Basic Programming I

Conditions

DigitalSkola
Uncover The World Of Digital Skills With Us

Quote of The Day

It's time to start living the life you've imagined.

- Henry James

DigitalSkola
Uncover The World Of Digital Skills With Us

Table of Content

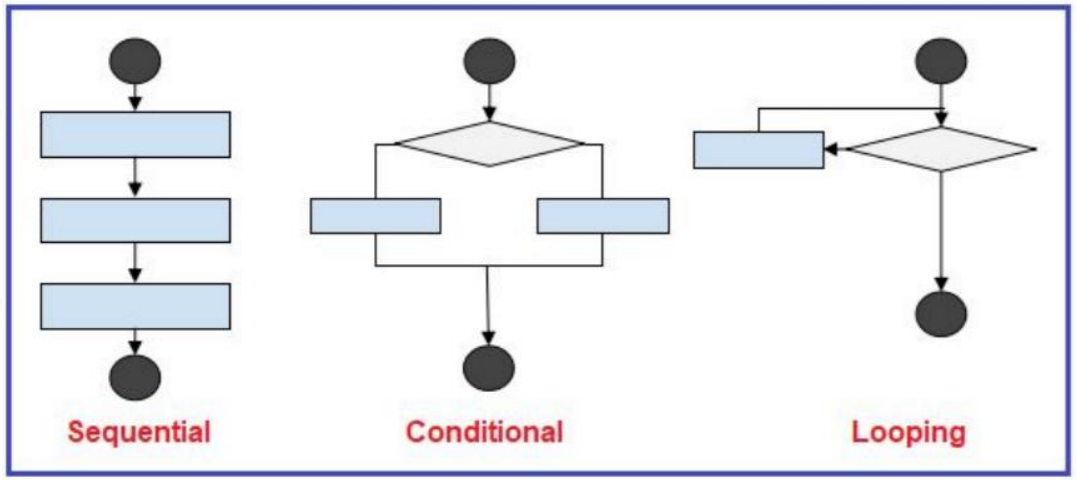
What will We Learn Today?

1. Introduction to Condition and Control Flow
2. Flowchart of Conditions
3. If statement
4. Nested if
5. Switch Case

Condition & Control Flow

Control Flow is the order in which the program's code executes.

The control flow of a Python program is regulated by conditional statements, loops, and function calls.

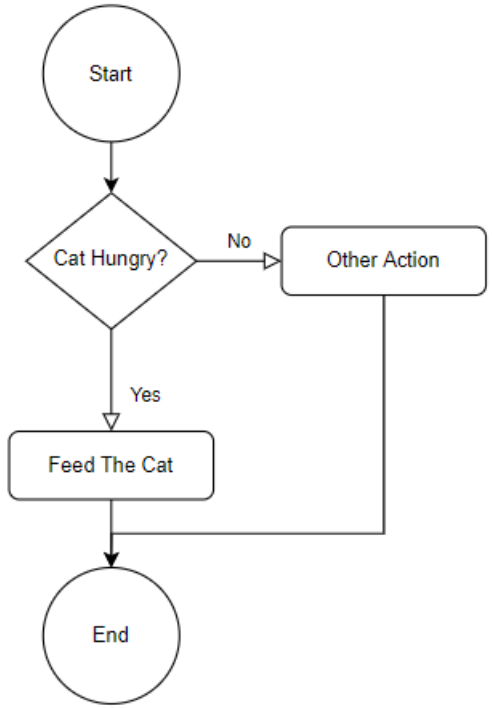


DigitalSkola
Uncover The World Of Digital Skills With Us

Condition & Control Flow

We often only want to run code when certain conditions has been met.

For example, if my cat is hungry (some condition), then I will feed my cat (some action)



Condition & Control Flow

In Python, the if statement is how you perform this sort of decision-making.

It contains a code which runs only when the condition given in the if statement is true / meet the condition.

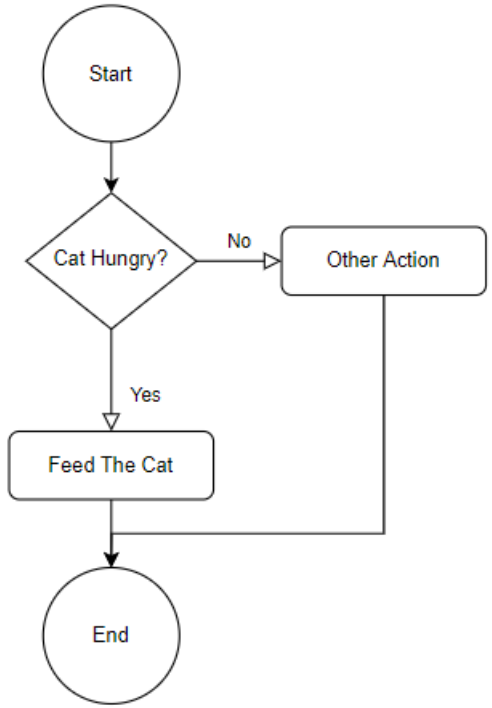
If the condition is false, then the optional else statement runs which contains some code for the else condition.

```
if <expression>:
        <statement>
else:
        <statement>
```

Condition & Control Flow

```
if cat is hungry then
        feed the cat
else
        do other things
```

```
if cat == 'hungry':
        feed_the_cat()
else:
        other_action()
```



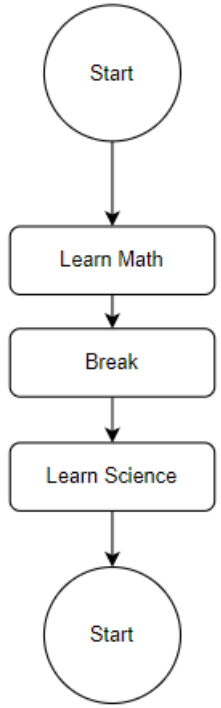
Condition & Control Flow

Sequential:

Learn Math

Break Time

Learn Science



Other Example:

```
a=20
b=10
c=a-b
print("a-b =",c)
```

Condition & Control Flow

Python supports the logical conditions from mathematics:

- 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

And also, here is the logical operators that can be used:

- True if both the operands are true: and
- True if either of the operands is true: or
- True if operand is false (complements the operand): not

DigitalSkola
Uncover The World Of Digital Skills With Us

“IF” Statement

- Single Condition (If)

```
a = 3
if a == 3:
    print("a is three")
```

output: a is three

- Two Condition (If -> Else)

```
a = 5
b = 5
if a != b:
    print("a and b is not the same")
else:
    print("a and b is the same")
```

output: a and b is the same

- Multiple Condition (If -> Elif -> Else)

```
a = 3
b = 5
if a > b:
    print("a is bigger than b")
elif a < b:
    print("a is smaller than b")
else:
    print("a and b is the same")
```

output: a is smaller than b

“IF” Statement

- Single Expression

```
a = 5
b = 7
if a <= b:
    print("a is smaller than b")
output: a is smaller than b
```

- Multiple Expression

```
a = 3
b = 7
c = 20
if a < b and b < c:
    print("a is smaller than b")
output: a is smaller than b
a = 3
b = 7
c = 20
if c > b or c < a:
    print("At least one of the expression is True")
output: At least one of the expression is True
```

“IF” Statement [Shorthand]

- Single Expression

```
a = 3
if a == 3:
    print("a is three")
```

=

```
a = 3
if a == 3: print("a is three")
```

```
if else
a = 5
b = 5
if a == b:
   print("same")
else:
   print("different")
```

=

```
a = 5
b = 5
print("same") if a==b else print("different")
```

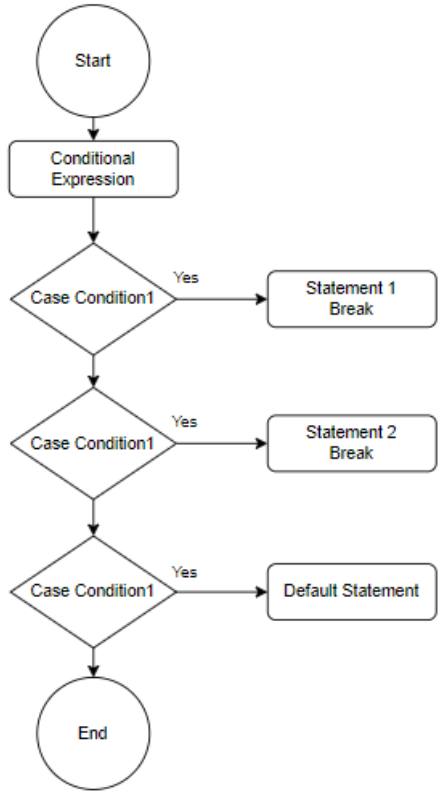
“IF” Statement [Nested]

```
x = 50
if x > 20:
   print("Above twenty, ")
   if x > 40:
   print("and also above 40")
 else:
   print("but not above 40")
else:
   print(“Below twenty”)
```

Output: Above twenty, and also above 40

Switch Case / Match Case

```
match x:
    case 'a':
        print(1)
    case 'b':
        print(2)
    case _:
          print(0)    # 0 is the default case if x is not found
```



DigitalSkola
Uncover The World Of Digital Skills With Us

Things We Learned

Congratulations! You have finished learning about conditions.

1. Introduction to Condition and Control Flow
2. If, Nested If
3. If - Shorthand
4. Switch Case

DigitalSkola

Uncover The World Of Digital Skills With Us

Thank YOU