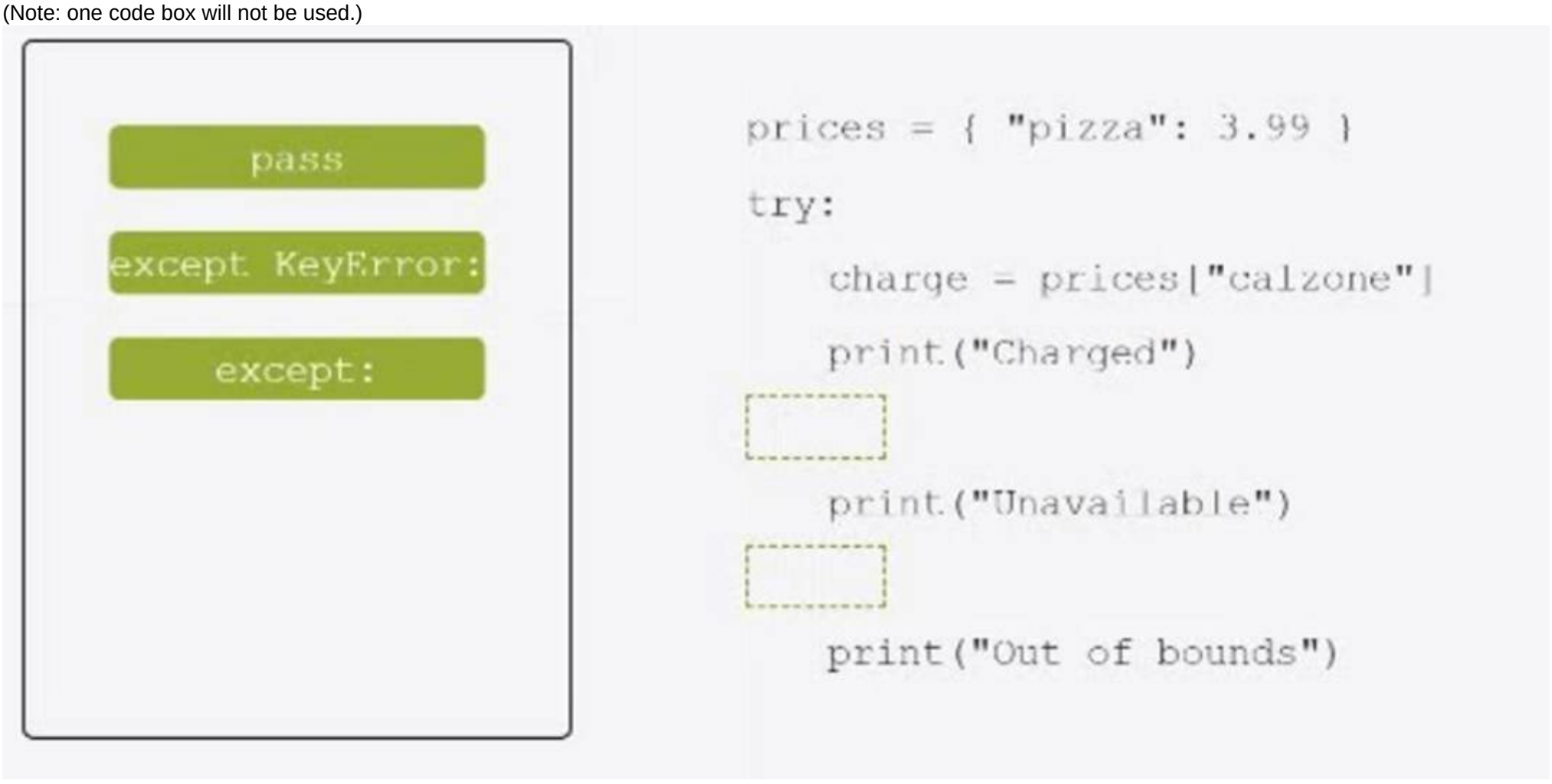
Python-Institute

Exam Questions PCEP-30-02

PCEP - Certified Entry-Level Python Programmer

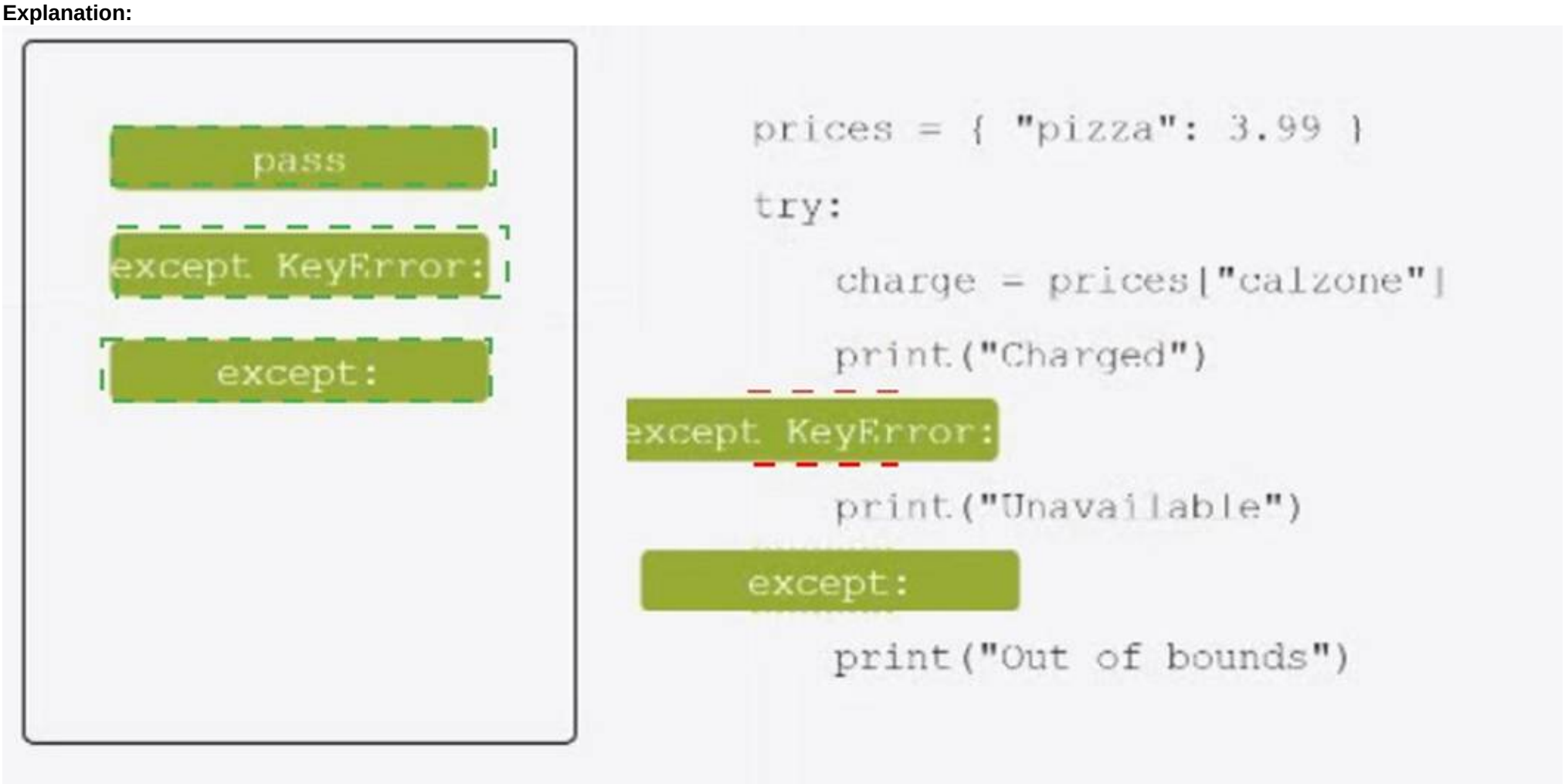
NEW QUESTION 1
DRAG DROP
Drag and drop the code boxes in order to build a program which prints Unavailable to the screen.
(Note: one code box will not be used.)



A. Mastered
B. Not Mastered

Answer: A

Explanation:



NEW QUESTION 2
What is the expected result of the following code?

```
rates = (1.2, 1.4, 1.0)
new = rates[3:]
for rate in rates[-2:]:
    new += (rate,)
print(len(new))
```

A. 5
B. 2
C. 1
D. The code will cause an unhandled

Answer: D

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.
Reference: Python - List Comprehension - W3SchoolsPython - List Comprehension -
GeeksforGeeksPython Exceptions: An Introduction – Real Python

NEW QUESTION 3
What happens when the user runs the following code?

```
total = 0
for i in range(4):
    if 2 * i < 4:
        total += 1
else:
    total += 1
print(total)
```

A. The code outputs 3.
B. The code outputs 2.
C. The code enters an infinite loop.
D. The code outputs 1.

Answer: B

Explanation:
The code snippet that you have sent is calculating the value of a variable ??total?? based on the values in the range of 0 to 3. The code is as follows:
total = 0 for i in range(0, 3): if i % 2 == 0: total = total + 1 else: total = total + 2 print(total) The code starts with assigning the value 0 to the variable ??total??. Then, it enters a for loop that iterates over the values 0, 1, and 2 (the range function excludes the upper bound). Inside the loop, the code checks if the current value of ??i?? is even or odd using the modulo operator (%). If ??i?? is even, the code adds 1 to the value of ??total??. If ??i?? is odd, the code adds 2 to the value of ??total??. The loop ends when ??i?? reaches 3, and the code prints the final
value of ??total?? to the screen.
The code outputs 2 to the screen, because the value of ??total?? changes as follows:
? When i = 0, total = 0 + 1 = 1
? When i = 1, total = 1 + 2 = 3
? When i = 2, total = 3 + 1 = 4
? When i = 3, the loop ends and total = 4 is printed Therefore, the correct answer is B. The code outputs 2.
Reference: [Python Institute - Entry-Level Python Programmer Certification]

NEW QUESTION 4
How many hashes (+) does the code output to the screen?

```
floor = 10
while floor != 0:
    floor //= 4
    print("#", end="")
else:
    print("#")
```

A. one

B. zero (the code outputs nothing)
C. five
D. three

Answer: C

Explanation:
The code snippet that you have sent is a loop that checks if a variable ??floor?? is less than or equal to 0 and prints a string accordingly. The code is as follows: floor = 5 while floor > 0: print(??+??) floor = floor - 1
The code starts with assigning the value 5 to the variable ??floor??. Then, it enters a while loop that repeats as long as the condition ??floor > 0?? is true. Inside the loop, the code prints a ??+?? symbol to the screen, and then subtracts 1 from the value of ??floor??. The loop ends when ??floor?? becomes 0 or negative, and the code exits.
The code outputs five ??+?? symbols to the screen, one for each iteration of the loop. Therefore, the correct answer is C. five.
Reference: [Python Institute - Entry-Level Python Programmer Certification]

NEW QUESTION 5
What is the expected output of the following code?

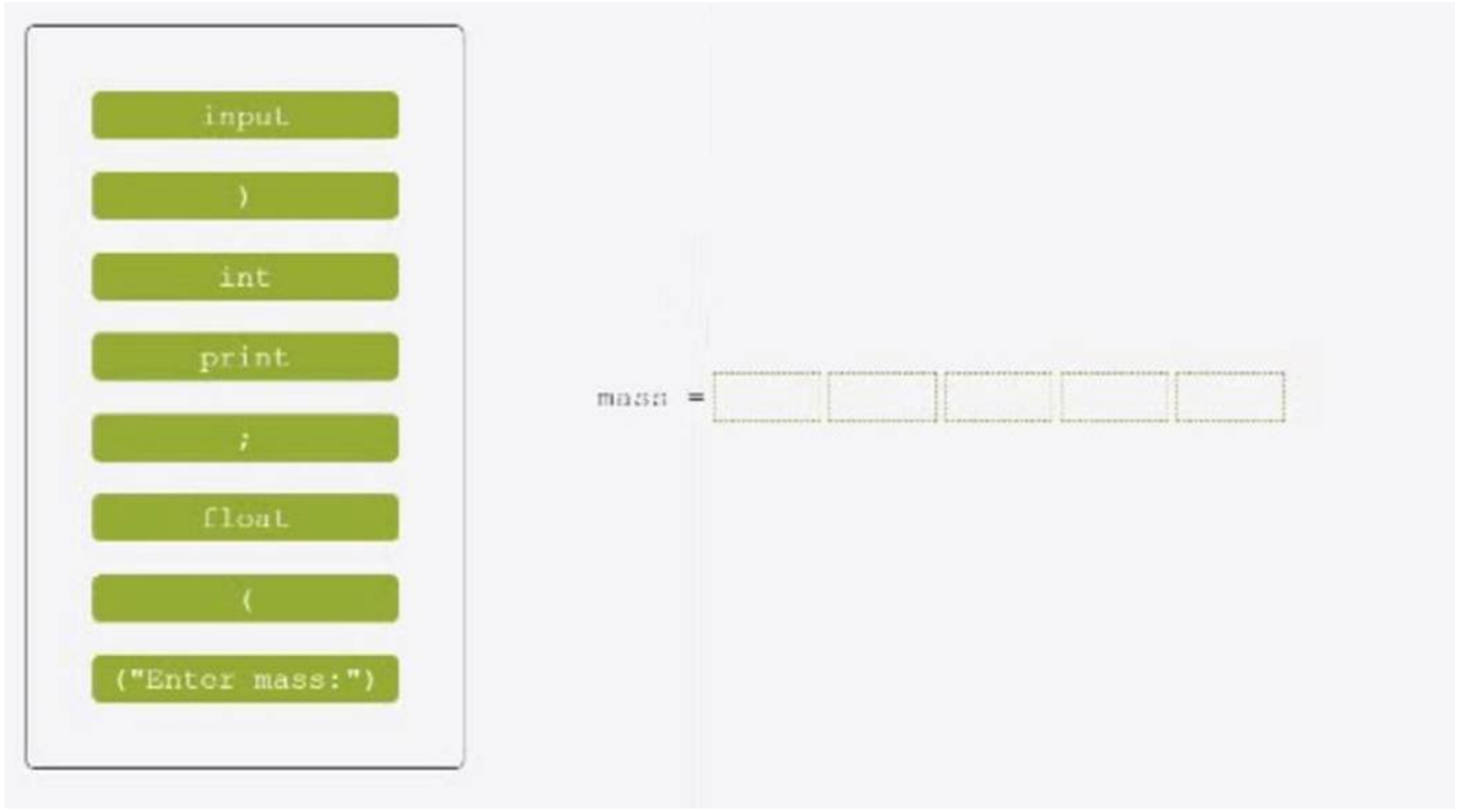
```
menu = {"pizza": 2.39, "pasta": 1.99, "folpetti": 3.99}

for value in menu:
    print(str(value)[0], end="")
```

A. The code is erroneous and cannot be run.
B. ppt
C. 213
D. pizzapastafolpetti

Answer: B

Explanation:
The code snippet that you have sent is using the slicing operation to get parts of a string and concatenate them together. The code is as follows:
pizza = ??pizza?? pasta = ??pasta?? folpetti = ??folpetti?? print(pizza[0] + pasta[0] + folpetti[0])
The code starts with assigning the strings ??pizza??, ??pasta??, and ??folpetti?? to the variables pizza, pasta, and folpetti respectively. Then, it uses the print function to display the result of concatenating the first characters of each string. The first character of a string can be accessed by using the index 0 inside square brackets. For example, pizza[0] returns ??p??. The concatenation operation is used to join two or more strings together by using the + operator. For example, ??a?? + ??b?? returns ??ab??. The code prints the result of pizza[0] + pasta[0] + folpetti[0], which is ??p?? + ??p?? + ??f??, which is ??ppt??.
The expected output of the code is ppt, because the code prints the first characters of each string. Therefore, the correct answer is B. ppt.
Reference: Python String Slicing - W3SchoolsPython String Concatenation - W3Schools

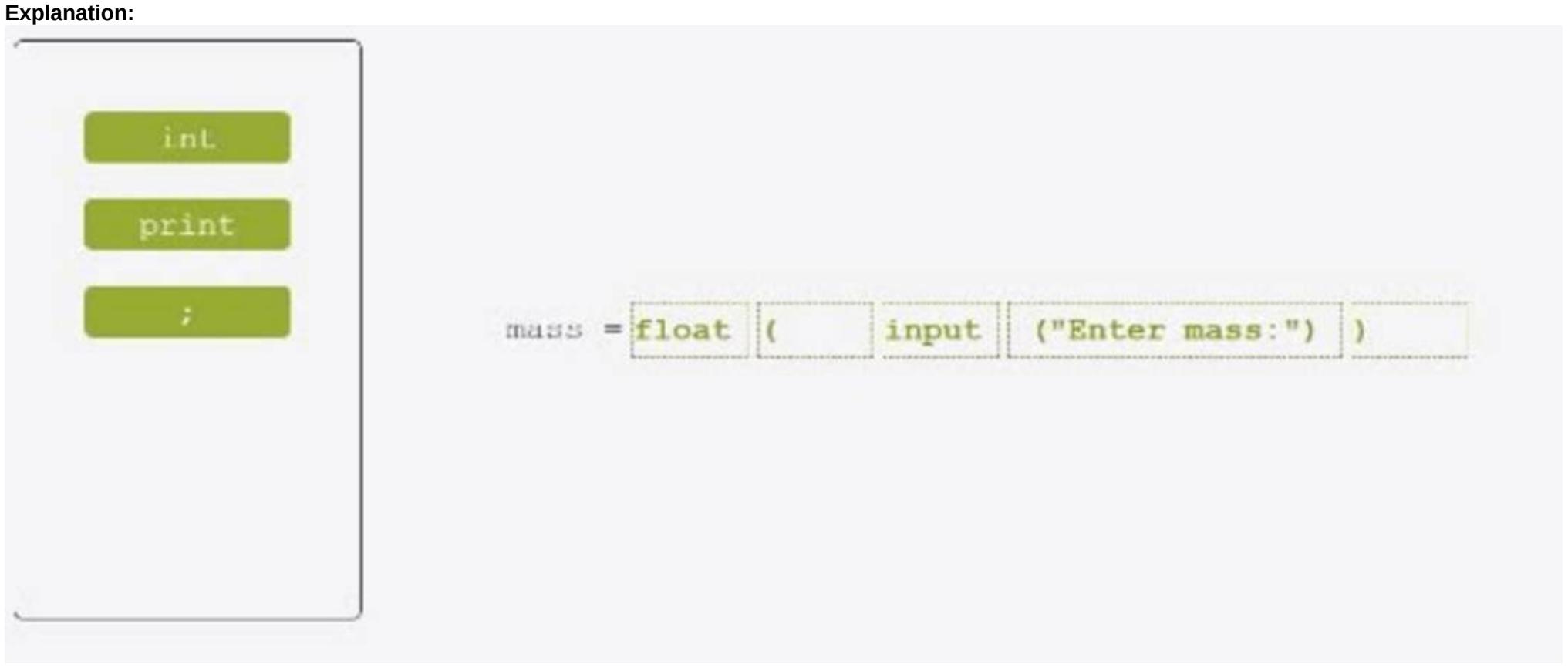
NEW QUESTION 6
DRAG DROP
Insert the code boxes in the correct positions in order to build a line of code which asks the user for a float value and assigns it to the mass variable.
(Note: some code boxes will not be used.)



A. Mastered
B. Not Mastered

Answer: A

Explanation:



One possible way to insert the code boxes in the correct positions in order to build a line of code that asks the user for a float value and assigns it to the mass variable is:
mass = float(input("Enter the mass: "))
This line of code uses the input function to prompt the user for a string value, and then uses the float function to convert that string value into a floating-point number. The result is then assigned to the variable mass.
You can find more information about the input and float functions in Python in the following references:
? [Python input() Function]
? [Python float() Function]

NEW QUESTION 7
Which of the following functions can be invoked with two arguments?
A)

```
def mu(Nonc):
    pass
```

B)

```
def iota(level, size = 0):
    pass
```

C)

```
def kappa(level):
    pass
```

D)

```
def lambda():
    pass
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: B

Explanation:
The code snippets that you have sent are defining four different functions in Python. A function is a block of code that performs a specific task and can be reused in the program. A function can take zero or more arguments, which are values that are passed to the function when it is called. A function can also return a value or None, which is the default return value in Python.
To define a function in Python, you use the def keyword, followed by the name of the function and parentheses. Inside the parentheses, you can specify the names of the parameters that the function will accept. After the parentheses, you use a colon and then indent the code block that contains the statements of the function. For example:
def function_name(parameter1, parameter2): # statements of the function return value To call a function in Python, you use the name of the function followed by parentheses.
Inside the parentheses, you can pass the values for the arguments that the function expects. The number and order of the arguments must match the number and order of the parameters in the function definition, unless you use keyword arguments or default values. For example:
function_name(argument1, argument2)
The code snippets that you have sent are as follows:
A) def my_function(): print(??Hello??)
B) def my_function(a, b): return a + b
C) def my_function(a, b, c): return a * b * c
D) def my_function(a, b=0): return a - b
The question is asking which of these functions can be invoked with two arguments. This means that the function must have two parameters in its definition, or one parameter with a default value and one without. The default value is a value that is assigned to a parameter if no argument is given for it when the function is called. For example, in option D, the parameter b has a default value of 0, so the function can be called with one or two arguments.
The only option that meets this criterion is option B. The function in option B has two parameters, a and b, and returns the sum of them. This function can be

invoked with two arguments, such as my_function(2, 3), which will return 5.
The other options cannot be invoked with two arguments. Option A has no parameters, so it can only be called with no arguments, such as my_function(), which will print ??Hello??. Option C has three parameters, a, b, and c, and returns the product of them. This function can only be called with three arguments, such as my_function(2, 3, 4), which will return 24. Option D has one parameter with a default value, b, and one without, a, and returns the difference of them. This function can be called with one or two arguments, such as my_function(2) or my_function(2, 3), which will return 2 or -1, respectively.
Therefore, the correct answer is B. Option B.

NEW QUESTION 8
What is true about tuples? (Select two answers.)

A. Tuples are immutable, which means that their contents cannot be changed during their lifetime.
B. The len { } function cannot be applied to tuples.
C. An empty tuple is written as { } .
D. Tuples can be indexed and sliced like lists.

Answer: AD

Explanation:
Tuples are one of the built-in data types in Python that are used to store collections of data. Tuples have some characteristics that distinguish them from other data types, such as lists, sets, and dictionaries. Some of these characteristics are:
? Tuples are immutable, which means that their contents cannot be changed during their lifetime. Once a tuple is created, it cannot be modified, added, or removed. This makes tuples more stable and reliable than mutable data types. However, this also means that tuples are less flexible and dynamic than mutable data types. For example, if you want to change an element in a tuple, you have to create a new tuple with the modified element and assign it to the same variable12
? Tuples are ordered, which means that the items in a tuple have a defined order and can be accessed by using their index. The index of a tuple starts from 0 for the first item and goes up to the length of the tuple minus one for the last item. The index can also be negative, in which case it counts from the end of the tuple. For example, if you have a tuple t = ("a", "b", "c"), then t[0] returns "a", and t[- 1] returns "c"12
? Tuples can be indexed and sliced like lists, which means that you can get a single item or a sublist of a tuple by using square brackets and specifying the start and end index. For example, if you have a tuple t = ("a", "b", "c", "d", "e"),
then t[2] returns "c", and t[1:4] returns ("b", "c", "d"). Slicing does not raise any exception, even if the start or end index is out of range. It will just return an empty tuple or the closest possible sublist12
? Tuples can contain any data type, such as strings, numbers, booleans, lists, sets,
dictionaries, or even other tuples. Tuples can also have duplicate values, which means that the same item can appear more than once in a tuple. For example, you can have a tuple t = (1, 2, 3, 1, 2), which contains two 1s and two 2s12
? Tuples are written with round brackets, which means that you have to enclose the
items in a tuple with parentheses. For example, you can create a tuple t = ("a", "b", "c") by using round brackets. However, you can also create a tuple without using round brackets, by just separating the items with commas. For example, you can create the same tuple t = "a", "b", "c" by using commas. This is called tuple packing, and it allows you to assign multiple values to a single variable12
? The len() function can be applied to tuples, which means that you can get the
number of items in a tuple by using the len() function. For example, if you have a tuple t = ("a", "b", "c"), then len(t) returns 312
? An empty tuple is written as (), which means that you have to use an empty pair of
parentheses to create a tuple with no items. For example, you can create an empty tuple t = () by using empty parentheses. However, if you want to create a tuple with only one item, you have to add a comma after the item, otherwise Python will not recognize it as a tuple. For example, you can create a tuple with one item t = ("a",) by using a comma12
Therefore, the correct answers are A. Tuples are immutable, which means that their contents cannot be changed during their lifetime. and D. Tuples can be indexed and sliced like lists.
Reference: Python Tuples - W3SchoolsTuples in Python - GeeksforGeeks

NEW QUESTION 10
.....

Thank You for Trying Our Product

We offer two products:

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questons and Answers in PDF Format

PCEP-30-02 Practice Exam Features:

* PCEP-30-02 Questions and Answers Updated Frequently

* PCEP-30-02 Practice Questions Verified by Expert Senior Certified Staff

* PCEP-30-02 Most Realistic Questions that Guarantee you a Pass on Your FirstTry

* PCEP-30-02 Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year