AICPA

PCEP-30-01

Certified Entry-Level Python Programmer

Question: 34

A function definition starts with the keyword:
A. def
B. function
C. fun

Answer: A

Explanation:

Topic: def

Try it yourself:

def my_first_function():

print('Hello')

my_first_function() # Hello

https://www.w3schools.com/python/python_functions.asp

Question: 35

Consider the following code snippet:

w = bool(23)

x = bool(")

y = bool(' ')

z = bool([False])

Which of the variables will contain False?
A. z
B. x
C. y
D. w

Answer: B

Explanation:

Topic: type casting with bool()

Try it yourself:

print(bool(23)) # True

print(bool(")) # False

print(bool(' ')) # True

print(bool([False])) # True

The list with the value False is not empty and therefore it becomes True

The string with the space also contain one character

and therefore it also becomes True

The values that become False in Python are the following:

print(bool(")) # False

print(bool(0)) # False

print(bool(0.0)) # False

print(bool(0j)) # False

print(bool(None)) # False

print(bool([])) # False

print(bool(())) # False

print(bool({})) # False

print(bool(set())) # False

print(bool(range(0))) # False

Question: 36

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. can be executed if and only if the tuple contains at least two elements
B. is illegal
C. may be illegal if the tuple contains strings
D. is fully correct

Answer: B

Explanation:

Topics: dictionary

Try it yourself:

my_tuple = (1, 2, 3)

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

TypeError: 'tuple' object does not support item assignment

A tuple is immutable and therefore you cannot

assign a new value to one of its indexes.

Question: 37

You develop a Python application for your company.

You have the following code.

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. None of the above.
D. (a + (b * c)) – d

Answer: D

Explanation:

Topics: addition operator multiplication operator

subtraction operator operator precedence

Try it yourself:

def main(a, b, c, d):

value = a + b * c – d # 3

value = (a + (b * c)) – d # 3

value = (a + b) * (c – d) # -3

value = a + ((b * c) – d) # 3

return value

print(main(1, 2, 3, 4)) # 3

This question is about operator precedence

The multiplication operator has the highest precedence and is therefore executed first.

That leaves the addition operator and the subtraction operator

They both are from the same group and therefore have the same precedence.

That group has a left-to-right associativity.

The addition operator is on the left and is therefore executed next.

And the last one to be executed is the subtraction operator

Question: 38

Which of the following variable names are illegal? (Select two answers)
A. TRUE
B. True
C. true
D. and

Answer: B, D

Explanation:

Topics: variable names keywords True and

Try it yourself:

TRUE = 23

true = 42

True = 7 # SyntaxError: cannot assign to True

and = 7   # SyntaxError: invalid syntax

You cannot use keywords as variable names.

Question: 39

Which of the following for loops would output the below number pattern?

11111

22222

33333

44444

55555
A. for i in range(0, 5):
print(str(i) * 5)
B. for i in range(1, 6):
print(str(i) * 5)
C. for i in range(1, 6):
print(i, i, i, i, i)
D. for i in range(1, 5):
print(str(i) * 5)

Answer: B

Explanation:

Topics: for range() str() multiply operator string concatenation

Try it yourself:

for i in range(1, 6):

print(str(i) * 5)

"""

11111

22222

33333

44444

55555

"""

print('————-')

for i in range(0, 5):

print(str(i) * 5)

"""

00000

11111

22222

33333

44444

"""

print('————-')

for i in range(1, 6):

print(i, i, i, i, i)

"""

1 1 1 1 1

2 2 2 2 2

3 3 3 3 3

4 4 4 4 4

5 5 5 5 5

"""

print('————-')

for i in range(1, 5):

print(str(i) * 5)

"""

11111

22222

33333

44444

"""

You need range (1, 6)

because the start value 1 is inclusive and the end value 6 is exclusive. To get the same numbers next to each other (without a space between them) you need to make a string and then use the multiply operator string concatenation

The standard separator of the print() function is one space. print(i, i, i, i, i) gives you one space between each number. It would work with print(i, i, i, i, i, sep='') but that answer is not offered here.

Question: 40

The digraph written as #! is used to:
A. tell a Unix or Unix-like OS how to execute the contents of a Python file.
B. create a docstring.
C. make a particular module entity a private one.
D. tell an MS Windows OS how to execute the contents of a Python file.

Answer: A

Explanation:

Topics: #! shebang

This is a general UNIX topic.

Best read about it here:

https://en.wikipedia.org/wiki/Shebang_(Unix)

*These questions are for demo purpose only. **Full version** is up to date and contains actual questions and answers.*

Killexams.com is an online platform that offers a wide range of services related to certification exam preparation. The platform provides actual questions, exam dumps, and practice tests to help individuals prepare for various certification exams with confidence. Here are some key features and services offered by Killexams.com:

<u>Actual Exam Questions</u>: Killexams.com provides actual exam questions that are experienced in test centers. These questions are updated regularly to ensure they are up-to-date and relevant to the latest exam syllabus. By studying these actual questions, candidates can familiarize themselves with the content and format of the real exam.

<u>Exam Dumps</u>: Killexams.com offers exam dumps in PDF format. These dumps contain a comprehensive collection of questions and answers that cover the exam topics. By using these dumps, candidates can enhance their knowledge and improve their chances of success in the certification exam.

<u>Practice Tests</u>: Killexams.com provides practice tests through their desktop VCE exam simulator and online test engine. These practice tests simulate the real exam environment and help candidates assess their readiness for the actual exam. The practice tests cover a wide range of questions and enable candidates to identify their strengths and weaknesses.

<u>Guaranteed Success</u>: Killexams.com offers a success guarantee with their exam dumps. They claim that by using their materials, candidates will pass their exams on the first attempt or they will refund the purchase price. This guarantee provides assurance and confidence to individuals preparing for certification exams.

<u>Updated Content:</u> Killexams.com regularly updates its question bank and exam dumps to ensure that they are current and reflect the latest changes in the exam syllabus. This helps candidates stay up-to-date with the exam content and increases their chances of success.

<u>Technical Support</u>: Killexams.com provides free 24x7 technical support to assist candidates with any queries or issues they may encounter while using their services. Their certified experts are available to provide guidance and help candidates throughout their exam preparation journey.