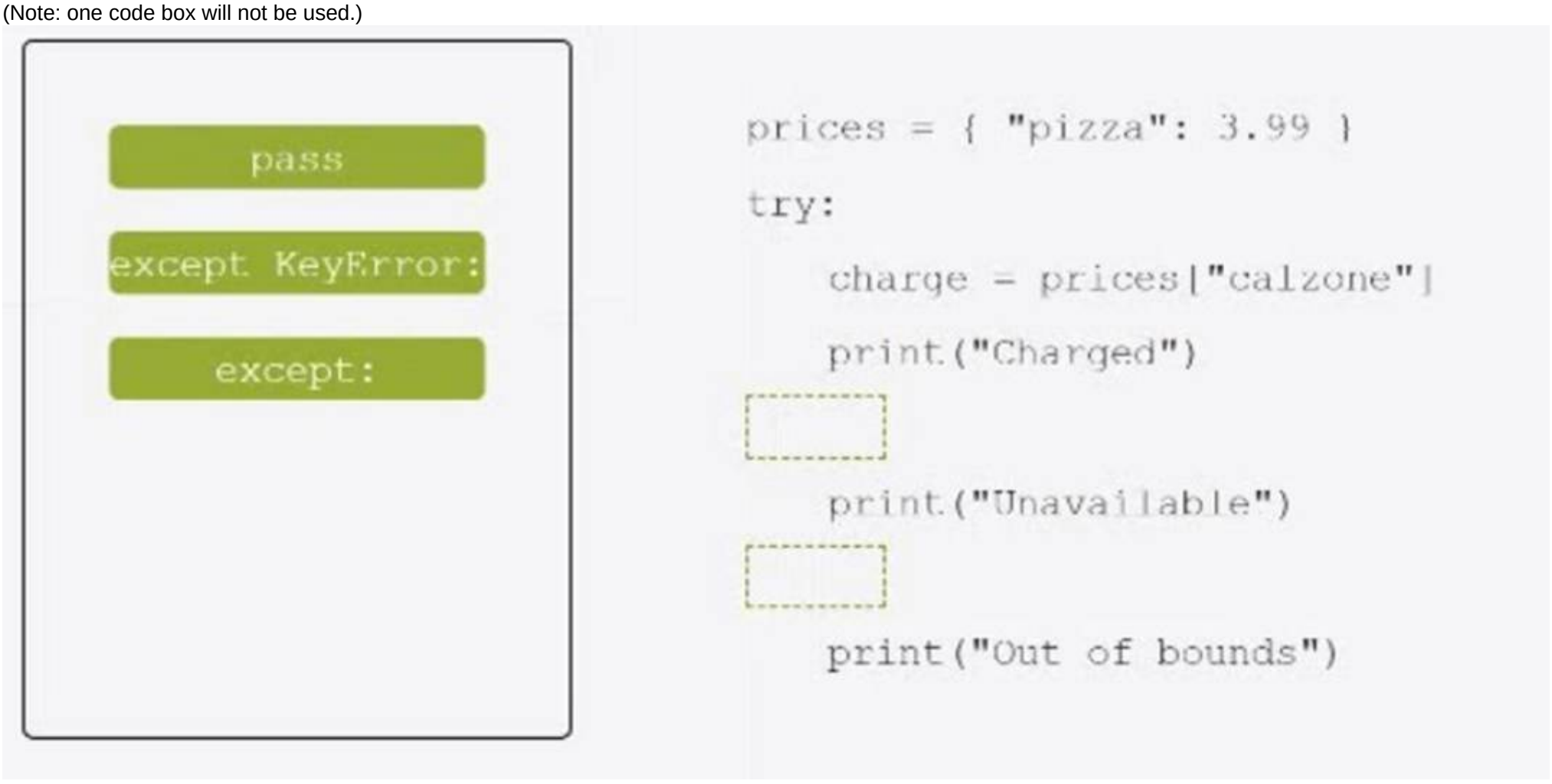
Python-Institute

Exam Questions PCEP-30-02

PCEP - Certified Entry-Level Python Programmer

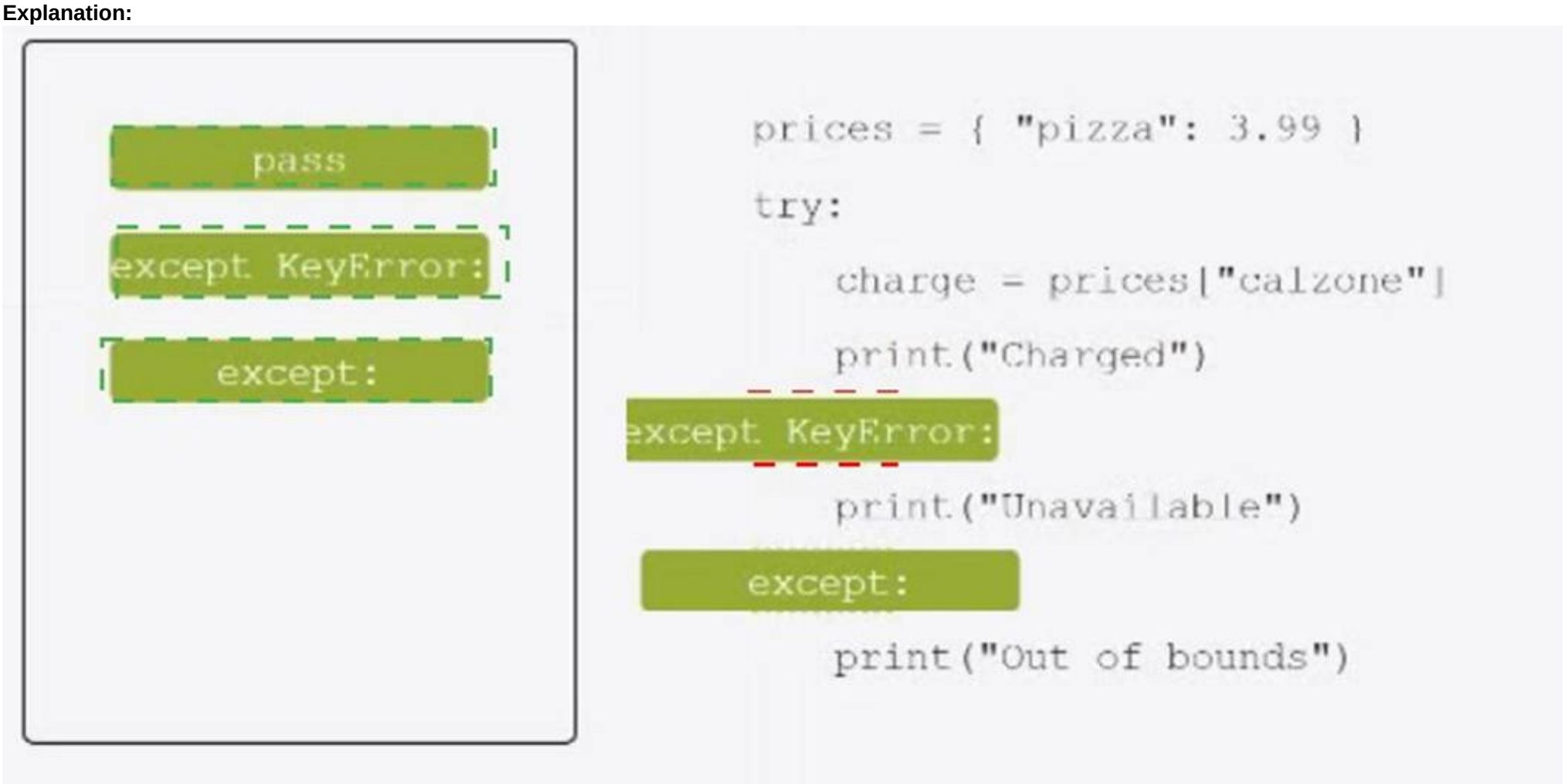
NEW QUESTION 1
DRAG DROP
Drag and drop the code boxes in order to build a program which prints Unavailable to the screen.
(Note: one code box will not be used.)



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?

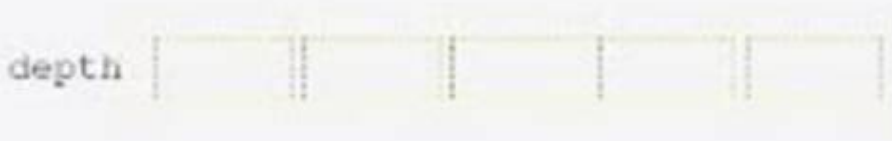
```
lolal = 0
for i in range(4):
    if 2 * i < 4:
        total += 1
else:
    total += 1
print(total)
```

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
DRAG DROP
Insert the code boxes in the correct positions in order to build a line of code which asks the user for an integer value and assigns it to the depth 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 which asks the user for an integer value and assigns it to the depth variable is:
depth = int(input("Enter the immersion depth: "))
This line of code uses the input function to prompt the user for a string value, and then uses the int function to convert that string value into an integer number. The result is then assigned to the variable depth.
You can find more information about the input and int functions in Python in the following references:
? [Python input() Function]
? [Python int() Function]

NEW QUESTION 5
What happens when the user runs the following code?

```
speed = 0
while speed < 30:
    speed *= 2
    if speed > 10:
        continue
    print("*", end="")
else:
    print("*")
```

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

Answer: D

Explanation:
The code snippet that you have sent is a while loop with an if statement and a print statement inside it. The code is as follows:
while True: if counter < 0: print(????) else: print(??**??)
The code starts with entering a while loop that repeats indefinitely, because the condition ??True?? is always true. Inside the loop, the code checks if the value of ??counter?? is less than 1. If yes, it prints a single asterisk () to the screen. If no, it prints three asterisks (**) to the screen. However, the code does not change the value of ??counter?? inside the loop, so the same condition is checked over and over again. The loop never ends, and the code enters an infinite loop.
The program outputs either one asterisk () or three asterisks (**) to the screen repeatedly, depending on the initial value of ??counter??. Therefore, the correct answer is D. The program enters an infinite loop.
Reference: [Python Institute - Entry-Level Python Programmer Certification]

NEW QUESTION 6
Which of the following are the names of Python passing argument styles? (Select two answers.)

A. keyword
B. reference
C. indicatory
D. positional

Answer: AD

Explanation:
Keyword arguments are arguments that are specified by using the name of the parameter, followed by an equal sign and the value of the argument. For example, print (sep='-', end='!') is a function call with keyword arguments. Keyword arguments can be used to pass arguments in any order, and to provide default values for some arguments1.
Positional arguments are arguments that are passed in the same order as the parameters of the function definition. For example, print ('Hello', 'World') is a function call with positional arguments. Positional arguments must be passed before any keyword arguments, and they must match the number and type of the parameters of the function2.
References: 1: 5 Types of Arguments in Python Function Definitions | Built In 2: python - What??s the pythonic way to pass arguments between functions ??

NEW QUESTION 7
What is the expected output of the following code?

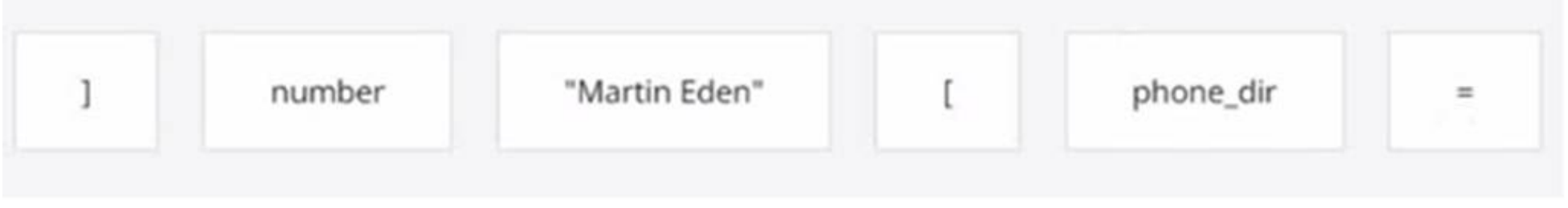
```
def traverse(stop):
    if stop == 0:
        return 0
    else:
        return stop * traverse(stop - 1)


print(traverse(2))
```

A. 2
B. 3
C. 1

Answer: D

Explanation:
The code snippet that you have sent is using the count method to count the number of occurrences of a value in a list. The code is as follows:
my_list = [1, 2, 3, 4, 5] print(my_list.count(1))
The code starts with creating a list called ??my_list?? that contains the numbers 1, 2, 3, 4, and 5. Then, it uses the print function to display the result of calling the count method on the list with the argument 1. The count method is used to return the number of times a value appears in a list. For example, my_list.count(1) returns 1, because 1 appears once in the list.
The expected output of the code is 1, because the code prints the number of occurrences of 1 in the list. Therefore, the correct answer is D. 1.
Reference: Python List count() Method - W3Schools

NEW QUESTION 8
DRAG DROP
Assuming that the phonc_dir dictionary contains namemumber pairs, arrange the code boxes to create a valid line of code which retrieves Martin Eden's phone number, and assigns it to the number variable.



A. Mastered
B. Not Mastered

Answer: A

Explanation:
number = phone_dir["Martin Eden"]
This code uses the square brackets notation to access the value associated with the key ??Martin Eden?? in the phone_dir dictionary. The value is then assigned to the variable number. A dictionary is a data structure that stores key-value pairs, where each key is unique and can be used to retrieve its corresponding value.
You can find more information about dictionaries in Python in the following references:
? [Python Dictionaries - W3Schools]
? [Python Dictionary (With Examples) - Programiz]
? [5.5. Dictionaries — How to Think Like a Computer Scientist ??]

NEW QUESTION 9
How many hashes (+) does the code output to the screen?

```
floor = 10
while floor != 0:
    floor //= 1
    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 10
What is the expected result of the following code?

```
def velocity(x=10):
    return speed + x

speed = 10
new_speed = velocity()
new_speed = velocity(new_speed)
print(new_speed)
```

A. The code is erroneous and cannot be run.
B. 20
C. 10
D. 30

Answer: A

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 10
What is the expected output of the following code?

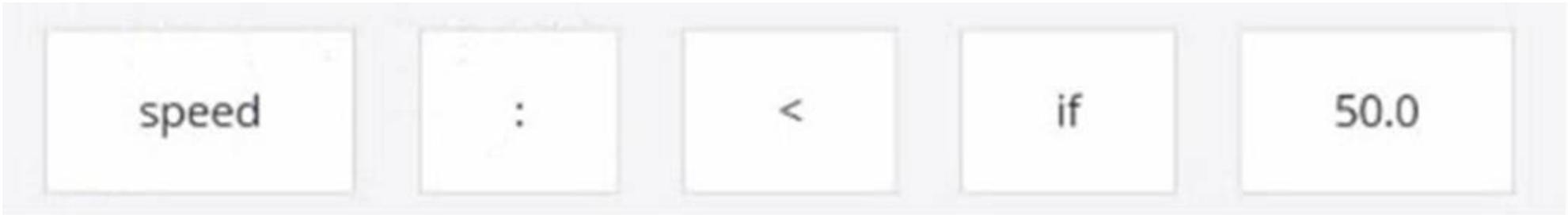
```
menu = {"pizza": 2.39, "pasta": 1.99, "folpetti": 3.99}

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 15
DRAG DROP
Arrange the code boxes in the correct positions to form a conditional instruction which guarantees that a certain statement is executed when the speed variable is less than 50.0.

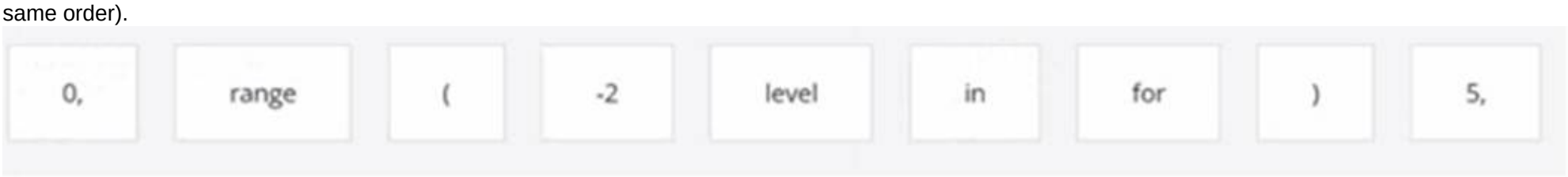


A. Mastered
B. Not Mastered

Answer: A

Explanation:
One possible way to arrange the code boxes in the correct positions to form a conditional instruction which guarantees that a certain statement is executed when the speed variable is less than 50.0 is: if speed < 50.0: print("The speed is low.")
This code uses the if keyword to create a conditional statement that checks the value of the variable speed. If the value is less than 50.0, then the code will print ??The speed is low.?? to the screen. The print function is used to display the output. The code is indented to show the block of code that belongs to the if condition.
You can find more information about the if statement and the print function in Python in the following references:
? Python If ?? Else
? Python Print Function

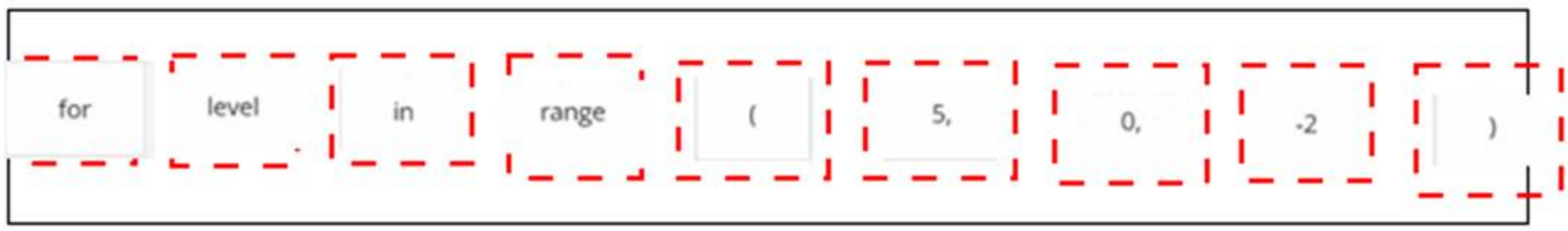
NEW QUESTION 19
DRAG DROP
Arrange the code boxes in the correct positions in order to obtain a loop which executes its body with the level variable going through values 5, 1, and 1 (in the same order).



A. Mastered
B. Not Mastered

Answer: A

Explanation:



NEW QUESTION 20
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 22
What is the expected result of running the following code?

```
def do_the_mess(parameter):
    parameter[0] += variable
    return parameter[0]



the_list = [x for x in range(2, 3)]
variable = -1
do_the_mess(the_list)
print(the_list[0])
```

A. The code prints 1 .
B. The code prints 2
C. The code raises an unhandled exception.
D. The code prints 0

Answer: C

Explanation:
The code snippet that you have sent is trying to use the index method to find the position of a value in a list. The code is as follows:
the_list = [1, 2, 3, 4, 5] print(the_list.index(6))
The code starts with creating a list called ??the_list?? that contains the numbers 1, 2, 3, 4, and 5. Then, it tries to print the result of calling the index method on the list with the argument 6. The index method is used to return the first occurrence of a value in a list. For example, the_list.index(1) returns 0, because 1 is the first value in the list.
However, the code has a problem. The problem is that the value 6 is not present in the list, so the index method cannot find it. This will cause a ValueError exception, which is an error that occurs when a function or operation receives an argument that has the right type but an inappropriate value. 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 find a value that does not exist in the list. Therefore, the correct answer is C. The code raises an unhandled exception.
Reference: Python List index() Method - W3SchoolsPython Exceptions: An Introduction – Real Python

NEW QUESTION 23
.....

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