

Loop Statements (for, while)

Statements	Syntax	Example	Meaning
while	while (Condition): statement(s)	count = 0 while (count < 3): count = count+1 print("Hello Bennettians")	while loop is used for iterators
		Output:	
		Hello Bennettians	
		Hello Bennettians	
		Hello Bennettians	
for	for iterator_var in sequence: statements(s)	<pre>I = ["bennett", "for", "bennettians"] for i in I: print(i) Output: bennett for bennettians</pre>	for can be used to iterate over iterators and a range.
range	for iterator_var in range(n):	for x in range(4): print(x) output: 0	It returns a sequence of numbers, starting from 0 by default, and increments by 1

ECSE105L: Computational Thinking and Programming





nested-for for iterator_var in sequence: for iterator_var in sequence: for iterator_var in sequence: statements(s) statements(s) nested-while while expression: print(i in range(1, 5): for i in range(1, 5): for i in range(i): print(i, end=' ') print() Output: 1 2 2 3 3 3 4 4 4 4 4 Python programming another for loop. Python programming another for loop. Python programming anguage allows use while loop inside another for loop.		T		T.,
nested-for for iterator_var in sequence: for iterator_var in sequence: for iterator_var in sequence: for iterator_var in sequence: statements(s) statements(s) Output: 1 2 2 3 3 3 4 4 4 4 nested-while while expression: while expression: statement(s) while i < 4: while j < 8: print(i, ",", j) j = j + 1 i = i + 1			1	(by default), and ends
nested-for for iterator_var in sequence: for iterator_var in sequence: for iterator_var in sequence: statements(s) statements(s) nested-while while expression: while expression: statement(s) statement(s) while i < 4: while j < 8: print(i, ",", j) j = j + 1 i = i + 1			2	•
sequence: for iterator_var in sequence: statements(s) statements(s) nested-while while expression: statement(s) statement(s) statement(s) while i < 4: while j < 8: print(i, end=' ') print(i) p			3	+
for iterator_var in sequence: statements(s) statements(s) output: 1 2 2 3 3 3 4 4 4 4 nested-while while expression: statement(s) statement(s) while i < 4: while j < 8: print(i, ",", j) j = j + 1 i = i + 1 while is of loop insignation in the specific print (i, end=' ') print(i, e	nested-for		for i in range(1, 5):	_ ·
for iterator_var in sequence: statements(s) print() Output: 1 2 2 3 3 3 4 4 4 4 4 nested-while while expression: statement(s) statement(s) while i < 4: while j < 8: print(i, end=' ') print(i) Python programmin language allows use while loop inside another while loop. while j < 8: print(i, ",", j) j = j + 1 i = i + 1		sequence:	for i in range(i):	
sequence: print(i, end='') statements(s) print() Output: 1 2 2 3 3 3 4 4 4 4 Python programmin language allows use while loop inside another while loop. statement(s) while i < 4:		for iterator var in	ior j iir range(i).	
statements(s) Output: 1 22 3333 4444 nested-while while expression: while expression: statement(s) statement(s) while i < 4: while j < 8: print(i, ",", j) j = j + 1 i = i + 1		_	print(i, end=' ')	another for loop.
statements(s) Output: 1 22 3333 4444 nested-while while expression: while expression: statement(s) statement(s) while i < 4: while j < 8: print(i, ",", j) j = j + 1 i = i + 1			n wint ()	
nested-while while expression: while expression: statement(s) statement(s) while j < 8: print(i, ",", j) j = j + 1 i = i + 1		statements(s)	print()	
nested-while while expression: $i = 1$		statements(s)	Output:	
nested-while while expression: $i = 1$ Python programming language allows use while loop inside another while loop. Statement(s) while $i < 4$: while $j < 8$: print(i, ",", j) $j = j + 1$ $j = i + 1$			1	
nested-while while expression: $i = 1$ Python programming language allows use while loop inside another while loop. Statement(s) while $i < 4$:			2 2	
nested-while while expression: $j=5$ while $i=1$ Python programming language allows use while loop inside another while loop. Statement(s) while $j<8$: $j=1$ Python programming language allows use while loop inside another while loop. $j=1$ $j=1$ $j=1$ $j=1$ $j=1$ $j=1$			3 3 3	
nested-while while expression: $i=1$			<i>A A A A</i>	
	nested-while	while expression:		Python programming
	nested wille	Willie expression.	1-1	
statement(s)		while expression:	j = 5	use while loop inside
print(i, ",", j) j = j + 1 i = i + 1		statement(s)	while i < 4:	another while loop.
j = j + 1 i = i + 1		statement(s)	while j < 8:	
i = i + 1			print(i, ",", j)	
			j = j + 1	
Output:			i = i + 1	
			Output:	
1,5			1,5	

ECSE105L: Computational Thinking and Programming



	I		I
		2,6	
		2 7	
		3,7	
Else in for loop	for iterator_var in	for x in range(6):	The else keyword in a
	sequence:	print(x)	for loop specifies a
	statements(s)	else:	block of code to be
	else	print("Finally finished!")	executed when the
	statements	Output:	loop is finished:
		0	
		1	
		2	
		3	
		4	
		5	
		Finally finished!	
Else in while	While condition:	x=0	The else keyword in a
loop	statements(s)	y=6	while loop specifies a
·	else	while y>x:	block of code to be
	statements	print(x)	executed when the
		x=x+1	loop is finished:
		else:	'
		print("Finally	
		finished!")Output:	
		0	
		1	
		2	
		3	
		4	
		5	
		Finally finished!	
		i many minanca:	

Control Statements (Continue, Break, Pass)

Statements	Example	Meaning
continue	for char in 'Pythn':	When the program encounters
	if (char == 'y'):	continue statement, it will skip the



	continue print("Current character: ", char) Output: Current character: P Current character: t Current character: h Current character: n	statements which are present after the continue statement inside the loop and proceed with the next iterations.
break	for char in 'Python': if (char == 'h'): break print("Current character: ", char) Output: Current character: P Current character: y Current character: t	The break statement is used to terminate the loop containing it, the control of the program will come out of that loop.
pass	for char in 'Python': if (char == 'h'): pass print("Current character: ", char) Output: Current character: P Current character: y Current character: t Current character: h Current character: o Current character: n	Pass statement is python is a null operation, which is used when the statement is required syntactically.



1. Predict the output:

```
j=1
while j<=10:
    print(j)
    j=j+1

Output:
    1
    2
    3
    4
    5
    6
    7
    8
    9
    10</pre>
```

2. Predict the output:

```
num = 10
while num > 6:
    print(num)
    num = num-1
else:
    print("loop is finished")
```

Output:

10

9

8

7

loop is finished

3. Predict the output

```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
    print(x)
```

Output:

apple

banana



cherry

```
4. Predict the output:
```

```
for x in "apple":
   print(x)

Output:
a
p
p
```

5. Predict the output:

```
batch = ["eb10", "eb12", "eb14"]
for x in batch:
    print(x)
    if x == "eb12":
        break

Output:
```

6. Predict the output:

eb10 eb12

```
batch = ["eb20", "eb21", "eb22"]
for x in batch:
  if x == "eb21":
    continue
  print(x)
```

Output:

eb20 eb22

7. Predict the output:

```
for x in range(2, 6, 2):
  print(x)

Output:
2
```



4

8. A series has been provided (1/1! + 2/2! + 3/3! + ...), calculate the sum of first 5 numbers of the series using While loop.

Solution:

```
x=1
sum=0
while(x<=5):
    j=x
    fact=1
    while(j>1):
        fact=fact*j
        j=j-1;
        sum = sum + x/fact
        x=x+1
print("Summation of series is: ", sum)
```

Output: Minimum value is 2.7083333333333333

9. Print the following pattern using for loop

```
1 2 3 4 5 6 7 8 9 10
```

Solution:

```
num=1
for i in range(4):
  for j in range(3,i,-1):
    print(" ", end=")
  for j in range(i+1):
    print(num,end=' ')
    num=num+1
  print("")
```

10. Calculate the summation of first 10 numbers using while loop, where x is a user input and value is 2.



$$\frac{x-1}{x} + \frac{1}{2} \left(\frac{x-1}{x} \right)^2 + \frac{1}{3} \left(\frac{x-1}{x} \right)^3 + \frac{1}{4} \left(\frac{x-1}{x} \right)^4 + \cdots$$

Solution:

```
x=1
in_var = 2
in_var=(in_var-1)/in_var
sum=0
for x in range(1,10):
    sum = sum + ((in_var)**x)/x
    x=x+1
print("Summation of ", sum)
```

11. Create a list of 10 elements and check whether a number is available in the list using for loop.

Output:

```
list=[1,2,3,7,9,4,56,23,12,3]
#value=7
value=int(input())
flag=0
for x in range(1, 10):
    if value==list[x]:
        print("number found at", x)
        flag=1
if flag==0:
    print("number not found")
```

12. Create a list of 10 elements and find out the minimum element using while loop.

Output:

```
list=[10,2,3,7,9,4,56,23,12,3]
min_value=list[0]
x=0
while(x<10):
   if min_value>list[x]:
      min_value = list[x]
   x=x+1
print("Minimum value is", min_value)
```

13. Enter the string of your name and print the ASCII value of it.

Solution:

```
print("Enter Name:", end="")
name = input()
```



Tutorials on Loop, and Control Structure

namelength = len(name)
for char in name:
 ascii = ord(char)
 print(char, "\t", ascii)