**Coding Nest MCQ Test -2**

**Python**

1. Create a Tuple with two items that both are lists. The first one holds a list of even numbers up to 10. The second one holds the odd numbers up to 20. How can you create this?

a. mytuple = ([\_ for \_ in range(10) if \_ % 2 == 0], [ \_ for \_ in range(10) if \_ % 2 != 0])  
b. mytuple = ([(\_) for \_ in range(10) if \_ % 2 == 0], [ (\_,) for \_ in range(20) if \_ % 2 != 0])  
c. mytuple = ([\_ for \_ in range(10) if \_ % 2 == 0], [ \_ for \_ in range(20) if \_ % 2 != 0])  
d. Tuples cannot contain lists.

2. Given a list:

arr = ['a', 'b', 'c', 'd']

How would you create a variable with the items in the list arr in a way that appending to the new list also appends to the original list arr?

a. arr\_copy = arr  
b. arr\_copy = arr.copy()  
c. arr\_copy = copy.copy(arr)  
d. arr\_copy = arr[:]

3. Given the following function:

def double(\*\*kwargs):

        return [ \_\*2 for \_ in range(kwargs)]

Upon calling the function double(10) you expect to see the numbers doubled until 10.

But get the following result:

Traceback (most recent call last):

  File "<stdin>", line 1, in <module>

TypeError: double() takes 0 positional arguments but 1 was given

What is wrong?

a. You need to yield the results instead of a return statement.  
b. You need to call the function with keyword arguments.  
c. You need to call the function like this: double(0,10).  
d. You need to replace the kwargs in the return statement with \*\*kwargs.

4. Given:

 class NoneClass:

  def \_\_eq\_\_(self, my\_object):

 # Override to always return true

  return True

none\_class = NoneClass()

What should be expected when running:

 >> >print(none\_class is None)

 >>> print(none\_class == None)

a. True

False

b. False

True

c. True

True

d. False

False

5. Which property of an object allows you to go through its elements with a for loop?

a. generator  
b. iterable  
c. iterator  
d. Cannot iterate over an object

6. Given the following function definition:

def my\_evens(list\_of\_numbers):

        for i in range(list\_of\_numbers):

                if i % 2 == 0:

                        yield i

You call the function with a print statement as follows:

print(my\_evens(100))

You receive the following output:

<generator object my\_evens at 0x006C3270>

What is the reason you can't see the list of integers you expect?

a. The yield statement needs to be replaced with the return statement.  
b. It prints only the odd numbers.  
c. Iterators use the next function to iterate over the elements, print does not implement it.  
d. You need to wrap it with the range function to iterate over the elements.

7. Output of the below Program:

class Test:

def \_\_init\_\_(self):

print("Hello World")

def \_\_init\_\_(self):

print("Bye World")

>>> obj = Test()

a. Hello World  
b. Compilation Error  
c. Bye World  
d. Ambiguity

8. Output of the below program:

>>> print(chr(ord('b')+1))

a. b

b. syntax error

c. c

d. b+1

9. What is output of following code −

x = 2

y = 10

x \* = y \* x + 1

a. 42

b. 41

c. 40

d. 39

10. What is output of following code −

def func(x, ans):

if(x==0):

return 0

else:

return func(x-1, x+ans)

print(func(2,0))

a. 0

b. 1

c. 2

d. 3

11. Find the output of the code?

def f(a, b = 1, c = 2):

print('a is: ',a, 'b is: ', b, 'c is: ', c)

>>> f(2, c = 2)

>>> f(c = 100, a = 110)

a. a is: 2 b is: 1 c is: 2

a is: 110 b is: 1 c is: 100

b. a is: 2 b is: 2 c is: 2

a is: 110 b is: 2 c is: 100

c. a is: 0 b is: 2 c is: 2

a is: 110 b is: 0 c is: 100

d. a is: 110 b is: 0 c is: 100

a is: 110 b is: 0 c is: 100

12. What is the output of the code?

try:

list = 5\*[0]+5\*[10]

x = list[9]

print(''Done!'')

except IndexError:

print(''Index out of Bond! '')

else:

print(''Nothing is wrong!'')

finally:

print(''Finally block!'')

a. ‘Finally Block!’

b. ‘Done!’ follow by ‘Nothing is wrong!’

c. ‘Nothing is wrong!’ followed by ‘Finally block!’

d. ‘Done!’ followed by ‘Nothing is wrong!’and then followed by ‘Finally block’.

13. Which options are correct to create an empty set in Python?

a. {}

b. ()

c. []

d. set()

14. What is the value of a, b, c in the given below code?

a, b = c = 2 + 2, ''TutorialsPoint''

a. a=4, 'TutorialsPoint'

b= 4, 'TutorialsPoint'

c= 4, 'TutorialsPoint'

b. a=2

b= 'TutorialsPoint'

c=4, 'TutorialsPoint'

c. a=4

b= 'TutorialsPoint'

c=4, 'TutorialsPoint'

d. a=4

b= 'TutorialsPoint'

c= NULL

15. What is setattr() used for?

a. To access the attribute of the object

b. To set an attribute

c. To check if an attribute exists or not

d. To delete an attribute

16. What will be the output of the following Python code?

re.findall('good', 'good is good')

re.findall('good', 'bad is good')

a.

[‘good’, ‘good’]

[‘good’]

b.

(‘good’, ‘good’)

(good)

c.

(‘good’)

(‘good’)

d.

[‘good’]

[‘good’]

17. What will be the output of the following Python code?

s = 'welcome home'

m = re.match(r'(.\*)(.\*?)', s)

print(m.group())

a. (‘welcome’, ‘home’)

b. [‘welcome’, ‘home’]

c. welcome home

d. [‘welcome’ // ‘home’ ]

18. What will be the output of the following Python code?

import re

s = "A new day"

m = re.match(r'(.\*)(.\*?)', s)

print(m.group(2))

print(m.group(0))

a.

No output

A new day

b.

No output

No output

c.

[‘A’, ‘new’, ‘day’]

(‘A’, ‘new’, ‘day’)

d.

Error

[‘A’, ‘new’, ‘day’]

19. The output of the following two Python codes are the same.

p = re.compile('hello')

r = p.match('hello everyone')

print(r.group(0))

r = re.match('hello', 'hello everyone')

print(r.group(0))

a. True

b. False

20. What will be the output of the following Python code?

class stud:

def \_\_init\_\_(self, roll\_no, grade):

self.roll\_no = roll\_no

self.grade = grade

def display (self):

print("Roll no : ", self.roll\_no, ", Grade: ", self.grade)

stud1 = stud(34, ‘S’)

stud1.age=7

print(hasattr(stud1, 'age'))

a. Error as age isn’t defined

b. True

c. False

d. 7

21. What will be the output of the following Python code?

class stud:

'''Base class for all students'''

def \_\_init\_\_(self, roll\_no, grade):

self.roll\_no = roll\_no

self.grade = grade

def display (self):

print("Roll no : ", self.roll\_no, ", Grade: ", self.grade)

print(student.\_\_doc\_\_)

a. Exception is thrown

b. \_\_main\_\_

c. Nothing is displayed

d. Base class for all students

22. Suppose B is a subclass of A, to invoke the \_\_init\_\_ method in A from B, what is the line of code you should write?

a. A.\_\_init\_\_(self)

b. B.\_\_init\_\_(self)

c. A.\_\_init\_\_(B)

d. B.\_\_init\_\_(A)

23. What will be the output of the following Python code?

class Test:

def \_\_init\_\_(self):

self.x = 0

class Derived\_Test(Test):

def \_\_init\_\_(self):

Test.\_\_init\_\_(self)

self.y = 1

def main():

b = Derived\_Test()

print(b.x,b.y)

main()

a. Error because class B inherits A but variable x isn’t inherited

b. 0 0

c. 0 1

d. Error, the syntax of the invoking method is wrong

24. What will be the output of the following Python code?

class A:

def \_\_init\_\_(self, x= 1):

self.x = x

class der(A):

def \_\_init\_\_(self,y = 2):

super().\_\_init\_\_()

self.y = y

def main():

obj = der()

print(obj.x, obj.y)

main()

a. Error, the syntax of the invoking method is wrong

b. The program runs fine but nothing is printed

c. 1 0

d. 1 2

25. What will be the output of the following Python code?

class A:

def one(self):

return self.two()

def two(self):

return 'A'

class B(A):

def two(self):

return 'B'

obj1=A()

obj2=B()

print(obj1.two(),obj2.two())

a. A A

b. A B

c. B B

d. An exception is thrown

26. What will be the output of the following Python code?

class A:

def \_\_init\_\_(self,x):

self.x = x

def count(self,x):

self.x = self.x+1

class B(A):

def \_\_init\_\_(self, y=0):

A.\_\_init\_\_(self, 3)

self.y = y

def count(self):

self.y += 1

def main():

obj = B()

obj.count()

print(obj.x, obj.y)

>>> main()

a. 3 0

b. 3 1

c. 0 1

d. An exception in thrown

27. What will be the output of the following Python code?

>>> class A:

pass

>>> class B(A):

pass

>>> obj=B()

>>> isinstance(obj,A)

a) True

b) False

c) Wrong syntax for isinstance() method

d) Invalid method for classes

28. What will be the output of the following Python code?

class A:

def \_\_init\_\_(self):

self.\_\_x = 1

class B(A):

def display(self):

print(self.\_\_x)

def main():

obj = B()

obj.display()

main()

a. 1

b. 0

c. Error, invalid syntax for object declaration

d. Error, private class member can’t be accessed in a subclass

29. What will be the output of the following Python code?

class A:

def \_\_init\_\_(self):

self.\_x = 5

class B(A):

def display(self):

print(self.\_x)

def main():

obj = B()

obj.display()

main()

a. Error, invalid syntax for object declaration

b. Nothing is printed

c. 5

d. Error, private class member can’t be accessed in a subclass

30. What will be the output of the following Python code?

class A:

def test1(self):

print(" test of A called ")

class B(A):

def test(self):

print(" test of B called ")

class C(A):

def test(self):

print(" test of C called ")

class D(B,C):

def test2(self):

print(" test of D called ")

obj=D()

obj.test()

a.

test of B called

test of C called

b.

test of C called

test of B called

c. test of B called

d. Error, both the classes from which D derives has same method test()