**Loops**

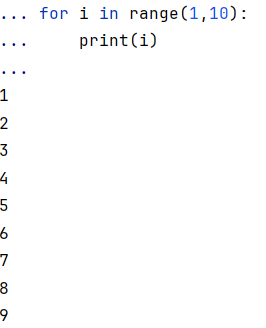
In Python, loops are used to execute a block of code repeatedly until a certain condition is met. Python supports two types of loops:

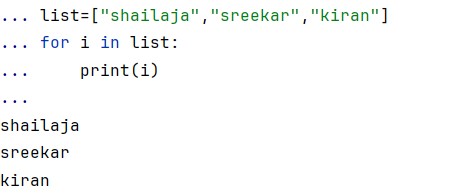
**1. for Loop**

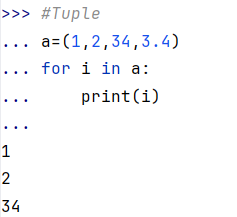
The for loop is used for iterating over a sequence (like a list, tuple, dictionary, set, or string).

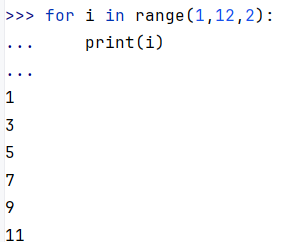
for variable in sequence:

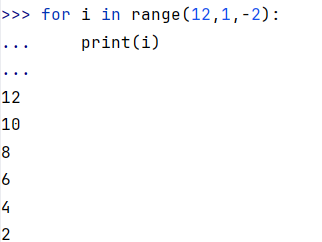
# Code to execute in each iteration

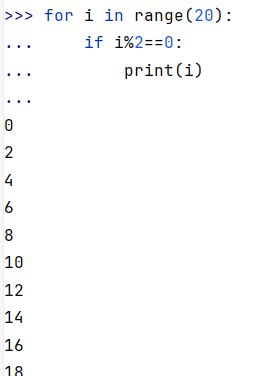
*#Loops*for i in range(1,10):  
 print(i)  
  
*#List*list=["shailaja","sreekar","kiran"]  
for i in list:  
 print(i)

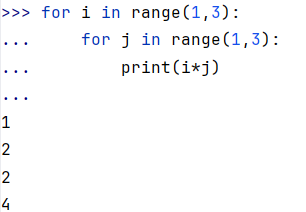
  
  
*#Tuple*a=(1,2,34,3.4)  
for i in a:  
 print(i)

  
  
for i in range(1,12,2):  
 print(i)

  
for i in range(12,1,-2):  
 print(i)

  
  
for i in range(20):  
 if i%2==0:  
 print(i)

  
for i in range(1,5):  
 for j in range(1,5):  
 print(i\*j)



**2. while Loop**

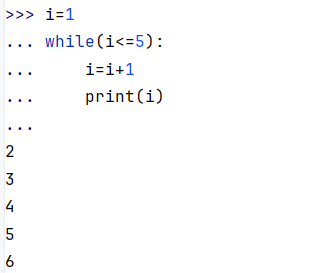
The while loop is used to execute a block of code as long as a condition is true.

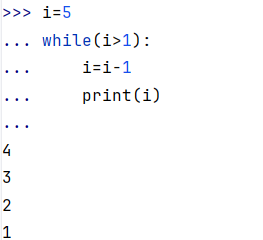
while condition:

# Code to execute

Examples:

i=1  
while(i>=5):  
 i=i+1  
 print(i)

  
  
i=5  
while(i>1):  
 i=i-1  
 print(i)

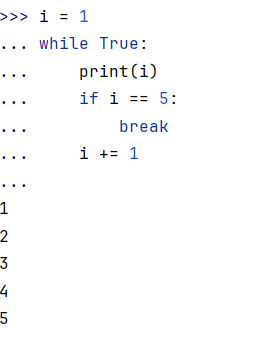


**Loop Control Statements**

These statements alter the flow of execution in loops.

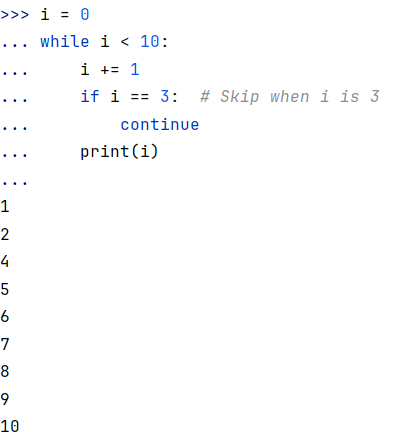
1. **break**: Terminates the loop immediately.

i = 1  
while True:  
 print(i)  
 if i == 5:  
 break  
 i += 1



1. **continue**: Skips the current iteration and moves to the next.

i = 0  
while i < 10:  
 i += 1  
 if i == 3: *# Skip when i is 3* continue  
 print(i)



**Nested Loops**

A loop inside another loop.

i=1  
while(i<3):  
 print(i)  
 i=i+1  
 j=1  
 while(j<3):  
 print(j)  
 j=j+1

