# Python-PrepTerm
# Quiz

| **Code:** | MT2020013 |
|---|---|

1. Which of the following operator in python evaluates to true if it does not finds a variable in the specified sequence and false otherwise?

    1. $**$
    2. //
    3. **is**
    4. **not in**

2. What is the following function reverses objects of list in place?

    1. **list** . reverse ()
    2. **list** . sort ([func])
    3. **list** . pop(obj=**list**[−1])
    4. **list** . remove(obj)

3. When is the finally block executed?

    1. when there is no exception
    2. when there is an exception
    3. only if some condition that has been specified is satisfied
    4. always

4. Using the pack manager, how you can you put the components in a container in the same row?

    1. Component.pack(side= ''LEFT'')
    2. Component.pack(''Left '')
    3. Component.pack(side=LEFT)
    4. Component.pack(Left−side)

5. Which of the following statements can be used to check, whether an object obj is an instance of class A or not?

    1. obj.**isinstance**(A)
    2. A.**isinstance**(obj)

3. **isinstance**(obj, A)
4. **isinstance**(A, obj)

6. There are different basic operators in python and work according to the order of their precedence.

   Arrange the order of precedence of the following operators:

   1. Division
   2. Multiplication
   3. Parentheses
   4. Exponential
   5. Addition
   6. Subtraction

   1. i, ii, iii, iv, v, vi.
   2. iv, iii, ii, i, vi, v.
   3. iii, iv, i, ii, v, vi.
   4. iv, iii, i, ii, v, vi.

7. What is output of following code:

   a = (1, 2) a[0] +=1

   1. (1,1,2)
   2. 2
   3. Type Error
   4. Syntax Error

8. Which of the following function sets the integer starting value used in generating random numbers?

   1. choice(seq)
   2. randrange ([start ,] stop [, step])
   3. random()
   4. seed([x])

9. Which of the following is required to create a new instance of the class?

   1. A constructor
   2. A class
   3. A value-returning method
   4. A None method

10. What is the following function compares elements of both dictionaries dict1, dict2?

    1. dict1.**cmp**(dict2)

2. dict1.sort(dict2)
3. **cmp**(dict1, dict2)
4. None of the above.

11. What is the output of the following code?

```
def nprint(message, n):
while(n > 0):
   print(message)
n-=1
nprint('z', 5)
```

1. zzzz
2. zzzzz
3. Syntax Error
4. Infinite Loop

12. Which of the following statements are correct about the given code snippet?

```
class A:
    def __init__(self, i = 0):
        self.i = i

class B(A):
    def __init__(self, j = 0):
        self.j = j

def main():
    b = B()
    print(b.i)
    print(b.j)

main()
```

1. Class B inherits A, but the data field `'i'` in A is not inherited.
2. Class B inherits A, thus automatically inherits all data fields in A.
3. When you create an object of B, you have to pass an argument such as B(5).
4. The data field `'j'` cannot be accessed by object b.

13. What should be given in range of the given below code to print nothing in output?

```
for i in range(?):
   print(i)
```

1. 0.1
2. 0
3. NULL
4. 1

14. Syntax error in python is detected by _____ at _____

    1. compiler/ compile time
    2. interpreter/ run time
    3. compiler/ run time
    4. interpreter/ compile time

15. What is the following function returns item from the list with max value?

    1. **cmp**(**list**)
    2. **len**(**list** )
    3. **max**(**list**)
    4. **min**(**list**)

16. What will be the output of the code?

    z = "Best website is Tutorials Point" z.find("Tutorials")

    1. 3
    2. 13
    3. 17
    4. 16

17. How many except statements can a try-except block have?

    1. zero
    2. one
    3. more than one
    4. more than zero

18. What is the output for:

    `'you are doing well'`[2:999]

    1. `'you are doing well'`
    2. `' '`
    3. Index error.
    4. `'u are doing well'`

19. What will be the output of the following code?

```
for i in ['t', 'n', 'i ', 'o', 'p'][::-1]:
    print(i)
```

    1. t n i o p
    2. p o i n t
    3. t n i o p 1 0 −1

4. p o i n t 1 0 −1

20. What is the following function inserts an object at given index in a list?

    1. **list** .index(obj)
    2. **list** . insert (index, obj)
    3. **list** .pop(obj=**list**[−1])
    4. **list** .remove(obj)