

DUMPSQUEEN

Certified Associate in Python Programming

Python Institute PCAP-31-03

Version Demo

Total Demo Questions: 10

Total Premium Questions: 144

Buy Premium PDF

<https://dumpsqueen.com>

support@dumpsqueen.com

dumpsqueen.com

QUESTION NO: 1

What is the expected behavior of the following snippet?

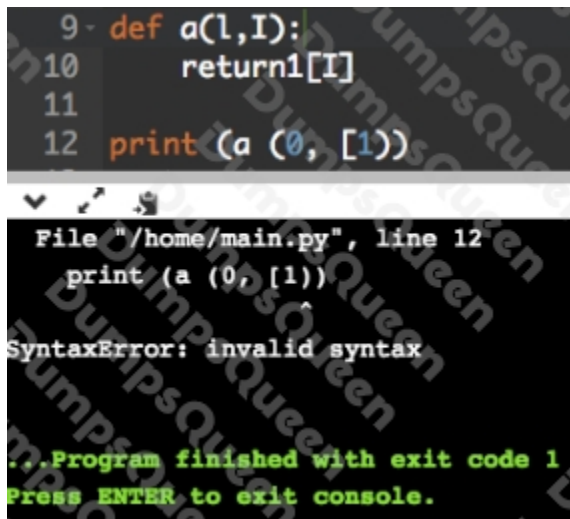
```
def a (l, I) :  
    return l [I]  
  
print (a (0, [1] )
```

It will:

- A. cause a runtime exception
- B. print 1
- C. print 0 , [1]
- D. print [1]

ANSWER: A

Explanation:



```
9 def a(l,I):  
10     return l[I]  
11  
12 print (a (0, [1])  
  
File "/home/main.py", line 12  
    print (a (0, [1])  
SyntaxError: invalid syntax  
  
...Program finished with exit code 1  
Press ENTER to exit console.
```

QUESTION NO: 2

What is the expected output of the following code?

```
def foo(x,y,z):
```

```
return x(y) - x(z)
```

```
print(f00(lambda x: x % 2, 2, 1))
```

- A. 1
- B. 0
- C. -1
- D. an exception is raised

ANSWER: C

QUESTION NO: 3

There is a stream named `s` open for writing. What option will you select to write a line to the stream"

- A. `s.write("Hello\n")`
- B. `write(s, "Hello")`
- C. `s.writel("Hello")`
- D. `s.writeline("Hello")`

ANSWER: A

Explanation:

Reference: https://en.wikibooks.org/wiki/Python_Programming/Input_and_Output

QUESTION NO: 4

Which of the following lines of code will work flawlessly when put independently inside the `add_new()` method in order to make the snippet's output equal to `[0, 1, 1]`? (Select two answers)

```
class MyClass:
    def __init__(self, initial):
        self.store = initial

    def put(self, new):
        self.store.append(new)

    def get(self):
        return self.store

    def dup(self):
        # insert the line of code here

Object = MyClass([0])
Object.put(1)
Object.dup()
print(Object.get())
```

- A. put self.store(1)
- B. self.put store(1)
- C. self .put self.get () [-1]
- D. self .put (self.store[1])

ANSWER: B D

QUESTION NO: 5

What is true about Python class constructors? (Select two answers)

- A. the constructor's first parameter identifies an object currently being created
- B. the constructor cannot use the default values of the parameters
- C. the constructor can be invoked directly under strictly defined circumstances
- D. super-class constructor is invoked implicitly during constructor execution

ANSWER: A C

QUESTION NO: 6

A two-parameter lambda function raising its first parameter to the power of the second parameter should be declared as:

- A. lambda (x, y) = x ** y

- B. lambda (x, y): x ** y
- C. def lambda (x, y): return x ** y
- D. lambda x, y: x ** y

ANSWER: D

QUESTION NO: 7

Assuming that the following code has been executed successfully, selected the expression which evaluate to True (Select two answers)

```
def f(x, y):  
    nom, denom = x, y  
    def g():  
        return nom / denom  
    return g  
  
a = f(1, 2)  
b = f(3, 4)
```

- A. a() == 4
- B. a is not None
- C. b() == 4
- D. a != b

ANSWER: A B

QUESTION NO: 8

What is true about Python packages? (Select two answers)

- A. the sys.path variable is a list of strings
- B. `__pycache__` is a folder that stores semi-completed Python modules

- C. a package contents can be stored and distributed as an mp3 file
- D. a code designed to initialize a package's state should be placed inside a file named init.py

ANSWER: A B

QUESTION NO: 9

If you want to transform a string into a list of words, what invocation would you use? (Select two answers)

Expected output:

```
The, Catcher, in, the Rye,
```

Code:

```
S = "The Catcher in the Rye"  
l = # put a proper invocation here  
For w in l:  
    Print (w, end=',') # outputs: The, Catcher, in, the Rye,
```

- A. s.split ()
- B. split (s, "" "")
- C. s.split (" """)
- D. split (s)

ANSWER: A C

QUESTION NO: 10

What is the expected output of the following code?

```
str = 'abcdef'  
def fun (s) :  
    del s [2]  
    return s  
  
print (fun (str) )
```

- A. abcef
- B. The program will cause a runtime exception error
- C. acdef
- D. abdef

Answer: B

Explanation:

```
9 str='abcdef'
10 def fun(s):
11     del s[2]
12     return s
13
14 print(fun(str))
```

Traceback (most recent call last):
File "/home/main.py", line 14, in <module>
 print(fun(str))
File "/home/main.py", line 11, in fun
 del s[2]
TypeError: 'str' object doesn't support item deletion

...Program finished with exit code 1
Press ENTER to exit console.

- A. The program will cause a runtime exception error
- B. acdef
- C. abdef

ANSWER: A

Explanation:

```
9 str='abcdef'
10 def fun(s):
11     del s[2]
12     return s
13
14 print(fun(str))
```

```
Traceback (most recent call last):
  File "/home/main.py", line 14, in <module>
    print(fun(str))
  File "/home/main.py", line 11, in fun
    del s[2]
TypeError: 'str' object doesn't support item deletion

...Program finished with exit code 1
Press ENTER to exit console.
```