



# Python Class

#Python Notes

# I/O Function Conditional Statements



# Input/Output

- `input()` function is used to receive input from the console.

- **Syntax:**

```
input ( [prompt] )
```

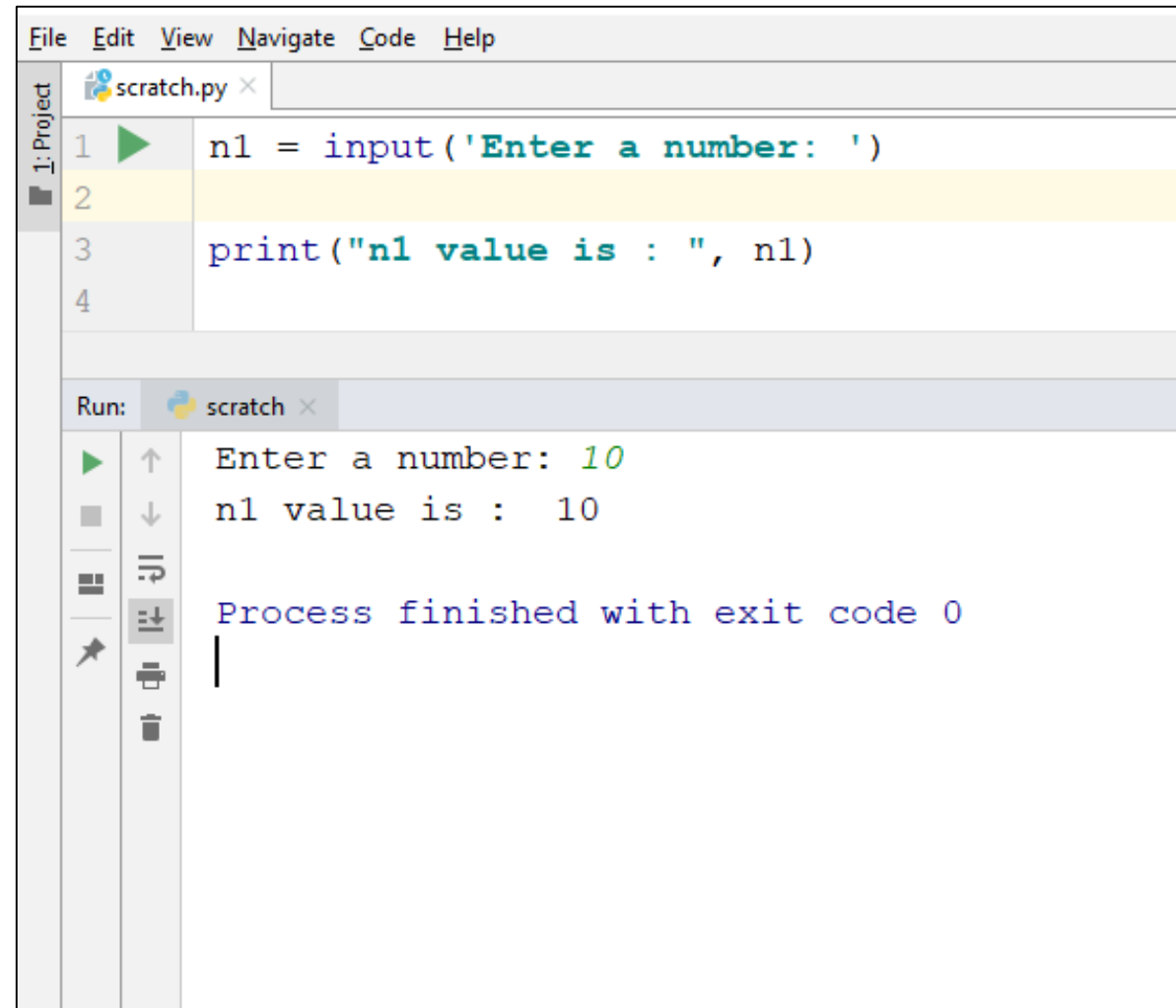
- `print()` function is used to display output to the console.

- **Syntax:**

```
print ( [message] )
```



# Example



The screenshot shows a Python IDE with a menu bar (File, Edit, View, Navigate, Code, Help) and a toolbar. The main editor window displays a file named 'scratch.py' with the following code:

```
1 n1 = input('Enter a number: ')\n2 \n3 print("n1 value is : ", n1)\n4
```

Below the editor is a 'Run' panel with a toolbar and a text area. The text area shows the output of the program:

```
Enter a number: 10\nn1 value is : 10\n\nProcess finished with exit code 0
```

## Example 2

```
scratch.py x
1  ▶ n1 = int(input('Enter number1:'))
2    n2 = int(input('Enter number2:'))
3
4    ans=n1+n2
5
6    print(ans)
7
8
9
10
11
```

```
Enter number1:10
Enter number2:20
30
```

# Conditional Statements

- if Statement
- if...else Statement
- if...elif...else Statement
- Nested if statements



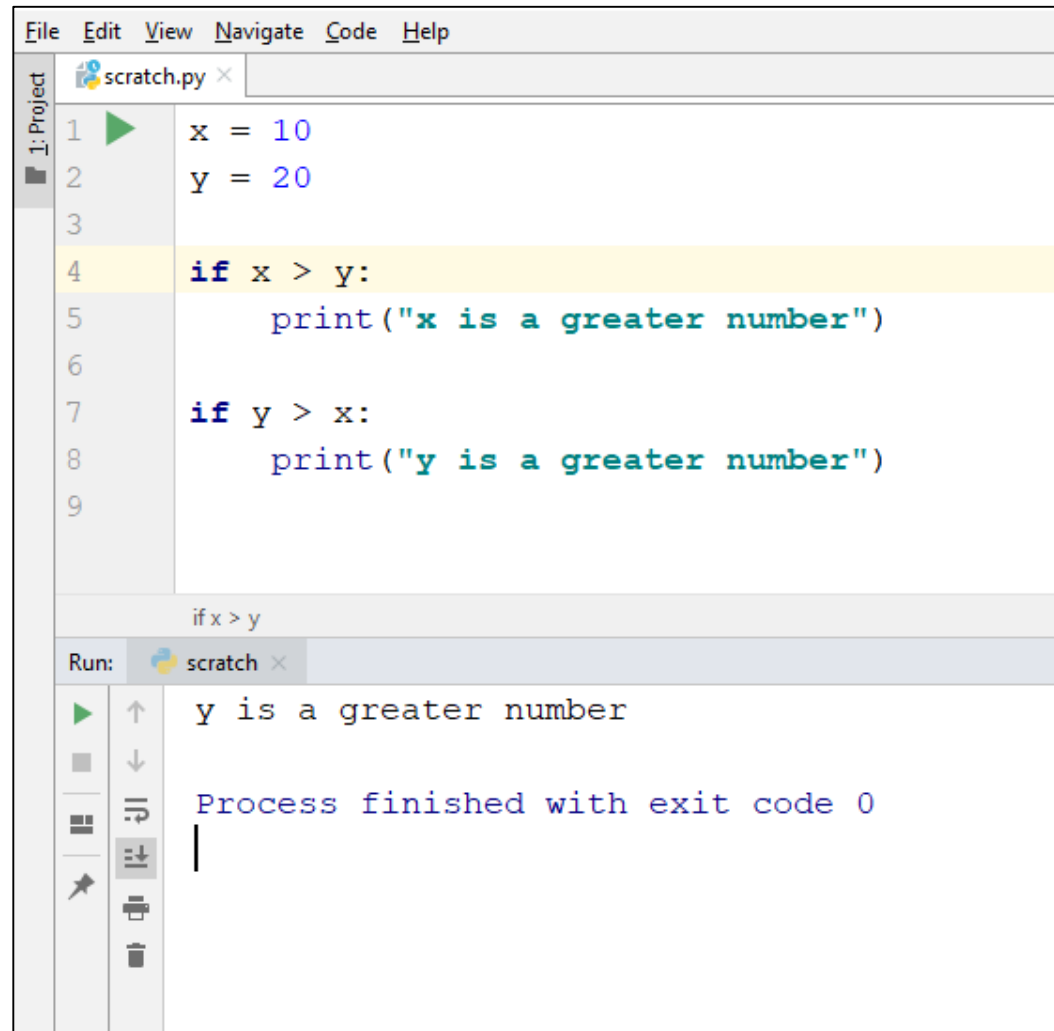
# if Statement

- **Syntax :-**

if condition:  
    statement(s)



# Example



```
File Edit View Navigate Code Help
scratch.py x
1 x = 10
2 y = 20
3
4 if x > y:
5     print("x is a greater number")
6
7 if y > x:
8     print("y is a greater number")
9

if x > y
Run: scratch x
y is a greater number
Process finished with exit code 0
```



# if...else Statement

- **Syntax :-**

if condition:

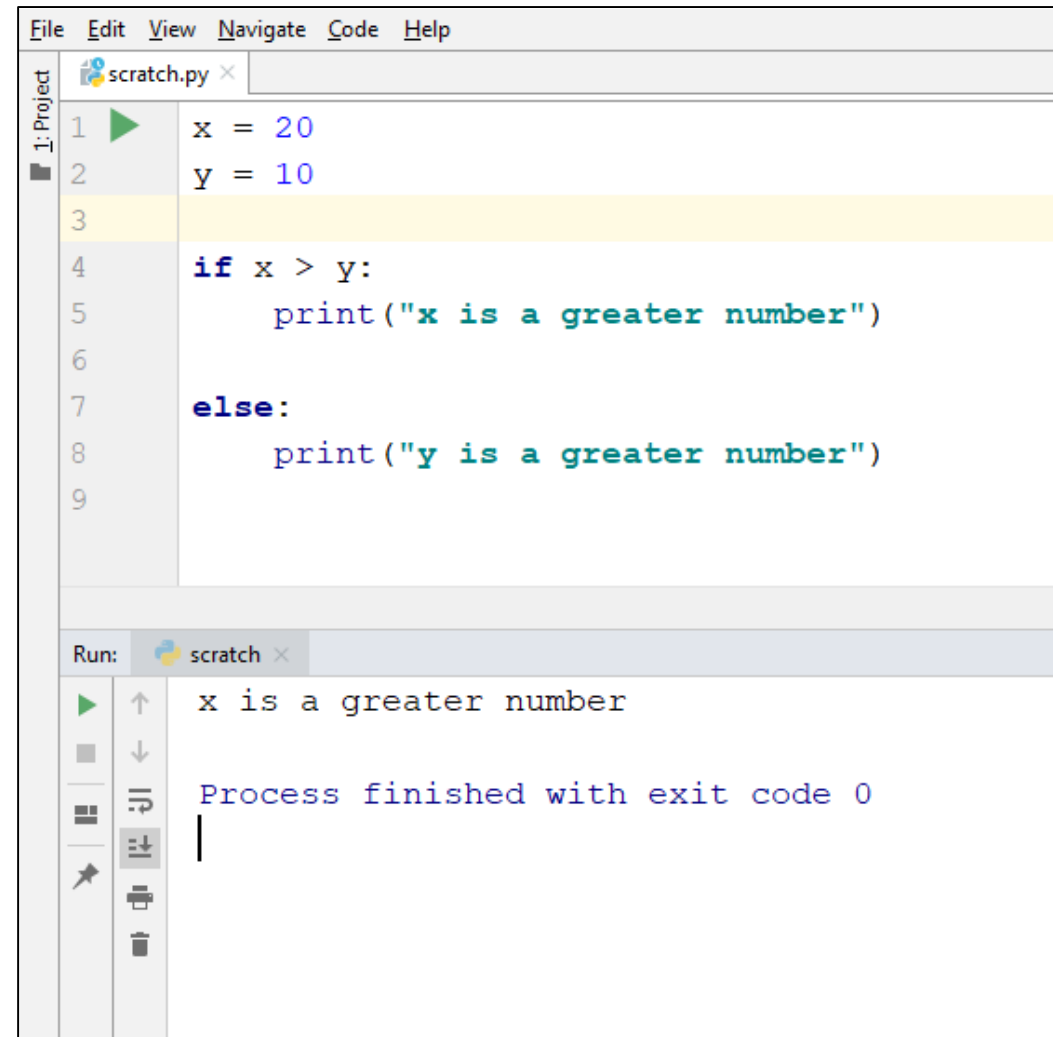
Statements to be executed

else:

Statements to be executed



# Example



The screenshot shows a Python IDE with a menu bar (File, Edit, View, Navigate, Code, Help) and a project pane on the left labeled '1: Project'. The main editor displays the following code in 'scratch.py':

```
1 x = 20
2 y = 10
3
4 if x > y:
5     print("x is a greater number")
6
7 else:
8     print("y is a greater number")
9
```

Below the editor is a 'Run' pane for 'scratch'. It shows the output of the script:

```
x is a greater number

Process finished with exit code 0
```

# if...elif...else Statement

- **Syntax :-**

if condition:

Statements to be executed

elif condition:

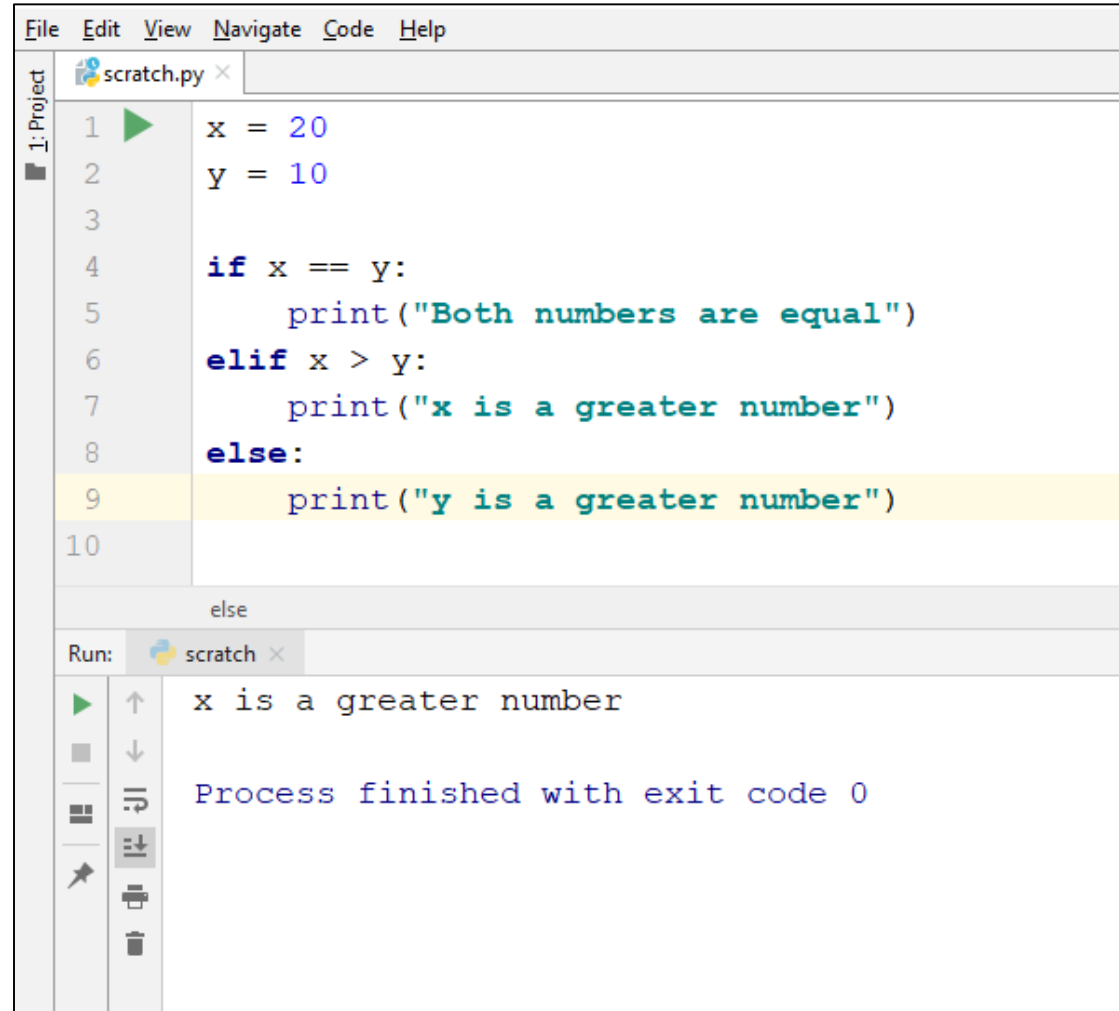
Statements to be executed

else:

Statements to be executed

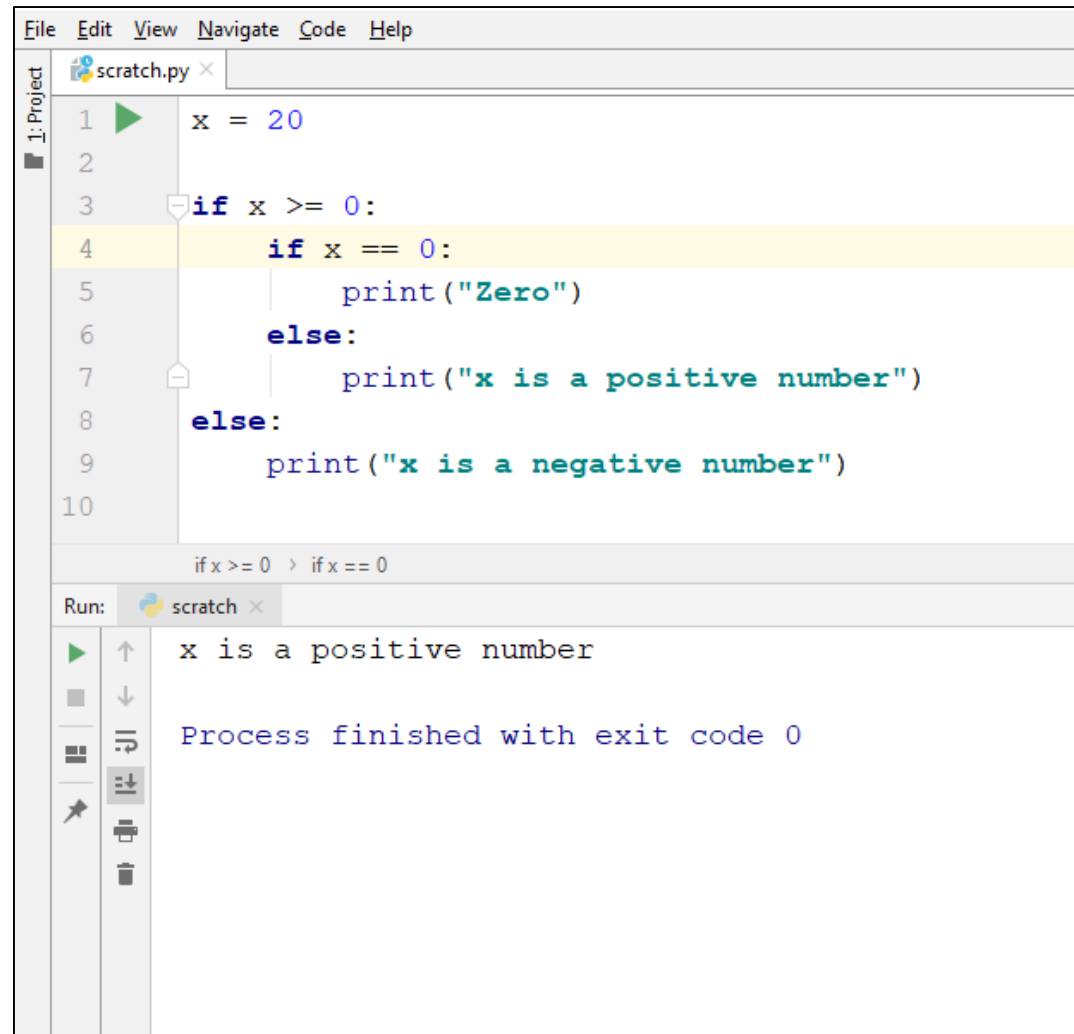


# Example



```
File Edit View Navigate Code Help
scratch.py x
1: Project
1 x = 20
2 y = 10
3
4 if x == y:
5     print("Both numbers are equal")
6 elif x > y:
7     print("x is a greater number")
8 else:
9     print("y is a greater number")
10
else
Run: scratch x
x is a greater number
Process finished with exit code 0
```

# Nested if statements



The screenshot shows a Python IDE with a file named `scratch.py`. The code defines a variable `x = 20` and uses a nested if statement to check its value. The first if statement checks `x >= 0`. If true, it enters a second if statement that checks `x == 0`. If that is also true, it prints "Zero". Otherwise, it prints "x is a positive number". If the first condition is false, it prints "x is a negative number". The output window shows that the program executed successfully and printed "x is a positive number".

```
File Edit View Navigate Code Help
scratch.py x
1: Project
1 x = 20
2
3 if x >= 0:
4     if x == 0:
5         print("Zero")
6     else:
7         print("x is a positive number")
8 else:
9     print("x is a negative number")
10

if x >= 0 > if x == 0
Run: scratch x
x is a positive number
Process finished with exit code 0
```

# Loops

- While Loop Statements
- For Loop Statements
- Nested Loops Statements



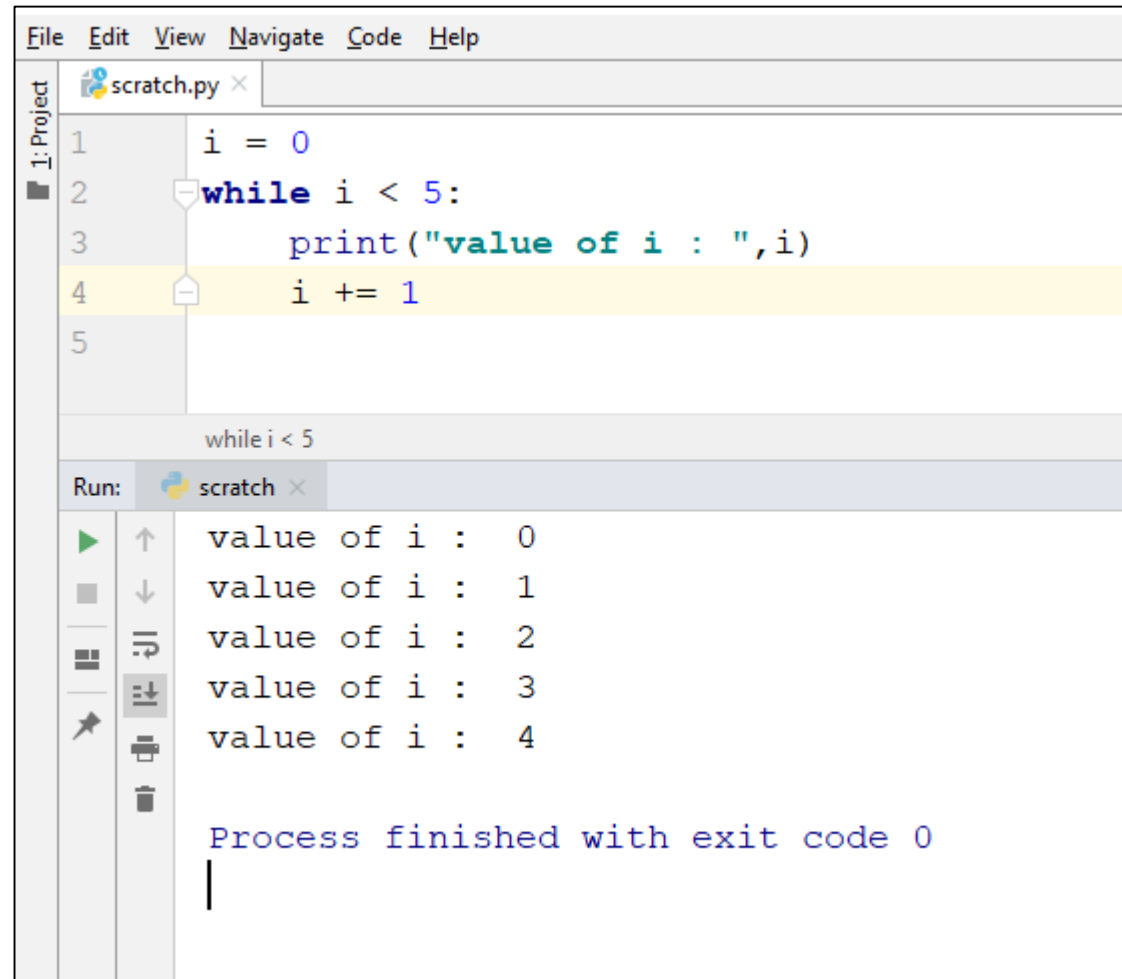
# while Loop Statements

- **Syntax:-**

```
while expression:  
    statement(s)
```



# Example



The screenshot shows a Python IDE with a file named `scratch.py`. The code in the editor is a while loop that prints the value of `i` from 0 to 4. The fourth line, `i += 1`, is highlighted in yellow. Below the code editor, the 'Run' panel shows the output of the program, which is five lines of 'value of i : ' followed by the numbers 0 through 4. The panel also indicates that the process finished with exit code 0.

```
File Edit View Navigate Code Help
scratch.py x
1 i = 0
2 while i < 5:
3     print("value of i : ",i)
4     i += 1
5
while i < 5
Run: scratch x
value of i : 0
value of i : 1
value of i : 2
value of i : 3
value of i : 4
Process finished with exit code 0
|
```

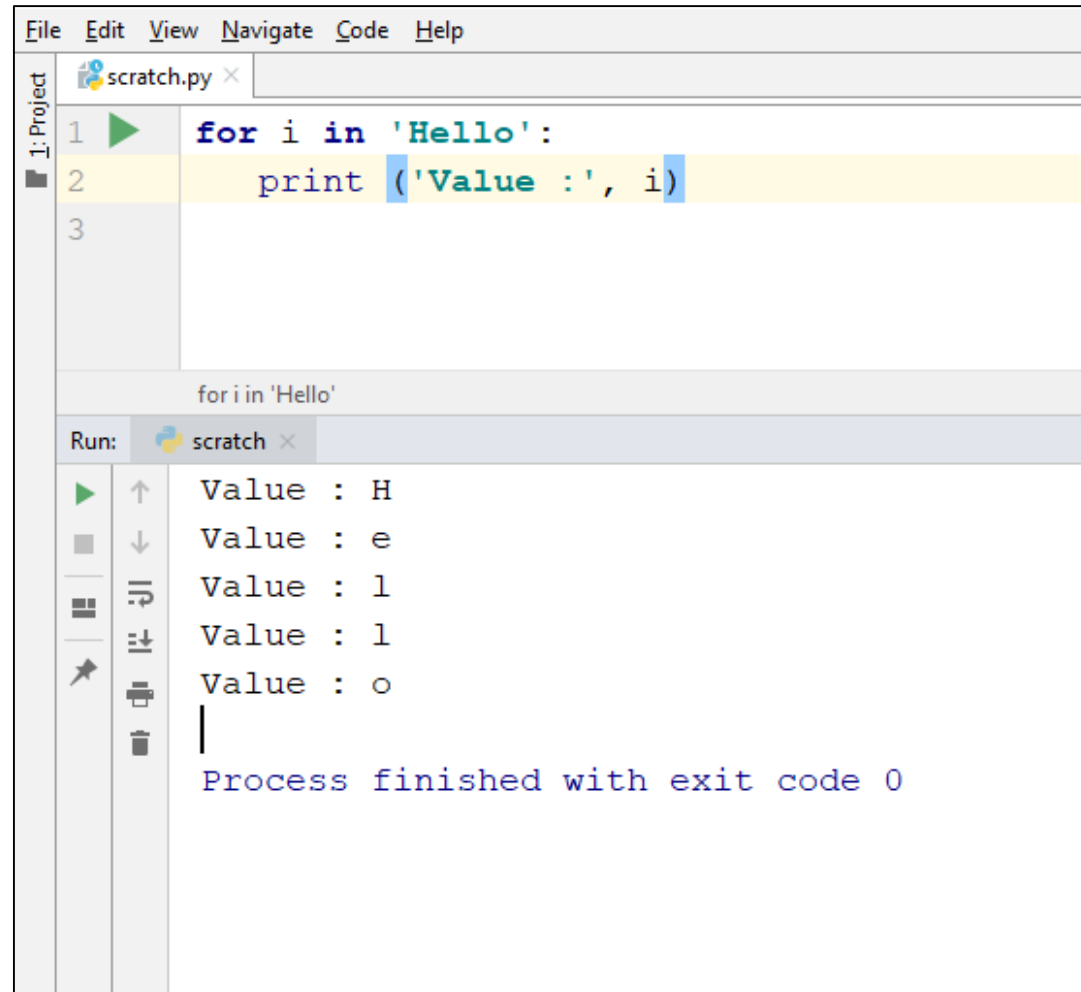


# for loop Statements

- The for loop in Python is used in list, tuple, string.
- **Syntax :-**  
for val in sequence:  
    Statements to be executed

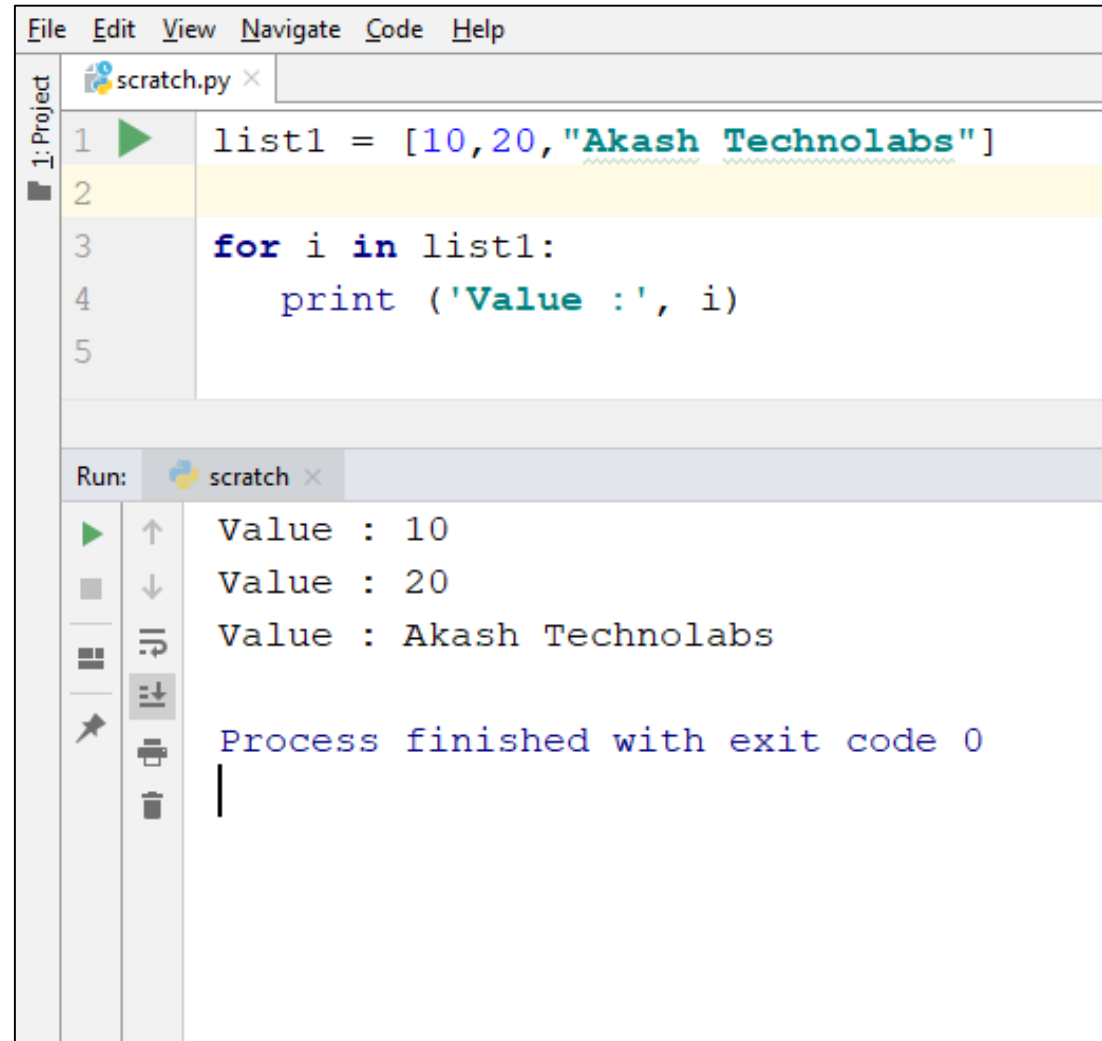


# Example



```
File Edit View Navigate Code Help
scratch.py x
1: Project
1 1 ▶ for i in 'Hello':
2 2   print ('Value :', i)
3 3
for i in 'Hello'
Run: scratch x
▶ ↑ Value : H
■ ↓ Value : e
□ ↶ Value : l
□ ↷ Value : l
□ ↶ Value : o
|
Process finished with exit code 0
```

# Example With List Datatype



The screenshot shows a Python IDE with a menu bar (File, Edit, View, Navigate, Code, Help) and a project explorer on the left. The main editor window displays a file named 'scratch.py' with the following code:

```
1 list1 = [10, 20, "Akash Technolabs"]
2
3 for i in list1:
4     print ('Value :', i)
5
```

Below the editor is a 'Run' console window for the 'scratch' file. It shows the output of the program:

```
Value : 10
Value : 20
Value : Akash Technolabs

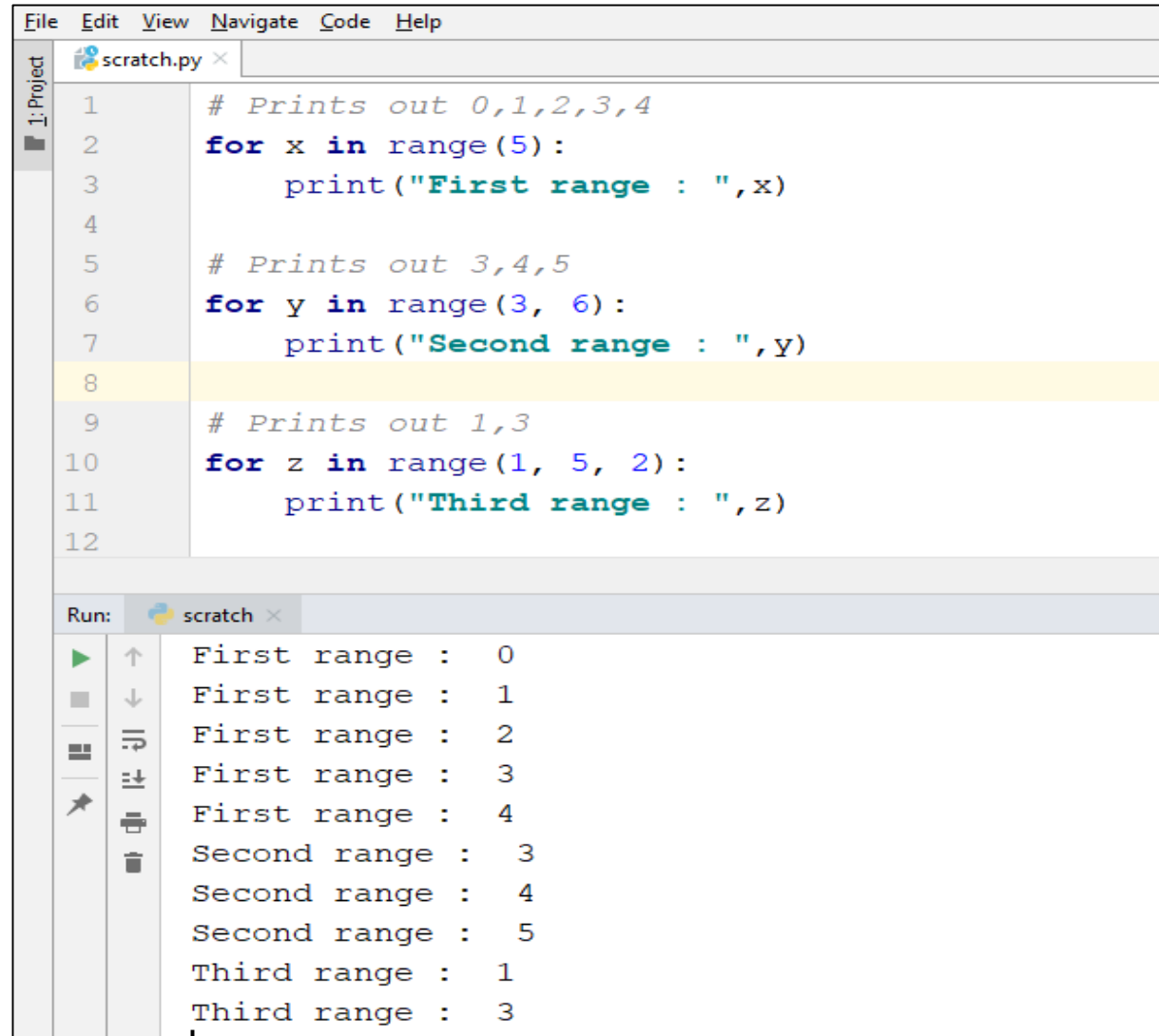
Process finished with exit code 0
```

# The range() function

- We can generate a sequence of numbers using range() function.
- **Example :- range(10) will generate numbers from 0 to 9 (10 numbers).**
- **We can also define the start, stop and step size as range(start,stop,step size). Step size defaults to 1 if not provided.**



# Example



The screenshot shows a Python IDE with a menu bar (File, Edit, View, Navigate, Code, Help) and a tab for 'scratch.py'. The code is as follows:

```
1 # Prints out 0,1,2,3,4
2 for x in range(5):
3     print("First range : ",x)
4
5 # Prints out 3,4,5
6 for y in range(3, 6):
7     print("Second range : ",y)
8
9 # Prints out 1,3
10 for z in range(1, 5, 2):
11     print("Third range : ",z)
12
```

Below the code editor is a 'Run' panel with a tab for 'scratch'. It shows the output of the code execution:

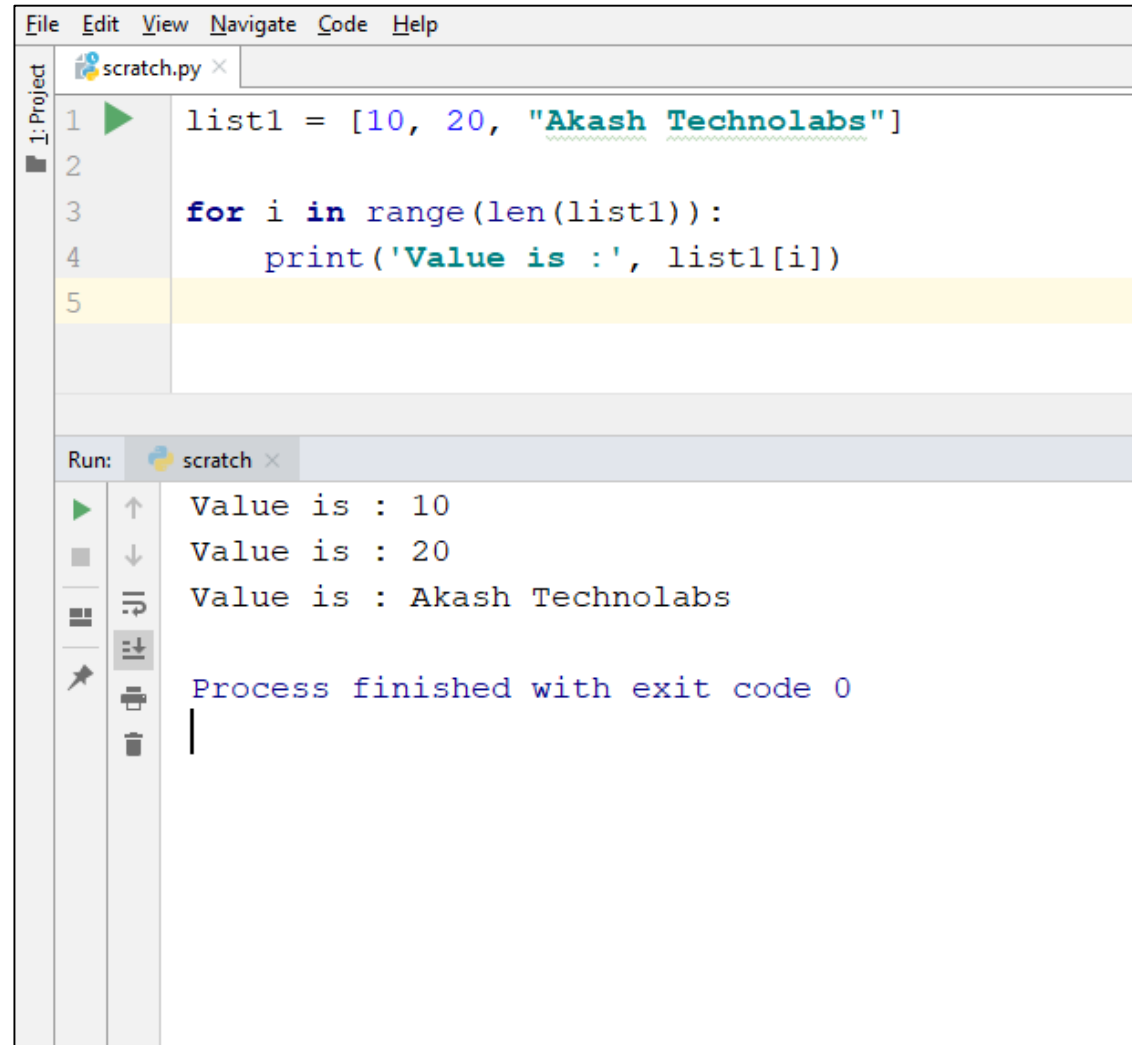
```
First range : 0
First range : 1
First range : 2
First range : 3
First range : 4
Second range : 3
Second range : 4
Second range : 5
Third range : 1
Third range : 3
```

## Example 2

- An alternative way of iterating through each item is by index.
- For that we have to use `range()` function along with `len()` function.
- The `len()` function provides the total number of elements in the tuple, the `range()` function give us the actual sequence to iterate over.



# Example



The screenshot shows a Python IDE with a menu bar (File, Edit, View, Navigate, Code, Help) and a tab for 'scratch.py'. The code editor contains the following Python code:

```
1 list1 = [10, 20, "Akash Technolabs"]
2
3 for i in range(len(list1)):
4     print('Value is :', list1[i])
5
```

Below the code editor is a 'Run' panel with a tab for 'scratch'. It displays the output of the code execution:

```
Value is : 10
Value is : 20
Value is : Akash Technolabs

Process finished with exit code 0
```

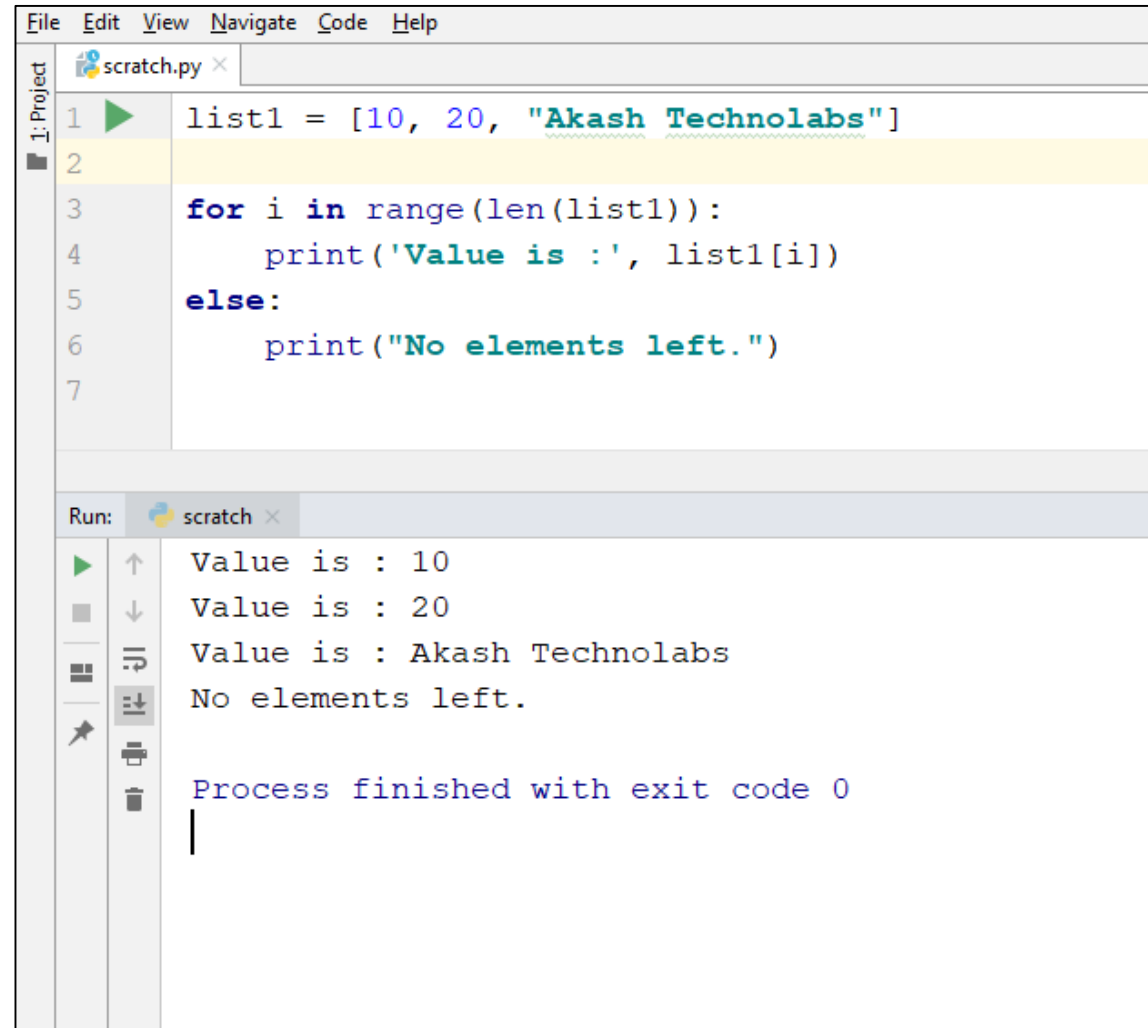
# Loop with else

- A loop can have an optional else block as well.
- If the **else** statement is used with a **for** loop, the **else** statement is executed when the loop has exhausted iterating the list.
- If the **else** statement is used with a **while** loop, the **else** statement is executed when the condition becomes false.





# Example



The screenshot shows a Python IDE with a menu bar (File, Edit, View, Navigate, Code, Help) and a tab for 'scratch.py'. The code editor contains the following Python code:

```
1 list1 = [10, 20, "Akash Technolabs"]
2
3 for i in range(len(list1)):
4     print('Value is :', list1[i])
5 else:
6     print("No elements left.")
7
```

Below the code editor is a 'Run' panel with a tab for 'scratch'. It shows the output of the code execution:

```
Value is : 10
Value is : 20
Value is : Akash Technolabs
No elements left.

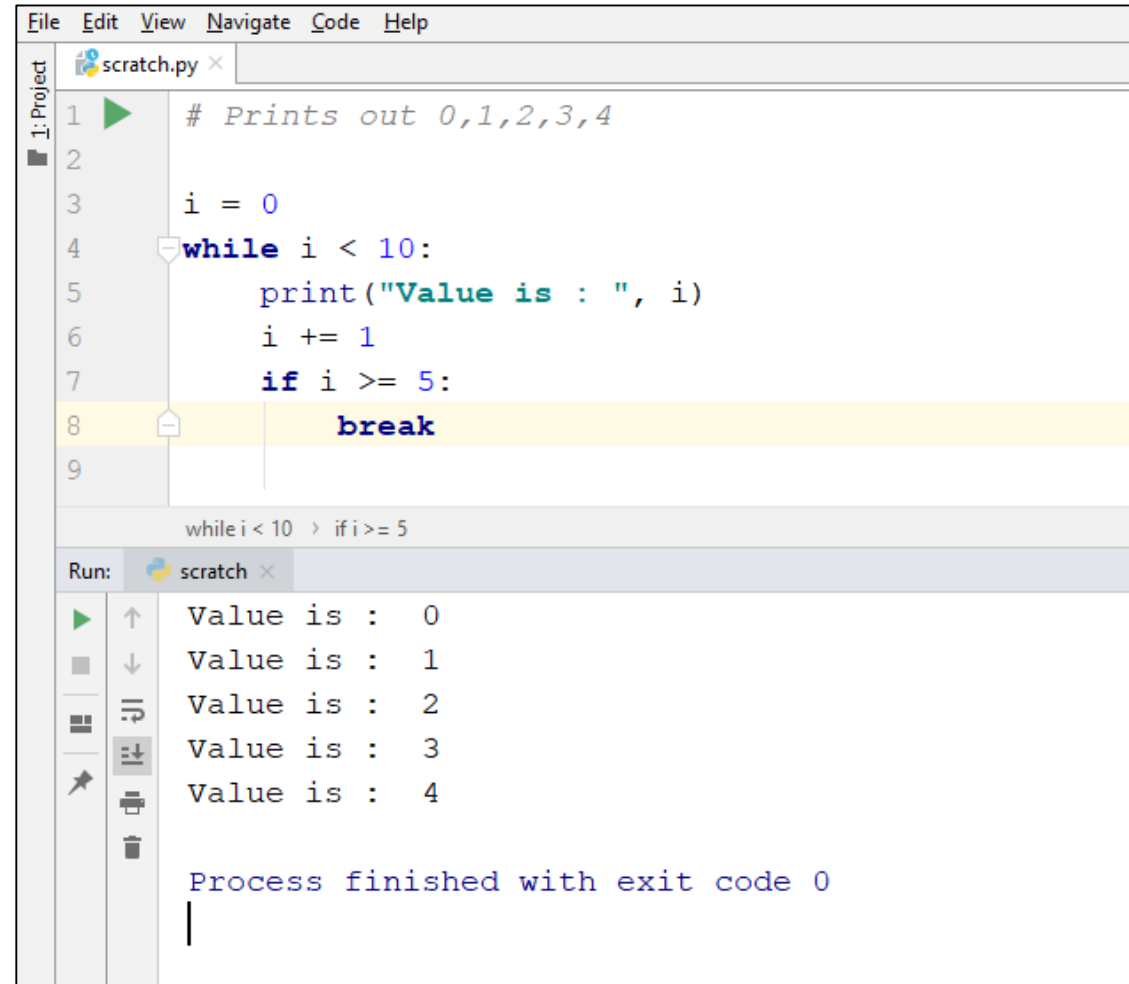
Process finished with exit code 0
```

# "break" and "continue" statements

- **break** is used to exit a for loop or a while loop
- **continue** is used to skip the current block, and return to the "for" or "while" statement.

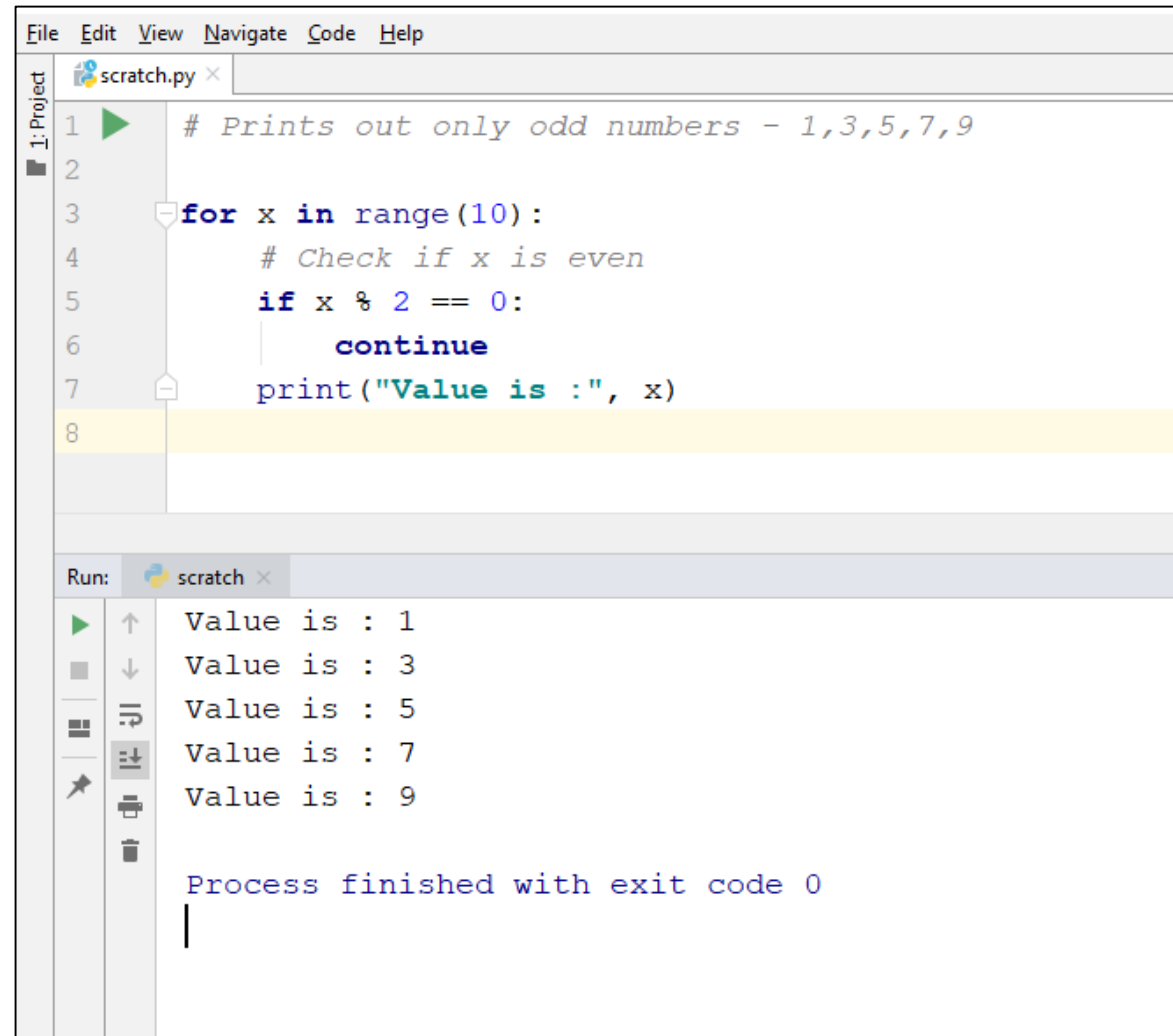


# Example of break statement



```
File Edit View Navigate Code Help
scratch.py x
1 # Prints out 0,1,2,3,4
2
3 i = 0
4 while i < 10:
5     print("Value is : ", i)
6     i += 1
7     if i >= 5:
8         break
9
while i < 10 > if i >= 5
Run: scratch x
↑ Value is : 0
↓ Value is : 1
↶ Value is : 2
↷ Value is : 3
↵ Value is : 4
Process finished with exit code 0
```

# Example of continue statement



```
File Edit View Navigate Code Help
scratch.py x
1 # Prints out only odd numbers - 1,3,5,7,9
2
3 for x in range(10):
4     # Check if x is even
5     if x % 2 == 0:
6         continue
7     print("Value is :", x)
8
Run: scratch x
Value is : 1
Value is : 3
Value is : 5
Value is : 7
Value is : 9
Process finished with exit code 0
```

# pass Statement in Python

- pass is a null statement. The difference between a comment and pass statement in Python is that, while the interpreter ignores a comment entirely, pass is not ignored.
- It is used as a placeholder.
- **Syntax :-**  
    Pass

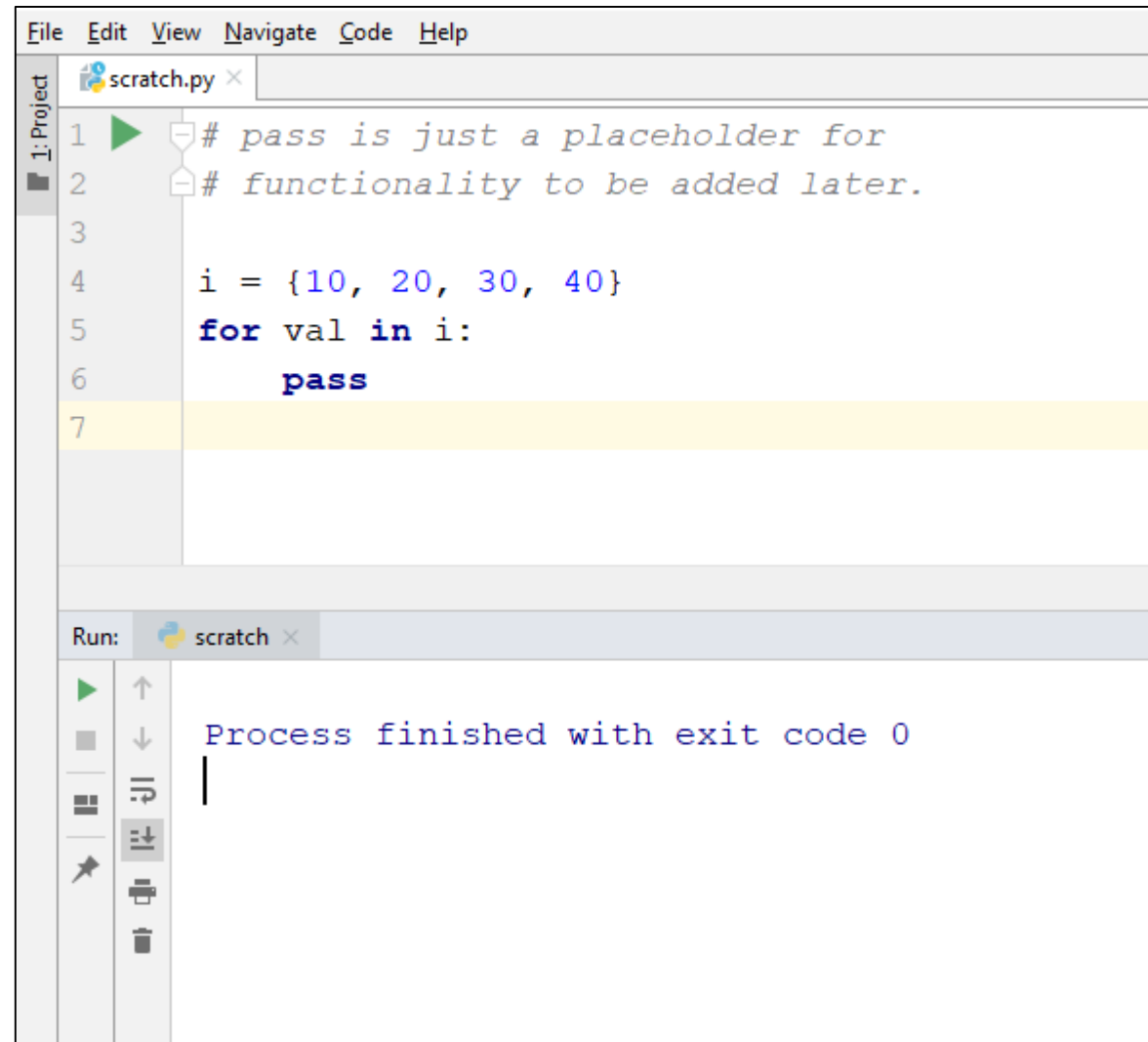


# When to use pass statement?

- Suppose we have a loop or a function that is not implemented yet, but we want to implement it in the future.
- They cannot have an empty body. The interpreter would complain.
- So, we use the pass statement to construct a body that does nothing.



# Example



The screenshot shows an IDE window with a menu bar (File, Edit, View, Navigate, Code, Help) and a tab for 'scratch.py'. The code editor contains the following Python code:

```
1 # pass is just a placeholder for
2 # functionality to be added later.
3
4 i = {10, 20, 30, 40}
5 for val in i:
6     pass
7
```

Line 7 is highlighted in yellow. Below the code editor is a 'Run:' panel with a tab for 'scratch'. It shows the output of the script execution:

```
Process finished with exit code 0
```

On the left side of the IDE, there is a 'Project' pane showing a folder icon and the text '1: Project'.

# Get Exclusive Video Tutorials



[www.apptutorials.com](http://www.apptutorials.com)

<https://www.youtube.com/user/Akashtips>







Get More Details

[www.akashsir.com](http://www.akashsir.com)



# If You Liked It !

## Rating Us Now



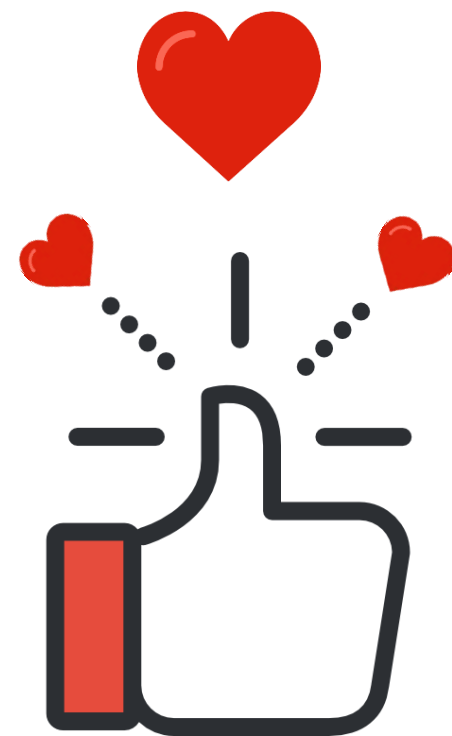
**Just Dial**

[https://www.justdial.com/Ahmedabad/Akash-Technolabs-Navrangpura-Bus-Stop-Navrangpura/079PXX79-XX79-170615221520-S5C4\\_BZDET](https://www.justdial.com/Ahmedabad/Akash-Technolabs-Navrangpura-Bus-Stop-Navrangpura/079PXX79-XX79-170615221520-S5C4_BZDET)



**Sulekha**

<https://www.sulekha.com/akash-technolabs-navrangpura-ahmedabad-contact-address/ahmedabad>



# Connect With Me



Akash Padhiyar  
#AkashSir

[www.akashsir.com](http://www.akashsir.com)

[www.akashtechlabs.com](http://www.akashtechlabs.com)

[www.akashpadhiyar.com](http://www.akashpadhiyar.com)

[www.aptutorials.com](http://www.aptutorials.com)

## # Social Info



Akash.padhiyar



Akashpadhiyar



Akash\_padhiyar



+91 99786-21654



#Akashpadhiyar  
#aptutorials