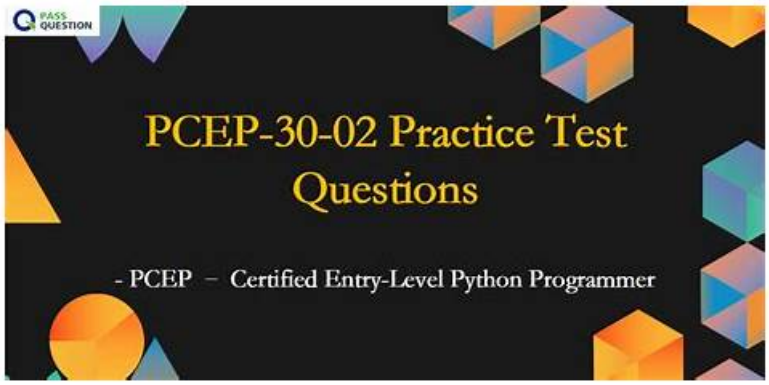
Latest Python Institute PCEP-30-02 Study Plan, Latest Test PCEP-30-02 Experience



P.S. Free 2026 Python Institute PCEP-30-02 dumps are available on Google Drive shared by ITPassLeader: https://drive.google.com/open?id=1PzGCXBtv550VLFxSWUdlcFHnI0E_H8wm

A whole new scope opens up to you and you are immediately hired by reputed firms. Even though the Python Institute PCEP-30-02 certification boosts your career options, you have to pass the PCEP-30-02 Exam. This Python Institute PCEP-30-02 exam serves to filter out the capable from incapable candidates.

Python Institute PCEP-30-02 Exam Syllabus Topics:

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

>> Latest Python Institute PCEP-30-02 Study Plan <<

Latest Test PCEP-30-02 Experience & PCEP-30-02 Valid Cram Materials

We are never complacent about our achievements, so all content of our PCEP-30-02 exam questions are strictly researched by proficient experts who absolutely in compliance with syllabus of this exam. Accompanied by tremendous and popular compliments around the world, to make your feel more comprehensible about the PCEP-30-02 study prep, all necessary questions of knowledge concerned with the exam are included into our PCEP-30-02 simulating exam.

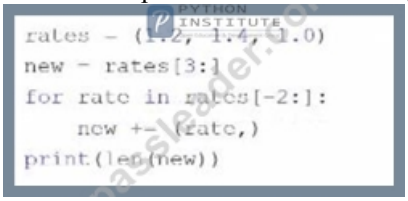
Python Institute PCEP - Certified Entry-Level Python Programmer Sample Questions (Q37-Q42):

NEW QUESTION # 37
A program written in a high-level programming language is called:

- A. machine code
- B. a source code
- C. the ASCI I code
- D. a binary code

Answer: B

NEW QUESTION # 38
What is the expected result of the following code?



- A. 0
- B. 1
- C. The code will cause an unhandled
- D. 2

Answer: C

Explanation:
Explanation
The code snippet that you have sent is trying to use a list comprehension to create a new list from an existing list. The code is as follows:
my_list = [1, 2, 3, 4, 5] new_list = [x for x in my_list if x > 5]
The code starts with creating a list called "my_list" that contains the numbers 1, 2, 3, 4, and 5. Then, it tries to create a new list called "new_list" by using a list comprehension. A list comprehension is a concise way of creating a new list from an existing list by applying some expression or condition to each element. The syntax of a list comprehension is:
new_list = [expression for element in old_list if condition]
The expression is the value that will be added to the new list, which can be the same as the element or a modified version of it. The element is the variable that takes each value from the old list. The condition is an optional filter that determines which elements will be included in the new list. For example, the following list comprehension creates a new list that contains the squares of the even numbers from the old list:
old_list = [1, 2, 3, 4, 5, 6] new_list = [x ** 2 for x in old_list if x % 2 == 0] new_list = [4, 16, 36]The code that you have sent is trying to create a new list that contains the elements from the old list that are greater than 5. However, there is a problem with this code. The problem is that none of the elements in the old list are greater than 5, so the condition is always false. This means that the new list will be empty, and the expression will never be evaluated. However, the expression is not valid, because it uses the variable x without defining it. This will cause a NameError exception, which is an error that occurs when a variable name is not found in the current scope. 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 tries to use an undefined variable in an expression that is never executed. Therefore, the correct answer is D. The code will cause an unhandled exception.

NEW QUESTION # 39
What is the expected output of the following code?

- A. 0
- B. 1
- C. 2
- D. The code raises an exception and outputs nothing.

Answer: D

Explanation:
Explanation
The code snippet that you have sent is trying to print the combined length of two lists, "collection" and "duplicate". The code is as follows:
collection = [] collection.append(1) collection.insert(0, 2) duplicate = collection duplicate.append(3) print(len(collection) + len(duplicate)) The code starts with creating an empty list called "collection" and appending the number 1 to it. The list now contains [1]. Then, the code inserts the number 2 at the beginning of the list. The list now contains [2, 1].
Then, the code creates a new list called "duplicate" and assigns it the value of "collection". However, this does not create a copy of the list, but rather a reference to the same list object. Therefore, any changes made to "duplicate" will also affect "collection", and vice versa. Then, the code appends the number 3 to "duplicate".

The list now contains [2, 1, 3], and so does "collection". Finally, the code tries to print the sum of the lengths of "collection" and "duplicate". However, this causes an exception, because the len function expects a single argument, not two. The code does not handle the exception, and therefore outputs nothing.
The expected output of the code is nothing, because the code raises an exception and terminates. Therefore, the correct answer is D. The code raises an exception and outputs nothing.

NEW QUESTION # 40
What happens when the user runs the following code?

- A. The program outputs five asterisks (*****) to the screen.
- B. The program outputs one asterisk (*) to the screen.
- C. The program enters an infinite loop.
- D. The program outputs three asterisks(***) to the screen.

Answer: B

NEW QUESTION # 41
Assuming that the following assignment has been successfully executed:
My_list - [1, 1, 2, 3]
Select the expressions which will not raise any exception.
(Select two expressions.)

- A. my_List- [0:1]
- B. my list [6]
- C. my_list[-10]
- D. my_list|my_Li1st | 3| I

Answer: A,D

Explanation:
The code snippet that you have sent is assigning a list of four numbers to a variable called "my_list". The code is as follows:
my_list = [1, 1, 2, 3]
The code creates a list object that contains the elements 1, 1, 2, and 3, and assigns it to the variable "my_list".
The list can be accessed by using the variable name or by using the index of the elements. The index starts from 0 for the first element and goes up to the length of the list minus one for the last element. The index can also be negative, in which case it counts from the end of the list. For example, my_list[0] returns 1, and my_list[-1] returns 3.
The code also allows some operations on the list, such as slicing, concatenation, repetition, and membership.
Slicing is used to get a sublist of the original list by specifying the start and end index. For example, my_list[1:
3] returns [1, 2]. Concatenation is used to join two lists together by using the + operator. For example, my_list
+ [4, 5] returns [1, 1, 2, 3, 4, 5]. Repetition is used to create a new list by repeating the original list a number of times by using the * operator. For example, my_list * 2 returns [1, 1, 2, 3, 1, 1, 2, 3]. Membership is used to check if an element is present in the list by using the in operator. For example, 2 in my_list returns True, and 4 in my_list returns False.
The expressions that you have given are trying to access or manipulate the list in different ways. Some of them are valid, and some of them are invalid and will raise an exception. An exception is an error that occurs when the code cannot be executed properly. The expressions are as follows:
A). my_list[-10]: This expression is trying to access the element at the index -10 of the list. However, the list only has four elements, so the index -10 is out of range. This will raise an IndexError exception and output nothing.
B). my_list|my_Li1st | 3| I: This expression is trying to perform a bitwise OR operation on the list and some other operands. The

bitwise OR operation is used to compare the binary representation of two numbers and return a new number that has a 1 in each bit position where either number has a 1. For example, 3 | 1 returns
3, because 3 in binary is 11 and 1 in binary is 01, and 11 | 01 is 11. However, the bitwise OR operation cannot be applied to a list, because a list is not a number. This will raise a TypeError exception and output nothing.
C). my list [6]: This expression is trying to access the element at the index 6 of the list. However, the list only has four elements, so the index 6 is out of range. This will raise an IndexError exception and output nothing.
D). my_List- [0:1]: This expression is trying to perform a subtraction operation on the list and a sublist. The subtraction operation is used to subtract one number from another and return the difference. For example, 3 -
1 returns 2. However, the subtraction operation cannot be applied to a list, because a list is not a number. This will raise a TypeError exception and output nothing.
Only two expressions will not raise any exception. They are:
B). my_list|my_Li1st | 3| I: This expression is not a valid Python code, but it is not an expression that tries to access or manipulate the list. It is just a string of characters that has no meaning. Therefore, it will not raise any exception, but it will also not output anything.
D). my_List- [0:1]: This expression is a valid Python code that uses the slicing operation to get a sublist of the list. The slicing operation does not raise any exception, even if the start or end index is out of range. It will just return an empty list or the closest possible sublist. For example, my_list[0:10] returns [1, 1, 2, 3], and my_list[10:20] returns []. The expression my_List- [0:1] returns the sublist of the list from the index 0 to the index 1, excluding the end index. Therefore, it returns [1]. This expression will not raise any exception, and it will output [1].
Therefore, the correct answers are B. my_list|my_Li1st | 3| I and D. my_List- [0:1].
Reference: [Python Institute - Entry-Level Python Programmer Certification]

NEW QUESTION # 42
.....

myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, wefunder.com, www.stes.tyc.edu.tw, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, Disposable vapes

DOWNLOAD the newest ITPassLeader PCEP-30-02 PDF dumps from Cloud Storage for free: https://drive.google.com/open?id=1PzGCXBtv550VLFxSWUdlcFHnI0E_H8wm