Python-PrepTerm
Quiz

Code:	MT2020163

1. 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

2. What is the output of **print str**[2:5] **if str** = `'Hello World!'`?

    1. llo World!
    2. H
    3. llo
    4. None of the above.

3. 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

4. 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

5. What will be the output of the below given code?

```
colors = ["white", "Black", "Grey"]
x = "Red" not in colors
```

   1. Yes
   2. No
   3. Error: not in not defined
   4. True

6. 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**)

7. 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

8. 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

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

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

10. 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)

11. What is output for **min**("hello world")

    1. e
    2. a blank space character
    3. w
    4. None of the above.

12. 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

13. 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'

14. 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

15. 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

16. 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])

17. Which of the following function of dictionary gets all the keys from the dictionary?

    1. getkeys()
    2. key()
    3. keys()
    4. None of the above.

18. 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.

19. What is the output of **print** tinylist * 2 **if** tinylist = [123, 'john']?

    1. [123, 'john', 123, 'john']\lstinline
    2. [123, 'john'] * 2\lstinline
    3. Error
    4. None of the above.

20. 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.