

Topic 1 - Exam A

Question #1

Topic 1

What is true about Python packages? (Choose two.)

- A. a code designed to initialize a package's state should be placed inside a file named init.py
- B. a package's contents can be stored and distributed as an mp3 file
- C. `__pycache__` is a folder that stores semi-compiled Python modules
- D. the `sys.path` variable is a list of strings

Correct Answer: AC

- DKAT2023** 1 month, 2 weeks ago
sorry, only C and D are corrects , for A the correct file name is `_init_.py` not `init.py`
upvoted 2 times
- DKAT2023** 1 month, 3 weeks ago
A, C and D
upvoted 1 times
- Damon54** 1 month, 3 weeks ago
PASS Exam Today ! The contents and concepts are contained in these 40 and 141 questions PCAP , but there are several questions taken up 30% with modifications, so be careful, it is necessary to study!!!
There are no syntax errors in the exam questions
upvoted 1 times
- Damon54** 2 months, 2 weeks ago
A C e D are correct !!
upvoted 1 times
- surendrapattipati** 2 months, 3 weeks ago
A. a code designed to initialize a package's state should be placed inside a file named `init.py`
upvoted 1 times

What is the expected output of the following code?

```
import sys
import math
b1 = type(dir(math)) is list
b2 = type(sys.path) is list
print(b1 and b2)
```

- A. None
- B. True
- C. 0
- D. False

Correct Answer: D

Community vote distribution

B (100%)

 **earawiran** Highly Voted 3 months ago

Selected Answer: B

```
import sys
import math
b1 = type(dir(math)) is list
b2 = type(sys.path) is list
print(b1 and b2)
```

True

upvoted 5 times

 **Dave304409** Most Recent 1 month, 2 weeks ago

Selected Answer: B

is correct

upvoted 1 times

 **DKAT2023** 1 month, 3 weeks ago

B True

upvoted 1 times

 **a4129fd** 2 months, 3 weeks ago

B is the right one: True

upvoted 2 times

 **Nikhil_Durgesh** 2 months, 4 weeks ago

Correct Answer : B

```
import sys
import math
b1 = type(dir(math)) is list
b2 = type(sys.path) is list
print(b1 and b2)
```

Output :

True

upvoted 4 times

A Python package named pypack includes a module named pymod.py which contains a function named pyfun(). Which of the following snippets will let you invoke the function? (Choose two.)

- A. `from pypack.pymod import pyfun`
`pyfun()`
- B. `import pypack`
`pymod.pyfun()`
- C. `from pypack import *`
`pyfun()`
- D. `import pypack`
`import pypack.pymod`
`pypack.pymod.pyfun()`

Correct Answer: AD

Community vote distribution

AD (100%)

  **Dave304409** 1 month, 2 weeks ago

Selected Answer: AD

is correct

upvoted 1 times

  **DKAT2023** 1 month, 3 weeks ago

A and D

upvoted 1 times

Assuming that the code below has been executed successfully, which of the following expressions will always evaluate to True? (Choose two.)

```
import random v1 = random.random() v2 = random.random()
```

- A. `len(random.sample([1,2,3],1)) > 2`
- B. `v1 == v2`
- C. `random.choice([1,2,3]) > 0`
- D. `v1 > 1`

Correct Answer: BC

DKAT2023 1 month, 3 weeks ago

C is the correct
upvoted 1 times

Nikhil_Durgesh 2 months, 3 weeks ago

Option C is correct
Explanation:
`import random`
`v1 = random.random()`
`v2 = random.random()`
`random.choice([1,2,3]) > 0` will return TRUE.`

All others options will return output as FALSE.
upvoted 2 times

narayanan010 3 months ago

Only option C, ``random.choice([1,2,3]) > 0` will return TRUE.`
All other options are FALSE.
upvoted 3 times

With regards to the directory structure below, select the proper forms of the directives in order to import module_c. (Choose two.)

```
pypack (dir)
├── upper (dir)
│   ├── lower (dir)
│   │   └── module_c.py (file)
│   └── module_b.py (file)
└── module_a.py (file)
```

- A. from pypack.upper.lower import module_c
- B. import pypack.upper.lower.module_c
- C. import upper.module_c
- D. import upper.lower.module_c

Correct Answer: AD

Community vote distribution

AB (100%)

Dave304409 1 month, 2 weeks ago

Selected Answer: AB

is correct

upvoted 1 times

DKAT2023 1 month, 3 weeks ago

A and B

upvoted 1 times

herrmann69 2 months, 2 weeks ago

Selected Answer: AB

A and B are correct.

from pypack.upper.lower import module_c

import import pypack.upper.lower.module_c

C is missing "lower" between "upper" and "module_c"

D is missing "pypack" as the parent of "upper"

upvoted 4 times

Which one of the platform module functions should be used to determine the underlying platform name?

- A. platform.processor()
- B. platform.uname()
- C. platform.python_version()
- D. platform.platform()

Correct Answer: D

Community vote distribution

D (100%)

  **Dave304409** 1 month, 2 weeks ago

Selected Answer: D

is correct

upvoted 1 times

  **DKAT2023** 1 month, 3 weeks ago

D is the correct

upvoted 1 times

  **herrmann69** 2 months, 2 weeks ago

Selected Answer: D

Answer D is correct. platform.platform()

code:

```
import platform
```

```
print(platform.processor())
print(platform.uname())
print(platform.python_version())
print(platform.platform())
```

output:

```
Intel64 Family 6 Model 140 Stepping 1, GenuineIntel
uname_result(system='Windows', node='xxx', release='11', version='10.0.22621', machine='AMD64')
3.12.3
Windows-11-10.0.22621-SP0
```

upvoted 3 times

What is the expected behavior of the following code?

```
s = '2A'

try:
    n = int(s)
except ValueError:
    n = 2
except ArithmeticError:
    n = 1
except:
    n = 0

print(n)
```

- A. the code is erroneous and it will not execute
- B. it outputs 1
- C. it outputs 2
- D. it outputs 0

Correct Answer: C

Community vote distribution

C (100%)

 **kino_1994** 16 hours, 8 minutes ago

Are you blind? s is lowercase, not uppercase, throughout the entire code snippet. Therefore, the output is 2.

Try yourself:

```
s = '2A'
try:
    n = int(s)
except ValueError:
    n = 2
except ArithmeticError:
    n = 1
except:
    n = 0
```

print(n)

upvoted 1 times

 **eskimolight** 3 weeks, 6 days ago

Analysis

Variable Name Case Sensitivity:

The variable S is defined as '2A', but within the try block, s (lowercase) is used. Since Python is case-sensitive, s is not defined, and this will raise a NameError.

Error Handling:

The NameError is not explicitly caught by any of the except blocks (ValueError, ArithmeticError). Therefore, the generic except block will catch it. Output:

Due to the NameError, the generic except block will execute, and n will be assigned the value 0.

Given this information, the expected behavior of the code is:

D. it outputs 0

This is because the NameError caused by the case mismatch between S and s will be caught by the generic except block, assigning n the value 0.

Additionally, there is a minor formatting issue in the code snippet you provided, which should be corrected for the code to run properly. However that doesn't change the output in this case.

upvoted 1 times

🗨️ **eskimolight** 3 weeks, 6 days ago

I feel the output should be 0 so in my opinion the answer should be D
upvoted 1 times

🗨️ **Dave304409** 1 month, 2 weeks ago

Answer C is correct. Print 2

#Explanation:
S = '2A'

```
try:  
n = int(S)  
except ValueError:  
n = 2  
except ArithmeticError:  
n = 1  
except:  
n = 0
```

```
print(n)  
upvoted 1 times
```

🗨️ **DKAT2023** 1 month, 3 weeks ago

C is the correct
upvoted 1 times

🗨️ **herrmann69** 2 months, 2 weeks ago

Selected Answer: C

Answer C is correct. It prints 2.

Trying to parse "2A" to an int will result in a ValueError, which will then set n to 2 in its except branch. No further excepts or assignments to n are done afterwards and n is printed with its value 2.

upvoted 3 times

🗨️ **Nikhil_Durgesh** 2 months, 3 weeks ago

Option "D = it outputs 0" is correct.

Explanation:
S = '2A'

```
try:  
n = int(s)  
except ValueError:  
n = 2  
except ArithmeticError:  
n = 1  
except:  
n = 0  
print(n)
```

Returns output 0.
upvoted 1 times

🗨️ **Dave304409** 1 month, 2 weeks ago

S=!s see your answer
upvoted 1 times

Which of the following snippets will execute without raising any unhandled exceptions? (Choose two.)

- A. `try:`
`print(-1/1)`
`except:`
`print(0/1)`
`else:`
`print(1/1)`
- B. `try:`
`x = 1`
`except:`
`x = x + 1`
`else:`
`x = x + 2`
- C. `try:`
`x = y + 1`
`except (NameError, SystemError):`
`x = y + 1`
`else:`
`y = x`
- D. `try:`
`x = 1 / 0`
`except NameError:`
`x = 1 / 1`
`else:`
`x = x + 1`

Correct Answer: AD

Community vote distribution

AB (100%)

Dave304409 1 month, 2 weeks ago

A and B are correct.
 upvoted 1 times

DKAT2023 1 month, 3 weeks ago

A and B are correct
 upvoted 1 times

Nikhil_Durgesh 2 months ago

A and B are correct.

Because,
 C will throw a NameError
 D will throw a ZeroDivisionError
 upvoted 2 times

kstr 2 months, 2 weeks ago

A and B try it
 try:
 print(-1/1)
 except:
 print(0/1)
 else:
 print(1/1)
 B. try:
 x = 1

except:

`x = x + 1`

else:

`x = x + 1`

upvoted 1 times

  **herrmann69** 2 months, 2 weeks ago

Selected Answer: AB

A and B are correct.

C will throw a NameError

D will throw a ZeroDivisionError

upvoted 4 times

  **a4129fd** 2 months, 3 weeks ago

Selected Answer: AB

B works well in my IDE

upvoted 1 times



What is the expected behavior of the following code?

```
m = 0

def foo(n):
    global m
    assert m == 0
    try:
        return 1/n
    except ArithmeticError:
        m += 1
        raise

try:
    foo(0)
except ArithmeticError:
    m += 2
except:
    m += 1

print(m)
```

- A. it outputs 3
- B. it outputs 1
- C. it outputs 2
- D. the code is erroneous and it will not execute

Correct Answer: A

Community vote distribution

A (100%)

eskimolight 3 weeks, 6 days ago

Confirmed it's Output 3
upvoted 1 times

Dave304409 1 month, 2 weeks ago

Selected Answer: A

is correct
upvoted 1 times

DKAT2023 1 month, 3 weeks ago

A is correct
upvoted 1 times

FasterN8 2 months ago

Tested, A is correct. The raise keyword here does not raise a generic exception.

"When we do not give any exception class name with the raise keyword, it re-raises the exception that last occurred."

So that line in the function re-raises the ArithmeticError a second time.

upvoted 3 times

What is true about the following snippet? (Choose two.)

```
class E(Exception):
    def __init__(self, message):
        self.message = message
    def __str__(self):
        return "it's nice to see you"

try:
    print("I feel fine")
    raise E("what a pity")
except E as e:
    print(e)
else:
    print("the show must go on")
```

- A. the string what a pity will be seen
- B. the string it's nice to see you will be seen
- C. the code will raise an unhandled exception
- D. the string I feel fine will be seen

Correct Answer: AB

Community vote distribution

BD (75%)

AB (25%)

DKAT2023 1 month, 1 week ago

Selected Answer: BD

first D will see , after B will see
upvoted 3 times

Dave304409 1 month, 1 week ago

Selected Answer: AB

is correct
upvoted 1 times

Dave304409 1 month, 2 weeks ago

A, D is correct
upvoted 2 times

Dave304409 1 month, 2 weeks ago

A, B is correct
upvoted 1 times

DKAT2023 1 month, 3 weeks ago

A and D are corrects
upvoted 2 times

kstr 2 months, 2 weeks ago

AD
CODE:
class E(Exception):
def __init__(self, message):
self.message = message

def __str__(self):
return "It's nice to see you"

try:
print("I feel fine")
raise Exception("what a pity")
except Exception as e:

```
print(e)
else:
print("the show must go on")
upvoted 3 times
```

  **DKAT2023** 1 month, 1 week ago

you write the code error, it should be : raise E("what a pity") instead of raise Exception("what a pity") and except E as e: instead of except Exception as e: , so the corrects are : B and D, not A and D
upvoted 2 times

  **kstr** 1 month, 1 week ago

```
You are right
lass E(Exception):
def __init__(self, message):
self.message = message
```

```
def __str__(self):
return "It's nice to see you"
```

```
try:
print("I feel fine")
raise E("what a pity")
except Exception as e:
print(e)
else:
print("the show must go on")
```

Correct is BD
upvoted 1 times

What is the expected behavior of the following code?

```
my_list = [1, 2, 3]

try:
    my_list[3] = my_list[2]
except BaseException as error:
    print(error)
```

- A. it outputs error
- B. it outputs
- C. the code is erroneous and it will not execute
- D. it outputs list assignment index out of range

Correct Answer: C

Community vote distribution

D (100%)

IshanKarpe 2 weeks, 2 days ago

Selected Answer: D

BaseException is expressing it simply so its D
upvoted 1 times

eskimolight 3 weeks, 6 days ago

The Answer is D
upvoted 1 times

JeyTlenJey 3 weeks, 6 days ago

D is correct
upvoted 1 times

Dave304409 1 month, 2 weeks ago

D is correct
upvoted 2 times

DKAT2023 1 month, 3 weeks ago

D is the correct
upvoted 2 times

kstr 2 months, 2 weeks ago

D tested:

```
my_list = [1, 2, 3]
```

```
try:
    my_list[3] = my_list[2]
except BaseException as error:
    print(error)
```

upvoted 2 times

Which of the following expressions evaluate to True? (Choose two.)

- A. `ord("0") - ord("9") == 10`
- B. `len("") == 2`
- C. `chr(ord('z') - 1) == 'y'`
- D. `len("1234") == 4`

Correct Answer: AC

Community vote distribution

CD (100%)

- eskimolight** 3 weeks, 6 days ago
B, C D = All the 3 are correct.
upvoted 1 times
- Dave304409** 1 month, 2 weeks ago
C, D is correct
upvoted 1 times
- DKAT2023** 1 month, 3 weeks ago
C and D are corrects
upvoted 1 times
- kstr** 2 months, 2 weeks ago
BCD are TRUE TESTED:
`print(ord("0") - ord("9") == 10)`
`print(len("") == 2)`
`print(chr(ord('z') - 1) == 'y')`
`print(len("1234") == 4)`
upvoted 3 times
- a4129fd** 2 months, 3 weeks ago
Selected Answer: CD
As tested in my IDE
upvoted 1 times

Which of the following expressions evaluate to True? (Choose two.)

- A. `'xYz'.lower() > 'XY'`
- B. `'8'+ '8' != 2 * '8'`
- C. `float('3.14') == str('3.' + '14')`
- D. `121 + 1 == int('1' + 2 * '2')`

Correct Answer: *BD*

Community vote distribution

AD (100%)

Dave304409 1 month, 2 weeks ago

A, D is correct
upvoted 2 times

DKAT2023 1 month, 3 weeks ago

A and D are corrects
upvoted 1 times

kstr 2 months, 2 weeks ago

AD:
`print('xYz'.lower() > 'XY')`
`print('8'+ '8' != 2 * '8')`
`print(float('3.14') == str('3.' + '14'))`
`print(121 + 1 == int('1' + 2 * '2'))`
upvoted 1 times

a4129fd 2 months, 3 weeks ago

Selected Answer: AD

AD are the right ones, as validated in my IDE
upvoted 2 times

What is the expected behavior of the following code?

```
string = str(1/3)
dummy = ''
for character in string:
    dummy = dummy + character
print (dummy [-1])
```

- A. it outputs 3
- B. it outputs 'None'
- C. it outputs 0
- D. it raises an exception

Correct Answer: D

Dave304409 1 month, 2 weeks ago

A is correct
upvoted 2 times

DKAT2023 1 month, 3 weeks ago

A is correct
upvoted 1 times

kstr 2 months, 2 weeks ago

Correct answer is A
string = str(1/3)
dummy = ""
for character in string:
dummy = dummy + character
print (dummy [-1])
upvoted 2 times

What is the expected behavior of the following code?

```
the_list = "alpha;beta;gamma".split(";")
the_string = ".join(the_list)
print(the_string.isalpha())
```

- A. it outputs True
- B. it outputs False
- C. it outputs nothing
- D. it raises an exception

Correct Answer: A

  **eskimolight** 3 weeks, 6 days ago

The Output should be TRUE - So Option A is the right. Both my IDE and ChatGPT gives the same answer.
upvoted 1 times

  **Dave304409** 1 month, 2 weeks ago

D is correct
upvoted 2 times

  **Dave304409** 1 month, 2 weeks ago

Sorry the answer correct is A
upvoted 2 times

  **DKAT2023** 1 month, 3 weeks ago

D is the correct ,
the_string = ".join(the_list) should be :
the_string = ""join(the_list)
so it raises an exception
upvoted 2 times

Which of the following invocations are valid? (Choose two.)

- A. `sort("python")`
- B. `"python".find("")`
- C. `"python".sort()`
- D. `sorted("python")`

Correct Answer: AD

Community vote distribution

BD (100%)

- Dave304409** 1 month, 2 weeks ago
B, D is correct
upvoted 1 times
- DKAT2023** 1 month, 3 weeks ago
B and D are corrects.
Sort for list not string
upvoted 2 times
- a4129fd** 2 months, 3 weeks ago
Selected Answer: BD
BD are the right ones
upvoted 2 times

Which of the following expressions evaluate to True? (Choose two.)

- A. `'in' in 'in'`
- B. `'in' in 'Thames'`
- C. `'in not' in 'not'`
- D. `'t'.upper() in 'Thames'`

Correct Answer: AD

- Dave304409** 1 month, 2 weeks ago
A, D is correct
upvoted 2 times

Assuming that the snippet below has been executed successfully, which of the following expressions will evaluate to True? (Choose two.)

```
string = 'python'[:2]
string = string[-1] + string[-2]
```

- A. `len(string) == 3`
- B. `string[0] == 'o'`
- C. `string[0] == string[-1]`
- D. `string is None`

Correct Answer: AB

Dave304409 1 month, 2 weeks ago

A, B is correct

```
string = 'python'[:2]
print(string)
pto
R: print(len(string) == 3 )
```

```
string = string[-1] + string[-2]
print(string)
ot
print(string[0] == 'o' )
upvoted 2 times
```

DKAT2023 1 month, 3 weeks ago

B is the correct
upvoted 1 times

kstr 2 months, 2 weeks ago

Sorry only B:
string = 'python'[:2]
string = string[-1] + string[-2]

```
print(len(string) == 3 )
print(string[0] == 'o' )
print( string[0] == string[-1] )
print( string is None)
upvoted 2 times
```

kstr 2 months, 2 weeks ago

Only C is True

```
string = 'python'[:2]
string = string[-1] + string[-2]
```

```
print(len(string) == 3 )
print(string[0] == 'o' )
print( string[0] == string[-1] )
print( string is None)
upvoted 1 times
```

Which of the following statements are true? (Choose two.)

- A. an escape sequence can be recognized by the / sign put in front of it
- B. ASCII is a subset of UNICODE
- C. II in ASCII stands for Internal Information
- D. a code point is a number assigned to a given character

Correct Answer: AB

Community vote distribution

BD (100%)

 **Dave304409** 1 month, 2 weeks ago

Selected Answer: BD

is correct

upvoted 2 times

 **DKAT2023** 1 month, 3 weeks ago

B and D are correct

upvoted 1 times

 **FasterN8** 2 months ago

Selected Answer: BD

BD I say.

upvoted 3 times

 **kstr** 2 months, 2 weeks ago

BD

A:BACKSLASH

C: II is Information Interchange

upvoted 3 times

A property that stores information about a given class's super-classes is named:

- A. `__bases__`
- B. `__super__`
- C. `__upper__`
- D. `__ancestors__`

Correct Answer: A

Community vote distribution

A (100%)

Dave304409 1 month, 2 weeks ago

Selected Answer: A

is correct

upvoted 1 times

DKAT2023 1 month, 3 weeks ago

A is correct

upvoted 2 times

Assuming that the code below has been executed successfully, which of the following expressions evaluate to True? (Choose two.)

```
class Class:
    var = data = 1
    def __init__(self, value):
        self.prop = value
```

```
Object = Class(2)
```

- A. `'var' in Class.__dict__`
- B. `'data' in Object.__dict__`
- C. `len(Class.__dict__) == 1`
- D. `'data' in Class.__dict__`

Correct Answer: AD

Community vote distribution

AD (100%)

Dave304409 1 month, 2 weeks ago

Selected Answer: AD

is correct

upvoted 1 times

DKAT2023 1 month, 3 weeks ago

A and D are corrects

upvoted 1 times

What is the expected behavior of the following code?

```
class Super:
    def make(self):
        return 0
    def doit(self):
        return self.make()

class Sub_A(Super):
    def make(self):
        return 1

class Sub_B(Super):
    pass

a = Sub_A()
b = Sub_B()
print(a.doit() + b.doit())
```

- A. it outputs 1
- B. it outputs 0
- C. it raises an exception
- D. it outputs 2

Correct Answer: D

  **Dave304409** 1 month, 2 weeks ago

A is correct

```
class Super:
def make(self):
return 0
def doit(self):
return self.make()
```

```
class Sub_A(Super):
def make(self):
return 1
```

```
class Sub_B(Super):
pass
```

```
a = Sub_A()
b = Sub_B()
print(a.doit() + b.doit())
```

upvoted 2 times

  **DKAT2023** 1 month, 3 weeks ago

A is the correct
upvoted 2 times

Assuming that the code below has been placed inside a file named code.py and executed successfully, which of the following expressions evaluate to True? (Choose two.)

```
class ClassA:
    var = 1
    def __init__(self, prop):
        prop1 = prop2 = prop

class ClassB(ClassA):
    def __init__(self, prop):
        prop3 = prop ** 2
        super().__init__(prop)
```

```
Object = ClassB(2)
```

- A. `str(Object) == 'Object'`
- B. `__name__ == '__main__'`
- C. `len(ClassB.__bases__) == 1`
- D. `ClassA.__module__ == 'ClassA'`

Correct Answer: BC

Community vote distribution

BC (100%)

 **Dave304409** 1 month, 2 weeks ago

Selected Answer: BC

is correct

upvoted 1 times

 **DKAT2023** 1 month, 3 weeks ago

B and C are corrects

upvoted 2 times

Assuming that the following inheritance set is in force, which of the following classes are declared properly? (Choose two.)

```
class A:  
    pass
```

```
class B(A):  
    pass
```

```
class C(A):  
    pass
```

- A. class Class_3(A,C): pass
- B. class Class_2(B,C): pass
- C. class Class_4(A,B): pass
- D. class Class_1(C,B): pass

Correct Answer: BC

Community vote distribution

BC (100%)

  **Dave304409** 1 month, 2 weeks ago

Selected Answer: BC

is correct

upvoted 1 times

  **DKAT2023** 1 month, 3 weeks ago

B and D are corrects

upvoted 4 times

Assuming that the following piece of code has been executed successfully, which of the expressions evaluate to True? (Choose two.)

```
class A:
    VarA = 1
    def __init__(self):
        self.prop_a = 1

class B(A):
    VarA = 2
    def __init__(self):
        super().__init__()
        self.prop_b = 2
```

```
obj_a = A()
obj_aa = A()
obj_b = B()
obj_bb = obj_b
```

- A. A.VarA == 1
- B. isinstance(obj_b,A)
- C. B.VarA == 1
- D. obj_a is obj_aa

Correct Answer: AB

Community vote distribution

AB (100%)

 **Dave304409** 1 month, 2 weeks ago

Selected Answer: AB

is correct

upvoted 1 times

 **DKAT2023** 1 month, 3 weeks ago

A and B are corrects

upvoted 2 times

What is the expected output of the following snippet?

```
class Upper:
    def __init__(self):
        self.property = 'upper'

class Lower(Upper):
    def __init__(self):
        super().__init__()

Object = Lower()
print(isinstance(Object, Lower), end=' ')
print(Object.property)
```

- A. False upper
- B. True upper
- C. False lower
- D. True lower

Correct Answer: B

Community vote distribution

B (100%)

 **Dave304409** 1 month, 2 weeks ago

Selected Answer: B

is correct

upvoted 1 times

 **DKAT2023** 1 month, 3 weeks ago

B is correct

upvoted 3 times

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]`? (Choose two.)

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

Correct Answer: AB

Community vote distribution

AB (100%)

 **Dave304409** 1 month, 2 weeks ago

Selected Answer: AB

is correct

upvoted 1 times

 **DKAT2023** 1 month, 3 weeks ago

A and B are corrects

upvoted 3 times

What is the expected behavior of the following code?

```
class Class:
    Variable = 0
    def __init__(self):
        self.value = 0

object_1 = Class()
Class.Variable += 1
object_2 = Class()
object_2.value += 1
print(object_2.Variable + object_1.value)
```

- A. it raises an exception
- B. it outputs 2
- C. it outputs 0
- D. it outputs 1

Correct Answer: B

Community vote distribution

D (100%)

DKAT2023 1 month, 1 week ago

Selected Answer: D

class. Variable +=1 so that the outputs is 1(D)
upvoted 2 times

Dave304409 1 month, 2 weeks ago

Selected Answer: D

is correct
upvoted 1 times

DKAT2023 1 month, 2 weeks ago

sorry, C is correct
upvoted 1 times

DKAT2023 1 month, 3 weeks ago

D is correct
upvoted 3 times

What is true about Object-Oriented Programming in Python? (Choose two.)

- A. encapsulation allows you to hide a whole class inside a package
- B. a class is a recipe for an object
- C. each object of the same class can have a different set of properties
- D. the arrows on a class diagram are always directed from a superclass towards its subclass

Correct Answer: BC

Community vote distribution

BC (100%)

kino_1994 14 hours, 54 minutes ago

I'm not sure about C. Each object of the same class has the same set of properties but can have different values.
upvoted 1 times

Dave304409 1 month, 2 weeks ago

Selected Answer: BC

is correct
upvoted 1 times

DKAT2023 1 month, 3 weeks ago

B and C are corrects
upvoted 1 times

What is true about Python class constructors? (Choose two.)

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

Correct Answer: AB

Community vote distribution

AB (100%)

Dave304409 1 month, 2 weeks ago

Selected Answer: AB

is correct
upvoted 1 times

DKAT2023 1 month, 3 weeks ago

A and B are corrects
upvoted 1 times

What is the expected behavior of the following code?

```
class Class:
    __Var = 0
    def foo(self):
        Class.__Class__Var += 1
        self.__prop = Class.__Class__Var

o1 = Class()
o1.foo()
o2 = Class()
o2.foo()
print(o2.__Class__Var + o1.__Class__prop)
```

- A. it outputs 6
- B. it raises an exception
- C. it outputs 1
- D. it outputs 3

Correct Answer: *D*

 **DKAT2023** 1 month, 2 weeks ago

B is correct
upvoted 2 times

 **Dave304409** 1 month, 2 weeks ago

D is correct
upvoted 1 times

What is the expected output of the following code?

```
myli = range (-2,2)
```

```
m = list(filter(lambda x: True if abs(x) < 1 else False, myli)) print(len(m))
```

- A. 4
- B. 1
- C. an exception is raised
- D. 16

Correct Answer: B

Community vote distribution

B (100%)

  **Dave304409** 1 month, 1 week ago

Selected Answer: B

is correct

upvoted 1 times

  **DKAT2023** 1 month, 2 weeks ago

B is correct

upvoted 2 times

What is true about lambda functions? (Choose two.)

- A. they are called anonymous functions
- B. they cannot return the None value as a result
- C. they must contain the return keyword
- D. they must have a non-zero number of parameters

Correct Answer: AD

Community vote distribution

AD (100%)

 **JeyTienJey** 3 weeks, 4 days ago

```
x = lambda : 10
print(x())
# returns 10 - it can have any number of argument , zero included
```

```
y = lambda x: None
print(y(1))
# returns None
```

A is correct :(
upvoted 1 times

 **Dave304409** 1 month, 2 weeks ago

Selected Answer: AD

is correct
upvoted 1 times

 **DKAT2023** 1 month, 2 weeks ago

A and D are corrects
upvoted 2 times