Loops and Lists and Strings

Many interesting problems involve manipulating sequences of data. In order to do this, we have to be able to loop through the sequence of data. In this set of activities, you will look at how we create loops in Python and some of the standard collections of sequenced data that we use in Python.

Model 1 The range Function

The Python range function will generate a list of numbers. The range function can take up to three numbers as arguments. Fill in the table below by typing the code into a Python Shell:

Python code	Shell output
range(5)	
list(range(5))	
list(range(10))	
list(range(5, 10))	
list(range(-3, 4))	
list(range(4, 10, 2))	
list(range(4, 11, 2))	
list(range(4, 22, 5))	

Questions

1. If the argument of the range function specifies a single number (x):
   a) What will be the first number listed?

   b) What will be the last number listed?

   c) How many numbers will be in the list?

   d) Use the range function to generate the sequence 0, 1, 2, 3.

2. If the argument of the range function specifies two numbers (*x*, *y*):
   a) What will be the first number listed?

   b) What will be the last number listed?

   c) How many numbers will be in the list?

   d) Use the range function to generate the sequence 1, 2, 3, 4.

3. If the argument of the range function specifies three numbers (*x*, *y*, *z*):
   a) What will be the first number listed?

   b) What does the third argument represent?

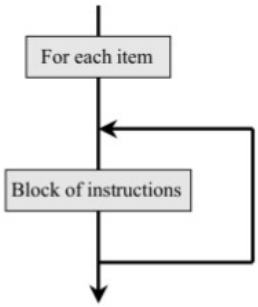
   c) Approximately how many numbers will be in the list?

   d) Use the range function to generate the sequence 1, 3, 5, 7.

Model 2 Two ways to loop

A for loop executes the same block of code “for each item in a sequence”. Create a new file named loops1.py, and enter the following code:

```
print("hello")
for x in range(5):
    print("The number is " + str(x))
print("goodbye")
```



Questions

4. Run the loops1.py program. How many times does the indented line of code execute under the for loop?

5. How many times does the line of code NOT indented execute after the for loop?

6. Identify the value of x each time the indented line of code is executed.
   a) 1st time:
   b) 2nd time:
   c) 3rd time

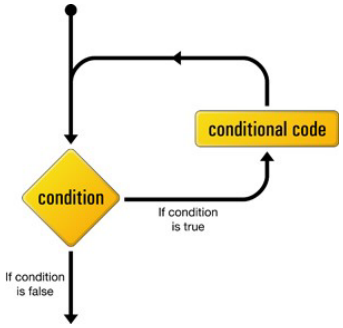
7. Modify the range() function in the following ways and rerun the program each time. Indicate how many times the for loop executes.
   d) range(5, 20)
   e) range(20, 5)
   f) range(5, 20, 2)
   g) range(20, 5, -2)

8. In general, what determines the number of times that the loop repeats?

9. What determines the value of the variable x? Explain your answer in terms of what is assigned (x = ...) each time the loop runs.

A more general looping structure is the while statement Create a new file named loops2.py, and enter the following code:

```
print("hello")
index = 0
while index<5:
    print("The number is " + str(index))
    index = index + 1
print("goodbye")
```



17. What must the value of the Boolean expression (after the while) be in order for the first "The number is " print statement to execute?

18. Circle the statement that changes the variable index in the above code.

19. What happens to the value of the variable index during the execution of the loop?

20. Explain why the loop body does not execute again after it outputs “the number is 4”.

21. Reverse the order of the statements in the loop body:

```
while index<5:
    index = index + 1
    print("The number is " + str(index))
```

a) How does the order impact the output displayed by the print function?

b) Does the order impact the total number of lines that are output?

22. Change the code so that it will perform in the following ways. Record what changes you needed to make for this to happen.
    a) Print the values from 5 to 19

b) Print the values from 5 to 19 counting up by 2s

c) Print the values from 19 to 5 counting down by 2s

23. Describe the three parts of a while loop that control the number of times the loop executes.

24. Replace the code in loops2.py with:

```
print("hello")
index = 0
while index<5:
    print("The number is " + str(index))
print("goodbye")
```

and run the module.

Then press Ctrl-C (hold down the Ctrl key and press C). Describe the behavior you see, and explain why it happened.

When writing a while loop, it's helpful to answer a few questions before you start:

- *What needs to be initialized before the loop?*
- *What condition must be true for the loop to repeat?*
- *What will change so that the loop eventually ends?*

Model 3 Working with Lists

Recall that a variable can hold multiple values in the form of a list. In Python, the values are separated by commas and wrapped in square brackets.

One at a time, and in the order presented below, type the following Python commands into the shell in Thonny. Record the results for each command.

Python commands	Shell output
rolls = [4, 6, 3, 2, 6]	
len(rolls)	
print(rolls[0])	
print(rolls[2])	
print(rolls[4])	
print(rolls[5])	
rolls.append(1)	
print(rolls)	
print(rolls[5])	
rolls.remove(6)	
print(rolls)	
lucky.append(5)	
lucky = [ ]	
len(lucky)	
lucky.append(5)	
print(lucky)	
len(lucky)	
students = ["Amy" , "Ben", "Chris"]	
len(students)	
print(students[1])	
students[1] = "Bill"	
print(students)	
students[3] = "Dave"	

Questions

25. If there are five items in a list, what index values "access" the five values?

26. If there are N items in a list, what index values "access" the N values?

27. What is the result of calling the append method on a list?

28. What must be defined prior to using a method like append?

29. Explain why two lines in Model 3 caused an IndexError.

30. Describe the similarities and differences between using a list method like append or remove and Python built-in functions like print or len.

31. The final command, an attempt to add a fourth student to the list, failed. What command would add the name "Dave" to the list?

Model 4 Working with Strings

A string is a sequence of characters in single quotes (') or double quotes ("). Depending on the application, we can treat a String as a single value (e.g., code), or we can access individual characters using square brackets (e.g., code[0]).

One at a time, and in the order presented below, type the following Python commands into the shell in Thonny. Record the results for each command.

Python commands	Shell output
code = "Python"	
len(code)	
print(code[0])	
print(code[3])	
print(code[5])	
print(code[6])	
code.append("!")	
code = code + "!"	
print(code)	
print(code[6])	
name = name + "Ben"	
name = ""	
len(name)	
name = name + "Ben"	
print(name)	
print(name[1])	
name[1] = "a"	

Questions

32. If there are six characters in a String, what index values "access" the six characters?

33. If there are N characters in a String, what index values "access" the N characters?

34. Explain what caused the IndexError.

35. Explain what caused the AttributeError.

36. Explain what caused the NameError.

37. Explain what caused the TypeError.

Model 5 Indexing and Slicing

In addition to using a single index to access We can also use ***slice notation*** (e.g., dna[4:8]) to refer to a range of characters. In fact, all types of sequences (including list and tuple) support indexing and slicing.

Python code	Shell output
dna = "CTGAGGACGT"	
dna[5]	
dna[10]	
len(dna)	
triplet = dna[2:5]	
print(triplet)	
dna[0:5]	
dna[5:10]	
dna[:5]	
dna[5:]	
dna[-1]	
dna[-2]	
dna[-3]	
dna[-9]	
dna[-10]	
dna[:-4]	
dna[-4:]	
triplet = dna[-4:-1]	
print(triplet)	

Questions

38. What is the *positive* index of each character in the dna string? Check your answers above.

Character:	C	T	G	A	G	G	A	C	G	T
Index:										

39. What is the *negative* index of each character in the dna string Check your answers above.

Character:	C	T	G	A	G	G	A	C	G	T
Index:										

40. Based on the previous questions, what are dna[3] and dna[-3]? Explain your answers.

41. What is the simplest way to get the first three characters of dna? What is the simplest way to get the last three characters?

42. Write a Python expression that slices `'GACG'` from dna using positive indexes. Then write another expression that slices the same string using negative indexes.

43. Write a Python assignment statement that uses the `len` function to assign the last letter of dna to the variable last.

44. Write a Python assignment statement that uses a negative index to assign the last letter of dna to the variable last.