are not eligible to Vote')
else:
    print('You are not eligible to Vote')
```

This works, but it is **nested** and **repetitive**.

Can we simplify it ?

Decision making

Logical operators

YES, we can simplify the previous code using **Logical operators**.

Logical operators allows to combine multiple conditions into a single statement

Logical Operator	Meaning	Python
and	True if both conditions are True	if age >= 18 and citizen == "yes":
or	True at least one condition is True	if age < 18 or student == "yes":
not	Reverse a condition	if not is_admin:

Decision making

Logical operator: and

Python code Example: Revised version for the previous code

```
Age=int(input('Enter your Age :'))
CitizenStatus=input('Are you a citizen ? YES/NO :')

if Age>=18 and CitizenStatus=='YES':
    print('You are eligible to Vote')
else:
    print('You are not eligible to Vote')
```

Sample output:

```
Enter your Age :35
Are you a citizen ? YES/NO :NO
You are not eligible to Vote
```

```
Enter your Age : 15
Are you a citizen ? YES/NO : YES
You are not eligible to Vote
```

```
Enter your Age :45
Are you a citizen ? YES/NO :YES
You are eligible to Vote
```

Decision making

Logical operator: or

Python code Example: A person gets a discount if they are below 18 or a student.

```
Age=int(input('Enter your Age :'))
CitizenStatus=input('Are you a citizen ? YES/NO :')

if Age<=18 and CitizenStatus=='YES':
    print('You are eligible to Vote')
else:
    print('You are not eligible to Vote')
```

Sample output:

```
Enter your Age :15
Are you a student ? YES/NO :YES
You get a discount
```

```
Enter your Age :12
Are you a student ? YES/NO :NO
You get a discount
```

```
Enter your Age :56
Are you a student ? YES/NO :NO
No discount available
```

```
Enter your Age :45
Are you a student ? YES/NO :YES
You get a discount
```

Why or ? : The person **only needs to satisfy one condition** to get the discount.

Decision making

Logical operator: not

Python code Example: to check is a number is zero.

```
Number= int(input('Enter a number :'))

if not (Number == 0):
    print ('the number is not ZERO')
else:
    print('the number is ZERO')
```

Sample output:

```
Enter a number 8
the number is not ZERO
```

```
Enter a number :0
the number is ZERO
```

Why or ? : The person **only needs to satisfy one condition** to get the discount.

Decision making

Logical operators

- They simplify conditions, reducing unnecessary if statements.
- They make code cleaner and easier to read.
- They allow us to combine multiple conditions efficiently.

Exercise 4

Write a Python program that:

- Find the largest among three numbers.
- Find the smallest of three numbers.
- Check if a given year is a leap year.
- Check if a Number is Divisible by 3 and 5.