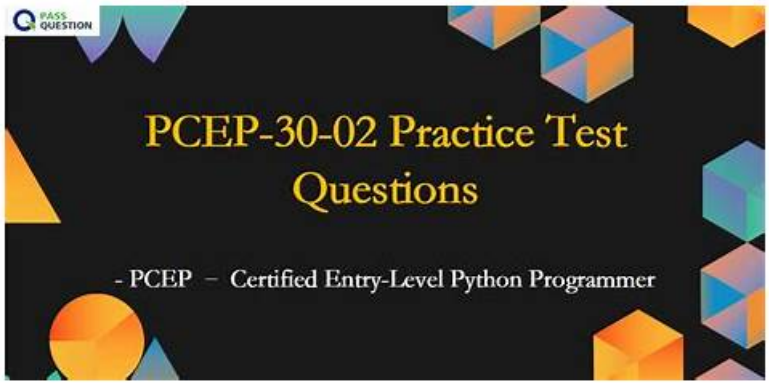
Python Institute PCEP-30-02 Best Preparation Materials - New PCEP-30-02 Test Online



BTW, DOWNLOAD part of PassTestking PCEP-30-02 dumps from Cloud Storage: https://drive.google.com/open?id=1sHLyLwScDuZrBxmy9DgKwgqde_4kNx0a

To get the PCEP-30-02 certification takes a certain amount of time and energy. Even for some exam like PCEP-30-02, the difficulty coefficient is high, the passing rate is extremely low, even for us to grasp the limited time to efficient learning. So how can you improve your learning efficiency? Here, I would like to introduce you to a very useful product, our PCEP-30-02 practice materials, through the information and data provided by it, you will be able to pass the PCEP-30-02 qualifying examination quickly and efficiently as the pass rate is high as 99% to 100%.

Python Institute PCEP-30-02 Exam Syllabus Topics:

Topic	Details
Topic 1	• Functions and Exceptions: This part of the exam covers the definition of function and invocation
Topic 2	• Control Flow: This section covers conditional statements such as if, if-else, if-elif, if-elif-else
Topic 3	• parameters, arguments, and scopes. It also covers Recursion, Exception hierarchy, Exception handling, etc.
Topic 4	• Data Collections: In this section, the focus is on list construction, indexing, slicing, methods, and comprehensions; it covers Tuples, Dictionaries, and Strings.

>> Python Institute PCEP-30-02 Best Preparation Materials <<

New PCEP-30-02 Test Online | Sample PCEP-30-02 Exam

we will provide you with the best Python Institute PCEP-30-02 exam dumps. You can pass the Python Institute PCEP-30-02 exam with high marks with the help of the Python Institute PCEP-30-02 exam questions. These Python Institute PCEP-30-02 exam practice questions are designed and verified by experienced and qualified PCEP-30-02 Exam Preparation trainers. They work together and put all their expertise and knowledge while verifying PCEP-30-02 exam questions all the time.

Python Institute PCEP - Certified Entry-Level Python Programmer Sample Questions (Q43-Q48):

NEW QUESTION # 43
Arrange the code boxes in the correct positions to form a conditional instruction which guarantees that a certain statement is executed when the temperature variable is equal to 0. 0.

Answer:

Explanation:
if temperature == 0.0:
Explanation:
* if
* temperature
* ==
* 0.0
* :
Arrange the boxes in this order:
This checks if temperature is exactly 0.0, and if so, runs the code inside the if block.

NEW QUESTION # 44
Which of the following expressions evaluate to a non-zero result? (Select two answers.)

- A. 4 / 2 * * 3 - 2
- B. 1 * 4 // 2 ** 3
- C. 2 ** 3 / A - 2
- D. 1 * * 3 / 4 - 1

Answer: A,C

Explanation:
Explanation
In Python, the ** operator is used for exponentiation, the / operator is used for floating-point division, and the // operator is used for integer division. The order of operations is parentheses, exponentiation, multiplication/division, and addition/subtraction. Therefore, the expressions can be evaluated as follows:
A). 2 ** 3 / A - 2 = 8 / A - 2 (assuming A is a variable that is not zero or undefined)
B). 4 / 2 * * 3 - 2 = 4 / 8 - 2 = 0.5 - 2 = -1.5 C. 1 * * 3 / 4 - 1 = 1 / 4 - 1 = 0.25 - 1 = -0.75 D. 1 * 4 // 2 ** 3 = 4 // 8 = 0 Only expressions A and B evaluate to non-zero results.

NEW QUESTION # 45
Which of the following expressions evaluate to a non-zero result? (Select two answers.)

- A. 4 / 2 * * 3 - 2
- B. 1 * 4 // 2 ** 3
- C. 2 ** 3 / A - 2
- D. 1 * * 3 / 4 - 1

Answer: A,C

Explanation:
In Python, the ** operator is used for exponentiation, the / operator is used for floating-point division, and the // operator is used for integer division. The order of operations is parentheses, exponentiation, multiplication /division, and addition/subtraction. Therefore, the expressions can be evaluated as follows:
A). 2 ** 3 / A - 2 = 8 / A - 2 (assuming A is a variable that is not zero or undefined) B. 4 / 2 * * 3 - 2 = 4 / 8 - 2 = 0.5 - 2 = -1.5 C. 1 * * 3 / 4 - 1 = 1 / 4 - 1 = 0.25 - 1 = -0.75 D. 1 * 4 // 2 ** 3 = 4 // 8 = 0 Only expressions A and B evaluate to non-zero results.
Reference: [Python Institute - Entry-Level Python Programmer Certification]

NEW QUESTION # 46
What is the expected result of the following code?

- A. 0
- B. The code is erroneous and cannot be run.
- C. 1

- D. 2

Answer: B

Explanation:
The code snippet that you have sent is trying to use the global keyword to access and modify a global variable inside a function. The code is as follows:
speed = 10 def velocity(): global speed speed = speed + 10 return speed print(velocity()) The code starts with creating a global variable called "speed" and assigning it the value 10. A global variable is a variable that is defined outside any function and can be accessed by any part of the code. Then, the code defines a function called "velocity" that takes no parameters and returns the value of "speed" after adding 10 to it. Inside the function, the code uses the global keyword to declare that it wants to use the global variable
"speed", not a local one. A local variable is a variable that is defined inside a function and can only be accessed by that function. The global keyword allows the function to modify the global variable, not just read it. Then, the code adds 10 to the value of "speed" and returns it. Finally, the code calls the function "velocity" and prints the result.
However, the code has a problem. The problem is that the code uses the global keyword inside the function, but not outside. The global keyword is only needed when you want to modify a global variable inside a function, not when you want to create or access it outside a function. If you use the global keyword outside a function, you will get a SyntaxError exception, which is an error that occurs when the code does not follow the rules of the Python language. The code does not handle the exception, and therefore it will terminate with an error message.
The expected result of the code is an unhandled exception, because the code uses the global keyword incorrectly. Therefore, the correct answer is A. The code is erroneous and cannot be run.
Reference: Python Global Keyword - W3SchoolsPython Exceptions: An Introduction - Real Python The code is erroneous because it is trying to call the "velocity" function without passing any parameter, which will raise a TypeError exception. The "velocity" function requires one parameter "x", which is used to calculate the return value of "speed" multiplied by "x". If no parameter is passed, the function will not know what value to use for "x".
The code is also erroneous because it is trying to use the "new_speed" variable before it is defined. The
"new_speed" variable is assigned the value of 20 after the first function call, but it is used as a parameter for the second function call, which will raise a NameError exception. The variable should be defined before it is used in any expression or function call.
Therefore, the code will not run and will not produce any output.
The correct way to write the code would be:
# Define the speed variable
speed = 10
# Define the velocity function
def velocity(x):
return speed * x
# Define the new_speed variable
new_speed = 20
# Call the velocity function with new_speed as a parameter
print(velocity(new_speed))
Copy
This code will print 200, which is the result of 10 multiplied by 20.
References:
[Python Programmer Certification (PCPP) - Level 1]
[Python Programmer Certification (PCPP) - Level 2]
[Python Programmer Certification (PCPP) - Level 3]
[Python: Built-in Exceptions]
[Python: Defining Functions]
[Python: More on Variables and Printing]

NEW QUESTION # 47
What is the expected output of the following code?

- A. The code produces no output.
- B. *
- C. * *
- D. * * *

Answer: C

Explanation:
The code snippet that you have sent is a conditional statement that checks if a variable "counter" is less than
0, greater than or equal to 42, or neither. The code is as follows:
if counter < 0: print("") elif counter >= 42: print("") else: print("") The code starts with checking if the value of "counter" is less than 0. If yes, it prints a single asterisk () to the screen and exits the statement. If no, it checks if the value of "counter" is greater than or equal to 42. If yes, it prints three asterisks () to the screen and exits the statement. If no, it prints two asterisks () to the screen and exits the statement.
The expected output of the code depends on the value of "counter". If the value of "counter" is 10, as shown in the image, the code will print two asterisks (**) to the screen, because 10 is neither less than 0 nor greater than or equal to 42. Therefore, the correct answer is C. * * Reference: [Python Institute - Entry-Level Python Programmer Certification]

NEW QUESTION # 48
.....

Success is has method. You can be successful as long as you make the right choices. PassTestking's Python Institute PCEP-30-02 exam training materials are tailored specifically for IT professionals. It can help you pass the exam successfully. If you're still catching your expertise to prepare for the exam, then you chose the wrong method. This is not only time-consuming and laborious, but also is likely to fail. But the remedy is not too late, go to buy PassTestking's Python Institute PCEP-30-02 Exam Training materials quickly. With it, you will get a different life. Remember, the fate is in your own hands.