Python-PrepTerm
Quiz

Code:	MT2020147

1. 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.

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. 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

4. Which of the following environment variable for Python contains the path of an initialization file containing Python source code?

    1. PYTHONPATH
    2. PYTHONSTARTUP
    3. PYTHONCASEOK

   4. PYTHONHOME

5. 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)

6. How to create a frame in Python?

   1. Frame = new.window()
   2. Frame = frame.new()
   3. Frame = Frame()
   4. Frame = window.new()

7. Which of the following function convert a string to a float in python?

   1. **int**(x [, base])
   2. **long**(x [,base])
   3. **float**(x)
   4. **str**(x)

8. Which of the following operator in python evaluates to true if the variables on either side of the operator point to the same object and false otherwise?

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

9. 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)

10. 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

11. 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.

12. What is output of following code:

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

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

13. 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

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 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

16. 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.

17. 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)

18. Which of the following function converts a string to all lowercase?

    1. lower()
    2. lstrip ()
    3. **max**(**str**)
    4. **min**(**str**)

19. 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.

20. What happens in the below code?

```
class A:
   def __init__(self, i=100):
      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 all the data fields of class A.
2. Class B needs an Argument.
3. The data field 'j' cannot be accessed by object b.
4. Class B is inheriting class A but the data field 'i' in A cannot be inherited.