Python for Data Science
Asg 10.1 Tuples & Sets (MC) Name _____

You will be emailed a Google Forms Virtual Scantron to use to submit your answers.

1. Select all true statements regarding the Python tuple.

a. We can remove the item from tuple but we cannot update items of the tuple
b. We cannot delete the tuple
c. We cannot remove the items from the tuple
d. We cannot update items of the tuple.
e. Both c and d are correct

2. What is the output of the following code?

```
tuple1 = (1120, 'a')
print(max(tuple1))
```

a. TypeError
b. 1120
c. 'a'

3. Choose the correct way to access value 20 from the following tuple.

```
aTuple = ("Orange", [10, 20, 30], (5, 15, 25))
```

a. aTuple[1:2][1]
b. aTuple[1:2](1)
c. aTuple[1:2][1]
d. aTuple[1][1]

4. What is the output of the following tuple operation?

```
aTuple = (100, 200, 300, 400, 500)
aTuple.pop(2)
print(aTuple)
```

a. (100, 200, 400, 500)
b. (100, 300, 400, 500)
c. AttributeError

5. What is the output of the following tuple operation?

```
aTuple = (100,)
print(aTuple * 2)
```

a. TypeError
b. (100, 100)
c. (200)

6. Select which of the following options are true for Python tuples.

a. A tuple maintains the order of items
b. A tuple is unordered
c. We cannot change the tuple once created
d. We can change the tuple once created
e. Both a and c are correct

7. What is the output of the following?

```
aTuple = "Yellow", 20, "Red"
a, b, c = aTuple
print(a)
```

a. ('Yellow', 20, 'Red')
b. TyepeError
c. Yellow

8. What is the output of the following code?

```
aTuple = (100, 200, 300, 400, 500)
aTuple[1] = 800
print(aTuple)
```

a. TypeError
b. (100, 800, 200, 300, 400, 500)
c. (800, 100, 200, 300, 400, 500)

9. What is the type of the following variable?

```
aTuple = ("Orange")
print(type(aTuple))
```

a. list
b. tuple
c. array
d. str

10. What is the output of the following?

```
aTuple = (10, 20, 30, 40, 50, 60, 70, 80)
print(aTuple[2:5], aTuple[:4], aTuple[3:])
```

a. (30, 40, 50) (10, 20, 30, 40) (40, 50, 60, 70, 80)
b. (20, 30, 40, 50) (10, 20, 30, 40) (30, 40, 50, 60, 70, 80)

11. A Python tuple can also be created without using parentheses.

a. False
b. True

12. What is the output of the following code?

```
aTuple = (100, 200, 300, 400, 500)
print(aTuple[-2])
print(aTuple[-4:-1])
```

a. IndexError: tuple index out of range
b. 400
c. (200, 300, 400)
d. 400
   (200, 300, 400)
e. 400
   ( )

13. For tuples and lists which is correct?

a. Lists and tuples both are mutable.
b. Lists are mutable whereas tuples are immutable.
c. Lists and tuples both are immutable.
d. Lists is immutable whereas tuples are mutable

14. Which of the following are correctly declared tuples?

a. x = ("orange", "yellow", "red")
b. x = "orange", "yellow", "red"
c. x = ["orange", "yellow", "red"]
d. x = "orangeyellowred"
e. Both a and b are correct

15. Which line of code will give you an error? b = (4, 5, 6, 7, 8)

a. b[2]
b. b[0] = 1
c. b[:3]
d. b[-2]

16. How would you refer to 3 in the following tuple? c = ((7,5),(5,8),(0,-1),(4,3))

a. c[0][0]
b. c(3)
c. c[8]
d. c[3][1]

17. What is the index of the following tuple: y = ()

a. 0
b. 1
c. There is none. It is an empty tuple

18. What is the output of the following?

```
set1 = {10, 20, 30, 40, 50}
set2 = {60, 70, 10, 30, 40, 80, 20, 50}

print(set1.issubset(set2))
print(set2.issuperset(set1))
```

a. False

   False

b. True

   True

19. Which of the following choices is true?

a. Option#1
   - Sets are unordered
   - set doesn't allow duplicate
   - sets are written with curly brackets {}

b. Option #2
   - set object does support indexing
   - set is mutable

c. Both Option #1 and Option #2 are correct

d. Neither Option #1 nor Option #2 are correct

20. What is the output of the following set operation?

```
sampleSet = {"Yellow", "Orange", "Black"}
sampleSet.update(["Blue", "Green", "Red"])
print(sampleSet)
```

a. {'Yellow', 'Orange', 'Red', 'Black', 'Green', 'Blue'}
b. {'Yellow', 'Orange', 'Black', ["Blue", "Green", "Red"]}
c. {'Green', 'Black', 'Yellow', 'Blue', 'Red', 'Orange'}
d. TypeError: update() doesn't allow list as a argument.

21. What is the output of the following?

```
sampleSet = {"Yellow", "Orange", "Black"}
sampleSet.discard("Blue")
print(sampleSet)
```

a. {'Yellow', 'Orange', 'Black'}

b. KeyError: 'Blue'

22. What is the output of the following set operation.?

```
set1 = {"Yellow", "Orange", "Black"}
set2 = {"Orange", "Blue", "Pink"}

set1.difference_update(set2)
print(set1)
```

a. {'Black', 'Yellow'}
b. {'Yellow', 'Orange', 'Black', 'Blue', 'Pink'}

23. What is the output of the following set operation.?

```
set1 = {"Yellow", "Orange", "Black"}
set2 = {"Orange", "Blue", "Pink"}

set3 = set2.difference(set1)
print(set3)
```

a. {'Yellow', "Black', 'Pink', 'Blue'}
b. {'Pink', 'Blue'}
c. {'Yellow', "Black'}

24. The `isdisjoint()` method returns True if none of the items are present in both sets, otherwise, it returns False

a. True
b. False

25. What is the output of the following union operation?

```
set1 = {10, 20, 30, 40}
set2 = {50, 20, "10", 60}

set3 = set1.union(set2)
print(set3)
```

a. {40, 10, 50, 20, 60, 30}
b. {40, '10', 50, 20, 60, 30}
c. {40, 10, '10', 50, 20, 60, 30}
d. SynatxError: Different types cannot be used with sets

26. What is the output of the following code?

```
aSet = {1, 'PYnative', ['abc', 'xyz'], True}
print(aSet)
```

a. {1, 'PYnative', ['abc', 'xyz']}
b. {1, 'PYnative', ['abc', 'xyz'], True}
c. TypeError

27. The `union()` method returns a new set with all items from both sets by removing duplicates.

a. True
b. False

28. The `symmetric_difference()` method returns a set that contains all items from both sets, but not the items that are present in both sets.

a. True
b. False

29. What is the output of the following code?

```
sampleSet = {"Yellow", "Orange", "Black"}
print(sampleSet[1])
```

a. Yellow
b. Syntax Error
c. Orange
d. Type Error

30. Select the correct option(s) to remove “Orange” from the set.

```
sampleSet = {"Yellow", "Orange", "Black"}
```

a. sampleSet.pop("Orange")
b. sampleSet.discard("Orange")
c. del sampleSet ["Orange"]

31. What is the output of the following code?

```
aSet = {1, 'PYnative', ('abc', 'xyz'), True}
print(aSet)
```

a. TypeError
b. {‘PYnative’, 1, (‘abc’, ‘xyz’), True}
c. {‘PYnative’, 1, (‘abc’, ‘xyz’)}

32. What is the output of the following code?

```
sampleSet = {"Yellow", "Orange", "Black"}
sampleSet.add("Blue")
sampleSet.add("Orange")
print(sampleSet)
```

a. {‘Blue’, ‘Orange’, ‘Yellow’, ‘Orange’, ‘Black’}
b. {‘Blue’, ‘Orange’, ‘Yellow’, ‘Black’}

33. Select all the correct ways to copy two sets

Which of the following choices is true?

a. Option#1
   - set2 = set1.copy()
   - set2 = set(set1)
   - set2.update(set1)

b. Option #2
   - set2 = set1

c. Both Option #1 and Option #2 are correct

d. Neither Option #1 nor Option #2 are correct