

# Python-PrepTerm Quiz

<b>Code:</b>	MT2020140
--------------	-----------

1. What is the following function sorts a list?
  1. `list.reverse()`
  2. `list.sort([func])`
  3. `list.pop(obj=list[-1])`
  4. `list.remove(obj)`
2. Which of the following environment variable for Python is an alternative module search path?
  1. PYTHONPATH
  2. PYTHONSTARTUP
  3. PYTHONCASEOK
  4. PYTHONHOME
3. What is output for:  
`a = ['hat', 'mat', 'rat']`  
`'rhyme'.join(a)`
  1. `['hat','mat','rat','rhyme']`
  2. `'hatmatratrhyme'`
  3. `['hat mat rat rhyme']`
  4. `'hatrhymematr rhyme rat'`
4. For tuples and list which is correct?
  1. List and tuples both are mutable.
  2. List is mutable whereas tuples are immutable.
  3. List and tuples both are immutable.
  4. List is immutable whereas tuples are mutable.
5. What is output for `min("hello world")`
  1. e
  2. a blank space character

3. w
4. None of the above.

6. Analyze the code:

```
print("Recursive Function")
def factorial(n):
    return(n*factorial(n-1))
factorial(4)
```

1. Recursive Function 24.
  2. Recursive Function.
  3. Function runs infinitely and causes a StackOverflowError.
  4. Syntax Error.
7. Pylab is a package that combine \_\_\_\_\_, \_\_\_\_\_ and \_\_\_\_\_ into a single namespace.
1. Numpy, scipy and matplotlib
  2. Numpy, matplotlib and pandas
  3. Numpy, pandas and matplotlib
  4. Numpy, scipy and pandas
8. 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)`
9. 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
10. Is the following Python code valid?
- ```
try :
    # Do something
except :
    # Do something
finally :
    # Do something
```
1. no, there is no such thing as finally

2. no, finally cannot be used with except
  3. no, finally must come before except
  4. yes
11. 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.
12. What will be the output of the following code?
- ```
minidict = { 'name': 'TutorialsPoint', 'name': 'website' }  
print(minidict[ 'name' ])
```
1. TutorialsPoint
  2. Website
  3. ('TutorialsPoint', 'website')
  4. It will show an Error.
13. Which of the following function converts a string to all lowercase?
1. lower()
  2. lstrip ()
  3. **max(str)**
  4. **min(str)**
14. What will be the output of the following Python code?

```
try:  
    if '1' != 1:  
        raise "someError"  
    else:  
        print("someError has not occurred")  
except "someError":  
    print ("someError has occurred")
```

1. someError has occurred
  2. someError has **not** occurred
  3. invalid code
  4. none of the mentioned
15. 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'
16. What is the following function removes an object from a list?
1. `list.index(obj)`
  2. `list.insert(index, obj)`
  3. `list.pop(obj=list[-1])`
  4. `list.remove(obj)`
17. 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.
18. What is output of following code:
- ```
num=3
while True:
    if (num%0o12 == 0):
        break
    print(num)
    num += 1
```

1. 3 4 5 6 7 8 9 10 11 12
  2. 3 4 5 6 7 8 9
  3. 3 4 5 6 7 8 9 10 11
  4. None of the above
19. Name the error that doesn't cause program to stop/end, but the output is not the desired result or is incorrect.
1. Syntax error
  2. Runtime error
  3. Logical error
  4. All of the above
20. Essential thing to create a window screen using tkinter Python?
1. call tk() function
  2. create a button
  3. To define a geometry
  4. All of the above