Question #1

Topic 1

Insert the correct snippet so that the program produces the expected output.

Expected output:

```
True
```

Code:

```
list = [False, True, "2", 3, 4, 5]
# insert code here
print(b)
```

A. b = 0 not in list

B. b = list[0]

C. b = 0 in list

D. b = False

Correct Answer: *C*

Community vote distribution

C (70%) A (30%)

vmse10 Highly Voted 5 months, 3 weeks ago

I tested the code:

list = [False, True, "2", 3, 4, 5]
b = 0 in list
print(b)

The answer is "C":

upvoted 7 times

Wolfincat Most Recent 1 month ago

Selected Answer: C

0 equal False
so choose C

upvoted 1 times

Vihaan_C 2 months ago

Selected Answer: C

0=False in python and false is there hence it is C

upvoted 1 times

Vano1 2 months ago

Selected Answer: C

False is seen by Python as 0, therefore the correct answer is C.

upvoted 1 times

MukomaGyie 2 months, 2 weeks ago

The correct answer is C

upvoted 1 times

Maurioasis 2 months, 3 weeks ago

The answer is "C":
if 0 in list #return True, so b=True and print b return True

upvoted 1 times

anvitan 2 months, 3 weeks ago

Selected Answer: C

It is C. 0 denotes false in python. So b in list is false, which returns true

upvoted 2 times

Poojasr 4 months ago

option A

upvoted 1 times

JullioSanntos 4 months ago

Selected Answer: C

I second yan45

upvoted 1 times

yan45 4 months ago

Selected Answer: C

c is the correct answer because in Python, False is equivalent to 0, and True is equivalent to 1. When checking if 0 is in a list using the in operator, Python will return True if False is present in the list because 0 == False.

upvoted 2 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

JoeJin 5 months, 3 weeks ago

A is correct
In Python, the comparison 0 in list evaluates to True because False is treated as equivalent to 0. Here's an explanation:

False is equivalent to 0 in Python (and True is equivalent to 1).
When checking 0 in list, Python finds False in the list, which is equivalent to 0.
Thus, the condition 0 in list returns True.

upvoted 2 times

ellene 6 months, 3 weeks ago

Let's evaluate each option:

Option A will result in b = 0 not in list which checks if 0 is not in the list. Since 0 is not in the list, this will result in True.
Option B will result in b = list[0] which assigns False to b.
Option C will result in b = 0 in list which checks if 0 is in the list. Since 0 is not in the list, this will result in False.
Option D will directly assign False to b.
Since we want the output to be True, option A is the correct one

upvoted 2 times

skala2 6 months, 3 weeks ago

Correct Answer is C
True boolean value is 0, so it can be considered as 0 is in the list.
Please try in the python lab to check it if you are confused.

upvoted 4 times

herrmann69 7 months ago

Selected Answer: A

A is correct.
0 is not in the list, hence its True

upvoted 3 times

Question #2

Topic 1

Assuming that the tuple is a correctly created tuple, the fact that tuples are immutable means that the following instruction:

```
my_tuple[1] = my_tuple[1] + my_tuple[0]
```

A. is illegal

B. may be illegal if the tuple contains strings

C. can be executed if and only if the tuple contains at least two elements

D. is fully correct

Correct Answer: *A*

Community vote distribution

A (100%)

Vihaan_C 2 months ago

Selected Answer: A

Because tuples are immutable

upvoted 1 times

consultsk 3 months ago

Selected Answer: A

Explanation:

Tuples are immutable: This means that once a tuple is created, its elements cannot be changed. Any attempt to modify an element in a tuple, as shown in the code, will result in an error.

Error: The code will raise a TypeError because Python does not allow the modification of elements in a tuple.

upvoted 2 times

Poojasr 4 months ago

option A

upvoted 1 times

christostz03 4 months, 2 weeks ago

a is the correct answer

upvoted 1 times

Supatekooma 4 months, 4 weeks ago

my_tuple = (3, 5, 7)

new_tuple = (my_tuple[0], my_tuple[1] + my_tuple[0], my_tuple[2])

3,5 + 3,7

print(new_tuple)

# The Given Instruction:

# my_tuple[1] = my_tuple[1] + my_tuple[0]

# This instruction tries to change the element at index 1 of my_tuple by adding the element at index 0 to it.

# Attempting to Modify a Tuple:

# Since tuples are immutable, any attempt to directly change an element in a tuple will result in a TypeError.

upvoted 1 times

herrmann69 7 months ago

Selected Answer: A

A is correct.

TypeError: 'tuple' object does not support item assignment

upvoted 3 times

Question #3 *Topic 1*

What is the expected output of the following code?

```
x = [0, 1, 2]
x.insert(0, 1)
del x[1]
print(sum(x))
```

A. 2

B. 4

C. 5

D. 3

Correct Answer: *B*

Community vote distribution

B (100%)

hovnival 1 week, 6 days ago

Selected Answer: B

let's break down the code step by step to understand the result:
Code:

x = [0, 1, 2] # Step 1: x is initialized as [0, 1, 2].
x.insert(0, 1) # Step 2: Inserts 1 at index 0. x becomes [1, 0, 1, 2].
del x[1] # Step 3: Deletes the element at index 1. x becomes [1, 1, 2].
print(sum(x)) # Step 4: Calculates the sum of elements in x, which is 1 + 1 + 2 = 4.

Result:

The output of the code is 4.

upvoted 1 times

Vihaan_C 2 months ago

Selected Answer: B

x=[0,1,2]
x=[1,0,1,2]
x=[1,1,2]
sum=1+1+2=4

upvoted 2 times

consultsk 3 months ago

Selected Answer: B

The given code performs the following operations:

1. `x = [0, 1, 2]`: Initializes the list `x` with the elements `[0, 1, 2]`.
2. `x.insert(0, 1)`: Inserts the value `1` at index `0`, so the list becomes `[1, 0, 1, 2]`.
3. `del x[1]`: Deletes the element at index `1`, so the list becomes `[1, 1, 2]`.
4. `print(sum(x))`: Calculates the sum of the elements in the list `x` and prints it.

Now, the list `x` is `[1, 1, 2]`, and the sum of these elements is `1 + 1 + 2 = 4`.

Expected output: `4`

upvoted 2 times

megan_mai 4 months, 1 week ago

Selected Answer: B

x.insert(0,1) >> insert value 1 to x[0] >> The new x = [1,0,1,2]

del x[1] >> delete x[1] >> delete value 0 in x

sum(x) = 1+1+2 = 4

upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #4 *Topic 1*

What is the expected output of the following code?

```python
list1 = [1, 3]
list2 = list1
list1[0] = 4
print(list2)
```

A. [1, 3]

B. [1, 4]

C. [4, 3]

D. [1, 3, 4]

Correct Answer: *C*

Community vote distribution

C (100%)

hovnival 1 week, 6 days ago

Selected Answer: C

C. [4, 3]

Explanation:

list1 = [1, 3]

This creates a list list1 with elements [1, 3].

list2 = list1

This does not create a new list. Instead, list2 is a reference to the same list object that list1 points to. Both list1 and list2 now refer to the same memory location.

list1[0] = 4

This modifies the first element of the list referred to by list1 (and list2, since they refer to the same object). The list is now [4, 3].

print(list2)

Since list2 refers to the same list as list1, it will also reflect the change. Therefore, the output will be [4, 3].

upvoted 1 times

Vihaan_C 2 months ago

Selected Answer: C

list1=[1,3]

list2=list1 which equals to [1,3]

list1=[4,3] and list2=[4,3]

upvoted 1 times

consultsk 3 months ago

Selected Answer: C

The code performs the following steps:

1. `list1 = [1, 3]`: Initializes `list1` with the elements `[1, 3]`.
2. `list2 = list1`: Assigns `list2` to refer to the same list object as `list1`. Now both `list1` and `list2` refer to the same list in memory.
3. `list1[0] = 4`: Modifies the first element of `list1` to be `4`. Since `list2` refers to the same list, this modification is reflected in `list2` as well.
4. `print(list2)`: Prints `list2`.

Since `list2` refers to the same list as `list1`, after modifying `list1`, `list2` will also reflect the changes.

Expected output: `[4, 3]`

upvoted 2 times

megan_mai 4 months, 1 week ago

Selected Answer: C

After assigning list2 = list 1 >> both refer to the same list in memory

>> changing list1 value will affect list2 value

>> answer is [4,3]

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

herrmann69 7 months ago

Selected Answer: C

C is correct.

list1 and list2 point to the same location in memory.

upvoted 3 times

Question #5 *Topic 1*

What is the expected output of the following code?

```
data = ['Peter', 404, 3.03, 'Wellert', 33.3]
print(data[1:3])
```

A. ['Peter', 404, 3.03, 'Wellert', 33.3]

B. None of the above.

C. [404, 3.03]

D. ['Peter', 'Wellert']

Correct Answer: *C*

Community vote distribution

C (100%)

consultsk 3 months ago

Selected Answer: C

The code performs the following steps:

1. `data = ['Peter', 404, 3.03, 'Wellert', 33.3]`: Initializes the list `data` with the elements `['Peter', 404, 3.03, 'Wellert', 33.3]`.
2. `print(data[1:3])`: Slices the list `data` starting from index `1` up to, but not including, index `3`, and prints the resulting sublist.

The slice `data[1:3]` will include the elements at indices `1` and `2`, which are `404` and `3.03`.

Expected output: `[404, 3.03]`

upvoted 1 times

megan_mai 4 months, 1 week ago

Selected Answer: C

data[1,3] takes the element from data[1] to but no include data[3] >> answer is C. [404,3.03]

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

vmse10 5 months, 3 weeks ago

The print(data[1:3]) statement uses list slicing to extract a sublist from the data list. In Python, list slicing is done using the syntax list[start:stop], where:

start is the index where the slice begins (inclusive).
stop is the index where the slice ends (exclusive).
For the given list data:

The element at index 1 is 404.
The element at index 2 is 3.03.
The element at index 3 is 'Wellert' (this element is not included in the slice because the stop index is exclusive).
Therefore, data[1:3] will include the elements at indices 1 and 2.

Result:
The resulting sublist will be [404, 3.03].

Answer Choices:
A. ['Peter', 404, 3.03, 'Wellert', 33.3]
B. None of the above.
C. [404, 3.03]

D. ['Peter', 'Wellert']

The correct answer is:
C. [404, 3.03]
upvoted 4 times

Question #6 *Topic 1*

Take a look at the snippet, and choose the true statements: (Choose two.)

```python
nums = [1, 2, 3]
vals = nums
del vals[1:2]
```

A. nums is longer than vals

B. nums and vals are of the same length

C. vals is longer than nums

D. nums and vals refer to the same list

Correct Answer: *BD*

⊟ **hovnival** 1 week, 6 days ago

Selected Answer: BD

Let's analyze the code snippet step by step:

nums = [1, 2, 3]
Creates a list nums with three elements: [1, 2, 3].

vals = nums
The variable vals is assigned the same reference as nums. Now both vals and nums point to the same list object in memory.

del vals[1:2]
The slice vals[1:2] selects the element at index 1 (value 2).
The del statement removes this element from the list. Since vals and nums point to the same list, the change affects both.

After this operation, the list becomes [1, 3].

Now evaluate the options:

A. nums is longer than vals
False. Both nums and vals refer to the same list, which now has a length of 2.

B. nums and vals are of the same length
True. Both nums and vals are the same list, so their lengths are equal.

C. vals is longer than nums
False. Both nums and vals refer to the same list.

D. nums and vals refer to the same list
True. nums and vals point to the same memory location, so they refer to the same list.

Correct Answers:

B. nums and vals are of the same length
D. nums and vals refer to the same list
upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

b & d are correct answers
upvoted 1 times

⊟ **froster02** 5 months, 2 weeks ago

vals = nums -> this statement means both the Vals & nums will share same memory pool

vals = noms[:] -> this means both have different memory pool

upvoted 2 times

Question #7 *Topic 1*

What is the output of the following snippet?

```
dct = {}
dct['1'] = (1, 2)
dct['2'] = (2, 1)

for x in dct.keys():
    print(dct[x][1], end='')
```

A. 12

B. (2, 1)

C. (1, 2)

D. 21

Correct Answer: *D*

Community vote distribution

D (100%)

consultsk 3 months ago

Selected Answer: D

To determine the output of this code, let's go through it step by step:

dct = {} creates an empty dictionary.
dct['1'] = (1, 2) adds a key-value pair to the dictionary. The key is the string '1', and the value is the tuple (1, 2).
dct['2'] = (2, 1) adds another key-value pair. The key is the string '2', and the value is the tuple (2, 1).
The for loop iterates over the keys of the dictionary using dct.keys().
For each key x, it prints dct[x][1] with end=''. This means it's printing the second element of each tuple (index 1), and the end='' argument prevents a newline after each print.

So, when we iterate over the keys:

For key '1', it will print 2 (the second element of (1, 2))
For key '2', it will print 1 (the second element of (2, 1))

These will be printed on the same line because of end=''.
Therefore, the output will be: 21
upvoted 3 times

megan_mai 4 months, 1 week ago

Selected Answer: D

dct = {}
dct['1'] = (1,2) #second element of this key is 2
dct['2'] = (2,1) #second element of this key is 1
for x in dct.keys():
print(dct[x][1], end='') #print out the second element of each key continuously
#>>> the answer is 21 (D)
upvoted 1 times

christostz03 4 months, 2 weeks ago

d is the correct answer
upvoted 1 times

froster02 5 months, 2 weeks ago

answer : 21
2 is of the dct['1'] and 1 is of dct['2']

upvoted 1 times

Question #8 *Topic 1*

What is the expected output of the following code?
print(list('hello'))

A. hello

B. [h, e, l, l, o]

C. ['h', 'e', 'l', 'l', 'o']

D. ['h' 'e' 'l' 'l' 'o']

E. None of the above.

Correct Answer: *C*

Community vote distribution

C (85%) E (15%)

consultsk 3 months ago

Selected Answer: C

This code does the following:

The string 'hello' is passed to the list() function.
list() converts the string into a list, where each character of the string becomes an individual element of the list.
The resulting list "['h', 'e', 'l', 'l', 'o']" is then printed.

Each character from the string 'hello' becomes a separate string element in the list, preserving the order of the original string. The output shows this list of single-character strings.

upvoted 2 times

megan_mai 4 months, 1 week ago

Selected Answer: C

Tested on Databricks
list('Hello') >>> a list containing all the character of the work "Hello"
print(list('Hello')) >>> print the list with the right format, this is what i got after running the code: ['h', 'e', 'l', 'l', 'o'] (C)

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

AlexNS653 4 months, 3 weeks ago

Selected Answer: C

Correct answer is C. Anyone who is guessing the answer is 'E' is incorrect, but an easy mistake to make. You are getting the Traceback TypeError traceback because you have a value assigned to the variable 'list' from one of the previous questions. So, when you type 'list' you aren't calling the function list, you are referencing the variable 'list'. Restart your shell and try the print(list('hello')) statement again and it will return ['h', 'e', 'l', 'l', 'o'] as it should. This is why we are taught to never use a reserved word in Python as a variable name.

upvoted 3 times

AlexNS653 4 months, 3 weeks ago

Anyone who is guessing the answer is 'E' is incorrect, but an easy mistake to make. You are getting the Traceback TypeError traceback because you have a value assigned to the variable 'list' from one of the previous questions. So, when you type 'list' you aren't calling the function list, you are referencing the variable 'list'. Restart your shell and try the print(list('hello')) statement again and it will return ['h', 'e', 'l', 'l', 'o'] as it should. This is why we are taught to never use a reserved word in Python as a variable name.

upvoted 1 times

tolliik 5 months, 2 weeks ago

Who got the error: "TypeError: 'list' object is not callable" clear your terminal session. In the question 1, you probably used a variable list, so there is a conflict between the function "list" and a variable "list". So C is right. Not E

upvoted 1 times

froster02 5 months, 2 weeks ago

C, as it is the list so answer will be format of C option only

upvoted 1 times

nieups 5 months, 3 weeks ago

Selected Answer: C

C is the right answer

upvoted 3 times

JoeJin 5 months, 3 weeks ago

Selected Answer: E

```
TypeError Traceback (most recent call last)
Cell In[6], line 1
----> 1 print(list('hello'))

TypeError: 'list' object is not callable
```

upvoted 1 times

JoeJin 5 months, 3 weeks ago

E is correct according to jupitor notebook...

```
print(list('hello'))

TypeError Traceback (most recent call last)
Cell In[6], line 1
----> 1 print(list('hello'))

TypeError: 'list' object is not callable
```

upvoted 1 times

BAune 6 months ago

Selected Answer: C

C is correct, tested in Pycharm.

upvoted 3 times

vlad29 6 months, 1 week ago

Selected Answer: E

The restul is: TypeError: 'list' object is not callable

upvoted 1 times

Question #9 *Topic 1*

What will be the output of the following code snippet?

```
a = [1, 2, 3, 4, 5, 6, 7, 8, 9]
print(a[::2])
```

A. [1, 3, 5, 7, 9]

B. [8, 9]

C. [1, 2, 3]

D. [1, 2]

Correct Answer: *A*

Community vote distribution

A (100%)

hovnival 1 week, 5 days ago

Selected Answer: A

The output of the code snippet is:

[1, 3, 5, 7, 9]

Explanation:

a[::2] uses Python slicing syntax:
The first : indicates the starting point (default is the beginning of the list).
The second : indicates the stopping point (default is the end of the list).
The 2 after the second colon is the step size, which means it skips every second element.
Starting from the first element (1), it selects every second element: 1, 3, 5, 7, 9.

upvoted 1 times

megan_mai 4 months, 1 week ago

Selected Answer: A

a = [1,2,3,4,5,6,7,8,9]
print(a[::2]) >> printing the whole list a by the step 2 >> a[0], a[2], a[4], a[6], a[8], remeber to start counting from 0

upvoted 3 times

christostz03 4 months, 2 weeks ago

a is the correct answer

upvoted 1 times

froster02 5 months, 2 weeks ago

[start : stop : step]
inclusive, exclusive respectively

upvoted 1 times

Question #10 *Topic 1*

What will be the output of the following code snippet?

```
d = {}
d[1] = 1
d['1'] = 2
d[1] += 1

sum = 0

for k in d:
    sum += d[k]

print(sum)
```

A. 3

B. 2

C. 4

D. 1

Correct Answer: *C*

Community vote distribution

C (100%)

hovnival 1 week, 5 days ago

Selected Answer: C

Here's the breakdown of the code and its execution:

Code Breakdown:

d = {}

Initializes an empty dictionary d.

d[1] = 1

Adds a key-value pair to the dictionary: key 1 (an integer) with value 1.

d['1'] = 2

Adds another key-value pair to the dictionary: key '1' (a string) with value 2.

Note: 1 (integer) and '1' (string) are considered different keys in Python.

d[1] += 1

Increments the value associated with the key 1 (integer) by 1.

The value of d[1] becomes 2.

Current State of d:

The dictionary now contains:

d = {1: 2, '1': 2}

sum = 0

Initializes a variable sum to 0.

for k in d: sum += d[k]

Iterates through the dictionary's keys (1 and '1') and adds their corresponding values to sum:

For k = 1, sum += d[1] → sum = 0 + 2 = 2
For k = '1', sum += d['1'] → sum = 2 + 2 = 4

Final Output:

The value of sum is 4.
Output:

4
upvoted 1 times

consultsk 3 months ago

Selected Answer: C

Let's analyze this code step by step:

d = {} creates an empty dictionary.
d[1] = 1 adds a key-value pair where the key is the integer 1 and the value is 1.
d['1'] = 2 adds another key-value pair where the key is the string '1' and the value is 2.
d[1] += 1 increments the value associated with the key 1. So now d[1] becomes 2.
sum = 0 initializes a variable sum to 0.
The for loop for k in d: iterates over the keys in the dictionary.
In each iteration, it adds the value associated with the current key to sum.

After the loop, sum will contain:

2 (from d[1])
2 (from d['1'])

Therefore, the final value of sum will be 4.

print(sum) will output this final value.

So, the output of this code will be: 4
upvoted 3 times

megan_mai 3 months ago

Selected Answer: C

d = {}
d[1] = 1 >> value of the key integer 1 is integer 1
d['1'] = 2 >> value of the key string '1' is 2
d[1] +=1 >> update the value of the key integer 1 to 1+1 =2
>>> final list is {1: 2, '1': 2}

sum = 0
for k in d:
sum += d[k] >> sum the values that associate to all the keys >> the answer is 2+2 = 4 (C)
print(sum)
print(d)
upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer
upvoted 1 times

froster02 5 months, 2 weeks ago

d = {} # Create an empty dictionary d
d[1] = 1 # Add a key-value pair where key is 1 and value is 1
d['1'] = 2 # Add a key-value pair where key is the string '1' and value is 2
d[1] += 1 # Increment the value associated with the key 1 by 1

```
{1: 2, '1': 2}
```

```
sum = 0 # Initialize sum to 0

for k in d: # Iterate over each key in the dictionary d
sum = sum + d[k] # Add the value associated with key k to sum
```

upvoted 1 times

Question #11 *Topic 1*

What is the output of the following snippet?

```
dictionary = {'one': 'two', 'three': 'one', 'two': 'three'}
v = dictionary['one']

for k in range(len(dictionary)):
    v = dictionary[v]

print(v)
```

A. two

B. one

C. (‘one’, ‘two’, ‘three’)

D. three

Correct Answer: *A*

Community vote distribution

A (100%)

hovnival 1 week, 5 days ago

Selected Answer: A

Step-by-step Execution:

Initialization:
dictionary is a dictionary: {'one': 'two', 'three': 'one', 'two': 'three'}.
v is initialized to dictionary['one'], which is 'two'.

For Loop Iterations:

First iteration (k = 0):
Current value of v is 'two'.
v = dictionary['two'] → 'two' maps to 'three'.
So, v = 'three'.

Second iteration (k = 1):
Current value of v is 'three'.
v = dictionary['three'] → 'three' maps to 'one'.
So, v = 'one'.

Third iteration (k = 2):
Current value of v is 'one'.
v = dictionary['one'] → 'one' maps to 'two'.
So, v = 'two'.

Final Value of v:
After 3 iterations, the value of v is 'two'.

Output:
The print(v) statement outputs 'two'.

upvoted 1 times

consultsk 3 months ago

Selected Answer: A

Let's analyze this code step by step:

A dictionary is created with key-value pairs:
'one': 'two'
'three': 'one'
'two': 'three'
v is initially set to dictionary['one'], which is 'two'.
The for loop runs for range(len(dictionary)), which is range(3) since the dictionary has 3 key-value pairs.
Inside the loop, v = dictionary[v] is executed 3 times:

First iteration: v = dictionary['two'] = 'three'
Second iteration: v = dictionary['three'] = 'one'
Third iteration: v = dictionary['one'] = 'two'

After the loop, the final value of v is 'two'.
print(v) outputs this final value.

Therefore, the output of this code will be:
two
upvoted 2 times

⊟ **megan_mai** 4 months, 1 week ago

Selected Answer: A

dictionary = {'one': 'two', 'three': 'one', 'two': 'three'}
v = dictionary['one'] >> v = 'two' (the value of key 'one')

for k in range(len(dictionary)): >> this loop will be run 3 times when k=0, k=1 and k=2
v = dictionary[v] >> loop 1 - k=0: v='two' >> dictionary(v) = 'three' >> new v = 'three'
>> loop 2 - k=1: v= 'three' >> dictionary(v) = 'one' >> new v = 'one'
>> loop 3 - k=2: v='one' >> dictionary(v) = 'two' >> new v = 'two'
print(v) >> printed value is the newest v = 'two' (A)
upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

a is the correct answer
upvoted 1 times

Question #12 *Topic 1*

What is the expected output of the following code?

```python
nums = [3, 4, 5, 20, 5, 25, 1, 3]
nums.pop(1)
print(nums)
```

A. [3, 1, 25, 5, 20, 5, 4]

B. [1, 3, 4, 5, 20, 5, 25]

C. [3, 5, 20, 5, 25, 1, 3]

D. [1, 3, 3, 4, 5, 5, 20, 25]

E. [3, 4, 5, 20, 5, 25, 1, 3]

Correct Answer: *C*

Community vote distribution

C (100%)

⊟ **hovnival** 1 week, 5 days ago

Selected Answer: C

Here's how the code works step by step:

The list nums is initialized as:

nums = [3, 4, 5, 20, 5, 25, 1, 3]

The method pop(1) is called. The pop() method removes the element at the specified index and returns it. Here, 1 is the index, so the element at index 1 (which is 4) is removed.

After nums.pop(1), the list becomes:

nums = [3, 5, 20, 5, 25, 1, 3]

The list is printed using print(nums).

Expected Output:

[3, 5, 20, 5, 25, 1, 3]

upvoted 1 times

⊟ **consultsk** 4 months, 2 weeks ago

Selected Answer: C

Let's break down this code:

nums = [3, 4, 5, 20, 5, 25, 1, 3] creates a list of integers.
nums.pop(1) removes the element at index 1 from the list and returns it. In this case, it removes the number 4.
The pop() method modifies the original list by removing the specified element.
print(nums) then prints the modified list.

After pop(1), the list nums becomes [3, 5, 20, 5, 25, 1, 3].

upvoted 3 times

⊟ **christostz03** 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #13 *Topic 1*

Which of the following sentences is true?

```
str1 = 'Peter'
str2 = str1[:]
```

A. str1 and str2 are different (but equal) strings.

B. str1 and str2 are different names of the same strings.

C. str1 is longer than str2

D. str2 is longer than str1

Correct Answer: *B*

Community vote distribution

A (67%) B (33%)

Billybob0604 2 weeks, 3 days ago

Selected Answer: B

The answer is B. For sure. Since strings are immutable and they have the same contents they both refer to the same memory location.

upvoted 1 times

Vano1 2 months ago

Selected Answer: A

Since there is a slicing the str2 made a copy of str1.

upvoted 1 times

consultsk 3 months ago

Selected Answer: A

let's analyze what's happening in the given code:

str1 = 'Peter' assigns the string 'Peter' to the variable str1.
str2 = str1[:] creates a slice of str1 that includes all characters (from beginning to end). This effectively creates a copy of the string.

In Python, strings are immutable, which means when you create a copy of a string using slicing, it creates a new string object with the same content.
Therefore, the correct answer is:
A. str1 and str2 are different (but equal) strings.
This is true because:

str1 and str2 contain the same characters ('Peter')
They are equal in terms of content
However, they are separate string objects in memory

Options B, C, and D are incorrect:

B is wrong because they are not the same string object, but two different objects with the same content.
C and D are incorrect because both strings have the same length.

upvoted 4 times

FedeXD 3 months, 1 week ago

A is correct. Slicing creates a copy of the string. Therefore, they are different strings, but if you compare them for equality you the result will be True as they have the same content.

upvoted 1 times

sandy589 3 months, 2 weeks ago

B is the correct answer, because their memory addresses are same if you check in your IDE.
hence B. str1 and str2 are different names of the same strings.

upvoted 1 times

christostz03 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

froster02 5 months, 2 weeks ago

```
str1 = 'Peter'
str2 = str1

print(id(str1), id(str2)) #same memory location :
123262391172144, 123262391172144

str2 = 'hell'
print(str1, str2)

print(id(str1), id(str2)) #on manipulation memory changes :
123262391172144, 123262391705072
```

hence, Str1 & Str2 are same Strings but different name.

upvoted 2 times

Vano1 2 months ago

Please pay attention to the slice [:] at the end of the second row. It copies the content of the first str to another memory address. Therefore, the right answer is A.

upvoted 1 times

nieups 5 months, 3 weeks ago

Selected Answer: B

B is the answer

upvoted 2 times

Sanela 7 months, 2 weeks ago

The correct answer is A, since str2 get all the elements from str1, but it's a new variable, new string. If you change str1, str2 remains unchanged

upvoted 2 times

Sanela 7 months, 2 weeks ago

Since there is no change made, they refer to the same string (have same id, I checked it). So I must correct my previous answer, it should be B

upvoted 1 times

Question #14 *Topic 1*

The fact that tuples belong to sequence types means:

A. they can be modified using the del instruction

B. they can be extended using the .append() method

C. they are actually lists

D. they can be indexed and sliced like lists

Correct Answer: *D*

hovnival 1 week, 5 days ago

Selected Answer: D

The correct answer is:

D. they can be indexed and sliced like lists

Tuples are immutable sequence types in Python, meaning they cannot be modified (no item assignment or deletion) and do not have methods like .append() for extension. However, like other sequence types such as strings and lists, tuples support indexing and slicing to access their elements.

upvoted 1 times

Sam_sgv 3 weeks, 5 days ago

Selected Answer: D

in Python, it is entirely possible to index and slice tuples

upvoted 1 times

Nfyughguygjhg 4 weeks ago

Selected Answer: C

c is the correct answer

upvoted 1 times

christostz03 4 months, 2 weeks ago

d is the correct answer

upvoted 3 times

Question #15 *Topic 1*

What is the output of the following code?

```python
my_list = [3, 1, -1]
my_list[-1] = my_list[-2]
print(my_list)
```

A. [1, 1, 1]

B. [3, -1, 1]

C. [3, 1, 1]

Correct Answer: *C*

Community vote distribution

C (100%)

hovnival 1 week, 5 days ago

Selected Answer: C

Let's analyze the code step by step:

Initialization of the list:
my_list = [3, 1, -1]
At this point, my_list is [3, 1, -1].

Assignment operation:
my_list[-1] = my_list[-2]
Here:
my_list[-1] refers to the last element of the list (-1).
my_list[-2] refers to the second-to-last element of the list (1).

The operation assigns the value of my_list[-2] (which is 1) to my_list[-1].
After this, the list becomes [3, 1, 1].

Print the list:
print(my_list)
The output is [3, 1, 1].

Final Output:

[3, 1, 1]

upvoted 1 times

consultsk 4 months, 1 week ago

Selected Answer: C

The code performs the following operations:

my_list = [3, 1, -1] initializes a list with the elements [3, 1, -1].
my_list[-1] = my_list[-2] assigns the value of the second last element in the list (my_list[-2]) to the last element (my_list[-1]). Since my_list[-2] is 1, the list becomes [3, 1, 1].
print(my_list) prints the updated list.
The output will be: [3,1,1]

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #16 *Topic 1*

What is the expected output of the following code?

```
data = ((1, 2),) * 7
print(len(data[3:8]))
```

A. The code is erroneous.

B. 6

C. 5

D. 4

Correct Answer: *D*

Community vote distribution

D (100%)

⊟ **hovnival** 1 week, 5 days ago

Selected Answer: D

Let's break down the code:

Creating the data variable:

data = ((1, 2),) * 7

The tuple (1, 2) is wrapped inside another tuple, ((1, 2),).
This tuple is repeated 7 times using the * operator, creating:

data = ((1, 2), (1, 2), (1, 2), (1, 2), (1, 2), (1, 2), (1, 2))

Slicing data:

data[3:8]

Slicing starts from index 3 (inclusive) and ends at index 8 (exclusive).
However, since data only has 7 elements (index 0 to 6), the slice data[3:8] only includes elements from index 3 to the end of the tuple:

((1, 2), (1, 2), (1, 2), (1, 2))

Calculating the length of the slice:

len(data[3:8])

The slice contains 4 elements, so len(data[3:8]) returns 4.

Final Output:

4

upvoted 1 times

⊟ **consultsk** 4 months, 1 week ago

Selected Answer: D

Let's break down the code step by step to understand the output:

data = ((1, 2),) * 7:

((1, 2),) creates a tuple containing a single tuple element (1, 2).

* 7 repeats this tuple 7 times.

So, data will be ((1, 2), (1, 2), (1, 2), (1, 2), (1, 2), (1, 2), (1, 2)).

print(len(data[3:8])):

data[3:8] slices the list starting from index 3 up to but not including index 8.

Since the tuple has 7 elements, the slice data[3:8] will include elements at indices 3, 4, 5, and 6, which gives you ((1, 2), (1, 2), (1, 2), (1, 2)).

The len() function then calculates the length of this slice, which is 4.

upvoted 4 times

⊟ **christostz03** 4 months, 2 weeks ago

d is the correct answer

upvoted 2 times

Question #17 *Topic 1*

What is the expected output of the following code?

```
data = {'Peter': 30, 'Paul': 31}
print(list(data.keys()))
```

A. ('Peter': 30, 'Paul': 31)

B. ('Peter', 'Paul')

C. ['Peter': 30, 'Paul': 31]

D. ['Peter', 'Paul']

Correct Answer: *D*

Community vote distribution

D (100%)

hovnival 1 week, 5 days ago

Selected Answer: D

The expected output of the code is:

['Peter', 'Paul']

Explanation:

data.keys() returns a view object that displays a list of all the keys in the dictionary data.
list(data.keys()) converts this view object into a list, which contains the keys of the dictionary.
The print statement outputs this list.

upvoted 1 times

consultsk 4 months, 1 week ago

Selected Answer: D

The output of this code will be:
['Peter', 'Paul']
This is because:

data is a dictionary with two key-value pairs: 'Peter': 30 and 'Paul': 31.
The keys() method returns a view object of all the keys in the dictionary.
The list() function converts this view object into a list.
Finally, the print() function outputs this list of keys.

The order of the keys in the output may vary, as dictionaries in Python 3.7+ maintain insertion order, but in earlier versions, the order was not guaranteed. However, based on the code shown, this order is likely to be preserved.

upvoted 3 times

christostz03 4 months, 2 weeks ago

d is the correct answer

upvoted 1 times

Question #18 *Topic 1*

What is the output of the following snippet?

```
tup = (1, ) + (1, )
tup = tup + tup
print(len(tup))
```

A. 2

B. 4

C. The snippet is erroneous (invalid syntax)

Correct Answer: *B*

⊟ **hovnival** 1 week, 5 days ago

Selected Answer: B

Let's break this down step by step:
Initial Code:

tup = (1,) + (1,)

(1,) is a tuple with a single element, 1.
(1,) + (1,) concatenates two tuples, resulting in (1, 1).

So now:

tup = (1, 1)

tup = tup + tup

tup is (1, 1).
Concatenating tup + tup results in (1, 1, 1, 1).

So now:

tup = (1, 1, 1, 1)

print(len(tup))

len(tup) calculates the number of elements in the tuple, which is 4.

Final Output:

4
upvoted 1 times

⊟ **vestersly** 4 months, 2 weeks ago

Explanation:
Tuple Creation and Concatenation:

python

tup = (1,) + (1,)
(1,) is a tuple with one element, 1.
The expression (1,) + (1,) concatenates these two tuples, resulting in a new tuple:
python
Copy code

```
tup = (1, 1)
```

Doubling the Tuple:

python

Copy code

```
tup = tup + tup
```

The tuple tup is concatenated with itself.

After the concatenation, tup becomes:

python

Copy code

```
tup = (1, 1) + (1, 1) = (1, 1, 1, 1)
```

Length Calculation:

python

Copy code

```
print(len(tup))
```

The len() function returns the number of elements in the tuple tup.

Since tup is now (1, 1, 1, 1), its length is 4.

Conclusion:

The correct answer is:

B. 4

upvoted 3 times

christostz03 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #19 *Topic 1*

What is the expected output of the following code?

```
data = (1, 2, 4, 8)
data = data[-2:-1]
data = data[-1]
print(data)
```

A. (4)

B. 4

C. (4,)

D. 44

Correct Answer: *B*

hovnival 1 week, 5 days ago

Selected Answer: B

Let's break down the code step by step:

Initial data assignment:

data = (1, 2, 4, 8)

data is a tuple containing (1, 2, 4, 8).

Slicing the tuple:

data = data[-2:-1]

Here, data[-2:-1] means:
Start slicing at index -2 (the second-to-last element, which is 4).
Stop slicing before index -1 (the last element, which is 8).
This results in a tuple with a single element: (4,).

Accessing the last element of the new tuple:

data = data[-1]

Now data is (4,).
data[-1] accesses the last (and only) element of the tuple, which is 4.
As a result, data becomes the integer 4.

Printing the result:

print(data)

The output will be:

4
upvoted 1 times

tbelap16 1 month, 1 week ago

Selected Answer: B

It is b because it makes the most sense
upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #20 *Topic 1*

What is the output of the following snippet?

```python
my_list = [1, 2]

for v in range(2):
    my_list.insert(-1, my_list[v])

print(my_list)
```

A. [1, 1, 2, 2]

B. [1, 1, 1, 2]

C. [1, 2, 1, 2]

D. [1, 2, 2, 2]

Correct Answer: *B*

Community vote distribution

B (100%)

hovnival 1 week, 5 days ago

Selected Answer: B

Let's analyze the given code step by step:
Initial list:

my_list = [1, 2]

Loop:

for v in range(2):
my_list.insert(-1, my_list[v])

The loop runs two times (v = 0 and v = 1) because range(2) generates values 0 and 1.
Inside the loop, the insert(-1, my_list[v]) method inserts my_list[v] into the second-to-last position (-1 index).

Iteration Breakdown:

Iteration 1 (v = 0):
my_list[v] = my_list[0] = 1
insert(-1, 1) inserts the value 1 just before the last element. The list becomes:

[1, 1, 2]

Iteration 2 (v = 1):

Now my_list = [1, 1, 2], so my_list[v] = my_list[1] = 1.
insert(-1, 1) again inserts the value 1 before the last element. The list becomes:

[1, 1, 1, 2]

Final list:

After both iterations, the list my_list becomes [1, 1, 1, 2].
Output:

Thus, the output of print(my_list) is:

[1, 1, 1, 2]

upvoted 1 times

Sam_sgv 3 weeks, 3 days ago

Selected Answer: B

In Python, when using the insert() method with lists, the index -1 behaves differently from how negative indexing works for accessing elements.
Negative index: With insert(-1, value), Python interprets -1 as "insert before the element at index -1". Since -1 refers to the last element for access, inserting before this position means the new element goes before the current last element, not at the very end.

This might seem counterintuitive at first, but it's consistent with how Python treats indexing for modification versus access:

Access: -1 gets you the last element.
Insertion: -1 means insert just before where you would access with -1.

upvoted 1 times

burritochewer 1 month, 1 week ago

why is insert(-1, ...) mean insert at the second to last position?

wouldn't insert(0, ...) insert at the first place as it's meant for 0 of index place? therefore, -1 is the last digit... is it not?

upvoted 1 times

consultsk 3 months ago

Selected Answer: B

Let's step through this code to understand what's happening:

We start with my_list = [1, 2]
The loop for v in range(2): will run twice, with v taking values 0 and 1.
In each iteration, we're using my_list.insert(-1, my_list[v]):

insert(-1, ...) means insert at the second-to-last position
my_list[v] is the element we're inserting

Let's go through the iterations:
First iteration (v = 0):

We insert my_list[0] (which is 1) at index -1
my_list becomes [1, 1, 2]

Second iteration (v = 1):

We insert my_list[1] (which is now 1) at index -1
my_list becomes [1, 1, 1, 2]

After the loop, we print my_list

Therefore, the output will be:
[1,1,1,2]

upvoted 3 times

megan_mai 4 months, 1 week ago

Selected Answer: B

my_list = [1,2]
for v in range (2):
my_list.insert(-1,my_list[v]) >>> loop 1: v=0 >> insert my_list[0] in the position my_list[-1] >> my_list = [1,1,2]
>>> loop 2: v=1 >> insert my_list[1] in the position my_list[-1] >> my_list = [1,1,1,2]

print(my_list) >>>> [1,1,1,2] (B)

upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #21 *Topic 1*

What is the expected output of the following code?

```
data = [1, 2, 3, None, (), [], ]
print(len(data))
```

A. 4

B. 6

C. 5

D. 3

Correct Answer: *B*

hovnival 1 week, 5 days ago

Selected Answer: B

The code defines a list data containing the following elements:

data = [1, 2, 3, None, (), [],]

The elements of the list are 1, 2, 3, None, an empty tuple (), and an empty list [].
The len() function returns the number of elements in the list, regardless of the types or contents of those elements.

In this case, the list data contains 6 elements: 1, 2, 3, None, (), and [].

Thus, the output of the code will be:

6

upvoted 1 times

christostz03 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #22 *Topic 1*

A data structure described as LIFO is actually a:

A. stack

B. tree

C. list

D. heap

Correct Answer: *A*

hovnival 1 week, 5 days ago

Selected Answer: A

The correct answer is:

A. stack

LIFO stands for "Last In, First Out," which is a characteristic of a stack data structure. In a stack, the last element added is the first one to be removed.

upvoted 1 times

christostz03 4 months, 2 weeks ago

a is the correct answer

upvoted 1 times

Question #23

Topic 1

How would you remove all the items from the d dictionary?

Expected output:

```
{}
```

Code:

```
d = {'A' : 1, 'B' : 2, 'C' : 3}
```

A. d.del()

B. d.remove()

C. del d

D. d.clear()

Correct Answer: *D*

⊟ **hovnival** 1 week, 5 days ago

Selected Answer: D

The correct option to remove all items from the dictionary d and result in an empty dictionary {} is:

D. d.clear()

Explanation:

d.clear() removes all items from the dictionary without deleting the dictionary itself.

Other options:

d.del() and d.remove() are invalid as these methods do not exist for dictionaries.

del d deletes the dictionary entirely, resulting in d no longer being defined.

Example:

d = {'A': 1, 'B': 2, 'C': 3}

d.clear()

print(d) # Output: {}

upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

d is the correct answer

upvoted 1 times

Question #24 *Topic 1*

What is the expected output of the following code?

```
data = {'one': 'two', 'two': 'three', 'three': 'one'}
res = data['three']

for _ in range(len(data)):
    res = data[res]

print(res)
```

A. three

B. ('one', 'two', 'three')

C. two

D. one

Correct Answer: *D*

hovnival 1 week, 5 days ago

Selected Answer: D

Let's analyze the code step by step to determine the expected output.
Code breakdown:

Initialization:

data = {'one': 'two', 'two': 'three', 'three': 'one'}
res = data['three'] # res = 'one'

Loop execution:

for _ in range(len(data)): # len(data) = 3
res = data[res]

The loop runs 3 times (since len(data) = 3).
Iterations:
First iteration: res = data[res] = data['one'] = 'two'
Second iteration: res = data[res] = data['two'] = 'three'
Third iteration: res = data[res] = data['three'] = 'one'

Final value of res: After 3 iterations, res is 'one'.

Output:

print(res) # Outputs 'one'

Expected Output:

one
upvoted 1 times

christostz03 4 months, 2 weeks ago

d is correct answer
upvoted 1 times

Question #25 *Topic 1*

What is the expected output of the following code?

```python
data = {'name': 'Peter', 'age': 30}
person = data.copy()
print(id(data) == id(person))
```

A. False

B. 1

C. 0

D. True

Correct Answer: *A*

Community vote distribution

A (100%)

hovnival 1 week, 4 days ago

Selected Answer: A

The expected output of the code is:

False

Explanation:

data.copy() creates a shallow copy of the dictionary data. The new dictionary (person) contains the same key-value pairs but resides at a different memory location.
id(data) retrieves the memory address of the data dictionary.
id(person) retrieves the memory address of the person dictionary.
Since the two dictionaries are separate objects, id(data) is not equal to id(person), and the comparison id(data) == id(person) evaluates to False.

upvoted 1 times

Ragoo 2 months, 2 weeks ago

Create a dictionary: data = {'name': 'Peter', 'age': 30} creates a dictionary with two key-value pairs.

Make a copy: person = data.copy() creates a new dictionary person that is a copy of data. This means person has its own memory address, separate from data.

Compare IDs: id(data) == id(person) compares the memory addresses of data and person. Since they are different objects, their IDs will not be equal, resulting in False.

Selected Answer: A

upvoted 1 times

megan_mai 4 months, 1 week ago

Selected Answer: A

This method doesn't modify the original, dictionary just returns copy of the dictionary.
if we do person = data >>> result would be True

upvoted 1 times

christostz03 4 months, 2 weeks ago

a is correct answer / copy and [:] are swallow copies so id changes

upvoted 1 times

Question #26 *Topic 1*

Which one of the lines should you put in the snippet below to match the expected output?

Expected output:

```
[4, 1, 7, 2, 'A']
```

Code:

```
list = ['A', 2, 7, 1, 4]

# enter code here

print(list)
```

A. reverse(list)

B. list.reversed()

C. list.reverse()

D. reversed(list)

Correct Answer: *C*

hovnival 1 week, 4 days ago

Selected Answer: C

The correct answer is:

C. list.reverse()

Explanation:

list.reverse(): This is the correct method to reverse a list in place (modifies the original list). It doesn't return a new list but instead reverses the list itself. The expected output [4, 1, 7, 2, 'A'] matches this behavior.

Why not the other options?

A. reverse(list): This is invalid syntax; there's no standalone function called reverse.

B. list.reversed(): This is incorrect as list objects do not have a method named reversed().

D. reversed(list): This returns an iterator, not a list. To achieve the desired effect, you'd need to convert the iterator back to a list using list(reversed(list)).

Correct Code:

list = ['A', 2, 7, 1, 4]

list.reverse()

print(list)

upvoted 1 times

Nfyughguygjhg 4 weeks ago

Selected Answer: A

c is correct answer

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is correct answer

upvoted 2 times

Question #27 *Topic 1*

What is the expected output of the following code?

```
data1 = '1', '2'
data2 = ('3', '4')
print(data1 + data2)
```

A. ['1', '2', '3', '4']

B. (1, 2, 3, 4)

C. ('1', '2', '3', '4')

D. The code is erroneous.

Correct Answer: *C*

hovnival 1 week, 4 days ago

Selected Answer: C

The expected output of the given code is:
C. ('1', '2', '3', '4')
Explanation:

data1 = '1', '2': This is a tuple assignment. In Python, writing values separated by commas without parentheses creates a tuple. So, data1 becomes ('1', '2').

data2 = ('3', '4'): This explicitly creates a tuple with parentheses. So, data2 becomes ('3', '4').

data1 + data2: The + operator concatenates two tuples. So, ('1', '2') + ('3', '4') results in ('1', '2', '3', '4').

print(data1 + data2): This prints the concatenated tuple ('1', '2', '3', '4').

Hence, the correct answer is C.

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is correct answer

upvoted 1 times

Question #28 *Topic 1*

What is the expected output of the following code?

```
data = (1, 2, 4, 8)
data = data[1:-1]
data = data[0]
print(data)
```

A. (2)

B. (2,)

C. 2

D. The code is erroneous.

Correct Answer: *C*

hovnival 1 week, 4 days ago

Selected Answer: C

Let's break down the code step by step:

data = (1, 2, 4, 8)
Here, data is a tuple: (1, 2, 4, 8).

data = data[1:-1]
This slices the tuple starting from index 1 and ending at index -1. So, it extracts the elements between index 1 and index -1, excluding the element at index -1. The result is: (2, 4).

data = data[0]
Now, data is a tuple (2, 4). The expression data[0] takes the first element of the tuple, which is 2.

print(data)
Finally, the code prints the value of data, which is now 2.

Therefore, the expected output is:

C. 2

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is correct answer

upvoted 1 times

Question #29

Topic 1

What is the output of the following snippet?

```
my_list = [3, 1, -2]
print(my_list[my_list[-1]])
```

A. -2

B. 3

C. -1

D. 1

Correct Answer: *D*

hovnival 1 week, 4 days ago

Selected Answer: D

Let's break down the code to determine the output:

my_list = [3, 1, -2]
print(my_list[my_list[-1]])

my_list[-1] retrieves the last element of the list, which is -2.
Then, my_list[my_list[-1]] becomes my_list[-2], which is the second-to-last element of the list.
my_list[-2] is 1 (the second-to-last element in the list [3, 1, -2]).

Therefore, the output of the snippet is:

D. 1

upvoted 1 times

christostz03 4 months, 2 weeks ago

d is correct answer

upvoted 1 times

Question #30 *Topic 1*

An alternative name for a data structure called a stack is:

A. LIFO

B. FIFO

C. FOLO

Correct Answer: *A*

hovnival 1 week, 4 days ago

Selected Answer: A

The correct answer is:

A. LIFO

LIFO stands for "Last In, First Out," which describes the behavior of a stack. The last element added to the stack is the first one to be removed.

upvoted 1 times

christostz03 4 months, 2 weeks ago

a is correct answer

upvoted 1 times

> **Nfyughguygjhg** 4 weeks ago
>
> why? is a the correct answer?
>
> upvoted 1 times

Question #31 *Topic 1*

What is the expected output of the following code?

```
w = [7, 3, 23, 42]
x = w[1:]
y = w[1:]
z = w
y[0] = 10
z[1] = 20
print(w)
```

A. [7, 3, 23, 42]

B. [7, 20, 23, 42]

C. [10, 20, 42]

D. [10, 20, 23, 42]

Correct Answer: *B*

Community vote distribution

B (100%)

⊟ **hovnival** 1 week, 4 days ago

Selected Answer: B

Let's analyze the code step by step:

w = [7, 3, 23, 42]
x = w[1:]
y = w[1:]
z = w
y[0] = 10
z[1] = 20
print(w)

w = [7, 3, 23, 42]: A list w is created with the values [7, 3, 23, 42].

x = w[1:]: This creates a new list x that is a slice of w, starting from index 1, i.e., x = [3, 23, 42]. Note that this is a shallow copy, meaning changes to x do not affect w.

y = w[1:]: This creates another new list y which is also a slice of w, similar to x. So y = [3, 23, 42]. Like x, this is also a shallow copy.

z = w: Here, z is assigned the reference to w, meaning any changes made to z will directly modify w.

y[0] = 10: This modifies the first element of y, so y becomes [10, 23, 42]. However, since y is a separate copy, w remains unaffected by this change.

z[1] = 20: This modifies the second element of z (and w, since z refers to w), so w becomes [7, 20, 23, 42].

Finally, when print(w) is called, it outputs:

[7, 20, 23, 42]

So the correct answer is B. [7, 20, 23, 42].

upvoted 1 times

⊟ **scriptnone** 2 months, 2 weeks ago

Selected Answer: B

Answer is b:

```
w = [7, 3, 23, 42]
x = w[1:]
y = w[1:]
z = w
y[0] = 10
z[1] = 20
print(w) # [7, 20, 23, 42]
```

upvoted 2 times

christostz03 4 months, 2 weeks ago

b is correct answer

upvoted 1 times

Question #32 *Topic 1*

Take a look at the snippet and choose one of the following statements which is true:

```
nums = []
vals = nums
vals.append(1)
```

A. vals is longer than nums

B. nums and vals are of the same length

C. nums is longer than vals

Correct Answer: *B*

⊟ **hovnival** 1 week, 4 days ago

Selected Answer: B

The correct answer is:

B. nums and vals are of the same length

Explanation:

In Python, lists are mutable objects. The assignment vals = nums does not create a new list; it simply creates a reference to the same list object.

Therefore, both nums and vals point to the same underlying list.

When vals.append(1) is executed, it modifies the shared list.

As a result, both nums and vals will have the same length (and the same content) because they are essentially two names for the same object.

upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

b is correct answer

upvoted 1 times

⊟ **Nfyughguygjhg** 4 weeks ago

how do you know?

upvoted 1 times

Question #33 *Topic 1*

What is the expected output of the following code?

```python
data = {'a': 1, 'b': 2, 'c': 3}
print(data['a', 'b'])
```

A. (1, 2)

B. The code is erroneous.

C. {'a':1, 'b':2}

D. [1,2]

Correct Answer: *B*

hovnival 1 week, 4 days ago

Selected Answer: B

The correct answer is:

B. The code is erroneous.
Explanation:

In the given code:

data = {'a': 1, 'b': 2, 'c': 3}
print(data['a', 'b'])

data is a dictionary with keys 'a', 'b', and 'c'.
When trying to access dictionary values, the key provided must be a single hashable object. In this case, ['a', 'b'] is a tuple (('a', 'b')), not a valid key in the dictionary.
Since the dictionary data does not contain the key ('a', 'b'), Python raises a KeyError.

Error:

KeyError: ('a', 'b')

upvoted 1 times

FedeXD 3 months, 3 weeks ago

You can't index with multiple keys like this. You get a KeyError.

upvoted 1 times

Question #34 *Topic 1*

How many elements does the L list contain?

```
L = [i for i in range(-1, -2)]
```

A. one

B. two

C. three

D. zero

Correct Answer: *D*

⊟ **hovnival** 1 week, 4 days ago

Selected Answer: D

The given list comprehension is:

L = [i for i in range(-1, -2)]

Explanation:

The range(start, stop) function generates numbers starting from start and stops before stop.
In this case, range(-1, -2) starts at -1 but stops before -2, and since -1 > -2, the range produces no numbers.
As a result, the list comprehension produces an empty list.

When you print L, it will output [].
Correct Answer:

D. zero

upvoted 1 times

⊟ **VinzBetel7** 2 weeks, 5 days ago

Selected Answer: D

-1 is larger than -2

upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

d is the correct answer

upvoted 1 times

Question #35 *Topic 1*

What is the output of the following snippet?

```python
my_list = [0, 1, 2, 3]
x = 1
for elem in my_list:
    x *= elem
print(x)
```

A. 1

B. 0

C. 6

Correct Answer: *B*

Community vote distribution

B (100%)

hovnival 1 week, 4 days ago

Selected Answer: B

The given Python snippet multiplies the value of x by each element in my_list. Let's break down the execution step by step:

my_list = [0, 1, 2, 3]
This initializes the list my_list with the elements [0, 1, 2, 3].

x = 1
This initializes x with the value 1.

for elem in my_list:
This starts a loop where elem will take the value of each element in my_list.

x *= elem
Inside the loop, x is multiplied by the current element (elem).

Here's the loop breakdown:

In the first iteration, elem = 0. So, x = 1 * 0 = 0.
In the second iteration, elem = 1. So, x = 0 * 1 = 0.
In the third iteration, elem = 2. So, x = 0 * 2 = 0.
In the fourth iteration, elem = 3. So, x = 0 * 3 = 0.

After the loop finishes, x remains 0.

print(x)
This will print 0.

Output:

0

upvoted 1 times

[Removed] 4 months, 2 weeks ago

Selected Answer: B

B is the correct answer!

upvoted 1 times

christostz03 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #36 *Topic 1*

What is the expected output of the following code?

```
nums = [1, 2, 3]
data = ('Peter',) * (len(nums) - nums[::-1][0])
print(data)
```

A. ('Peter', 'Peter',)

B. PeterPeter

C. The code is erroneous.

D. ('Peter')

E. ()

Correct Answer: *E*

Community vote distribution

E (100%)

hovnival 1 week, 4 days ago

Selected Answer: E

Let's break down the code step by step:

nums = [1, 2, 3]
This creates a list nums with the elements 1, 2, and 3.

nums[::-1]
This reverses the list nums. So, nums[::-1] becomes [3, 2, 1].

nums[::-1][0]
This accesses the first element of the reversed list, which is 3.

len(nums) - nums[::-1][0]
The length of nums is 3 (since there are three elements). Subtracting nums[::-1][0] (which is 3), we get:

len(nums) - nums[::-1][0] = 3 - 3 = 0

('Peter',) * 0
This creates a tuple ('Peter',) and multiplies it by 0. In Python, multiplying a tuple by 0 results in an empty tuple:

('Peter',) * 0 = ()

print(data)
Finally, this prints the value of data, which is an empty tuple.

Expected Output:

()
upvoted 1 times

LunaMeadows1 4 months ago

Selected Answer: E

E is the correct answer.
Breakdown of data = ('Peter',) * (len(nums) - nums[::-1][0])
len(nums) = 3
nums[::-1] reverses the list so nums would become [3,2,1]
nums[0] would then return 3 as the first element

When multiplying a tuple, you create that many versions of the current entry. 3-3 becomes 0 and multiplying anything by 0 becomes 0 so there would be 0 entries in the tuple returning ()

upvoted 3 times

⊟ **christostz03** 4 months, 2 weeks ago

e is the correct answer

upvoted 2 times

Question #37 *Topic 1*

What is the expected output of the following code?

```
t1 = (1, 4, 9)
t2 = ('A', 'D', 'Z')
t3 = (True, False, None)
t4 = (5.0, 7.5, 9.9)

t1, t3 = t2, t4
print(t1)
```

A. The program will cause an error.

B. (1, 4, 9)

C. ('A', 'D', 'Z')

D. (5.0, 7.5, 9.9)

Correct Answer: *C*

⊟ **hovnival** 1 week, 4 days ago

Selected Answer: C

Let's analyze the given Python code step-by-step:

Four tuples are defined:
t1 = (1, 4, 9)
t2 = ('A', 'D', 'Z')
t3 = (True, False, None)
t4 = (5.0, 7.5, 9.9)

The next line performs tuple unpacking:

t1, t3 = t2, t4

Here, t1 is assigned the value of t2 and t3 is assigned the value of t4. After this line:

t1 now refers to ('A', 'D', 'Z')
t3 now refers to (5.0, 7.5, 9.9)

Finally, the code prints t1:

print(t1)

Since t1 has been reassigned to the value of t2, the output will be:

('A', 'D', 'Z')

upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #38 *Topic 1*

What is the expected output of the following code?

```
data = ()
print(data.__len__())
```

A. The code is erroneous.

B. 1

C. 0

D. None

Correct Answer: *C*

Community vote distribution

A (100%)

hovnival 1 week, 4 days ago

Selected Answer: C

The given Python code snippet defines an empty tuple and prints its length using the __len__() method.
Code:

data = ()
print(data.__len__())

Explanation:

data = () initializes an empty tuple.
data.__len__() is a method that returns the number of items in the tuple.
Since the tuple is empty, the length is 0.

Expected Output:

0

upvoted 1 times

VinzBetel7 2 weeks, 3 days ago

Selected Answer: A

the code is erroneous. Correct answer is A

upvoted 1 times

Nfyughguygjhg 4 weeks ago

Selected Answer: A

it hass to be A!

upvoted 1 times

raYda 1 month ago

Selected Answer: C

C is the correct answer

upvoted 1 times

Ameer 1 month ago

Selected Answer: C

C is the correct answer

upvoted 1 times

Dont_Stop_Learning 2 months, 1 week ago

Selected Answer: A

The correct answer is A

upvoted 2 times

⊟ **christostz03** 4 months, 2 weeks ago

AttributeError: 'tuple' object has no attribute '_len_'

upvoted 1 times

⊟ **Vano1** 2 months ago

Please try with two underscores instead of one. .__len__ exist.

upvoted 1 times

Question #39 *Topic 1*

What is the expected output of the following code?

```python
fruits1 = ['Apple', 'Pear', 'Banana']
fruits2 = fruits1
fruits3 = fruits1[:]

fruits2[0] = 'Cherry'
fruits3[1] = 'Orange'

res = 0

for i in (fruits1, fruits2, fruits3):
    if i[0] == 'Cherry':
        res += 1
    if i[1] == 'Orange':
        res += 10

print(res)
```

A. 22

B. 12

C. 0

D. 11

Correct Answer: *B*

hovnival 1 week, 4 days ago

Selected Answer: B

Result Calculation: The for loop iterates over (fruits1, fruits2, fruits3) and checks conditions:
If i[0] == 'Cherry', add 1 to res.
If i[1] == 'Orange', add 10 to res.

Details:

For fruits1 = ['Cherry', 'Pear', 'Banana']:
i[0] == 'Cherry': True → res += 1 → res = 1.
i[1] == 'Orange': False → no change.

For fruits2 = ['Cherry', 'Pear', 'Banana']:
i[0] == 'Cherry': True → res += 1 → res = 2.
i[1] == 'Orange': False → no change.

For fruits3 = ['Apple', 'Orange', 'Banana']:
i[0] == 'Cherry': False → no change.
i[1] == 'Orange': True → res += 10 → res = 12.

Output: The final value of res is 12.

Final Output

12

upvoted 1 times

christostz03 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #40 *Topic 1*

What is the expected output of the following code?

```
data = (1,) * 3
data[0] = 2
print(data)
```

A. (2, 1, 1)

B. (1, 1, 1)

C. (2, 2, 2)

D. The code is erroneous.

Correct Answer: *D*

hovnival 1 week, 4 days ago

Selected Answer: D

The code will raise a TypeError.

Here's the explanation:

The tuple data = (1,) * 3 creates a tuple containing three elements: (1, 1, 1).
Tuples in Python are immutable, meaning their elements cannot be modified after creation.
The statement data[0] = 2 attempts to modify the first element of the tuple, which is not allowed.

Because of the immutability of tuples, attempting to modify an element will result in a TypeError.

Thus, the expected output is an error message similar to:

TypeError: 'tuple' object does not support item assignment

upvoted 1 times

FedeXD 3 months, 3 weeks ago

Tuples are immutable, therefore you can't modify its elements.

upvoted 4 times

imran04120 4 months, 1 week ago

TypeError: 'tuple' object does not support item assignment

If it was a list then it would be [2,1,1]

upvoted 2 times

Question #41 *Topic 1*

Which one of the lines should you put in the snippet below to match the expected output?

Expected output:

```
[1, 2, 4, 7]
```

Code:

```
list = [2, 7, 1, 4]

# enter code here

print(list)
```

A. sorted(list)

B. sort(list)

C. list.sort()

D. list.sorted()

Correct Answer: *C*

Community vote distribution

C (50%) A (50%)

yagebin 4 days, 21 hours ago

Selected Answer: C

The correct one is C, only sort method can save the changement

upvoted 1 times

hovnival 1 week, 2 days ago

Selected Answer: C

To achieve the expected output [1, 2, 4, 7] from the list [2, 7, 1, 4], you should use the list.sort() method, which sorts the list in place.

Here is the complete code snippet:

Copy

list = [2, 7, 1, 4]
list.sort() # This sorts the list in place
print(list)

So, the correct line to use is:

C. list.sort()

upvoted 1 times

Nfyughguygjhg 4 weeks ago

Selected Answer: A

A is correct

upvoted 1 times

rcptryk 1 month, 2 weeks ago

Selected Answer: C

C is correct because the output expectation must come from print(list) not from the print(sorted(list)). you have to write comment before the print which is list.sort() or you have to write list=sorted(list) which is not option

upvoted 2 times

mldprasad 1 month, 2 weeks ago

Selected Answer: A

list = [2, 7, 1, 4]

print(sorted(list))

So A is correct answer. C output is None

upvoted 2 times

Vano1 2 months ago

Selected Answer: C

The correct answer is C. Checked in Python

upvoted 1 times

scriptnone 2 months, 2 weeks ago

Selected Answer: A

list = [2, 7, 1, 4]

print(sorted(list))

So A is correct answer. C output is None

upvoted 2 times

SujPractise 3 months, 1 week ago

C will return the sorted list

[1, 2, 4, 7]

upvoted 2 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 2 times

Biju1 4 months, 2 weeks ago

list.sort() will show none

upvoted 2 times

Biju1 4 months, 2 weeks ago

A is correct

upvoted 2 times

Question #42 *Topic 1*

Which of the following sentences are true about the code? (Choose two.)

```
nums = [1, 2, 3]
vals = nums
```

A. nums and vals are different lists

B. vals is longer than nums

C. nums and vals are different names of the same list

D. nums and vals have the same length

Correct Answer: *CD*

Community vote distribution

CD (100%)

hovnival 1 week, 2 days ago

Selected Answer: CD

The two true sentences about the code are:

C. nums and vals are different names of the same list
D. nums and vals have the same length

Explanation:

When you assign vals = nums, vals becomes a reference to the same list that nums refers to. Therefore, any changes made to the list through either nums or vals will affect the same underlying list.
Since vals and nums are references to the same list, they both have the same length.

upvoted 1 times

scriptnone 2 months, 2 weeks ago

Selected Answer: CD

C and D.

upvoted 2 times

christostz03 4 months, 2 weeks ago

correct answers are c & d

upvoted 1 times

Question #43 *Topic 1*

The second assignment:

```
vals = [0, 1, 2]
vals[0], vals[1] = vals[1], vals[2]
```

A. extends the list

B. doesn't change the list's length

C. shortens the list

Correct Answer: *B*

hovnival 1 week, 2 days ago

Selected Answer: B

The correct answer is:

B. doesn't change the list's length

Explanation:
The code vals[0], vals[1] = vals[1], vals[2] swaps the values at indices 0 and 1 with the values at indices 1 and 2, respectively. This operation modifies the elements within the list but does not change the length of the list. The list still has the same number of elements after the assignment.

upvoted 1 times

christostz03 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #44 *Topic 1*

What is the expected output of the following code?

```
data = ['abc', 'def', 'abcde', 'efg']
print(max(data))
```

A. efg

B. abc

C. def

D. The code is erroneous.

E. abcde

F. None of the above.

Correct Answer: *A*

hovnival 1 week, 2 days ago

Selected Answer: A

The expected output of the code is:

A. efg

Explanation:
The max() function returns the largest item in an iterable. When applied to a list of strings, it compares the strings lexicographically (i.e., based on the ASCII values of the characters). In this case, the comparison is done character by character from left to right.

Here's how the comparison works for the given list:

'abc' vs 'def' -> 'def' is larger
'def' vs 'abcde' -> 'def' is larger
'def' vs 'efg' -> 'efg' is larger
Therefore, the largest string in the list according to lexicographical order is 'efg'.

The expected output of the code is: A. efg

This is because the max function, when applied to a list of strings, returns the string that would appear last if the list were sorted lexicographically.

upvoted 1 times

VinzBetel7 2 weeks, 2 days ago

Selected Answer: A

A is correct

upvoted 1 times

christostz03 4 months, 2 weeks ago

a is the correct answer. In terms of chars efg are greater than all other options.

upvoted 1 times

Question #45 *Topic 1*

Which function does in-place reversal of objects in a list?

A. list.sort([func])

B. list.pop(obj=list[-1])

C. list.remove(obj)

D. list.reverse()

Correct Answer: *D*

hovnival 1 week, 2 days ago

Selected Answer: D

The correct answer is:

D. list.reverse()

Explanation:
The list.reverse() method reverses the elements of the list in place, meaning it modifies the original list directly without creating a new list. This is the function that performs an in-place reversal of objects in a list.

upvoted 1 times

christostz03 4 months, 2 weeks ago

d is the correct answer

upvoted 1 times

Question #46 *Topic 1*

What is the output of the following snippet?

```
my_list_1 = [1, 2, 3]
my_list_2 = []
for v in my_list_1:
    my_list_2.insert(0, v)
print(my_list_2)
```

A. [1, 2, 3]

B. [3, 3, 3]

C. [3, 2, 1]

D. [1, 1, 1]

Correct Answer: *C*

⊟ **hovnival** 1 week, 2 days ago

Selected Answer: C

The correct answer is:

C. [3, 2, 1]

Explanation:
The code iterates over each element v in my_list_1 and inserts v at the beginning (index 0) of my_list2. This results in reversing the order of the elements from my_list_1 in my_list2. Here is the step-by-step process:

v = 1 -> my_list2.insert(0, 1) -> my_list2 becomes [1]
v = 2 -> my_list2.insert(0, 2) -> my_list2 becomes [2, 1]
v = 3 -> my_list2.insert(0, 3) -> my_list2 becomes [3, 2, 1]
Therefore, the final output of print(my_list2) is [3, 2, 1].

upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #47 *Topic 1*

What snippet would you insert in the line indicated below to print The highest number is 10 and the lowest number is 1. to the monitor?

```
data = [10, 2, 1, 7, 5, 6, 4, 3, 9, 8]
# insert your code here
print(
    ('The highest number is {} ' +
     'and the lowest number is {}.').format(high, low)
)
```

A. None of the above.

B.

```
def find_high_low(nums):
    nums.sort()
    return nums[0], nums[-1]


high, low = find_high_low(data)
```

C.

```
def find_high_low(nums):
    nums.sort()
    return nums[len(nums)], nums[0]


high, low = find_high_low(data)
```

D.

```
def find_high_low(nums):
    nums.sort()
    return nums[-1], nums[0]


high, low = find_high_low(data)
```

Correct Answer: *D*

hovnival 1 week, 2 days ago

Selected Answer: D

>>> # Question 47

>>> data = [10, 2, 1, 7, 5, 6, 4, 3, 9, 8]

>>>

>>> def find_high_low(nums):

... nums.sort()

... return nums[-1], nums[0]

...

>>> high, low = find_high_low(data)

>>>

>>> print(

... ('The highest number is {} ' +

... 'and the lowest number is {}.').format(high, low)

...)

The highest number is 10 and the lowest number is 1.

>>>

upvoted 1 times

christostz03 4 months, 2 weeks ago

d is the correct answer

upvoted 1 times

Question #48 *Topic 1*

What is the output of the following snippet?

```python
my_list = ['Mary', 'had', 'a', 'little', 'lamb']


def my_list(my_list):
    del my_list[3]
    my_list[3] = 'ram'


print(my_list(my_list))
```

A. ['Mary', 'had', 'a', 'little', 'lamb']

B. ['Mary', 'had', 'a', 'lamb']

C. ['Mary', 'had', 'a', 'ramb']

D. No output, the snippet is erroneous

Correct Answer: *D*

Community vote distribution

D (100%)

hovnival 1 week, 2 days ago

Selected Answer: D

There are a few issues with the provided code:

The list my_list is correctly defined, but the function my_list has the same name as the list my_list, which can cause confusion and errors.
The function my_list does not return anything, so print(my_list(my_list)) will print None.
Here is the corrected version of the code:

Copy

my_list = ['Mary', 'had', 'a', 'little', 'lamb']

def modify_list(input_list):
del input_list[3]
input_list[3] = 'ram'
return input_list

modified_list = modify_list(my_list)
print(modified_list)

In this corrected version:

The function is renamed to modify_list to avoid naming conflicts.
The function modify_list returns the modified list, which is then printed.
When you run this code, the output will be:

['Mary', 'had', 'a', 'ram']

upvoted 1 times

megan_mai 4 months, 1 week ago

Selected Answer: D

TypeError: 'function' object does not support item deletion

upvoted 2 times

⊟ 👤 **Sanela** 7 months, 2 weeks ago

D, because the name of the list is the same as the name of the function

upvoted 3 times

Question #49

Topic 1

What is the expected output of the following code?

```python
data = ['Peter', 'Paul', 'Mary']
print(data[int(-1 / 2)])
```

A. Paul

B. Mary

C. The code is erroneous.

D. None of the above.

E. Peter

Correct Answer: *E*

hovnival 1 week, 2 days ago

Selected Answer: E

The expression int(-1 / 2) prints Peter because of the way integer division and type conversion work in Python. Let's break it down step by step:

Division Operation:

When you perform the division -1 / 2, the result is -0.5.
Type Conversion:

The int function converts a floating-point number to an integer by truncating the decimal part. This means it removes the fractional part without rounding.
Therefore, int(-0.5) becomes 0.
List Indexing:

In Python, list indexing starts from 0. So, data[0] refers to the first element of the list.
Combining these steps, the expression data[int(-1 / 2)] is equivalent to data[0], which accesses the first element of the list data, which is 'Peter'.

So, the reason (-1 / 2) prints 'Peter' is because the expression evaluates to 0, and data[0] refers to the first element of the list.

upvoted 1 times

christostz03 4 months, 2 weeks ago

e is the correct answer

upvoted 1 times

Question #50 *Topic 1*

What is the expected output of the following code?

```
data = (1, 2, 3, 4)
data = data[-2:-1]
data = data[-1]
print(data)
```

A. 1

B. 4

C. 3

D. 2

Correct Answer: *C*

hovnival 1 week, 2 days ago

Selected Answer: C

Initialization:

Copy

data = (1, 2, 3, 4)

The variable data is initialized as a tuple containing the elements (1, 2, 3, 4).

Slicing the Tuple:

Copy

data = data[-2:-1]

The tuple data is sliced from the second-to-last element to the element just before the last element. In this case, data[-2:-1] will result in a new tuple containing only the third element of the original tuple, which is (3,).

Accessing the Last Element of the Tuple:

Copy

data = data[-1]

The last element of the tuple (3,) is accessed. Since the tuple contains only one element, data[-1] will be 3.

Printing the Result:

Copy

print(data)

The value of data, which is now 3, is printed.

upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #51 *Topic 1*

What is the expected output of the following code?

```python
data = {1: 0, 2: 1, 3: 2, 0: 1}
x = 0

for _ in range(len(data)):
    x = data[x]

print(x)
```

A. 0

B. 1

C. The code is erroneous.

D. 2

Correct Answer: *A*

hovnival 1 week, 2 days ago

Selected Answer: A

Let's go through the code step by step:

The dictionary data is defined as {1: 0, 2: 1, 3: 2, 0: 1}.
The variable x is initialized to 0.
The for loop runs len(data) times, which is 4 times since the length of the dictionary data is 4.
Let's trace the value of x through each iteration of the loop:

Iteration 1: x = data[0] which is 1. So x becomes 1.
Iteration 2: x = data[1] which is 0. So x becomes 0.
Iteration 3: x = data[0] which is 1. So x becomes 1.
Iteration 4: x = data[1] which is 0. So x becomes 0.
After completing the loop, the value of x is 0. Therefore, the output is:

0

upvoted 1 times

christostz03 4 months, 2 weeks ago

a is the correct answer

upvoted 1 times

Question #52 Topic 1

What is the expected output of the following code?

```
list = ['Peter', 'Paul', 'Mary']


def list(data):
    del data[1]
    data[1] = 'Jane'
    return data


print(list(list))
```

A. The code is erroneous.

B. ['Peter', 'Jane', 'Mary']

C. ['Peter', 'Jane']

D. ['Paul', 'Mary', 'Jane']

Correct Answer: *A*

Community vote distribution

A (100%)

hovnival 1 week, 2 days ago

Selected Answer: A

The provided code snippet results in a TypeError because the variable list (which is a list) and the function list (which is a function) have the same name. When you call print(list(list)), the interpreter gets confused and tries to treat the function list as if it were the list list.

To avoid this naming conflict, you should use different names for the variable and the function.

upvoted 1 times

pescamillam 2 months, 1 week ago

Selected Answer: A

Traceback (most recent call last):
File "main.py", line 8, in <module>
print (list(list))
File "main.py", line 4, in list
del data[1]
TypeError: 'function' object does not support item deletion

** Process exited - Return Code: 1 **
Press Enter to exit terminal

upvoted 1 times

christostz03 4 months, 2 weeks ago

a is the correct answer. We can not have list as a def name.

upvoted 1 times

NullNickname 7 months ago

Answer: C

upvoted 1 times

Question #53 *Topic 1*

What is the expected output of the following code?

```
data = {}
data['2'] = [1, 2]
data['1'] = [3, 4]

for i in data.keys():
    print(data[i][1], end=' ')
```

A. 2 4

B. 1 3

C. 3 1

D. 4 2

Correct Answer: *A*

⊟ **hovnival** 1 week, 2 days ago

Selected Answer: A

The expected output of the given code is:

24

Explanation:

The dictionary data is created with two keys: '2' and '1'.

data['2'] = [1, 2]

data['1'] = [3, 4]

The for loop iterates over the keys of the dictionary data.

For each key i, the code prints the second element (index 1) of the corresponding list.

For the key '2', data['2'][1] is 2.

For the key '1', data['1'][1] is 4.

The end='' argument in the print function ensures that the outputs are printed on the same line without any spaces in between.

Therefore, the output is 24.

upvoted 1 times

⊟ **vestersly** 4 months, 2 weeks ago

A, is correct because the index 1 in from the loop iteration is 2, and 4.

upvoted 2 times

⊟ **christostz03** 4 months, 2 weeks ago

a is the correct answer

upvoted 2 times

Question #54 Topic 1

What is the expected output of the following code?

```
data1 = (1, 2)
data2 = (3, 4)
[print(sum(x)) for x in [data1 + data2]]
```

A. 4

B. 10

C. Nothing gets printed.

D. The code is erroneous.

Correct Answer: *B*

hovnival 1 week, 2 days ago

Selected Answer: B

The given code concatenates two tuples data1 and data2 to form a new tuple and then sums up its elements. Here is the step-by-step explanation:

data1 = (1, 2)
data2 = (3, 4)
The expression data1 + data2 concatenates the two tuples to form a single tuple (1, 2, 3, 4).
The list comprehension [print(sum(x)) for x in [data1 + data2]] iterates over a list containing the single concatenated tuple [(1, 2, 3, 4)].
So, the list comprehension will execute print(sum((1, 2, 3, 4))).

The sum function calculates the sum of the elements in the tuple (1, 2, 3, 4), which is 1 + 2 + 3 + 4 = 10.
The print function prints the result 10.
Therefore, the expected output of the code is:

10

upvoted 1 times

christostz03 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #55 *Topic 1*

After execution of the following snippet, the sum of all vals elements will be equal to:

```
vals = [0, 1, 2]
vals.insert(0, 1)
del vals[1]
```

A. 4

B. 5

C. 2

D. 3

Correct Answer: *A*

hovnival 1 week, 2 days ago

Selected Answer: A

Let's analyze the given code snippet step-by-step to determine the sum of all elements in the vals list after execution:

vals = [0, 1, 2]

Initially, vals is set to [0, 1, 2].
vals.insert(0, 1)

This inserts the element 1 at index 0. After this operation, vals becomes [1, 0, 1, 2].
del vals[1]

This deletes the element at index 1. After this operation, vals becomes [1, 1, 2].
Now, let's calculate the sum of all elements in the vals list:

1 + 1 + 2 = 4
Therefore, the sum of all elements in the vals list after execution is 4.
upvoted 1 times

christostz03 4 months, 2 weeks ago

a is the correct answer
upvoted 2 times

Question #56 *Topic 1*

What is the expected output of the following code?

```
data = set([1, 2, 2, 3, 3, 3, 4, 4, 4, 4])
print(len(data))
```

A. 4

B. 2

C. 10

D. 0

E. 1

F. 3

Correct Answer: *A*

hovnival 1 week, 1 day ago

Selected Answer: A

The given code creates a set from a list containing duplicate elements and then prints the length of the set. Let's analyze it step-by-step:

data = set([1, 2, 2, 3, 3, 3, 4, 4, 4, 4])

This creates a set from the list [1, 2, 2, 3, 3, 3, 4, 4, 4, 4].
Since sets do not allow duplicate elements, the resulting set will be {1, 2, 3, 4}.
print(len(data))

This prints the length of the set data.
The set {1, 2, 3, 4} has 4 unique elements.
Therefore, the expected output of the code is:

4

upvoted 1 times

christostz03 4 months, 2 weeks ago

a is the correct answer

upvoted 1 times

Question #57 *Topic 1*

What is the output of the following snippet?

```python
my_list = [x * x for x in range(5)]

def fun(lst):
    del lst[lst[2]]
    return lst

print(fun(my_list))
```

A. [0, 1, 9, 16]

B. [0, 1, 4, 16]

C. [0, 1, 4, 9]

D. [1, 4, 9, 16]

Correct Answer: *C*

hovnival 1 week, 1 day ago

Selected Answer: C

Let's analyze the given code snippet step-by-step to determine the output:

my_list = [x * x for x in range(5)]

This creates a list comprehension that generates a list of the squares of numbers from 0 to 4.
The resulting list is [0, 1, 4, 9, 16].
The function fun takes a list lst as an argument.

del lst[lst[2]] removes the element at the index specified by the value of lst[2].
In my_list, lst[2] is 4. Therefore, del lst[4] removes the element at index 4 from the list lst.
The list lst is modified in place and then returned.
print(fun(my_list))

This calls the function fun with my_list as the argument and prints the result.
Let's see the steps within the function fun:

The initial list is [0, 1, 4, 9, 16].
lst[2] is 4, so del lst[4] removes the element at index 4, which is 16.
The modified list is [0, 1, 4, 9].
Therefore, the expected output of the code is:

[0, 1, 4, 9]
upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer
upvoted 2 times

Question #58 *Topic 1*

What is the expected output of the following code?

```
a = [1, 2, 3, 4, 5]
print(a[3:0:-1])
```

A. [4, 3, 2, 1]

B. [4, 3, 2]

C. [4, 3]

D. The code is erroneous.

Correct Answer: *B*

hovnival 1 week, 1 day ago

Selected Answer: B

Initial List
The initial list a is [1, 2, 3, 4, 5].

Slicing Operation
The slicing operation is a[3:0:-1].

The slicing syntax is list[start:stop:step]:

start is the index where the slice starts (inclusive).
stop is the index where the slice ends (exclusive).
step is the step size or the direction of the slice.
In this case:

start is 3, so the slice starts at index 3 (the element 4).
stop is 0, so the slice ends before index 0 (the element 1), and since the step is negative, it will not include the element at index 0.
step is -1, so the slice moves backward.
Extracting Elements
Following the slicing operation a[3:0:-1]:

Start at index 3 (the element 4).
Move backward with a step of -1.
Stop before reaching index 0.
The elements included in the slice are:

a[3] is 4
a[2] is 3
a[1] is 2
Therefore, the resulting sliced list is [4, 3, 2].

Output
[4, 3, 2]
upvoted 1 times

christostz03 4 months, 2 weeks ago

b is the correct answer
upvoted 1 times

Question #59 *Topic 1*

Take a look at the snippet and choose one of the following statements which is true:

```
nums = []
vals = nums[:]
vals.append(1)
```

A. nums and vals are of the same length

B. nums is longer than vals

C. vals is longer than nums

Correct Answer: *C*

hovnival 1 week, 1 day ago

Selected Answer: C

Let's analyze the given code snippet step-by-step to determine which statement is true:

nums = []
vals = nums[:]
vals.append(1)

nums = []

nums is initialized as an empty list: [].
vals = nums[:]

vals is created as a shallow copy of nums.
Since nums is an empty list, vals will also be an empty list: [].
vals.append(1)

1 is appended to the list vals.
After this operation, vals will be: [1].
Final State of the Lists
nums remains an empty list: [].
vals is now: [1].
Comparing the Lengths
The length of nums is 0.
The length of vals is 1.
Conclusion
vals is longer than nums.
Therefore, the correct statement is:

C. vals is longer than nums
upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer. we have a swallow copy here so vals is greater since nums is not getting appended but only vals.
upvoted 2 times

Question #60 *Topic 1*

What is the expected output of the following code?

```python
data = {'1': '0', '0': '1'}

for d in data.vals():
    print(d, end=' ')
```

A. 0 1

B. 1 0

C. The code is erroneous.

D. 0 0

Correct Answer: *C*

Community vote distribution

C (80%) A (20%)

hovnival 1 week, 1 day ago

Selected Answer: C

The code you provided contains an error. Specifically, the method vals() does not exist for dictionary objects in Python. The correct method to retrieve the values of a dictionary is values().

Here's the corrected version of the code:
data = {'1': '0', '0': '1'}
for d in data.values():
print(d, end=")

upvoted 1 times

Nfyughguygjhg 4 weeks ago

Selected Answer: A

its AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

upvoted 1 times

Vano1 2 months ago

Selected Answer: C

.vals() method doesn't exist. The correct method is .values().
Therefore, the correct answer is C.

upvoted 3 times

pescamillam 2 months, 1 week ago

Selected Answer: C

Traceback (most recent call last):
File "main.py", line 2, in <module>
for d in data.vals():
AttributeError: 'dict' object has no attribute 'vals'

upvoted 2 times

SarangM 2 months, 1 week ago

Selected Answer: A

A is the correct answer

upvoted 2 times

Ragoo 2 months, 2 weeks ago

Run the code it shows the out out 0 1

data = {'1': '0', '0': '1'}

```
for d in data.values():
print(d, end=' ')
```

The answer is A

upvoted 2 times

Vano1 2 months ago

in the code is "data.vals()" which is erronous. Therefore, the correct answer is C.

upvoted 2 times

megan_mai 4 months, 1 week ago

Selected Answer: C

AttributeError: 'dict' object has no attribute 'vals'

data.values() is the correct method

upvoted 2 times

Question #61 *Topic 1*

What code would you insert instead of the comment to obtain the expected output?

Expected output:

```
a
b
c
```

Code:

```python
dictionary = {}
my_list = ['a', 'b', 'c', 'd']

for i in range(len(my_list) - 1):
    dictionary[my_list[i]] = (my_list[i], )

for i in sorted(dictionary.keys()):
    k = dictionary[i]
    # Insert your code here.
```

A. print(k[0])

B. print(k)

C. print(k['0'])

D. print(k["0"])

Correct Answer: *A*

megan_mai 4 months, 1 week ago

dict = {}

list = ['a', 'b', 'c', 'd']

for i in range(len(list)-1):

dict[list[i]] = (list[i],) >>> adding list's element to the dictionary by creating a tuple containing a single list element

>>> dict = {'a': ('a',), 'b': ('b',), 'c': ('c',)}

for i in sorted(dict.keys()):

k = dict[i]

print(k[0])

>>>> a

b

c

print(k) >>> ('a',)

('b',)

('c',)

The other two options will generate errors

>>> Answer is A

upvoted 2 times

christostz03 4 months, 2 weeks ago

a is the correct answer

upvoted 1 times

brunoestudos 4 months, 3 weeks ago

From my perspective, each option doesn't match the requirements of this excercise. The correct answer is use a print(i) above the second loop for!

upvoted 1 times

Question #62 *Topic 1*

What is the expected output of the following code?

```
data = [
    [1, 2, 3, 4],
    [5, 6, 7, 8],
    [9, 10, 11, 12],
    [13, 14, 15, 16]
]
for i in range(0, 4):
    print(data[i].pop(), end=' ')
```

A. 1 5 9 13

B. 1 2 3 4

C. 13 14 15 16

D. 4 8 12 16

Correct Answer: *D*

hovnival 1 week, 1 day ago

Selected Answer: D

The provided code iterates over each sublist in the data list and removes (and prints) the last element from each sublist using the pop() method. The pop() method removes and returns the last item in the list if no index is specified.

the expected output of the code is:
4 8 12 16

upvoted 1 times

vedu04 2 weeks ago

Selected Answer: D

pop() removes and return last element in each list when it iterates 4 times each list's last element removed and return to console

upvoted 1 times

christostz03 4 months, 2 weeks ago

d is the correct answer

upvoted 1 times

Question #63 *Topic 1*

What is the expected output of the following code?

```python
data = [[0, 1, 2, 3] for i in range(2)]
print(data[2][0])
```

A. The code is erroneous.

B. 2

C. 1

D. 0

Correct Answer: *A*

Community vote distribution

A (100%)

⊟ **hovnival** 1 week, 1 day ago

Selected Answer: A

The code provided will result in an IndexError. Let's break it down step by step to understand why:

data = [[0, 1, 2, 3] for i in range(2)]
print(data[2][0])

The list comprehension [[0, 1, 2, 3] for i in range(2)] generates a list of lists. The range(2) means that the outer list will have 2 elements, each of which is the list [0, 1, 2, 3].

So, data will be:

data = [
[0, 1, 2, 3],
[0, 1, 2, 3]
]

The print(data[2][0]) statement attempts to access the element at index 2 of the data list. However, since data only has indices 0 and 1 (it has only 2 sublists), trying to access data[2] will raise an IndexError.
So, the expected output when you run the code will be:
IndexError: list index out of range
upvoted 1 times

⊟ **megan_mai** 4 months, 1 week ago

Selected Answer: A

data = [[0, 1, 2, 3], [0, 1, 2, 3]]
calling data[2][0] will be out of range >> IndexError: list index out of range
>>> The answer is A
upvoted 2 times

⊟ **christostz03** 4 months, 2 weeks ago

a is the correct answer
upvoted 1 times

Question #64 *Topic 1*

What is the expected output of the following code?

```
numbers = [1, 2, 3, 4, 5]
nums = numbers[2: ]
print(nums)
```

A. [2]

B. [3, 4, 5]

C. The program will cause an error.

D. [2, 3, 4, 5]

Correct Answer: *B*

⊟ **hovnival** 1 week, 1 day ago

Selected Answer: B

The code provided is slicing a list in Python. Let's break down the code to understand the expected output:

numbers = [1, 2, 3, 4, 5]
nums = numbers[2:]
print(nums)

Here, numbers is a list containing the elements [1, 2, 3, 4, 5].

The slicing operation numbers[2:] means "take all elements from index 2 to the end of the list". In Python, list indices start at 0, so index 2 refers to the third element in the list, which is 3.

Therefore, numbers[2:] will result in a sublist that starts from the element at index 2 and includes all subsequent elements.

The resulting sublist will be [3, 4, 5].

So, the expected output of the code is:

[3, 4, 5]

The correct answer is:
B. [3, 4, 5]
upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

b is the correct answer
upvoted 1 times

Question #65 *Topic 1*

You develop a Python application for your company.
A list named employees contains 200 employee names, the last five being company management. You need to slice the list to display all employees excluding management.
Which code segments can you use? (Choose two.)

A. employees[1:-5]

B. employees[0:-5]

C. employees[:-5]

D. employees[0:-4]

E. employees[1:-4]

Correct Answer: *BC*

hovnival 1 week, 1 day ago

Selected Answer: BC

To slice the list to display all employees excluding the last five who are company management, you can use the following code segments:

B. employees[0:-5]
C. employees[:-5]

Both of these options will correctly exclude the last five elements from the list.

upvoted 1 times

christostz03 4 months, 2 weeks ago

b & c are the correct answers

upvoted 2 times

Question #66 *Topic 1*

What is the expected output of the following code?

```
data = [1, 2, 3, 4, 5, 6]

for i in range(1, 6):
    data[i - 1] = data[i]

for i in range(0, 6):
    print(data[i], end=' ')
```

A. 1 1 2 3 4 5

B. 2 3 4 5 6 1

C. 2 3 4 5 6 6

D. 1 2 3 4 5 6

Correct Answer: *C*

hovnival 1 week, 1 day ago

Selected Answer: C

data = [1, 2, 3, 4, 5, 6]

In the first for loop, the code updates each element of data from index 0 to 4 with the value of the next element:

for i in range(1, 6):

data[i - 1] = data[i]

This loop will perform the following assignments:

When i = 1: data[0] = data[1] → data = [2, 2, 3, 4, 5, 6]

When i = 2: data[1] = data[2] → data = [2, 3, 3, 4, 5, 6]

When i = 3: data[2] = data[3] → data = [2, 3, 4, 4, 5, 6]

When i = 4: data[3] = data[4] → data = [2, 3, 4, 5, 5, 6]

When i = 5: data[4] = data[5] → data = [2, 3, 4, 5, 6, 6]

In the second for loop, the code prints each element of data from index 0 to 5:

for i in range(0, 6):

print(data[i], end=' ')

The output of this loop will be:

2 3 4 5 6 6

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #67 *Topic 1*

What is the expected output of the following code?

```
x = [1, 2, 3, 4, 5, 6, 7, 8, 9]
x[::2] = 10, 20, 30, 40, 50, 60
print(x)
```

A. [1, 10, 3, 20, 5, 30, 7, 40, 9, 50, 60]

B. [1, 2, 10, 20, 30, 40, 50, 60]

C. The code is erroneous.

D. [10, 2, 20, 4, 30, 6, 40, 8, 50, 60]

Correct Answer: *C*

hovnival 1 week, 1 day ago

Selected Answer: C

The code will result in an error because you are trying to assign 6 values (10, 20, 30, 40, 50, 60) to every other element of the list x, which has only 5 positions (1, 3, 5, 7, 9) available for assignment. The lengths do not match, and thus it raises a ValueError.

Therefore, the correct answer is:
C. The code is erroneous.

upvoted 1 times

christostz03 4 months, 2 weeks ago

ValueError: attempt to assign sequence of size 6 to extended slice of size 5. so c is correct

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer.

upvoted 1 times

NullNickname 7 months ago

Answer D

upvoted 1 times

Question #68 *Topic 1*

What is the expected output of the following code?

```
box = {}
jars = {}
crates = {}

box['biscuit'] = 1
box['cake'] = 3

jars['jam'] = 4

crates['box'] = box
crates['jars'] = jars

print(len(crates[box]))
```

A. 2

B. 1

C. The code is erroneous.

D. 3

E. 4

Correct Answer: *C*

hovnival 1 week, 1 day ago

Selected Answer: C

The code is erroneous.

The error occurs in the line print(len(crates[box])). The key box is not a string but a dictionary, which is not a valid key type for a dictionary in Python. Dictionary keys must be of an immutable type, such as strings, numbers, or tuples containing only immutable types.

To fix the code, you should use the string 'box' as the key instead of the dictionary box

upvoted 1 times

FedeXD 3 months, 1 week ago

C is correct. Dictionary keys have to be immutable, therefore you can't index a dictionary with a dictionary.

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #69 *Topic 1*

What is the expected output of the following code?

```python
data = 'Hello@Peter!!'
print(data.lower())
```

A. hellopeter

B. hello@Peter!!

C. hello@peter!!

D. None

Correct Answer: *C*

hovnival 1 week, 1 day ago

Selected Answer: C

The lower() method converts all uppercase letters in the string to lowercase. The special characters and numbers are not affected by this method.

So, the string 'Hello@Peter!!' is converted to 'hello@peter!!'.

Therefore, the output is:

hello@peter!!
Hence, the correct answer is C. hello@peter!!

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #70 *Topic 1*

Consider the following list.

data = [1, 5, 10, 19, 55, 30, 55, 99]

Which of the code snippets below would produce a new list like the following?

[1, 5, 10, 99]

A.
```
data.pop(5)
data.pop(19)
data.pop(55)
```

B.
```
data.pop(1)
data.pop(3)
data.pop(4)
data.pop(6)
```

C. None of the above.

D.
```
data.pop(5)
data.remove(19)
data.remove(55)
data.remove(55)
```

E.
```
data.remove(5)
data.remove(19)
data.remove(55)
```

Correct Answer: *D*

Community vote distribution

D (100%)

⊟ **hovnival** 1 week, 1 day ago

Selected Answer: D

# Question 70

data = [1, 5, 10, 19, 55, 30, 55, 99]

data.pop(5)

data.remove(19)

data.remove(55)

data.remove(55)

print(data)

upvoted 1 times

⊟ **sandy589** 2 months, 2 weeks ago

Selected Answer: D

D is the correct answer, below is the explanation:

data.pop(5) removes the element at index 5 (which is 30), leaving [1, 5, 10, 19, 55, 55, 99].

data.remove(19) removes the first occurrence of 19, leaving [1, 5, 10, 55, 55, 99].

data.remove(55) removes the first occurrence of 55, leaving [1, 5, 10, 55, 99].

data.remove(55) removes the second occurrence of 55, leaving [1, 5, 10, 99], which matches the target list.

upvoted 2 times

⊟ **s_tzoref** 2 months, 3 weeks ago

D would remove 5, which needs to remain, and would not remove 30?

upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

d is the correct answer

upvoted 1 times

Question #71 *Topic 1*

What is the expected output of the following code?

```python
data1 = 'a', 'b'
data2 = ('a', 'b')
print(data1 == data2)
```

A. 0

B. 1

C. False

D. True

Correct Answer: *D*

hovnival 1 week, 1 day ago

Selected Answer: D

The expected output of the given code is True.

Here's a breakdown of why:

data1 = 'a', 'b' creates a tuple with two elements, 'a' and 'b'.
data2 = ('a', 'b') also creates a tuple with two elements, 'a' and 'b'.
In Python, tuples can be created either by using parentheses or by simply separating elements with commas. Therefore, both data1 and data2 are tuples containing the same elements in the same order.

When we compare data1 == data2, we are checking if the two tuples are equal. Since both tuples contain the same elements in the same order, the comparison returns True.

So, the expected output is:

True

upvoted 1 times

christostz03 4 months, 2 weeks ago

d is the correct answer

upvoted 1 times

Question #72 *Topic 1*

What is the output of the following snippet?

```python
l1 = [1, 2, 3]

for v in range(len(l1)):
    l1.insert(1, l1[v])

print(l1)
```

A. [1, 2, 3, 1, 2, 3]

B. [3, 2, 1, 1, 2, 3]

C. [1, 1, 1, 1, 2, 3]

D. [1, 2, 3, 3, 2, 1]

Correct Answer: *C*

hovnival 1 week, 1 day ago

Selected Answer: C

1. You start with the list `l1 = [1, 2, 3]`.

2. You iterate through the range of the length of `l1`, which initially is 3.

3. During each iteration, you insert the value at the current index `v` into the second position of the list (index 1).

Here's the detailed iteration breakdown:

- Initial `l1 = [1, 2, 3]`.
- The length of `l1` is 3, so the range is `range(3)` which is `[0, 1, 2]`.

Iteration 1 (v = 0):

- `l1[0]` is 1.
- Insert `1` at position `1`.
- `l1` becomes `[1, 1, 2, 3]`.

Iteration 2 (v = 1):

- `l1[1]` is 1.
- Insert `1` at position `1`.
- `l1` becomes `[1, 1, 1, 2, 3]`.

Iteration 3 (v = 2):

- `l1[2]` is 1.
- Insert `1` at position `1`.
- `l1` becomes `[1, 1, 1, 1, 2, 3]`.

Finally, the list `l1` is `[1, 1, 1, 1, 2, 3]`.

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #73 *Topic 1*

Insert the correct snippet to convert the t tuple to a dictionary named d.

Expected output:

```
{'A': 1, 'B': 2, 'C': 3}
```

Code:

```python
t = (('A', 1), ('B', 2), ('C', 3))
# insert code here
print(d)
```

A. t >> d.dict

B. d = dict(t)

C. d.dict(t)

D. d = t(dict)

Correct Answer: *B*

hovnival 1 week, 1 day ago

Selected Answer: B

The correct snippet to convert the tuple `t` to a dictionary named `d` is:

B. `d = dict(t)`

So the full code will be:

t = (('A', 1), ('B', 2), ('C', 3))
d = dict(t)
print(d)

The expected output will be:
{'A': 1, 'B': 2, 'C': 3}
upvoted 1 times

christostz03 4 months, 2 weeks ago

b is the correct answer
upvoted 2 times

Question #74 *Topic 1*

How many elements does the my_list list contain?

```python
my_list = [0 for i in range(1, 3)]
```

A. one

B. three

C. two

Correct Answer: *C*

hovnival 1 week, 1 day ago

Selected Answer: C

The list my_list is created using a list comprehension. The expression [0 for i in range(1, 3)] generates a list containing the value 0 for each number in the range from 1 to 2 (inclusive of 1 and exclusive of 3).

So, the range range(1, 3) produces two numbers: 1 and 2.

Therefore, the list my_list will contain 2 elements, both of which are 0.

Hence, my_list contains 2 elements.

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #75 *Topic 1*

What is the expected output of the following code?

```
data = {'z': 23, 'x': 7, 'y': 42}

for _ in sorted(data):
    print(data[_], end=' ')
```

A. 7 23 42

B. 7 42 23

C. 42 23 7

Correct Answer: *B*

hovnival 1 week, 1 day ago

Selected Answer: B

The code provided will output the values of the dictionary data in the order of their keys sorted alphabetically. Here is a step-by-step explanation:

The dictionary data is defined as {'z': 23, 'x': 7, 'y': 42}.
The sorted(data) function sorts the keys of the dictionary alphabetically, resulting in the list ['x', 'y', 'z'].
The for loop iterates over this sorted list of keys.
During each iteration, the corresponding value from the dictionary is accessed using the key and printed with a space as the separator.
So, the expected output of the code is:

7 42 23

upvoted 1 times

christostz03 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #76 *Topic 1*

What is the expected output of the following code?

```
x = {(1, 2): 1, (2, 3): 2}
print(x[1, 2])
```

A. {(2, 3): 2}

B. {(1, 2): 1}

C. The code is erroneous.

D. 1

Correct Answer: *D*

hovnival 1 week, 1 day ago

Selected Answer: D

In the given code, you have a dictionary `x` where the keys are tuples and the values are integers. The dictionary is defined as:

x = {(1, 2): 1, (2, 3): 2}

When you access the value associated with the key `(1, 2)` using `x[1, 2]`, you are essentially looking for the value stored for the key `(1, 2)` in the dictionary `x`.

The key `(1, 2)` has an associated value of `1`.

So, the expected output of the code:

print(x[1, 2])

is:

1

upvoted 1 times

christostz03 4 months, 2 weeks ago

d is the correct answer

upvoted 1 times

Question #77 *Topic 1*

Take a look at the snippet and choose the true statement:

```
nums = [1, 2, 3]
vals = nums
del vals[:]
```

A. vals is longer than nums

B. nums is longer than vals

C. nums and vals have the same length

D. The snippet will cause a runtime error.

Correct Answer: *C*

hovnival 1 week, 1 day ago

Selected Answer: C

The correct answer is:

C. nums and vals have the same length

Explanation: When vals is assigned to nums, both vals and nums refer to the same list in memory. The del vals[:] operation deletes all elements from the list that both vals and nums reference. Therefore, after this operation, both vals and nums are empty lists, and thus have the same length (which is 0).

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #78 *Topic 1*

What is the expected output of the following code?

```python
data = {}
data[1] = 1
data['1'] = 2
data[1.0] = 4

res = 0
for d in data:
    res += data[d]

print(res)
```

A. 3

B. The code is erroneous.

C. 6

D. 7

Correct Answer: *C*

hovnival 1 week, 1 day ago

Selected Answer: C

The expected output of the given code is 6.

Here's a step-by-step explanation of how the code works:

An empty dictionary data is created.
The key 1 is assigned the value 1, so data becomes {1: 1}.
The key '1' (a string) is assigned the value 2, so data becomes {1: 1, '1': 2}.
The key 1.0 (a float) is assigned the value 4. In Python, the float 1.0 is considered equal to the integer 1, so this updates the value for the key 1 to 4. Now data becomes {1: 4, '1': 2}.
The dictionary data now contains two key-value pairs: {1: 4, '1': 2}.

The for loop iterates over the keys of the dictionary:

For the first key 1, it adds data[1] (which is 4) to res, so res becomes 4.
For the second key '1', it adds data['1'] (which is 2) to res, so res becomes 6.
Finally, the print(res) statement outputs 6.
upvoted 1 times

Question #1 *Topic 2*

What would you insert instead of ???, so that the program checks for even numbers?

```
if ???:
    print('x is an even number')
```

A. x % ‘even’ == True

B. x % 2 == 1

C. x % 2 == 0

D. x % x == 0

E. x % 1 == 2

Correct Answer: *C*

⊟ **christostz03** 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #2 *Topic 2*

What value will be assigned to the x variable?

```
z = 3
y = 7
x = y < z and z > y or y > z and z < y
```

A. 0

B. True

C. False

D. 1

Correct Answer: *B*

⊟ **hovnival** 1 week, 1 day ago

Selected Answer: B

1. Evaluate `y < z`:
7 < 3 => False

2. Evaluate `z > y`:
3 > 7 => False

3. Combine the first part with `and`:
False and False => False

4. Evaluate `y > z`:
7 > 3 => True

5. Evaluate `z < y`:
3 < 7 => True

6. Combine the second part with `and`:
True and True => True

7. Combine the results with `or`:
False or True => True

So, the value assigned to `x` is `True`.

upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

b is the correct answer / False or True => True

upvoted 1 times

Question #3 *Topic 2*

What is the expected output of the following code?

```
x = 1 // 5 + 1 / 5
print(x)
```

A. 0

B. 0.2

C. 0.0

D. 0.4

Correct Answer: *B*

hovnival 1 week ago

Selected Answer: B

First, 1 // 5 is calculated. The // operator performs integer (floor) division.

1 // 5 results in 0 because 1 divided by 5 is 0 with a remainder of 1, and the floor division discards the remainder.
Next, 1 / 5 is calculated. The / operator performs true division (floating-point division).

1 / 5 results in 0.2 because 1 divided by 5 is 0.2.
The results of these two operations are then added together:

0 (result of 1 // 5) + 0.2 (result of 1 / 5) = 0.2.
So, the value of x is 0.2.

Therefore, the expected output is:

B. 0.2
upvoted 1 times

christostz03 4 months, 2 weeks ago

b is the correct answer / 0 + 0.2 = 0.2
upvoted 2 times

Question #4 *Topic 2*

An operator able to check whether two values are not equal is coded as:

A. <>

B. !=

C. =/=

D. not ==

Correct Answer: *B*

christostz03 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #5 Topic 2

The result of the following addition:
123 + 0.0

A. cannot be evaluated

B. is equal to 123

C. is equal to 123.0

Correct Answer: *C*

hovnival 1 week ago

Selected Answer: C

The result of the addition 123 + 0.0 is equal to 123.0.

So, the correct answer is:
C. is equal to 123.0

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #6 *Topic 2*

What will be the output of the following code snippet?
print(3 / 5)

A. None of the above.

B. 0

C. 0.6

D. 6/10

Correct Answer: *C*

hovnival 1 week ago

Selected Answer: C

The output of the code snippet print(3 / 5) will be:

C. 0.6

upvoted 1 times

christostz03 4 months, 2 weeks ago

c is the correct answer

upvoted 1 times

Question #7 Topic 2

What will be the output of the following code snippet?

```
x = 2
y = 1
x *= y + 1
print(x)
```

A. 1

B. 4

C. 2

D. 3

E. None

Correct Answer: *B*

hovnival 1 week ago

Selected Answer: B

Let's break down the code snippet step by step:

x = 2 assigns the value 2 to the variable x.
y = 1 assigns the value 1 to the variable y.
x *= y + 1 is equivalent to x = x * (y + 1).
First, the expression inside the parentheses is evaluated: y + 1 which is 1 + 1 resulting in 2.
Then, the multiplication is performed: x * 2 which is 2 * 2 resulting in 4.
print(x) outputs the value of x, which is now 4.
So, the output of the code snippet will be:

4

upvoted 1 times

christostz03 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #8 *Topic 2*

What is the expected output of the following code?

```
x = 1 / 2 + 3 // 3 + 4 ** 2
print(x)
```

A. 17

B. 17.5

C. 8.5

D. 8

Correct Answer: *B*

hovnival 1 week ago

Selected Answer: B

Let's break down the expression step by step:

1 / 2 evaluates to 0.5 (division)
3 // 3 evaluates to 1 (integer division)
4 ** 2 evaluates to 16 (exponentiation)
Now we add these results together:

0.5 + 1 + 16 = 17.5

So, the correct output of the code is 17.5.

The correct option is:

B. 17.5

upvoted 1 times

christostz03 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #9 Topic 2

You develop a Python application for your company. You have the following code.

```python
def main(a, b, c, d):
    value = a + b * c - d
    return value
```

Which of the following expressions is equivalent to the expression in the function?

A. (a + b) * (c - d)

B. (a + (b * c)) - d)

C. (a + ((b * c) - d)

D. None of the above.

Correct Answer: *B*

Community vote distribution

D (100%)

hovnival 1 week ago

Selected Answer: B

The expression in the function is:
value = a + b * c - d

According to the order of operations (PEMDAS/BODMAS), multiplication is performed before addition and subtraction. Therefore, the expression can be rewritten with parentheses to make the order of operations explicit:

value = a + (b * c) - d

This matches option B:

B. (a + (b * c)) - d)

So, the correct answer is:

B. (a + (b * c)) - d)
upvoted 1 times

5cd303a 1 month, 2 weeks ago

Selected Answer: D

I would say D is the correct answer, as in B and C are each a parentheses missing.
upvoted 2 times

FedeXD 3 months, 1 week ago

B is correct. Multiplication has higher precedence than addition and substraction. Then for addition and substraction with the same precedence, left associativity is applied.
upvoted 1 times

Delroy2826 4 months ago

value = 1 + 2 * 3 - 4
print(value) #3
value = (1 + (2 * 3)) - 4
print(value) #3
Hence B is correct
upvoted 1 times

S1nthe0 4 months, 1 week ago

Why does the system say that the correct answer is B? It means the system answers can't be trusted?
upvoted 2 times

christostz03 4 months, 2 weeks ago

b is the correct answer

According to the order of operations (PEMDAS/BODMAS), multiplication is performed before addition and subtraction.

Specifically:

Multiplication: b * c

Addition: a + (result of b * c)

Subtraction: (result of a + b * c) - d

upvoted 1 times

herrmann69 7 months ago

Selected Answer: D

Answer D is correct. None of the above.

A is wrong since multiplication has higher precedence than addition/subtraction

B has a closing ")" too much

C has an opening "(" too much

upvoted 3 times

Question #10 *Topic 2*

What is the output of the following code?

```
a = 1
b = 0
x = a or b
y = not(a and b)
print(x + y)
```

A. The output cannot be predicted.

B. 2

C. The program will cause an error.

D. 1

Correct Answer: *B*

hovnival 1 week ago

Selected Answer: B

1. \`a\` is assigned the value \`1\`.
2. \`b\` is assigned the value \`0\`.

Next, we evaluate the expression \`x = a or b\`:
- The \`or\` operator returns the first truthy value it encounters, or the last value if none are truthy.
- \`a\` is \`1\`, which is truthy, so \`x\` will be \`1\`.

Then, we evaluate the expression \`y = not(a and b)\`:
- The \`and\` operator returns the first falsy value it encounters, or the last value if none are falsy.
- \`a\` is \`1\` (truthy) and \`b\` is \`0\` (falsy), so \`a and b\` is \`0\`.
- Applying the \`not\` operator to \`0\` (falsy) results in \`True\`, which is equivalent to \`1\` in Python.

Finally, we compute \`x + y\`:
- \`x\` is \`1\` and \`y\` is \`1\`.
- So, \`x + y\` is \`1 + 1\`, which equals \`2\`.

Therefore, the output of the code is:

2

upvoted 1 times

christostz03 4 months, 2 weeks ago

b is the correct answer
1 or 0 and not(a and b)=
= 1 and 1 =
=2

upvoted 1 times

Question #11 *Topic 2*

What is the expected output of the following code?

```
x = 28
y = 8
print(x / y)
print(x // y)
print(x % y)
```

A. 1 | 3.0
2 | 3
3 | 2

B. 1 | 3.5
2 | 3
3 | 4

C. 1 | 3
2 | 3.5
3 | 4

D. 1 | 3.5
2 | 3.5
3 | 2

Correct Answer: *B*

⊟ **hovnival** 1 week ago

Selected Answer: B

The expected output of the code is:

```
x = 28
y = 8
print(x / y) # This performs floating-point division
print(x // y) # This performs integer (floor) division
print(x % y) # This calculates the remainder of the division
```

Let's calculate each step:

1. `x / y` performs floating-point division:
\(28 / 8 = 3.5 \)

2. `x // y` performs integer (floor) division:
\(28 // 8 = 3 \)

3. `x % y` calculates the remainder of the division:
\(28 \% 8 = 4 \)

So, the output will be:

```
3.5
3
4
```

Therefore, the correct answer is:

B. 1 | 3.5

2 | 3

3 | 4

upvoted 1 times

⊟ **christostz03** 4 months, 2 weeks ago

b is the correct answer

upvoted 1 times

Question #12 *Topic 2*

Which of the following statements are true? (Choose two.)

A. The result of the / operator is always an integer value.

B. The ** operator uses right-sided binding.

C. The right argument of the % operator cannot be zero.

D. Addition precedes multiplication.

Correct Answer: *BC*

hovnival 1 week ago

Selected Answer: BC

The two statements that are true are:

B. The ** operator uses right-sided binding.
C. The right argument of the % operator cannot be zero.

Explanation:

Statement A is false because the / operator can result in a floating-point value if the operands are not both integers.
Statement D is false because, according to the order of operations (PEMDAS/BODMAS rules), multiplication precedes addition.

upvoted 1 times

Vano1 2 months ago

Where is the snippet? It is not displayed

upvoted 1 times

christostz03 4 months, 2 weeks ago

b & c are the correct answers

upvoted 2 times

Question #13 *Topic 2*

Which of the following statements is false?

A. The right argument of the % operator can not be zero.

B. The ** operator has right-to-left associativity.

C. Multiplication precedes addition.

D. The result of the / operator is always an integer value.

Correct Answer: *D*

hovnival 1 week ago

Selected Answer: D

D. The result of the / operator is always an integer value.

This statement is false. In many programming languages, the result of the / operator (division) can be a floating-point value, not necessarily an integer. For instance, in Python, dividing two integers using / results in a float. If integer division is needed, the // operator should be used instead.

upvoted 1 times

Sai_23_2000 1 week, 6 days ago

Selected Answer: A

How come A is correct
what happens if we have 0 on right side with % symbol

upvoted 1 times

Question #14 *Topic 2*

What is the expected output of the following code?
print(1 // 2 * 3)

A. 4.5

B. 0.16666666666666666

C. 0.0

D. 0

Correct Answer: *D*

hovnival 1 week ago

Selected Answer: D

Actually, in Python, both multiplication (*) and floor division (//) have the same precedence level, and they are evaluated from left to right.

So, in the given code print(1 // 2 * 3):

1 // 2 is evaluated first, resulting in 0 because 1 divided by 2 is 0.5 and the floor division of 0.5 is 0.
Then, 0 * 3 is evaluated, which is 0.
Therefore, the expected output is indeed 0.

upvoted 1 times

Question #15 *Topic 2*

What is the expected output of the following code?

```
x = True
y = False
x = x or y
y = x and y
x = x or y
print(x, y)
```

A. True True

B. True False

C. False False

D. False True

Correct Answer: *B*

hovnival 1 week ago

Selected Answer: B

Let's analyze the code step-by-step:

x = True
y = False
x = x or y
Since x is True and y is False, x or y evaluates to True. So, x remains True.
y = x and y
Now, x is True and y is False, x and y evaluates to False. So, y remains False.
x = x or y
Again, x is True and y is False, x or y evaluates to True. So, x remains True.
Finally, the print(x, y) statement will output True False.

So, the correct answer is:
B. True False
upvoted 1 times

Question #16 *Topic 2*

The ** operator:

A. performs duplicated multiplication

B. does not exist

C. performs floating-point multiplication

D. performs exponentiation

Correct Answer: *D*

hovnival 1 week ago

Selected Answer: D

D. performs exponentiation

In Python, the ** operator is used to raise a number to the power of another number. For example, 2 ** 3 results in 8.

upvoted 1 times

Question #17 *Topic 2*

Only one of the following statements is true - which one?

A. addition precedes multiplication

B. multiplication precedes addition

C. neither statement can be evaluated

Correct Answer: *B*

hovnival 1 week ago

Selected Answer: B

B. multiplication precedes addition

In the order of operations (PEMDAS/BODMAS), multiplication is performed before addition.

upvoted 1 times

Question #18 *Topic 2*

The += operator, when applied to strings, performs:

A. Subtraction

B. Concatenation

C. Multiplication

Correct Answer: *B*

hovnival 1 week ago

Selected Answer: B

B. Concatenation

The += operator, when applied to strings, appends the right-hand string to the left-hand string, effectively concatenating them.

upvoted 1 times

Question #19 *Topic 2*

What is the expected output of the following code?

```python
x = 9
y = 12
result = x // 2 * 2 / 2 + y % 2 ** 3
print(result)
```

A. 8

B. 7.0

C. 8.0

D. 9.0

Correct Answer: *C*

hovnival 1 week ago

Selected Answer: C

To determine the expected output of the given Python code, we need to follow the order of operations (PEMDAS/BODMAS) which stands for Parentheses, Exponents (Powers), Multiplication and Division (from left to right), and Addition and Subtraction (from left to right).

1. **Exponentiation**: `2 ** 3` is calculated first:
\[
2 ** 3 = 8
\]

2. **Modulo operation**: `y % 8`:
\[
12 % 8 = 4
\]

3. **Floor division**: `x // 2`:
\[
9 // 2 = 4
\]

4. **Multiplication**: `4 * 2`:
\[
4 * 2 = 8
\]

5. **Division**: `8 / 2`:
\[
8 / 2 = 4.0
\]

6. **Addition**: `4.0 + 4`:
\[
4.0 + 4 = 8.0
\]

So, the expected output of the code is:

8.0

upvoted 1 times

Question #20 *Topic 2*

Consider the following code.

```
x = 1
x = x == x
```

The value eventually assigned to x is equal to:

A. 1

B. 0

C. True

D. False

Correct Answer: *C*

hovnival 1 week ago

Selected Answer: C

In the given code:

x = 1
x = x == x

Here is the step-by-step explanation of what happens:

1. `x` is initially assigned the value `1`.
2. The expression `x == x` is evaluated. Since `x` is `1`, the expression `1 == 1` evaluates to `True`.
3. The result of the expression (`True`) is then assigned back to `x`.

Therefore, the value eventually assigned to `x` is `True`.

So, the correct answer is:
C. True

upvoted 1 times

Question #21 *Topic 2*

What is the output of the following code?

```python
a = 10
b = 20
c = a > b
print(not(c))
```

A. The program will cause an error.

B. False

C. None

D. True

Correct Answer: *D*

hovnival 1 week ago

Selected Answer: D

Let's analyze the code step by step:

1. `a = 10`: This assigns the value 10 to the variable `a`.
2. `b = 20`: This assigns the value 20 to the variable `b`.
3. `c = a > b`: This compares whether `a` is greater than `b`. Since `10` is not greater than `20`, the result is `False`. Therefore, `c` is assigned the value `False`.
4. `print(not(c))`: This prints the logical negation of `c`. Since `c` is `False`, `not(c)` is `True`.

Therefore, the output of the code is:

True

upvoted 1 times

Question #22 *Topic 2*

What is the expected output of the following code?
x = 1 + 1 // 2 + 1 / 2 + 2
print(x)

A. 3

B. 4.0

C. 4

D. 3.5

Correct Answer: *D*

hovnival 1 week ago

Selected Answer: D

Certainly! Let's break down the expression step by step, considering floor division in Python:

x = 1 + 1 // 2 + 1 / 2 + 2

1. The `//` operator (floor division) and `/` operator (float division) have higher precedence than the `+` operator, so they will be evaluated first.
2. Evaluate `1 // 2`:
- `1 // 2` results in `0` because floor division of 1 by 2 rounds down to 0.
3. Evaluate `1 / 2`:
- `1 / 2` results in `0.5` because it is float division.

Now substitute these results back into the expression:

x = 1 + 0 + 0.5 + 2

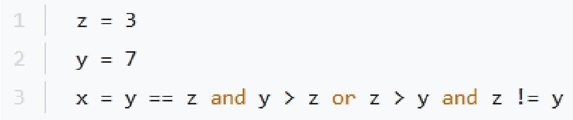
4. Next, evaluate the additions from left to right:
- `1 + 0` results in `1`
- `1 + 0.5` results in `1.5`
- `1.5 + 2` results in `3.5`

Therefore, the expected output of the code is:

D. 3.5
upvoted 1 times

Question #23 *Topic 2*

What value will be assigned to the x variable?



```
z = 3
y = 7
x = y == z and y > z or z > y and z != y
```

A. False

B. 0

C. 1

D. True

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #24 *Topic 2*

What is the expected output of the following code?

```
print(1 / 1)
```

A. 1

B. 1.0

C. This can not be predicted.

D. This can not be evaluated.

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #25 *Topic 2*

What is the expected output of the following code?

```python
list1 = [3, 7, 23, 42]
list2 = [3, 7, 23, 42]
print(list1 is list2)
print(list1 == list2)
```

A. 1 | False
2 | False

B. 1 | True
2 | True

C. 1 | False
2 | True

D. 1 | True
2 | False

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #26 *Topic 2*

What is the expected output of the following code?

```
print(3 * 'abc' + 'xyz')
```

A. 3abcxyz

B. abcabcxyzxyz

C. abcabcabcxyz

D. abcxyzabcxyzabcxyz

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #27 *Topic 2*

What is the expected output of the following code?

```python
nums = [3, 7, 23, 42]
alphas = ['p', 'p', 'm', 'j']

print(nums is alphas)
print(nums == alphas)

nums = alphas

print(nums is alphas)
print(nums == alphas)
```

A. 1 | True
2 | False
3 | True
4 | False

B. 1 | False
2 | False
3 | True
4 | True

C. 1 | False
2 | True
3 | True
4 | True

D. 1 | False
2 | True
3 | False
4 | True

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #28 *Topic 2*

What would you insert instead of ???, so that the program prints True to the monitor?

```
x = 'Peter'
y = 'Peter'
res = ???
print(res)
```

A. x < y

B. x is not y

C. x is y

D. x != y

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #29

Topic 2

What is the expected output of the following code?

```
x, y, z = 3, 2, 1
z, y, x = x, y, z
print(x, y, z)
```

A. 1 2 3

B. 1 2 2

C. 3 2 1

D. 2 1 3

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #30 *Topic 2*

What is the expected output of the following code?

```
x = 1
print(++++x)
```

A. 1

B. 2

C. 4

D. 3

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #31

Topic 2

What is the data type of x, y, z after executing the following snippet?

```
x = 23 + 42
y = '23' + '42'
z = '23' * 7
```

A. int, int, int

B. x is int,
y and z are invalid declarations

C. int, str, int

D. int, str, str

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Question #32 *Topic 2*

What is the expected output of the following code?

```
x = 2
y = 6
x += 2 ** 3
x //= y // 2 // 3
print(x)
```

A. 9

B. 10

C. 0

D. 11

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #33 *Topic 2*

The result of the following division:

```
1 / 1
```

A. cannot be evaluated

B. is equal to 1.0

C. is equal to 1

D. cannot be predicted

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #34 *Topic 2*

What is the expected output of the following code?

```
res = str(bool(1) + float(12) / float(2))
print(res)
```

A. 7

B. The code is erroneous.

C. 6.0

D. 7.0

E. 6

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Question #35 *Topic 2*

What is the expected output of the following code?

```
print('Mike' > 'Mikey')
```

A. False

B. True

C. 1

D. 0

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #36 *Topic 2*

Which of the following is the correct order of operator precedence?

A. Exponents -

Unary positive, negative, not -

Parentheses -

Multiplication and Division -

Addition and Subtraction -
And

B. Parentheses -

Exponents -

Unary positive, negative, not -

Addition and Subtraction -

Multiplication and Division -
And

C. Parentheses -

Exponents -

Unary positive, negative, not -

Multiplication and Division -

Addition and Subtraction -
And

D. Exponents -

Unary positive, negative, not -

Multiplication and Division -

Addition and Subtraction -

And -
Parentheses

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #37 *Topic 2*

What is the result of the following operation?

```
1 + 1.0
```

A. 2.0

B. 11.0

C. The operation is illegal in Python

D. 2

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #38

Topic 2

Consider the following code.

```
languages = ['English', 'Spanish', 'German']
more_languages = ['English', 'Spanish', 'German']
extra_languages = more_languages
```

Which statement will print True to the monitor? (Choose two.)

A. print(languages == more_languages)

B. print(languages is more_languages)

C. print(more_languages is extra_languages)

D. print(languages is extra_languages)

Correct Answer: *AC*

Currently there are no comments in this discussion, be the first to comment!

Question #39

Topic 2

Which expression evaluates to 7?

A. 9 % 3 + 7

B. 9 - 3 * 7

C. 9 // 3 - 7

D. 9 / 3 * 7

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #40 *Topic 2*

The // operator:

A. performs integer division

B. performs regular division

C. does not exist

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #41 *Topic 2*

Which of the following statements are correct? (Choose two.)

A. 7 + False evaluates to False

B. True or False evaluates to True

C. True + 1 evaluates to 2

D. True and False evaluates to True

E. type('') returns <class 'bool'>

Correct Answer: *BC*

Currently there are no comments in this discussion, be the first to comment!

Question #42 *Topic 2*

What is the output of the following snippet?

```
y = 2 + 3 * 5.
print(y)
```

A. 25.0

B. The snippet will cause an execution error.

C. 17.0

D. 17

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #43 *Topic 2*

What is the expected output of the following code?

```python
list1 = ['Peter', 'Paul', 'Mary', 'Jane']
list2 = ['Peter', 'Paul', 'Mary', 'Jane']

print(list1 is list2)
print(list1 == list2)

list1 = list2

print(list1 is list2)
print(list1 == list2)
```

A. 1 | False
2 | True
3 | False
4 | True

B. 1 | False
2 | True
3 | True
4 | True

C. 1 | False
2 | True
3 | True
4 | False

D. 1 | False
2 | False
3 | True
4 | True

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #44 *Topic 2*

Evaluate the following Python arithmetic expression:

(3 * (1 + 2) ** 2 - (2 ** 2) * 3)

What is the result?

A. 13

B. 3

C. 69

D. 15

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Question #45 *Topic 2*

What value will be assigned to the x variable?

```
z = 10
y = 0
x = z > y or z == y
```

A. 1

B. False

C. True

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #46

Topic 2

Consider the following code.

```
x = [0, 1, 2]
x[0], x[2] = x[2], x[0]
```

What does the second assignment do?

A. It shortens the list.

B. It reverses the list.

C. It doesn't change the list.

D. It extends the list.

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #47 *Topic 2*

An operator able to check whether two values are equal, is coded as:

A. ==

B. ===

C. is

D. =

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #48 *Topic 2*

What will be the output of the following snippet?

```
a = 1
b = 0
a = a ^ b
b = a ^ b
b = a ^ b
print(a, b)
```

A. 0 0

B. 1 1

C. 0 1

D. 1 0

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Question #49 *Topic 2*

What is the expected output of the following code?

```python
a = 1
b = 0
c = a & b
d = a | b
e = a ^ b
print(c + d + e)
```

A. 2

B. 0

C. 1

D. 3

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #50 *Topic 2*

Consider the following code.
x = float('23.42')
Which of the following expressions will evaluate to 2?

A. int(x) + False

B. bool(x) + True

C. str(x)

D. bool(x)

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #51 *Topic 2*

What is the expected output of the following code?

```
print(1 // 2)
```

A. 0.5

B. 0.0

C. None of the above.

D. 0

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Question #52 *Topic 2*

What is the result of the following code?

```
x = 7
y = x % 2
y += 1
print(y)
```

A. 1

B. 2

C. 5

D. 3

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #53 *Topic 2*

What is the expected output of the following code?

```python
list1 = ['Peter', 'Paul', 'Mary', 'Jane']
list2 = ['Peter', 'Paul', 'Mary', 'Jane']

print(list1 is not list2)
print(list1 != list2)

list1 = list2

print(list1 is not list2)
print(list1 != list2)
```

A. 1 | True
2 | True
3 | False
4 | False

B. 1 | True
2 | False
3 | False
4 | False

C. 1 | True
2 | False
3 | False
4 | True

D. 1 | True
2 | False
3 | True
4 | False

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #54 *Topic 2*

The expression:
'mike' > 'Mike'
is

A. erroneous

B. true

C. false

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #55 *Topic 2*

You are writing a Python program that evaluates an arithmetic formular.
The formular is described as b equals a multiplied by negative one, then raised to the second power, where a is the value that will be input and b is the result. a = eval(input('Enter a number for the equation: '))
Which of the following is a valid expression for the given requirement?

A. b = (-a) ** 2

B. b = (a-) ** 2

C. b = (a) ** -2

D. b = -(a) ** 2

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #56

Topic 2

What is the expected output of the following code?

```
print(2 ** 3 ** 2 ** 1)
```

A. 64

B. 16

C. 128.0

D. The code is erroneous.

E. 16.0

F. 512

Correct Answer: *F*

Currently there are no comments in this discussion, be the first to comment!

Question #57 *Topic 2*

Which of the following code snippets will print True to the monitor? (Choose two.)

A.

```
print('is' in 'This IS Python code.')
```

B.

```
x = 42
y = 42
print(x is not y)
```

C.

```
x = 'Peter Wellert'
y = 'Peter Wellert'.lower()
print(x is y)
```

D.

```
print('t' in 'Peter')
```

E.

```
x = ['Peter', 'Paul', 'Mary']
y = ['Peter', 'Paul', 'Mary']
print(x is y)
```

Correct Answer: *AD*

Currently there are no comments in this discussion, be the first to comment!

Question #58 *Topic 2*

What value will be assigned to the x variable?

```
z = 2
y = 1
x = y < z or z > y and y > z or z < y
```

A. False

B. 1

C. 0

D. True

Correct Answer: *D*

⊟ **mldprasad** 1 month, 2 weeks ago

Selected Answer: D

True or True and False or False

Applying logical operators:
Evaluate and first (higher precedence):

True and False evaluates to False.
The expression now becomes:

True or False or False.
Evaluate or from left to right:

True or False evaluates to True.
True or False evaluates to True.
upvoted 1 times

⊟ **ellene** 6 months, 3 weeks ago

True and False

Since the AND operator requires both conditions to be True for the whole expression to be True, and one of them is False, the overall expression evaluates to False.

Therefore, the value assigned to
x
x will be False, making the correct answer C (False).

So, the provided answer key is incorrect; it should be C (False) instead of D (True).
upvoted 1 times

⊟ **ellene** 6 months, 3 weeks ago

Let's break down the expression:

x = y < z or (z > y and y > z) or z < y
Now, we evaluate the sub-expressions:
y < z evaluates to 1 < 2, which is True.
z > y evaluates to 2 > 1, which is True.
y > z evaluates to 1 > 2, which is False.
z < y evaluates to 2 < 1, which is False.
So, plugging these values into the expression, we get:

x = True or (True and False) or False
Evaluate the and expression:

True and False is False.
Now the expression becomes:

x = True or False or False
Finally, evaluate the or expressions:
True or False is True.
So, the value assigned to x will be True.

The correct answer is:

D. True
upvoted 1 times

Question #59 *Topic 2*

The ** operator ...

A. does not exist.

B. performs floating-point multiplication.

C. performs duplicated multiplication.

D. performs exponentiation.

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Question #60 *Topic 2*

Right-sided binding means that the following expression

```
1 ** 2 ** 3
```

will be evaluated:

A. from right to left

B. in random order

C. from left to right

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #61 *Topic 2*

What is the expected output of the following code?

```
print(type(1 / 2))
```

A. <type 'double'>

B. <type 'int'>

C. <type 'number'>

D. <type 'tuple'>

E. <type 'float'>

Correct Answer: *E*

Currently there are no comments in this discussion, be the first to comment!

Question #62

Topic 2

What is the expected output of the following code?

```python
x = 0
y = 1
x = x ^ y
y = x ^ y
y = x ^ y
print(x, y)
```

A. 0 1

B. 1 0

C. The code is erroneous.

D. 0 0

E. 1 1

Correct Answer: *E*

Currently there are no comments in this discussion, be the first to comment!

Question #63 *Topic 2*

What is the output of the following code?

```python
x = "2"
y = 2 * x
print(y)
```

A. 2x

B. 22

C. The program will cause an error.

D. 4

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #64 *Topic 2*

What is the output of the following snippet if the user enters two lines containing 3 and 6 respectively?

```
x = input()
y = int(input())

print(x * y)
```

A. 333333

B. 18

C. 36

D. 666

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #65 *Topic 2*

What is the expected output of the following code?

```
x = True
y = False
z = False

if x or y and z:
    print('TRUE')
else:
    print('FALSE')
```

A. The code is erroneous.

B. TRUE

C. FALSE

D. None of the above.

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #66

Topic 2

What is the expected output of the following code if the user enters 2 and 4?

```
x = float(input())
y = float(input())
print(y ** (1 / x))
```

A. 1.0

B. 4.0

C. 2.0

D. 0.0

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #67 *Topic 2*

What is the expected output of the following code?

```
x = 4.5
y = 2
print(x // y)
```

A. 2.5

B. 2

C. 2.25

D. 2.0

Correct Answer: *D*

Billybob0604 1 month, 2 weeks ago

The answer should be B

upvoted 2 times

Question #1 *Topic 3*

The digraph written as #! is used to:

A. tell an MS Windows OS how to execute the contents of a Python file.

B. tell a Unix or Unix-like OS how to execute the contents of a Python file.

C. create a docstring.

D. make a particular module entity a private one.

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #2 *Topic 3*

What are the four fundamental elements that make a language?

A. An alphabet, a lexis, a syntax, and semantics

B. An alphabet, morphology, phonetics, and semantics

C. An alphabet, a lexis, phonetics, and semantics

D. An alphabet, phonetics, phonology, and semantics

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #3 *Topic 3*

What do you call a tool that lets you launch your code step-by-step and inspect it at each moment of execution?

A. An editor

B. A debugger

C. A console

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #4 *Topic 3*

What will be the output of the following code snippet?

```
x = 1
y = 2
z = x
x = y
y = z
print(x, y)
```

A. 1 2

B. 2 2

C. 1 1

D. 2 1

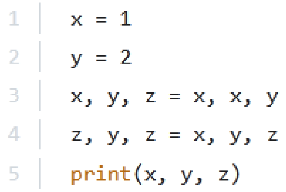
Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Question #5

Topic 3

What is the expected output of the following code?



```
x = 1
y = 2
x, y, z = x, x, y
z, y, z = x, y, z
print(x, y, z)
```

A. 1 2 1

B. 2 1 2

C. 1 1 2

D. 1 2 2

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #6 *Topic 3*

Python is an example of:

A. a machine language

B. a natural language

C. a high-level programming language

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #7 *Topic 3*

What is the expected output of the following code?

```
x = '\''
print(len(x))
```

A. 0

B. 1

C. The code is erroneous.

D. 2

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #8 *Topic 3*

Which of the following variable names are illegal? (Choose two.)

A. True

B. true

C. and

D. TRUE

Correct Answer: *AC*

Currently there are no comments in this discussion, be the first to comment!

Question #9 *Topic 3*

What is CPython?

A. It’s a programming language that is a superset of the C language, designed to produce Python-like performance with code written in C.

B. It's a default, reference implementation of the Python language, written in C.

C. It’s a programming language that is a superset of the Python, designed to produce C-like performance with code written in Python.

D. It's a default, reference implementation of the C language, written in Python.

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #10 *Topic 3*

Which of the following variable names is illegal?

A. In

B. IN

C. in_

D. in

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Question #11 *Topic 3*

What is the default value of encoding in the string function encode()?

A. qwerty

B. ascii

C. utf-16

D. utf-8

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Question #12 *Topic 3*

What is machine code?

A. A medium-level programming language consisting of the assembly code designed for the computer processor

B. A low-level programming language consisting of binary digits/bit that the computer reads and understands

C. A high-level programming language consisting of instruction lists that humans can read and understand

D. A low-level programming language consisting of hexadecimal digits that make up high-level language instructions

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #13 *Topic 3*

The pyc file contains ...

A. a Python interpreter.

B. Python source code.

C. compiled Python bytecode.

D. a Python compiler.

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #14 *Topic 3*

Select the true statements: (Choose two.)

A. You can use keywords as variable names in Python

B. You can use keywords as function names in Python

C. You cannot use keywords as variable names in Python

D. You cannot use keywords as function names in Python

Correct Answer: *CD*

Currently there are no comments in this discussion, be the first to comment!

Question #15 *Topic 3*

The escape character owes its name to the fact that it:

A. cannot be caught due to its high speed

B. escapes from source files into the computer memory

C. changes the meaning of the character next to it

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #16 *Topic 3*

What do you call a computer program which directly executes instructions written in a programming language?

A. A compiler

B. An interpreter

C. A translator

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #17 *Topic 3*

What do you call a command-line interpreter which lets you interact with your OS and execute Python commands and scripts?

A. A console

B. A compiler

C. Jython

D. An editor

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #18

Topic 3

A value returned by the input() function is:

A. an integer

B. a float

C. a string

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #19 *Topic 3*

Insert the correct piece of code, so that the program produces the expected output.

Expected output:

```
Andy
Brown
```

Code:

```
# insert code here
```

A. print(Andy\nBrown)

B. print("Andy Brown", end=" ")

C. print("Andy Brown", sep=" ")

D. print("Andy\nBrown")

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Question #20 *Topic 3*

The \n digraph forces the print() function to:

A. duplicate the character next to the digraph

B. stop its execution

C. break the output line

D. output exactly two characters: \ and n

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #21 *Topic 3*

What is true about compilation? (Choose two.)

A. The code is converted directly into machine code executable by the processor

B. It tends to be faster than interpretation

C. It tends to be slower than interpretation

D. Both you and the end user must have the code

Correct Answer: *AB*

Currently there are no comments in this discussion, be the first to comment!

Question #22 *Topic 3*

You execute the following command in the terminal.

python index.py Hello

You want the command to print out Hello.

What has to be inside of index.py?

A.

```
from sys import argv
print(argv[1])
```

B.

```
from sys import args
print(args[1])
```

C.

```
from sys import args
print(args[0])
```

D.

```
from sys import argv
print(argv[0])
```

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #23 *Topic 3*

The compiled Python bytecode is stored in files having names ending with:

A. py

B. pyb

C. pyc

D. pc

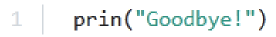
Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #24

Topic 3

What is the expected behavior of the following program?



```
prin("Goodbye!")
```

A. The program will generate an error message on the screen

B. The program will output "Goodbye!"

C. The program will output Goodbye!

D. The program will output ("Goodbye!")

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #25 *Topic 3*

You have the following file.

```
index.py:
from sys import argv
print(argv[1] + argv[2])
```

You run the file by executing the following command in the terminal. python index.py 42 3

What is the expected output?

A. 45

B. 424242

C. 423

D. 126

E. The code is erroneous.

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #26 *Topic 3*

You create a function to calculate the power of a number by using Python. You need to ensure that the function is documented with comments. You create the following code. Line numbers are included for reference only.

```
01 # The calc_power function calculates exponents
02 # x is the base
03 # y is the exponent
04 # The value of x raised to the y power is returned
05 def calc_power(x, y):
06     comment = "# Return the value"
07     return x ** y  # raise x to the y power
```

Which of the following statements are true? (Choose two.)

A. The string in Line 06 will be interpreted as a comment.

B. The pond sign (#) is optional for lines 02 and 03.

C. Line 07 contains an inline comment.

D. Lines 01 through 04 will be ignored for syntax checking.

Correct Answer: *CD*

Currently there are no comments in this discussion, be the first to comment!

Question #27

Topic 3

The print() function can output values of:

A. any number of arguments (including zero)

B. any number of arguments (excluding zero)

C. not more than five arguments

D. just one argument

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #28 *Topic 3*

What is the expected output of the following code?

```
x = '\\\'
print(len(x))
```

A. The code is erroneous.

B. 2

C. 3

D. 1

Correct Answer: *A*

5cd303a 1 month, 3 weeks ago

It is A --> erroneous.

if it is '\\' you have len(x) = 1
if it is '\\\' you have an error
if it is '\\\\' you have len(x) = 2
upvoted 2 times

ellene 6 months, 3 weeks ago

The code x = '\\\' is not erroneous. This is a valid Python string assignment. In Python, the backslash (\) is an escape character.

Here's how it works in this case:

The first backslash escapes the second backslash, resulting in a single backslash character.
The third backslash is not followed by any special character, so it remains as a backslash.
Therefore, the string x actually contains two backslashes: '\\'.

So, the length of the string x is 2.

Thus, the correct answer is:

B. 2
upvoted 1 times

Question #29 *Topic 3*

The print() function is an example of:

A. a Python built-in function

B. a user-defined function

C. an anonymous function

D. a Python output method

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #30 *Topic 3*

What is IDLE?

A. An acronym that stands for Integrated Development and Learning Environment for Python

B. An acronym that stands for Interactive Development and Learning Extension

C. A version of Python

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #31 *Topic 3*

You have the following file.

```
index.py:
from sys import argv
sum = 0
for i in range(2, len(argv)):
    sum += float(argv[i])
print(
    "The average score for {0} is {1:.2f}"
    .format(argv[1], sum/(len(argv)-2))
)
```

You want the following output: The average score for Peter is 200.00.

Which command do you have to execute in the command line?

A. python index.py Peter 100

B. python index.py Peter 100 200 300

C. The code is erroneous.

D. python index.py Peter 100 200

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #32 *Topic 3*

A keyword is a word: (Choose two.)

A. that cannot be used as a variable name

B. that cannot be used as a function name

C. is the most important word in the whole program

Correct Answer: *AB*

Currently there are no comments in this discussion, be the first to comment!

Question #33 *Topic 3*

What is CPython?

A. The default implementation of the Python programming language

B. Another name for CPython, a superset of the Python programming language

C. A compiled language used to perform high-level programming functions

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #34 *Topic 3*

ASCII is:

A. a character name

B. short for American Standard Code for Information Interchange

C. a predefined Python variable name

D. a standard Python module name

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #35 *Topic 3*

Which of the following variable names are illegal and will cause the SyntaxError exception? (Choose two.)

A. print

B. In

C. for

D. in

Correct Answer: *CD*

Currently there are no comments in this discussion, be the first to comment!

Question #36 *Topic 3*

UTF 8 is ...

A. the 9th version of the UTF Standard.

B. a synonym for "byte".

C. a Python version name.

D. an encoding form of the Unicode Standard.

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Question #37 *Topic 3*

A complete set of known commands is called:

A. a machine list

B. a low-level list

C. an instruction list

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #38 *Topic 3*

A code point is:

A. A point used to write a code.

B. A number which makes up a character.

C. A code containing a point.

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #39 *Topic 3*

How did Python, the programming language, get its name?

A. Guido van Roddum named it to honor Python of Catana, a dramatic poet of the time of Alexander the Grerat

B. Guido van Roddum named it to honor Monty Python's Flying Circus, a BBC comedy series popular in the 1970s

C. Guido van Roddum named it after the Pythonidae - a family of large, nonvenomous snakes

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #40 *Topic 3*

Which of the following variable names is illegal?

A. TRUE

B. True

C. tRUE

D. true

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #41 *Topic 3*

What is the best definition of a script?

A. It's a text file that contains sequences of zeroes and ones

B. It's a text file that contains instructions which make up a Python program

C. It's an error message generated by the interpreter

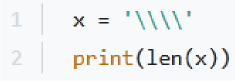
D. It's an error message generated by the compiler

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #42 *Topic 3*

What is the expected output of the following code?



```
x = '\\\\'
print(len(x))
```

A. 4

B. The code is erroneous.

C. 2

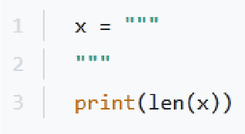
D. 1

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #43 *Topic 3*

What is the expected output of the following code?



```
x = """
"""
print(len(x))
```

A. 0

B. The code is erroneous.

C. 2

D. 1

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Question #44 *Topic 3*

What do you call a file containing a program written in a high-level programming language?

A. A code file

B. A target file

C. A machine file

D. A source file

Correct Answer: *D*

Currently there are no comments in this discussion, be the first to comment!

Question #45 *Topic 3*

Which one of the following is an example of a Python file extension?

A. pi

B. py

C. p

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!

Question #46 *Topic 3*

What is source code?

A. A program written in a high-level programming language

B. Another name for a source file

C. Machine code executed by computers

Correct Answer: *A*

Currently there are no comments in this discussion, be the first to comment!

Question #47 *Topic 3*

What is the expected output of the following code?

```
num = 2 + 3 * 5
print(Num)
```

A. 25

B. 17.0

C. The code is erroneous.

D. 17

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #48 Topic 3

The folder created by Python used to store pyc files is named:

A. __pyc__

B. __pycfiles__

C. __pycache__

D. __cache__

Correct Answer: *C*

Currently there are no comments in this discussion, be the first to comment!

Question #49 *Topic 3*

Select the true statements? (Choose two.)

A. Python is a good choice for low-level programming, e.g. when you want to implement an effective driver

B. Python is a good choice for creating and executing tests for applications

C. Python is free, open-source, and multiplatform

D. Python 3 is backwards compatible with Python 2

Correct Answer: *BC*

Currently there are no comments in this discussion, be the first to comment!

Question #50

Topic 3

You develop a Python application for your company. You want to add notes to your code so other team members will understand it.
What should you do?

A. Place the notes after the last list of code separated by a blank line.

B. Place the notes after the # sign on any line.

C. Place the notes inside of parentheses on any line.

D. Place the notes before the first line of code separated by a blank line.

Correct Answer: *B*

Currently there are no comments in this discussion, be the first to comment!