course_1_assessment_11

Due: 2018-11-25 01:25:00

Description: Assessment for Way of Programmer Week four.   Score: 11.0 of 11 = 100.0%

Questions

Which of these is a correct reference diagram following the execution of the following code?
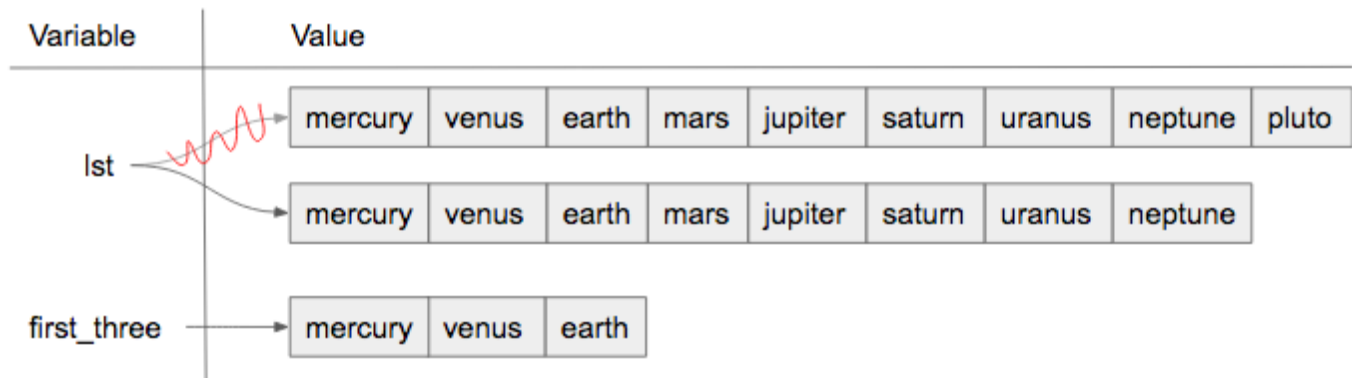
Score: 1.0 / 1

Comment: autograded

```
lst = ['mercury', 'venus', 'earth', 'mars', 'jupiter', 'saturn', 'ura
lst.remove('pluto')
first_three = lst[:3]
```

1.

2.

- ○ A. I.
- ○ B. II.
- ○ C. Neither is the correct reference diagram.

Check Me   Compare me

✔️ Yes, when we are using the remove method, we are just editing the existing list, not making a new copy.

Activity: 9.16.1 Multiple Choice (assess_question4_1_1_1)

Question in Context
(/runestone/books/published/fopp/TransformingSequences/week4a1.html#assess_question4_1_1_1)

What will be the value of `a` after the following code has executed?

```
a = ["holiday", "celebrate!"]
quiet = a
quiet.append("company")
```

The value of `a` will be

```
["holiday", "celebrate!", "con
```

[Check me] [Compare me]

Good work!

Score: 1.0 / 1

Comment: autograded

Activity: 9.16.1.1 Fill in the Blank (assess_question3_3_1_1)

Question in Context
(/runestone/books/published/fopp/TransformingSequences/week4a1.html#assess_question3_3_1_1)

Could aliasing cause potential confusion in this problem?

```
b = ['q', 'u', 'i']
z = b
b[1] = 'i'
z.remove('i')
print(z)
```

🔘 A. yes

⭕ B. no

[Check Me] [Compare me]

Score: 1.0 / 1

Comment: autograded

✔️ Yes, b and z reference the same list and changes are made using both aliases.

Activity: 9.16.1.2 Multiple Choice (assess_question3_3_1_2)

Question in Context
(/runestone/books/published/fopp/TransformingSequences/week4a1.html#assess_question3_3_1_2)

Given that we want to accumulate the total sum of a list of numbers, which of the following accumulator patterns would be appropriate?

Score: 1.0 / 1

Comment: autograded

1.

```
nums = [4, 5, 2, 93, 3, 5]
s = 0
for n in nums:
    s = s + 1
```

2.

```
nums = [4, 5, 2, 93, 3, 5]
s = 0
for n in nums:
    s = n + n
```

3.

```
nums = [4, 5, 2, 93, 3, 5]
s = 0
for n in nums:
    s = s + n
```

○ A. I.

○ B. II.

◉ C. III.

○ D. none of the above would be appropriate for the problem.

[Check Me] [Compare me]

✔️ Yes, this will solve the problem.

Activity: 9.16.4.1 Multiple Choice (assess_question5_2_1_1)

Question in Context
(/runestone/books/published/fopp/TransformingSequences/week4a1.html#assess_question5_2_1_1)

Given that we want to accumulate the total number of strings in the list, which of the following accumulator patterns would be appropriate?

Score: 1.0 / 1

Comment: autograded

1.

```
lst = ['plan', 'answer', 5, 9.29, 'order, items', [4]]
s = 0
for n in lst:
    s = s + n
```

2.

```
lst = ['plan', 'answer', 5, 9.29, 'order, items', [4]]
for item in lst:
    s = 0
    if type(item) == type("string"):
        s = s + 1
```

3.

```
lst = ['plan', 'answer', 5, 9.29, 'order, items', [4]]
s = ""
for n in lst:
    s = s + n
```

4.

```
lst = ['plan', 'answer', 5, 9.29, 'order, items', [4]]
s = 0
for item in lst:
    if type(item) == type("string"):
        s = s + 1
```

○ A. 1.

○ B. 2.

○ C. 3.

⦿ D. 4.

○ E. none of the above would be appropriate for the problem.

| Check Me | Compare me |

✔ Yes, this will solve the problem.

Activity: 9.16.4.2 Multiple Choice (assess_question5_2_1_2)

Question in Context
(/runestone/books/published/fopp/TransformingSequences/week4a1.html#assess_question5_2_1_2)

Which of these are good names for an accumulator variable? Select as many as apply.

Score: 1.0 / 1

Comment: autograded

☐ A. sum

☐ B. x

☑ C. total

☑ D. accum

☐ E. none of the above

| Check Me | Compare me |

✔
    C. Yes, total is a good name for accumulating numbers.
    D. Yes, accum is a good name. It's both short and easy to remember.

Activity: 9.16.4.3 Multiple Choice (assess_question5_2_1_3)

Question in Context
(/runestone/books/published/fopp/TransformingSequences/week4a1.html#assess_question5_2_1_3)

Which of these are good names for an iterator (loop) variable? Select as many as apply.

Score: 1.0 / 1

Comment: autograded

☑ A. item

☐ B. y

☑ C. elem

☑ D. char

☐ E. none of the above

| Check Me | Compare me |

✔
    A. Yes, item can be a good name to use as an iterator variable.
    C. Yes, elem can be a good name to use as an iterator variable, especially when iterating over lists.

D. Yes, char can be a good name to use when iterating over a string, because the iterator variable would be assigned a character each time.

Activity: 9.16.4.4 Multiple Choice (assess_question5_2_1_4)

Question in Context
(/runestone/books/published/fopp/TransformingSequences/week4a1.html#assess_question5_2_1_4)

Which of these are good names for a sequence variable? Select as many as apply.

Score: 1.0 / 1

Comment: autograded

☑ A. num_lst

☐ B. p

☑ C. sentence

☑ D. names

☐ E. none of the above

[Check Me]   [Compare me]

✔

A. Yes, num_lst is good for a sequence variable if the value is actually a list of numbers.
C. Yes, this is good to use if the for loop is iterating through a string.
D. Yes, names is good, assuming that the for loop is iterating through actual names and not something unrelated to names.

Activity: 9.16.4.5 Multiple Choice (assess_question5_2_1_5)

Question in Context
(/runestone/books/published/fopp/TransformingSequences/week4a1.html#assess_question5_2_1_5)

Given the following scenario, what are good names for the accumulator variable, iterator variable, and sequence variable? You are writing code that uses a list of sentences and accumulates the total number of sentences that have the word 'happy' in them.

Score: 1.0 / 1

Comment: autograded

○ A. accumulator variable: x | iterator variable: s | sequence variable: lst

○ B. accumulator variable: total | iterator variable: s | sequence variable: lst

○ C. accumulator variable: x | iterator variable: sentences | sequence variable: sentence_lst

◉ D. accumulator variable: total | iterator variable: sentence |sequence variable: sentence_lst

◯ E. none of the above

| Check Me | Compare me |

✔ Yes, this combination of variable names is the clearest.

Activity: 9.16.4.6 Multiple Choice (assess_question5_2_1_6)

Question in Context
(/runestone/books/published/fopp/TransformingSequences/week4a1.html#assess_question5_2_1_6)

Score: 1.0 / 1

Comment: autograded

For each character in the string saved in `ael` , append that character to a list that should be saved in a variable `app` .

| Save & Run | 15/09/2022, 22:52:08 - 2 of 2 | Show CodeLens |

```
1
2 ael = "python!"
3 app = []
4 for w in ael:
5     app.append(w)
6 print(app)
7
8
```

Activity: 9.16.4.7 ActiveCode (access_ac_5_2_1_1)

Question in Context (/runestone/books/published/fopp/TransformingSequences/week4a1.html#access_ac_5_2_1_1)

Score: 1.0 / 1

Comment: autograded

For each string in `wrds` , add 'ed' to the end of the word (to make the word past tense). Save these past tense words to a list called `past_wrds` .

Save & Run   15/09/2022, 22:52:18 - 2 of 2   Show CodeLens

```
1 wrds = ["end", 'work', "play", "start", "walk", "look",
2          "open", "rain", "learn", "clean"]
3
4 past_wrds = [word + "ed" for word in wrds]
5 print(past_wrds)
```

Activity: 9.16.4.8 ActiveCode (access_ac_5_2_1_2)

Question in Context (/runestone/books/published/fopp/TransformingSequences/week4a1.html#access_ac_5_2_1_2)

Score Me

username: girishkhule@gmail.com | Back to top