Python-PrepTerm
Quiz

Code:	MT2020061

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

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

2. 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

3. 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

4. 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

5. 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.

6. 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**

7. Which of the following environment variable for Python is an alternative module search path?

1. PYTHONPATH
2. PYTHONSTARTUP
3. PYTHONCASEOK
4. PYTHONHOME

8. 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

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

   def foo(): try: return 1 finally: return 2 k = foo() print(k)

   1. 1
   2. 2
   3. 3
   4. error, there is more than one return statement in a single try-finally block

10. 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)

11. 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.

12. 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'`

13. 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

14. How to create a frame in Python?

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

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

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

16. What is output of following code:

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

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

17. 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.

18. What is the output of the code?

```
def f():
    try:
        return(1)
    finally:
        return(2)
k=f()
print(k)
```

    1. 1 2
    2. 2 1
    3. 2
    4. Error

19. 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

20. nfig() in Python Tkinter are used for

    1. destroy the widget
    2. place the widget
    3. change property of the widget
    4. configure the widget