Welcome to English Programming

Week 4 Class 1

Asynchronous Class

Reading Challenge: Practice

Text

4. More Control Flow Tools — Python 3.9.4 documentation

4. More Control Flow Tools

Besides the while statement just introduced, Python uses the usual flow control statements known from other languages, with some twists.

4.1. _____

(Paragraph 1) Perhaps the most well-known statement type is the if statement.

(Paragraph 2) There can be zero or more elif parts, and the else part is optional. The keyword 'elif' is short for 'else if', and is useful to avoid excessive indentation. An if … elif … elif … sequence is a substitute for the switch or case statements found in other languages.

4.2. _____

(Paragraph 3) The for statement in Python differs a bit from what you may be used to in C or Pascal. Rather than always iterating over an arithmetic progression of numbers (like in Pascal), or giving the user the ability to define both the iteration step and halting condition (as C), Python's for statement iterates over the items of any sequence (a list or a string), in the order that they appear in the sequence.

(Paragraph 4) Code that modifies a collection while iterating over that same collection can be tricky to get right. Instead, it is usually more straight-forward to loop over a copy of the collection or to create a new collection:

4.3. _____

(Paragraph 5) If you do need to iterate over a sequence of numbers, the built-in function range() comes in handy. It generates arithmetic progressions:

```
>>> for i in range(5):
...     print(i)
...
0
1
2
3
4
```

Illustration 1

(Paragraph 6) The given end point is never part of the generated sequence; range(10) generates 10 values, the legal indices for items of a sequence of length 10. It is possible to let the range start at another number, or to specify a different increment (even negative; sometimes this is called the 'step'):

```
range(5, 10)
   5, 6, 7, 8, 9

range(0, 10, 3)
   0, 3, 6, 9

range(-10, -100, -30)
  -10, -40, -70
```

Illustration 2

(Paragraph 7) To iterate over the indices of a sequence, you can combine range() and len() as follows:

```
>>> a = ['Mary', 'had', 'a', 'little', 'lamb']
>>> for i in range(len(a)):
...     print(i, a[i])
...
0 Mary
1 had
2 a
3 little
4 lamb
```

Illustration 3

(Paragraph 8) In most such cases, however, it is convenient to use the enumerate() function, see Looping Techniques.

(Paragraph 9) A strange thing happens if you just print a range:

```
>>> print(range(10))
range(0, 10)
```

Illustration 4

(Paragraph 10) In many ways the object returned by range() behaves as if it is a list, but in fact it isn't. It is an object which returns the successive items of the desired sequence when you iterate over it, but it doesn't really make the list, thus saving space.

(Paragraph 11) We say such an object is iterable, that is, suitable as a target for functions and constructs that expect something from which they can obtain successive items until the supply is exhausted. We have seen that the for statement is such a construct, while an example of a function that takes an iterable is sum():

```
>>> sum(range(4))  # 0 + 1 + 2 + 3
6
```

Illustration 5

(Paragraph 12) Later we will see more functions that return iterables and take iterables as arguments. Lastly, maybe you are curious about how to get a list from a range. Here is the solution:

```
>>> list(range(4))
[0, 1, 2, 3]
```

Illustration 6

(Paragraph 13) In chapter Data Structures, we will discuss in more detail about list().

4.4. _____

(Paragraph 14) The break statement, like in C, breaks out of the innermost enclosing for or while loop.

(Paragraph 15) Loop statements may have an else clause; it is executed when the loop terminates through exhaustion of the iterable (with for) or when the condition becomes false (with while), but not when the loop is terminated by a break statement.

(Paragraph 16) When used with a loop, the else clause has more in common with the else clause of a try statement than it does with that of if statements: a try statement's else clause runs when no exception occurs, and a loop's else clause runs when no break occurs. For more on the try statement and exceptions, see Handling Exceptions.

(Paragraph 17) The continue statement, also borrowed from C, continues with the next iteration of the loop:

```
>>> for num in range(2, 10):
...     if num % 2 == 0:
...         print("Found an even number", num)
...         continue
...     print("Found an odd number", num)
Found an even number 2
Found an odd number 3
Found an even number 4
Found an odd number 5
Found an even number 6
Found an odd number 7
Found an even number 8
Found an odd number 9
```

Illustration 7

4.5. _____

(Paragraph 18) The pass statement does nothing. It can be used when a statement is required syntactically but the program requires no action. For example:

```
>>> while True:
...     pass  # Busy-wait for keyboard interrupt (Ctrl+C)
...
```

Illustration 8

(Paragraph 19) This is commonly used for creating minimal classes:

```
>>> class MyEmptyClass:
...     pass
...
```

Illustration 9

(Paragraph 20) Another place pass can be used is as a place-holder for a function or conditional body when you are working on new code, allowing you to keep thinking at a more abstract level. The pass is silently ignored:

```
>>> def initlog(*args):
...     pass   # Remember to implement this!
...
```

Illustration 10

1. Skimming

Match the titles with the corresponding section.

Section		Title
4.1 _____	a.	The range() Function
4.2 _____	b.	for Statements
4.3 _____	c.	pass Statements
4.4 _____	d.	if Statements
4.5 _____	e.	break and continue Statements, and else Clauses on Loops

2. Scanning

Complete the following sentences with words from the text.

1. The *if* statement is the most _____ statement.
2. The '*else if*' is the long version of the _____ keyboard.
3. The _____ function generates arithmetic progressions.
4. The combination of _____ and _____ are useful to repeat indices of a sequence.

3. Meaning from Context

Select the most appropriate answer to the following questions.

1. In the phrase *"[…] giving the user the ability to define […] the **iteration** step […],"* the word 'iteration' on paragraph 3 means:
    a. the process of repeating set of instructions again and again, each time applying it to the result of the previous stage
    b. a new version of a piece of computer software
    c. the act of repeating
2. In the phrase *"[…] giving the user the ability to define […] **halting** condition (as C) […],"* the word 'halting' on paragraph 3 means:
    a. Stutter
    b. Obstruct
    c. Stop
3. In the sentence "that same collection can be difficult to get right. Instead, it is usually more *straight-forward* to ***loop*** over a copy of the collection," the word 'loop' on paragraph 4 means:
    a. to form or bend something into a circular shape
    b. to move in a circular way
    c. to repeat a piece of code until a certain condition is reached
4. In the sentence "that same collection can be difficult to get right. Instead, it is usually more ***straight-forward*** to loop over a copy of the collection," the antonym or opposite idea of the word 'straight-forward' on paragraph 4 is:
    a. Long
    b. Difficult
    c. false

4. Parts of speech

Answer the following questions.

1. Which word can replace the word "*point*" on paragraph 6? Select the correct answer.
    a. They
    b. We
    c. it
2. Select the verbs in the following sentence from the text. *A strange thing happens if you just print a range*. (Paragraph 9)
    a. Strange
    b. Thing
    c. Happens
    d. You
    e. Print
    f. Range
3. What type of 'items' are the ones mentioned in the sentence: *"they can obtain successive items"*? Write the answer on the space provided. _____
4. According to the following sentence from the text, how is the pass statement used? *"This is commonly used for creating minimal classes*." (paragraph 19) Write the answer on the space provided. _____

5. Cognates

Write the best translation for the following cognates.

1. Convenient (paragraph 8) _____
2. Object (paragraph 10) _____
3. Clause (paragraph 16) _____
4. Exceptions (paragraph 16) _____

6. Reading comprehension

Are the following statements true (T) or false (F).

1. The flow controls used by Python are exactly the same as the ones used by other languages. T/F
2. The *for* statement in C and Rascal is not different from the one from Python. T/F
3. It is easier to create and repeat a copy of a collection than to modify that collection and at the same time iterate it. T/F
4. The final number in a range is not indicated in the sequence produced by the range function. T/F

Self-Assessment

Read the following statements and tick YES or NO in each case.

Statements	Yes	No
1. I can skim a text to find what a text, specific section, or paragraph is about.		
2. I can scan a text to find specific information.		
3. I can use the context to find the meaning of unfamiliar words.		
4. I can identify and understand the different parts of speech.		
5. I identify cognates in the text.		
6. I can understand what the text is about.		

Answer Key

1. Skimming

Match the titles with the corresponding section.

Section	Title
4.1 d	a. The range() Function
4.2 b	b. for Statements
4.3 a	c. pass Statements
4.4 e	d. if Statements
4.5 c	e. break and continue Statements, and else Clauses on Loops

2. Scanning

Complete the following sentences with words from the text.

5. The *if* statement is the most well-known statement.
6. The '*else if*' is the long version of the 'elif' keyboard.
7. The range function generates arithmetic progressions.
8. The combination of *range* and *len* are useful to repeat indices of a sequence.

3. Meaning from Context

Select the most appropriate answer to the following questions.

5. In the phrase *"[…] giving the user the ability to define […] the **iteration** step […],"* the word 'iteration' on paragraph 3 means:
   a. the process of repeating set of instructions again and again, each time applying it to the result of the previous stage

b. a new version of a piece of computer software

c. the act of repeating

6. In the phrase *"[…] giving the user the ability to define […]* ***halting*** *condition (as C) […],"* the word 'halting' on paragraph 3 means:

   a. Stutter

   b. Obstruct

   c. Stop

7. In the sentence "that same collection can be difficult to get right. Instead, it is usually more *straight-forward* to ***loop*** over a copy of the collection," the word 'loop' on paragraph 4 means:

   a. to form or bend something into a circular shape

   b. to move in a circular way

   c. to repeat a piece of code until a certain condition is reached

8. In the sentence "that same collection can be difficult to get right. Instead, it is usually more ***straight-forward*** to loop over a copy of the collection," the antonym or opposite idea of the word 'straight-forward' on paragraph 4 is:

   a. Long

   b. Difficult

   c. false

4. Parts of speech

Answer the following questions.

5. Which word can replace the word "*point*" on paragraph 6? Select the correct answer.

   a. They

   b. We

   c. it

5. Select the verbs in the following sentence from the text. *A strange thing happens if you just print a range*. (Paragraph 9)
    a. Strange
    b. Thing
    c. Happens
    d. You
    e. Print
    f. Range
6. What type of 'items' are the ones mentioned in the sentence: *"they can obtain successive items"*? Write the answer on the space provided. successive
7. According to the following sentence from the text, how is the `pass` statement used? *"This is commonly used for creating minimal classes*." (paragraph 19) Write the answer on the space provided. commonly

5. Cognates

Write the best translation for the following cognates.

1. Convenient (paragraph 8) conveniente
2. Object (paragraph 10) objeto
3. Clause (paragraph 16) clausula (fórmula lógica)
4. Exceptions (paragraph 16) excepciones

6. Reading comprehension

Are the following statements true (T) or false (F).

5. The flow controls used by Python are exactly the same as the ones used by other languages. T/F
6. The *for* statement in C and Rascal is not different from the one from Python. T/F

7. It is easier to create and repeat a copy of a collection than to modify that collection and at the same time iterate it. T/F
8. The final number in a range is not indicated in the sequence produced by the range function. T/F