Python-PrepTerm
Quiz

Code:	MT2020004

1. What is the output of **print str** $* 2$ **if str** $=$ 'Hello World!'?

    1. Hello World!Hello World!
    2. Hello World! $* 2$
    3. Hello World!
    4. None of the above.

2. 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

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. 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

5. 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

6. 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)

7. 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.

8. 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)

9. 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**

10. 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**)

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

```python
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

12. 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.

13. What is output of following code:

```python
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

14. 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

15. Is the following Python code valid?

```python
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

16. What is the output of **print** tinylist $* 2$ **if** tinylist $= [123, \text{'john'}]$?

   1. $[123, \text{'john'}, 123, \text{'john'}]\backslash$ lstinline
   2. $[123, \text{'john'}] * 2\backslash$ lstinline
   3. Error
   4. None of the above.

17. 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.

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

   print(type(1/2))

   1. <**class** 'float'>
   2. <**class** 'int'>
   3. NameError: '1/2'**is not** defined.
   4. 0.5

19. What is output for:

   a = ['hat', 'mat', 'rat']

   'rhyme'.join(a)

   1. ['hat','mat','rat','rhyme']
   2. 'hatmatratrhyme'
   3. ['hat mat rat rhyme']
   4. 'hatrhymematrhyme rat'

20. 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)