

Extra class correction

Henri Vandendriessche
henri.vandendriessche@ens.fr

2023-12-19

Q1. What does the following Python code do?

```
print("Bonjour")
```

Q1. What does the following Python code do?

```
print("Bonjour")
```

```
## Bonjour
```

Q2. What does the following Python code do?

```
print(3 + 2)
```

Q2. What does the following Python code do?

```
print(3 + 2)
```

```
## 5
```

Q3. What does the following Python code do?

```
print("3 + 2")
```

Q3. What does the following Python code do?

```
print("3 + 2")
```

```
## 3 + 2
```

Q4: What do the following Python code do?

```
print(x)
```


Q4: What do the following Python code do?

```
print(x)
```

```
## name 'x' is not defined
```

Q5. what does the following Python code do?

```
x = 5  
print(x - 3)
```

Q5. what does the following Python code do?

```
x = 5  
print(x - 3)
```

2

Q6. What does the following code print?

```
x = 0  
x = 1  
print(x)
```

Q6. What does the following code print?

```
x = 0  
x = 1  
print(x)
```

```
## 1
```

Q7. What does the following code print ?

```
x = 2  
x = x - 1  
print(x)
```

Q7. What does the following code print ?

```
x = 2  
x = x - 1  
print(x)
```

```
## 1
```

Q8. What does the following Python code print?

```
print(1 == 2)
```


Q8. What does the following Python code print?

```
print(1 == 2)
```

```
## False
```

Q9. What does the following code print?

```
x = 3 == 3  
print(x)
```

Q9. What does the following code print?

```
x = 3 == 3  
print(x)
```

```
## True
```

Q10. What does the following code print?

```
if 0 == 1:  
    print("a")  
  
print("b")
```

Q10. What does the following code print?

```
if 0 == 1:  
    print("a")  
  
print("b")
```

```
## b
```

Q11. What does the following code print?

```
if 0 == 1:  
    print("a")  
else:  
    print("b")
```

Q11. What does the following code print?

```
if 0 == 1:  
    print("a")  
else:  
    print("b")
```

b

Q12. What does the following code print?

```
if 0 == 1:  
    print("a")  
    print("b")  
else:  
    print("c")  
    print("d")  
  
print("e")
```


MCQ test correction

Q12. What does the following code print?

```
if 0 == 1:  
    print("a")  
    print("b")  
else:  
    print("c")  
    print("d")
```

```
## c  
## d
```

```
print("e")
```

```
## e
```

Q13. What does the following code print?

```
if 0 == 1:
    print("a")
else:
    if 1 == 1:
        print("b")
    else:
        print("c")
    print("d")
```

Q13. What does the following code print?

```
if 0 == 1:  
    print("a")  
else:  
    if 1 == 1:  
        print("b")  
    else:  
        print("c")  
    print("d")
```

```
## b  
## d
```

Q14. How many lines does the following code print?

```
n = 0
while n > 1:
    print("ok")
```

Q14. How many lines does the following code print?

```
n = 0
while n > 1:
    print("ok")
```

Q15. How many lines does the following code print?

```
n = 3
while n > 1:
    n = n - 1
    print("ok")
```

Q15. How many lines does the following code print?

```
n = 3
while n > 1:
    n = n - 1
    print("ok")
```

```
## ok
## ok
```

Q16. What does the following code print?

```
x = 1
y = -1
while x < 5:
    y = y - 1
    x = x * 2

print(x, y)
```


Q16. What does the following code print?

```
x = 1
y = -1
while x < 5:
    y = y - 1
    x = x * 2

print(x, y)
```

8 -4

Q17. What does the following code print?

```
n = 0
while n < 3:
    if n < 2:
        print("less")
    else:
        print("more")
    n = n + 1
```

Q17. What does the following code print?

```
n = 0
while n < 3:
    if n < 2:
        print("less")
    else:
        print("more")
    n = n + 1
```

```
## less
## less
## more
```

Q18. What does the following code print?

```
for x in [3, 1]:  
    for y in [2, 4]:  
        print(x, y)
```

Q18. What does the following code print?

```
for x in [3, 1]:  
    for y in [2, 4]:  
        print(x, y)
```

```
## 3 2  
## 3 4  
## 1 2  
## 1 4
```

Q19. What does the following code print?

```
def print_one():  
    print(1)
```

Q19. What does the following code print?

```
def print_one():  
    print(1)
```

Q20. What does the following code print?

```
def print_one():  
    print(1)  
  
print_one()
```


Q20. What does the following code print?

```
def print_one():  
    print(1)
```

```
print_one()
```

```
## 1
```

Q21. Consider the following code:

```
def print_sum(x, y):  
    print(x + y)
```

What is the name of the above-defined function?

Q21. Consider the following code:

What is the name of the above-defined function?

```
print_sum
```

Q22. Consider the following code:

```
def print_sum(x, y):  
    print(x + y)  
  
print_sum(5,6)
```

What are the parameters of the above-defined function?

Q22. Consider the following code:

What are the parameters of the above-defined function?

(x, y)

Q23. Consider the following code:

```
def print_sum(x, y):  
    print(x + y)
```

Which line(s) correspond(s) to the body of the above-defined function?

Q23. Consider the following code:

Which line(s) correspond(s) to the body of the above-defined function?

```
def print_sum(x, y):  
    print(x + y)
```

Q24. What does the following code print?

```
def sum(x, y):  
    print(x)  
    return x + y  
  
a = 1  
b = sum(a, -1)  
print(b)
```


MCQ test correction

Q24. What does the following code print?

```
def sum(x, y):  
    print(x)  
    return x + y
```

```
a = 1  
b = sum(a, -1)
```

```
## 1
```

```
print(b)
```

```
## 0
```

MCQ test correction

Q25. What does the following code print?

```
def fun_a(x):  
    print(x - 1)  
    return x + 1  
  
def fun_b(y):  
    print(y)  
    z = fun_a(x)  
    print(z)  
    return z * 2  
  
y = 1  
z = fun_b(y)  
print(z)
```

MCQ test correction

Q25. What does the following code print?

```
def fun_a(x):  
    print(x - 1)  
    return x + 1  
  
def fun_b(y):  
    print(y)  
    z = fun_a(x)  
    print(z)  
    return z * 2  
  
y = 1  
z = fun_b(y)
```

```
## 1  
## 0  
## 2
```

```
print(z)
```

```
## 4
```

Q26. How can we extract the name (JB Lewis) from aString?

```
aString = "A long complicated string containing a quote of someone important who said: 'JB Lewis'"
```

- aString[aString.index("-")+1:]
- aString[index("-")+1:]
- aString[index("-")+1:end]
- aString[aString.index("-")+1:end]

Q26. How can we extract the name (JB Lewis) from aString?

```
aString = "A long complicated string containing a quote of someone important who said  
print(aString[aString.index("-")+1:])
```

```
## EOL while scanning string literal (<string>, line 1)
```

Q27. What will the following code print?

```
listOfNums = ["2", "7", "3", "4"]  
result = "+".join(listOfNums)  
print(result)
```

Q27. What will the following code print?

```
listOfNums = ["2", "7", "3", "4"]  
result = "+".join(listOfNums)  
print(result)
```

2+7+3+4

Q28.What will the following code print?

```
start = "This is a test"  
end = " ".join(start.split(" "))  
print(start == end)
```


Q28.What will the following code print?

```
start = "This is a test"  
end = " ".join(start.split(" "))  
print(start == end)
```

```
## True
```

Q29. What will the following code print?

```
aStr = "This is a test"  
aStr.upper()  
print(aStr)
```

Q29. What will the following code print?

```
aStr = "This is a test"  
aStr.upper()
```

```
## 'THIS IS A TEST'
```

```
print(aStr)
```

```
## This is a test
```

Q30. What are the contents of the file 'test.txt' after the following executes?

```
file = open('test.txt', 'w')
file.write('This is a test, I think.')
anotherFile = open('test.txt', 'r')
contents = anotherFile.read()
yetAnotherOne = open('test.txt', 'a')
yetAnotherOne.write(' Yes')
```

MCQ test correction

Q30. What are the contents of the file 'test.txt' after the following executes?

```
file = open('test.txt', 'w')  
file.write('This is a test, I think.')
```

24

```
anotherFile = open('test.txt', 'r')  
contents = anotherFile.read()  
yetAnotherOne = open('test.txt', 'a')  
yetAnotherOne.write(' Yes')
```

4

Q31. In which directory is the file test.txt located?

```
file = open('test.txt', 'w')  
file.write("This is a test, I think.")  
file.close()
```

Q31. In which directory is the file test.txt located?

```
file = open('test.txt', 'w')  
file.write("This is a test, I think.")
```

```
## 24
```

```
file.close()
```

Q32. Should one generally prefer absolute or relative paths to files and directories?

- Absolute
- Relative
- It doesn't matter
- Randomly choose one of the two

Q32. Should one generally prefer absolute or relative paths to files and directories?

- Absolute
- **Relative**
- It doesn't matter
- Randomly choose one of the two

Q33. Why is it often useful to name a piece of data? e.g. (several options possible)

```
ipLine = data[2]
```

Q33. Why is it often useful to name a piece of data? e.g. (several options possible)

```
ipLine = data[2]
```

```
## name 'data' is not defined
```

Q34. In the following code, what is the value of fileName?

```
def wordCount(fileName, word):  
    file = open(fileName, 'r')  
    contents = file.read()  
    words = contents.split(' ')  
    wordCount = words.count(word)  
    print(wordCount)
```

MCQ test correction

Q34. In the following code, what is the value of fileName?

```
def wordCount(fileName, word):  
    file = open(fileName, 'r')  
    contents = file.read()  
    words = contents.split(' ')  
        wordCount = words.count(word)  
    print(wordCount)
```

inconsistent use of tabs and spaces in indentation (<string>, line 5)

Q35. What will be printed after this program runs?

```
def test():  
    print("This will print 12")
```

Q35. What will be printed after this program runs?

```
def test():  
    print("This will print 12")
```

Q36. What will be printed when the following program executes?

```
def aProgram():  
    print("This will print 12")  
aProgram()
```


Q36. What will be printed when the following program executes?

```
def aProgram():  
    print("This will print 12")  
aProgram()
```

expected an indented block (<string>, line 2)

Q37. What will be printed when the following program executes?

```
def aProgram():  
    print("This will print 12")  
aProgram()
```

Q37. What will be printed when the following program executes?

```
def aProgram():  
    print("This will print 12")  
aProgram()
```

```
## This will print 12
```

Q38. What will be printed when the following program is executed?

```
def myFun(aNum):  
    return aNum * 2  
  
myNumber = 4  
k = myFun(myNumber)  
print(k)
```

Q38. What will be printed when the following program is executed?

```
def myFun(aNum):  
    return aNum * 2  
  
myNumber = 4  
k = myFun(myNumber)  
print(k)
```

8

Q39. What will be printed when the following program is executed?

```
def myFun(aNum):  
    a = aNum * 2  
  
myNumber = 4  
k = myFun(myNumber)  
print(k)
```

Q39. What will be printed when the following program is executed?

```
def myFun(aNum):  
    a = aNum * 2  
  
myNumber = 4  
k = myFun(myNumber)  
print(k)
```

None

Q40. What will be printed when the following program is executed?

```
def myFun(aNum):  
    a = aNum * 2  
    return a  
myNumber = 4  
k = myFun(myNumber)  
print(k)
```


Q40. What will be printed when the following program is executed?

```
def myFun(aNum):  
    a = aNum * 2  
    return a  
myNumber = 4  
k = myFun(myNumber)  
print(k)
```

8

Q41. What will be printed when the following program is executed?

```
def myFun(aNum):  
    a = aNum * 2  
    print(a)  
  
myNumber = 4  
k = myFun(myNumber)  
print(k)
```

MCQ test correction

Q41. What will be printed when the following program is executed?

```
def myFun(aNum):  
    a = aNum * 2  
    print(a)  
  
myNumber = 4  
k = myFun(myNumber)
```

```
## 8
```

```
print(k)
```

```
## None
```

Q42. What's the value of the variable k after this program executes?

```
def myFun(aNum):  
    a = aNum * 2  
    print(a)  
  
myNumber = 4  
k = myFun(myNumber)  
print(k)
```

MCQ test correction

Q42. What's the value of the variable k after this program executes?

```
def myFun(aNum):  
    a = aNum * 2  
    print(a)  
  
myNumber = 4  
k = myFun(myNumber)
```

```
## 8
```

```
print(k)
```

```
## None
```

Q43. What is/are the correct way(s) to call the following function? (assume all variables exist. correct means here no error display on the terminal)

```
def readResults(resultFile):  
    file = open(resultFile, 'r');  
    contents = file.read()  
    return contents
```

Q43. What is/are the correct way(s) to call the following function? (assume all variables exist. correct means here no error display on the terminal)

```
def readResults(resultFile):  
    file = open(resultFile, 'r');  
    contents = file.read()  
    return contents
```

Q44. What will the following code print?

```
def functionA(aNumber):  
    return aNumber+1  
def functionB(aNumber):  
    return functionA(aNumber)+1  
print(functionB(1))
```


Q44. What will the following code print?

```
def functionA(aNumber):  
    return aNumber+1  
def functionB(aNumber):  
    return functionA(aNumber)+1  
print(functionB(1))
```

3

Q45. What will the following program print?

```
aVar = 12
def myFun():
    aVar = 7+3
print(aVar)
```

Q45. What will the following program print?

```
aVar = 12
def myFun():
    aVar = 7+3
print(aVar)
```

12

Q46. What will the following program print?

```
aVar = 12
def myFun():
    aVar = 7+3
print (aVar)
myFun()
```

Q46. What will the following program print?

```
aVar = 12
def myFun():
    aVar = 7+3
print (aVar)
```

12

```
myFun()
```

Q47. What will the following program print?

```
aVar = 12
def myFun():
    aVar = 7+3
    print (aVar)
myFun()
```

Q47. What will the following program print?

```
aVar = 12
def myFun():
    aVar = 7+3
    print (aVar)
myFun()
```

10

Q48. What will the following program print?

```
aVar = 12
def myFun():
    aVar = 7+3
    return aVar
print(myFun())
```


Q48. What will the following program print?

```
aVar = 12
def myFun():
    aVar = 7+3
    return aVar
print(myFun())
```

10

Q49. What will the following program print?

```
myVar = 12
def myFun():
    aVar = 7+3
    return aVar
myVar = myFun()
print(myVar)
```

Q49. What will the following program print?

```
myVar = 12
def myFun():
    aVar = 7+3
    return aVar
myVar = myFun()
print(myVar)
```

10

Q50. What will the following program print?

```
def myFun(aVar):  
    test = aVar+3  
    return test  
aVar = 15  
bVar = myFun(aVar)  
print(aVar)
```

MCQ test correction

Q50. What will the following program print?

```
def myFun(aVar):  
    test = aVar+3  
    return test  
aVar = 15  
bVar = myFun(aVar)  
print(aVar)
```

15

Q51. What will the following program print?

```
def myFun(aVar):  
    test = aVar+3  
    return test  
bVar = 15  
bVar = myFun(bVar)  
print(bVar)
```

Q51. What will the following program print?

```
def myFun(aVar):  
    test = aVar+3  
    return test  
bVar = 15  
bVar = myFun(bVar)  
print(bVar)
```

18

Q52. What will the following program print?

```
def myFun(aList):  
    aList[1] = 100  
  
test = [1, 2, 3]  
myFun(test)  
print(test[1])
```


Q52. What will the following program print?

```
def myFun(aList):  
    aList[1] = 100
```

```
test = [1, 2, 3]  
myFun(test)  
print(test[1])
```

```
## 100
```

Q53. What's the strangest part of the following function definition?

```
def myFun(aVar):  
    aVar = 3  
    return aVar + 5
```

Q53. What's the strangest part of the following function definition?

```
def myFun(aVar):  
    aVar = 3  
    return aVar + 5
```

Q54. What will the following code print?

```
def validateAge(userAge):  
    if userAge<18:  
        return False  
    else:  
        return True  
  
def isAdult():  
    age = input("Please type your age") # assume user input 19  
    return validateAge(int(age))  
  
print(isAdult())
```

Q54. What will the following code print?

```
def validateAge(userAge):  
    if userAge<18:  
        return False  
    else:  
        return True  
  
def isAdult():  
    age = input("Please type your age") # assume user input 19  
    return validateAge(int(age))  
  
print(isAdult())
```

Q55. What is the value of aVar after the following executes?

```
def someGreatFunction():  
    return 12  
aVar = someGreatFunction
```

Q55. What is the value of aVar after the following executes?

```
def someGreatFunction():  
    return 12  
aVar = someGreatFunction
```

Q56. What is the value of aVar after the following executes?

```
def someGreatFunction():  
    return 12  
bVar = someGreatFunction  
aVar = bVar()
```


Q56. What is the value of aVar after the following executes?

```
def someGreatFunction():  
    return 12  
bVar = someGreatFunction  
aVar = bVar()
```

Q57. In the following code what is the name of the module ?

```
from random import choice
```

Q57. In the following code what is the name of the module ?

```
from random import choice
```

Q58. After the following import statement, how can we use the shuffle function?

```
from random import shuffle
```

Q58. After the following import statement, how can we use the shuffle function?

```
from random import shuffle
```

```
shuffle()
```

```
## shuffle() missing 1 required positional argument: 'x'
```

Q59. After the following import statement, how can we use the randint function?

```
from random import *
```

Q59. After the following import statement, how can we use the randint function?

```
from random import *
```

```
randint(min, max)
```

```
## unsupported operand type(s) for +: 'builtin_function_or_method' and 'int'
```

Q60. After the following import statement, how can we use the randint function?

```
import random
```


MCQ test correction

Q60. After the following import statement, how can we use the randint function?

```
import random
```

```
random.randint(min, max)
```

```
## unsupported operand type(s) for +: 'builtin_function_or_method' and 'int'
```

Q61. What does the “*data” means in the following code

```
def someFun(*data):  
    print(data)
```

Q61. What does the “*data” means in the following code

```
def someFun(*data):  
    print(data)
```

Q62. What will be the output of this code

```
print(type(type(int)))
```

Q62. What will be the output of this code

```
print(type(type(int)))
```

```
## <class 'type'>
```

Q62. Which of these is not a core data type

- Tuple
- Integer
- Dict
- Datetime

Q62. Which of these is not a core data type

- Tuple
- Integer
- Dict
- **Datetime**

Q63. What is the correct way to instantiate an empty dictionary? (several option possible)

- `mydict = dict()`
- `mydict = {}`
- `mydict = ()`
- `mydict = dictionary{}`
- `mydict = dict{}`

Q63. What is the correct way to instantiate an empty dictionary? (several option possible)

- **mydict = dict()**
- **mydict = {}**
- mydict = ()
- mydict = dictionary{}
- mydict = dict{}

Q65. Which method can be used to separate a string in a list?

- `split()`
- `len()`
- `cut()`
- `strip()`

Q65. Which method can be used to separate a string in a list?

- **split()**
- len()
- cut()
- strip()

Q66. Which statement is used to stop a loop?

- exit
- break
- continue
- return

Q66. Which statement is used to stop a loop?

- exit
- **break**
- continue
- return

Q67. Which statement is used to skip an iteration of a loop?

- next
- jump
- continue
- skip

Q67. Which statement is used to skip an iteration of a loop?

- next
- jump
- **continue**
- skip

Q68. Which collection data type is ordered, changeable and allows duplicate items?

- Set
- List
- Tuple
- Dictionnary

Q68. Which collection data type is ordered, changeable and allows duplicate items?

- Set
- **List**
- Tuple
- Dictionnary

Q69. Which of the following options is provided by pdb? (several option possible)

- Executing the program line by line
- List all the errors in your script
- Checking variables values at any step of the code execution
- Set break points

Q69. Which of the following options is provided by pdb? (several option possible)

- **Executing the program line by line**
- List all the errors in your script
- **Checking variables values at any step of the code execution**
- **Set break points**

Q70. What are the possibilities of the logging module in python? (several option possible)

- Generating more precise error message
- Creating custom log levels according to the “criticality” of events in your script**
- Preventing your script from crashing
- Basic logging to track variables values

Q70. What are the possibilities of the logging module in python? (several option possible)

- Generating more precise error message
- **Creating custom log levels according to the “criticality” of events in your script**
- Preventing your script from crashing
- **Basic logging to track variables values**