Python

Variable is a reusable container for storing a value in a program. It acts as a named placeholder that can hold different values during the execution of the program. The value stored in a variable can be changed or updated as the program runs, making variables a dynamic and essential concept in programming

a variable behaves as if it were the value it contains. In programming, a variable is essentially a symbolic name or identifier associated with a storage location that holds a value. When you use a variable in operations or expressions, the program treats it as if it were the actual value it holds.

2 + x = 5 x = 3

In [154...
```
age = 21
print (age)
```

```
21
```

In [155...
```
print("You are " + str(age) + " years old") #put space after quotation mark
```

```
You are 21 years old
```

In [156...
```
print ("You are", age, "years old") # No need to put space after quotation mark
```

```
You are 21 years old
```

In [157...
```
print(f"You are {age} years old") # Python f-strings provide a quick way to interp
```

```
You are 21 years old
```

PYTHON DATA TYPES

INTEGER

In [158...
```
age = 21
players = 2
quantity = 5
```

In [159...
```
print(f"You are {age} years old")
print(f"There are {players} players online")
print(f"You would like to buy {quantity} items")
```

```
You are 21 years old
There are 2 players online
You would like to buy 5 items
```

FLOAT

In [160...
```
gpa = 3.8
distance = 2.5
price = 10.99
```

In [161...
```
print(f"Your gpa is {gpa}")
print(f"You ran {distance}Km")
print(f"The price is ${price}")
```

```
Your gpa is 3.8
You ran 2.5Km
The price is $10.99
```

STRING

In [162...
```
name = "Salim"
food = "Pizza"
email = "diwania.salim@gmail.com"
```

In [163...
```
print(f"Hello {name}")
print(f"You like {food}")
print(f"Your email is : {email}")
```

```
Hello Salim
You like Pizza
Your email is : diwania.salim@gmail.com
```

BOOLEAN

In [164...
```
online = True #Do not write boolean within a quotes
for_sale = False
running = True
```

In [166...
```
print(f"Are you online? : {online}")
print(f"Is the item for sale?: {for_sale}")
print(f"Game running : {running}")
```

```
Are you online? : True
Is the item for sale?: False
Game running : True
```

In [167...
```
x = 1
y = 2
z = 3
```

In [168...
```
print(x)
print(y)
print(z)
```

```
1
2
3
```

In [169...
```
x,y,z = 1,2,3
```

In [170...
```
print(x)
print(y)
print(z)
```

```
1
2
3
```

In [171...
```
x=y=z=0
```

In [172...
```
print(x)
print(y)
print(z)
```

```
0
0
0
```

TYPE CASTING

Typecasting is a fundamental concept in programming where you convert a value from one data type to another. This is often necessary when you want to perform operations that involve different data types or when you need a value in a particular format. (string, integer, float, boolean)

In [173...
```
name = "Salim"
age = 40
gpa = 3.8
UDOM_STAFF = True
```

In [174...
```
print(type(name))
print(type(age))
print(type(gpa))
print(type(UDOM_STAFF))
```

```
<class 'str'>
<class 'int'>
<class 'float'>
<class 'bool'>
```

Explicit typecasting

Explicit typecasting, also known as manual type conversion or casting, occurs when a programmer intentionally converts a variable from one data type to another using predefined functions or constructors. This conversion is performed by calling specific functions that represent the desired data type.

In [175...
```
age = float(age)
print(age)
```

```
40.0
```

In [176…
```
print(type(age))
```

```
<class 'float'>
```

In [177…
```
gpa = int(gpa)
print(gpa)
```

```
3
```

In [178…
```
UDOM_STAFF = str(UDOM_STAFF)
print(UDOM_STAFF)
```

```
True
```

In [179…
```
print(type(UDOM_STAFF))
```

```
<class 'str'>
```

In [180…
```
age = bool(age)
print(age)
```

```
True
```

In [181…
```
name = bool(name)
print(name)
```

```
True
```

Implicit Typecasting

Implicit typecasting, also known as automatic type conversion, occurs when the Python interpreter automatically converts one data type to another without the programmer explicitly requesting the conversion. This converstion takes place in certain situations, especially during operations involving different data types.

The goal of implicit typecasting is to facilitate the interaction between different data types in expressions and operations. Python automatically promotes or demotes data types as needed to perform the operation.

In [182…
```
x = 2
y = 2.0
x = x/y
print(x)
```

```
1.0
```

In [183…
```
result = 5 + 3.2 # Here the integer 5 is implicitly converted to a float 3.2
print(result)
```

```
8.2
```

In [184...
```
message = "Hello, " + "World!" #Here two string are concatenated using the '+' ope
print(message)
```

```
Hello, World!
```

In [185...
```
comparison_result = 10 > 5.5
print(comparison_result)
```

```
True
```

User input

In Python, you can use the 'input()' function to take user input. The 'input()' function reads a line from the user, converts it into a string, and returns that string.

user_input = input("Enter something: ")

print("You entered:", user_input)

In [186...
```
name = input("Enter your name: ")
age = input("Enter your age: ")
```

```
Enter your name: Salim Diwani
Enter your age: 40
```

In [187...
```
print(f"Hello {name}")
print(f"You are {age} years old")
```

```
Hello Salim Diwani
You are 40 years old
```

In [188...
```
name = input("Enter your name: ")
age = int(input("Enter your age: "))
age = age + 1
print(f"Hello {name}")
print(f"You are {age} years old")
```

```
Enter your name: Salim Diwani
Enter your age: 40
Hello Salim Diwani
You are 41 years old
```

In [99]:
```
name = input("Enter your name: ")
age = float(input("Enter your age: "))
age = age + 1
print(f"Hello {name}")
print(f"You are {age} years old")
```

```
Enter your name: salim
Enter your age: 40
Hello salim
You are 41.0 years old
```

Exercise: mad libs. Mad libs is were user develop a story and can enter verb, adjective,

or noun

In [189...

```
adjective1 = input("Enter an adjective: ")
noun = input("Enter a noun: ")
adjective2 = input("Enter an adjective: ")
verb = input("Enter a verb: ")
adjective3 = input("Enter an adjective: ")

print(f"Today I went to a {adjective1} zoo. ")
print(f"In an exhibit, I saw {noun}")
print(f"{noun} was {adjective2} and {verb}ing")
print(f"I was {adjective3}")
```

```
Enter an adjective: suspecious
Enter a noun: Mark Zuckerburg
Enter an adjective: berserk
Enter a verb: screech
Enter an adjective: amazed
Today I went to a suspecious zoo.
In an exhibit, I saw Mark Zuckerburg
Mark Zuckerburg was berserk and screeching
I was amazed
```

Exercise two: Calculate Area of Rectangle

In [190...

```
length = float(input("Enter the length of a rectangle: "))
width = float(input("Enter the width of a rectangle: "))
area = length * width
print(f"The area is: {area}cm^2")
```

```
Enter the length of a rectangle: 9
Enter the width of a rectangle: 15
The area is: 135.0cm^2
```

In [191...

```
length = float(input("Enter the length of a rectangle: "))
width = float(input("Enter the width of a rectangle: "))
height = float(input("Enter the height of a rectangle: "))
volume = length * width * height
print(f"The volume is: {volume}cm^3")
```

```
Enter the length of a rectangle: 9
Enter the width of a rectangle: 15
Enter the height of a rectangle: 7
The volume is: 945.0cm^3
```

Exercise three: Shopping cart

In [192...

```
item = input("What item would you like to buy?: ")
price = float(input("What is the price?: "))
quantity = int(input("How many would you like?: "))

total = price * quantity

print(f"You have bought {quantity} x {item}/s")
print(f"Your total is: ${round(total, 2)}")
```

```
What item would you like to buy?: Pizza
```

```
What is the price?: 4.99
How many would you like?: 9
You have bought 9 x Pizza/s
Your total is: $44.91
```

Maths in Python

In [104...

```
friends = 0
friends = friends + 1
print(friends)
```

```
1
```

augmented assignment operators provide a short way to perform a binary operation and assigning results back to one of the operands. The way to write an augmented operator is just to write that binary operator and assignment operator together. In Python, we have several different augmented assignment operators like +=, -=, *=, /=, //=, **=, |=, &=, >>=, <<=, %= and ^=.

In [193...

```
friends = 0
friends += 1 #Argumented assignment operator
print(friends)
```

```
1
```

In [194...

```
friends = 0
friends = friends - 2
print(friends)
```

```
-2
```

In [195...

```
friends = 0
friends -= 2
print(friends)
```

```
-2
```

In [196...

```
friends = 5
friends = friends * 3
print(friends)
```

```
15
```

In [197...

```
friends = 5
friends *= 3
print(friends)
```

```
15
```

In [198...

```
friends = 5
friends = friends / 2
print(friends)
```

```
2.5
```

In [199…
```
friends = 5
friends /= 2
print(friends)
```
2.5

In [200…
```
friends = 5
friends = friends **2
print(friends)
```
25

In [201…
```
friends = 5
friends **=  2
print(friends)
```
25

In [202…
```
friends = 10
reminder = friends % 3
print(reminder)
```
1

Built in function in Python

In [203…
```
x = 3.14
y = 4
z = 5
result = round(x)
print(result)
```
3

In [204…
```
x = 3.14
y = -4
z = 5
result = abs(y)
print(result)
```
4

In [205…
```
result = pow(4, 3)
print(result)
```
64

In [206…
```
x = 3.14
y = -4
z = 5
result= max(x, y, z)
print(result)
```
5

In [207…

```
x = 3.14
y = -4
z = 5
result = min(x, y, z)
print(result)
```

```
-4
```

In [208…

```
import math
print(math.pi)
```

```
3.141592653589793
```

In [209…

```
import math
print(math.e) #e is exponential constant. It is used in physics most of the time
```

```
2.718281828459045
```

In [210…

```
import math
x = 9
result = math.sqrt(x)
print(result)
```

```
3.0
```

In [211…

```
import math
x = 9.1
result = math.ceil(x) #round off the result up
print(result)
```

```
10
```

In [212…

```
import math
x = 9.9
result = math.floor(x) #round off the result down
print(result)
```

```
9
```

Exercise 1

calculate circumference of a circle

Formula for Circumference of a Circle: C = 2 *pi* r

In [213…

```
import math
radius = float(input("Enter the radius of a circle: "))
circumference = 2 * math.pi * radius
print(f"The circumference is: {circumference}")
```

```
Enter the radius of a circle: 10.5
The circumference is: 65.97344572538566
```

In [214...

```
import math
radius = float(input("Enter the radius of a circle: "))
circumference = 2 * math.pi * radius
print(f"The circumference is: {round (circumference, 2)}cm")
```

```
Enter the radius of a circle: 10.5
The circumference is: 65.97cm
```

calculate area of a circle

Formula for area of a circle: A = pi *r* r

In [215...

```
import math
radius = float(input("Enter the radius of a circle: "))
area = math.pi * pow(radius, 2)
print(f"The area of the circle is: {area}cm^2")
```

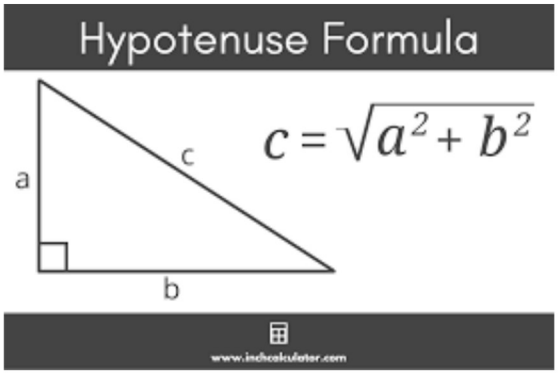
```
Enter the radius of a circle: 10.5
The area of the circle is: 346.3605900582747cm^2
```

In [216...

```
import math
radius = float(input("Enter the radius of a circle: "))
area = math.pi * pow(radius, 2)
print(f"The area of the circle is: {round(area, 2)}cm^2")
```

```
Enter the radius of a circle: 10.5
The area of the circle is: 346.36cm^2
```

Calculate hypotenuse of a right angle triangle



In [129...

```
import math
a = float(input("Enter side A: "))
b = float(input("Enter side B: "))
c = math.sqrt(pow(a, 2) + pow(b, 2))
print(f"Side C = {c}")
```

```
Enter side A: 5
Enter side B: 6
Side C = 7.810249675906654
```

If statements in Python (if, elif, else)

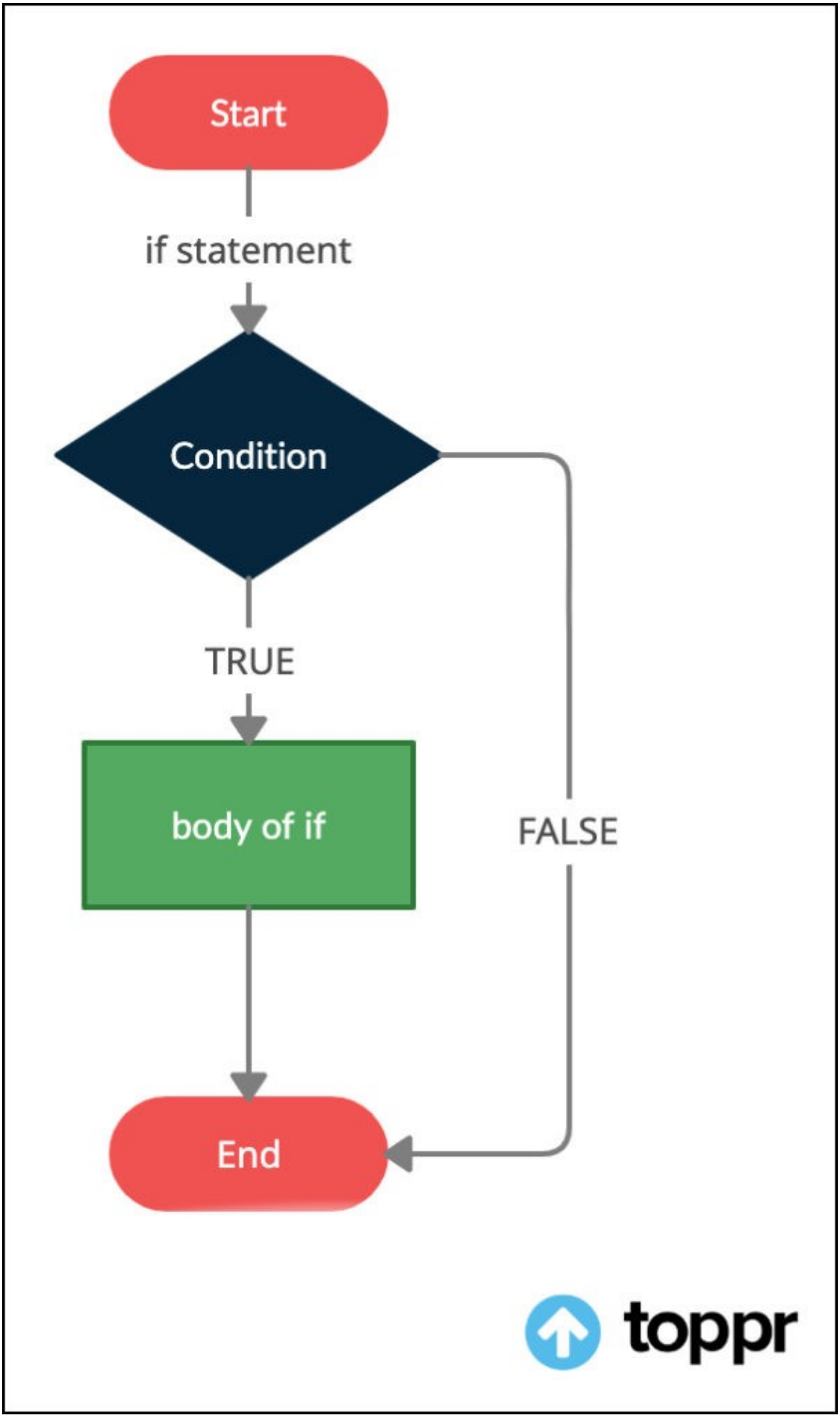
In Python, the If statement is a fundamental control flow statement that allows you to conditionally execute a block of code. it enables your program to make decisions and execute specific instructions based on whether a given condition is True or False.

Basic Syntax: if condition:

```
    # Code block to execute if the condition is True
```

if = Do some code only if some condition is True

```
if condition:
     statements(s)
```



The if statement in Python is used to make decisions. It comprises of a piece of code that only

executes when the if statement's condition is TRUE. If the condition is FALSE, then the condition will not be executed

Python supports the standard mathematical logical conditions:

Equals: a == b Not Equals: a != b Less than: a < b Less than or equal to: a <= b Greater than: a > b Greater than or equal to: a >= b

```
# Python program to illustrate different if conditions
age = 16
limit = 25

if (age == 16):
    print('Age is correct as mentioned')

if (age < 25):
    print('Below age')

if (age > 10):
    print('Teen')

if (age <= limit):
    print('Under age limit')

if (age >= limit):  # will not be printed because the condition is FALSE
    print('Above limit')
```

In [217…

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

if age >= 18:
    print("You are now signed up!")
```

```
Enter your age: 40
You are now signed up!
```

In [218…

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

if age >= 18:
    print("You are now signed up!")
```

```
Enter your age: 13
```

In [219…

```
x = 10
if x > 5:
    print("x is greater than 5")
```

```
x is greater than 5
```

In [220…

```
x = 4
if x > 5:
    print("x is greater than 5")
```

Else Clause

You can extend the if statement with an else clause to provide an alternative block of code to execute when the if condition is False

The basic syntax for if-else statements in Python

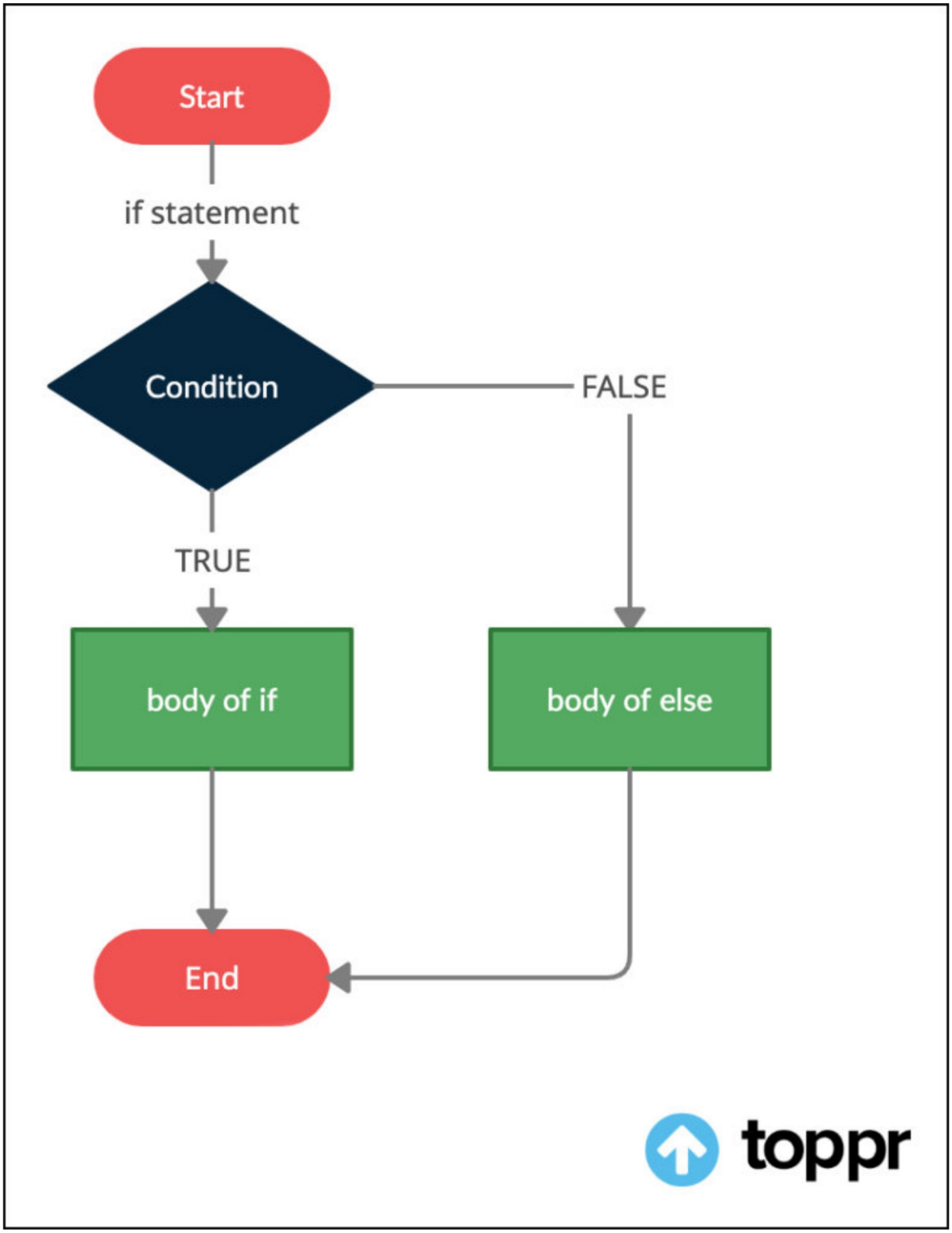
if condition:

```
    #code block to execute if the condition is True
```

else:

```
    #code block to execute if the condition is False
```

```
if condition:
     statements(s)
else:
     statements(s)
```



The if-else statement comprises of 2 blocks of statements out of which the first set of code executes only when the if statement's condition is true. If the condition is false, then the statements in the body of the else section will be executed.

In [221…

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

if age >= 18:
    print("You are now signed up!")
else:
    print("You must be 18+ to sign up")
```

```
Enter your age: 13
You must be 18+ to sign up
```

In [222...

```
y = 3
if y > 5:
    print("y is greater than 5")
else:
    print("y is not greater than 5")
```

```
y is not greater than 5
```

In [223...

```
y = 10
if y > 5:
    print("y is greater than 5")
else:
    print("y is not greater than 5")
```

```
y is greater than 5
```

In [224...

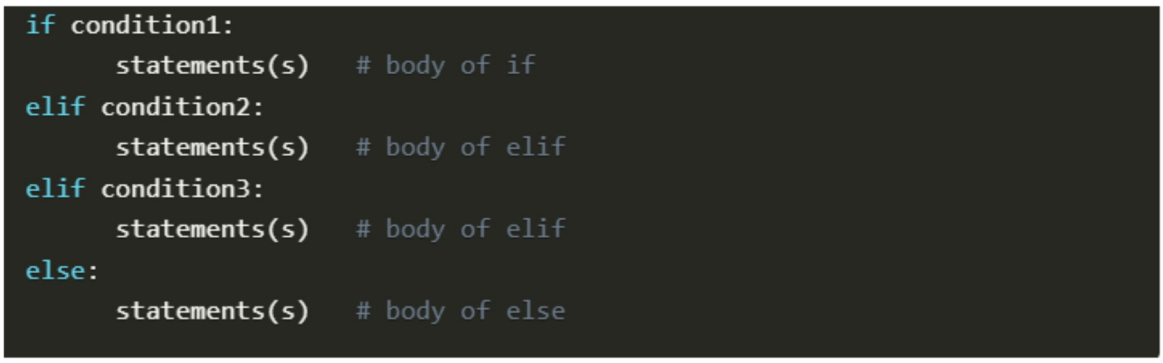
```
# Python program to compare between subject marks
python = 45
graphics = 57

if (python > graphics):
     print('Python marks greater than Graphics')
else:
     print('Graphics marks greater than Python')
```

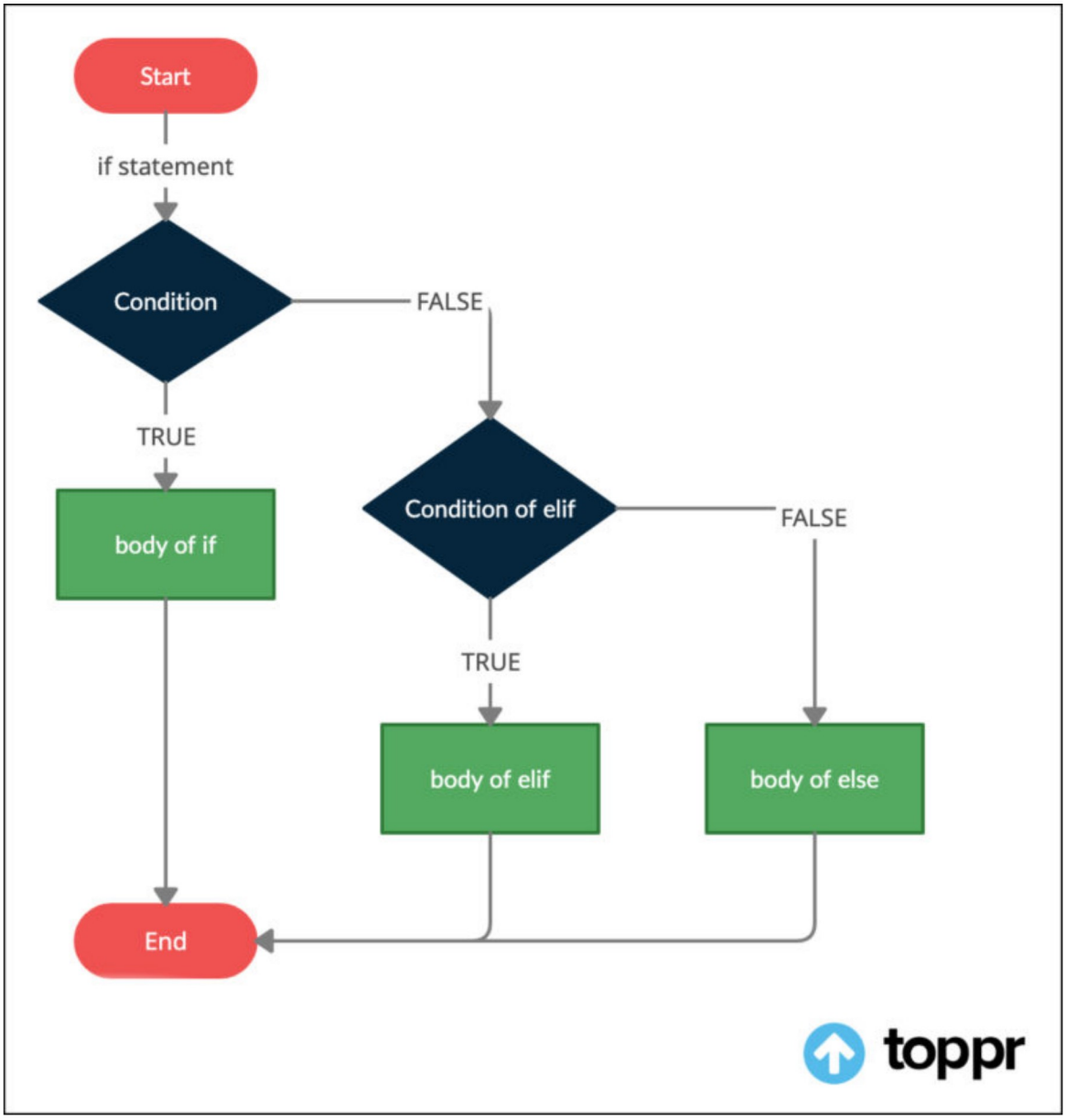
```
Graphics marks greater than Python
```

Elif Clause

Elif Clause: The elif (else if) statement is used to check multiple conditions. It comes after an if statement and before an optional else statement.



elif is short for else if. If the condition for if is FALSE then the elif condition is checked and executed if found TRUE or, the statements after elif are executed.



In [226…

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

if age >= 18:
    print("You are now signed up!")
elif age < 0:
    print("You haven't been born yet!")
else:
    print("You must be 18+ to sign up")
```

```
Enter your age: 20
You are now signed up!
```

In [227…

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

if age >= 18:
    print("You are now signed up!")
elif age < 0:
    print("You haven't been born yet!")
elif age >= 100:
    print("You are too old to sign up")
else:
    print("You must be 18+ to sign up")
```

```
Enter your age: -1
You haven't been born yet!
```

In [228…

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

if age >= 100:
    print("You are too old to sign up")
elif age >= 18:
     print("You are now sign up")
elif age < 0:
    print("You have not been born yet")
else:
    print("You must be 18+ to sign up")
```

```
Enter your age: 111
You are too old to sign up
```

In [229…

```
response = input("Would you like food? (Y/N): ")

if response == "Y":
    print("Have some food!")
else:
    print("No food for you!")
```

```
Would you like food? (Y/N): Y
Have some food!
```

In [230…

```
name = input("Enter your name: ")

if name == "":
    print("You did not type in your name!")
else:
    print(f"Hello {name}")
```

```
Enter your name: Salim Diwani
Hello Salim Diwani
```

In [231…

```
name = input("Enter your name: ")

if name == "":
    print("You did not type in your name!")
else:
    print(f"Hello {name}")
```

```
Enter your name:
You did not type in your name!
```

In [232…

```
for_sale = True

if for_sale:
    print("This item is for sale")
else:
    print("This item is NOT for sale")
```

```
This item is for sale
```

In [233…

```
online = False

if online:
    print("The user is online")
else:
    print("The user is offline")
```

```
The user is offline
```

In [234…

```
online = True

if online:
    print("The user is online")
else:
    print("The user is offline")
```

```
The user is online
```

Simple Calculator Program Application

In [235…

```
operator = input("Enter an operator (+ - * /): ")
num1 = float(input("Enter the 1st number: "))
num2 = float(input("Enter the 2nd number: "))

if operator == "+":
    result = num1 + num2
    print(round(result, 3))
elif operator == "-":
    result = num1 - num2
    print(round(result, 3))
elif operator == "*":
    result = num1 * num2
    print(round(result, 3))
elif operator == "/":
    result = num1 / num2
    print(round(result, 3))
else:
    print(f"{operator} is not a valid operator")
```

```
Enter an operator (+ - * /): +
Enter the 1st number: 3
Enter the 2nd number: 3.14
6.14
```

In [236…

```
operator = input("Enter an operator (+ - * /): ")
num1 = float(input("Enter the 1st number: "))
num2 = float(input("Enter the 2nd number: "))

if operator == "+":
    result = num1 + num2
    print(round(result, 3))
elif operator == "-":
    result = num1 - num2
    print(round(result, 3))
elif operator == "*":
    result = num1 * num2
    print(round(result, 3))
elif operator == "/":
    result = num1 / num2
    print(round(result, 3))
else:
    print(f"{operator} is not a valid operator")
```

```
Enter an operator (+ - * /): -
Enter the 1st number: 3
Enter the 2nd number: 3.14
-0.14
```

In [237…

```
operator = input("Enter an operator (+ - * /): ")
num1 = float(input("Enter the 1st number: "))
num2 = float(input("Enter the 2nd number: "))

if operator == "+":
    result = num1 + num2
    print(round(result, 3))
elif operator == "-":
    result = num1 - num2
    print(round(result, 3))
elif operator == "*":
    result = num1 * num2
    print(round(result, 3))
elif operator == "/":
    result = num1 / num2
    print(round(result, 3))
else:
    print(f"{operator} is not a valid operator")
```

```
Enter an operator (+ - * /): *
Enter the 1st number: 3
Enter the 2nd number: 3.14
9.42
```

In [238...

```
operator = input("Enter an operator (+ - * /): ")
num1 = float(input("Enter the 1st number: "))
num2 = float(input("Enter the 2nd number: "))

if operator == "+":
    result = num1 + num2
    print(round(result, 3))
elif operator == "-":
    result = num1 - num2
    print(round(result, 3))
elif operator == "*":
    result = num1 * num2
    print(round(result, 3))
elif operator == "/":
    result = num1 / num2
    print(round(result, 3))
else:
    print(f"{operator} is not a valid operator")
```

```
Enter an operator (+ - * /): /
Enter the 1st number: 3
Enter the 2nd number: 3.14
0.955
```

In [239...

```
operator = input("Enter an operator (+ - * /): ")
num1 = float(input("Enter the 1st number: "))
num2 = float(input("Enter the 2nd number: "))

if operator == "+":
    result = num1 + num2
    print(round(result, 3))
elif operator == "-":
    result = num1 - num2
    print(round(result, 3))
elif operator == "*":
    result = num1 * num2
    print(round(result, 3))
elif operator == "/":
    result = num1 / num2
    print(round(result, 3))
else:
    print(f"{operator} is not a valid operator")
```

```
Enter an operator (+ - * /): Pizza
Enter the 1st number: 3
Enter the 2nd number: 3.14
Pizza is not a valid operator
```

Weight Converter Program

In [240…

```
weight = float(input("Enter your weight: "))
unit = input("Kilograms or Pounds? (K or L): ")

if unit == "K":
    weight = weight * 2.205
    unit = "Lbs."
    print(f"Your weight is: {round(weight, 1)} {unit}")
elif unit == "L":
    weight = weight / 2.205
    unit = "Kgs."
    print(f"Your weight is: {round(weight, 1)} {unit}")
else:
    print(f"{unit} was not valid")
```

```
Enter your weight: 180
Kilograms or Pounds? (K or L): K
Your weight is: 396.9 Lbs.
```

In [241…

```
weight = float(input("Enter your weight: "))
unit = input("Kilograms or Pounds? (K or L): ")

if unit == "K":
    weight = weight * 2.205
    unit = "Lbs."
    print(f"Your weight is: {round(weight, 1)} {unit}")
elif unit == "L":
    weight = weight / 2.205
    unit = "Kgs."
    print(f"Your weight is: {round(weight, 1)} {unit}")
else:
    print(f"{unit} was not valid")
```

```
Enter your weight: 180
Kilograms or Pounds? (K or L): L
Your weight is: 81.6 Kgs.
```

Temperature Conversion program

In [153…

```
unit = input("Is this temperature in Celsius or Fahrenheight (C/F): ")
temp = float(input("Enter the temperature: "))

if unit == "C":
    temp = round((9 * temp) / 5 + 32, 1)
    print(f"The temperature in Fahrenheight is: {temp}F")
elif unit == "F":
    temp = round((temp - 32)* 5/9, 1)
    print(f"The temperature in Celsius is: {temp}C")
else:
    print(f"{unit} is an invalid unit of measurement")
```

```
Is this temperature in Celsius or Fahrenheight (C/F): C
Enter the temperature: 33
The temperature in Fahrenheight is: 91.4F
```

In [242…

```
unit = input("Is this temperature in Celsius or Fahrenheight (C/F): ")
temp = float(input("Enter the temperature: "))

if unit == "C":
    temp = round((9 * temp) / 5 + 32, 1)
    print(f"The temperature in Fahrenheight is: {temp}F")
elif unit == "F":
    temp = round((temp - 32)* 5/9, 1)
    print(f"The temperature in Celsius is: {temp}C")
else:
    print(f"{unit} is an invalid unit of measurement")
```

```
Is this temperature in Celsius or Fahrenheight (C/F): F
Enter the temperature: 33
The temperature in Celsius is: 0.6C
```

In [ ]: