What is the output of the following code?

```
string = "Hello World"
print(string[1:5])
```

- a. Hell
- b. Hello
- c. ello
- d. Hel

What does the strip() method do in Python?

- a. Removes all spaces from the string.
- b. Removes the first and last character of the string.
- c. Removes whitespace from the beginning and end of the string.
- d. Converts the string into a list of characters.

What is the output of the following code?

```
string = "Python Programming"
print(string.find("Pro"))
```

- a. 7
- b. 6
- c. -1
- d. None

What is the output of the following code?

```
string = "abcdef"
print(string[-3:])
```

- a. abc
- b. def
- c. cde
- d. ef

Which of the following methods is used to check if a string starts with a specific prefix?

- a. startswith()
- b. startswith_prefix()
- c. startswithwith()
- d. startswiths()

What is the output of the following code?

```
print(len("Python\nProgramming"))
```

- a. 19
- b. 18
- c. 17
- d. Error

\n is a whole character count

Which of the following is true about Python strings?

- a. Strings can be modified using indexing.
- b. Strings are immutable.
- c. Strings can only contain alphabetic characters.
- d. Strings must end with a newline character.

Which of the following methods is **not** a string method in Python?

- a. capitalize()
- b. startswith()
- c. pop()
- d. find()

What will be the result of the following code?

```
string = "Python"
print("-".join(string))
```

- a. P-y-t-h-o-n
- b. Python
- c. Error
- d. P-y-t-h-o-n-

What will the following code produce?

```
string = "hello WORLD"
print(string.swapcase())
```

- a. Hello world
- b. HELLO world
- c. hELLO world
- d. HELLO world

Which method is used to add an element to the end of a list?

- a. insert()
- b. append()
- c. extend()
- d. add()

What is the output of the following code?

```
my_list = [1, 2, 3]
my_list.extend([4, 5])
print(my_list)
```

- a. [1, 2, 3]
- b. [1, 2, 3, [4, 5]]
- c. [1, 2, 3, 4, 5]
- d. Error

What will be the result of the following code?

```
my_list = [1, 2, 3, 4]
my_list.insert(2, 10)
print(my_list)
```

- a. [1, 2, 3, 4]
- b. [1, 10, 2, 3, 4]
- c. [1, 2, 10, 3, 4]

What will be the output of the following code?

```
my_list = [1, 2, 3, 4, 5]
my_list.remove(3)
print(my_list)
```

- a. [1, 2, 3, 4, 5]
- b. [1, 2, 4, 5]
- c. [1, 2, 3, 4]

What happens if you attempt to access an index that is out of range in a list?

- a. It returns None.
- b. It raises an IndexError
- c. It creates a new element at that index.
- d. It returns an empty list.

What will the following code output?

```
my_list = [1, 2, 3]
my_list *= 2
print(my_list)
```

- a. [1, 2, 3]
- b. [1, 2, 3, 1, 2, 3]
- c. [2, 4, 6]
- d. Error

What will the following code output?

```
list1=[4,5,6]
list2=list1
list2[0]=3
print(list1)
```

- a. [4, 5, 6]
- b. [3, 5, 6,3]
- c. [3, 5, 6]
- d. [3,4, 5, 6]

What will the following code output?

```
my_list = [1, 2, [3, 4]]
my_list[2].append(5)
print(my_list)
```

- a. [4, 3, 2, 1]
- b. [4, 3, 2]
- c. [4, 3, 2, 1, 0]
- d. [4, 3, 2, 1, -1]

What will the following code output?

```
my_list = [10, 20, 30, 40]
print(my_list * 0)
```

- a. []
- b. [10, 20, 30, 40]
- c. Error
- d. None

What will this code output?

```
my_list = [1, 2, 3]
my_list.append([4, 5])
print(my_list)
```

- a. [1,2,3,4,5]
- b. [4,5,1,2,3]
- c. [[4,5],1,2,3]
- d. [1, 2, 3, [4, 5]]

What will the following code output?

```
my_list = [1, 2, 3, 4]
del my_list[1:3]
print(my_list)
```

- a. [1, 2, 3, 4]
- b. [1, 4]
- c. [1, 3, 4]
- d. Error

What will the following code output?

```
my_list = [ 1, 2, 3,0]
my_list.pop(0)
print(my_list)
```

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

What will the following code produce?

```
my_list = [ 0,1, 2, 3]
my_list.insert(8,12)
print(my_list)
```

- a. [0,1,2,3,8,12]
- B [0,1,2,3,8]
- c. [0, 1, 2, 3, 12]
- d. Error Index Not Found

What will be the output of the following code?

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

```
my_list = [ 0,1, 2, 3]
x=my_list.pop()
print(x)
```

What will be the output of the following code?

```
my_list = [0,2,1,1]
my_list.remove(1)
print(my_list)
```

- a. 2
- b. [0,2]
- c. [0, 2, 1]
- d. [0,1,1]

Which of the following is a valid way to create an empty tuple?

- a. empty_tuple = ()
- b. empty_tuple = tuple()
- c. empty_tuple = []
- d. Both A and B

What will the following code output?

```
t = (1, 2, 3)
t[1] = 4
```

- a. (1, 4, 3)
- b. TypeError
- c. [1, 4, 3]
- d. None

What will the following code output?

```
my_list = [0,2,1,[4,5]]
print(len(my_list))
```

- a. 4
- b. 5
- c. 6
- d. Error in Code

[4,5] count is one

What will the following code output?

```
my_list = [0,2,33,[4,5]]
print(len("my_list"))
```

- a. 4
- b. 5
- c. 6
- d. 7

Which of the following Statement will create a Tuple:

- a. t1=1,2,4
- b. t1=(1,)
- c. t1=tuple("123")
- d. All of these

What is the correct way to create a tuple with a single element?

- a. (1)
- b. (1,)
- c. [1]
- d. {1}

Which of the following methods can be used with a tuple?

- a. append()
- b. insert()
- c. index()
- d. remove()

What will the following code output?

```
tup = (1, 2, 3)
tup[1] = 5
print(tup)
```

- a. (1, 5, 3)
- b. (1, 2, 3)
- c. Error
- d. (1, 3)