Question1: what will be the output of below program?

```
if []:
    print("this")
else:
    print("that")
```

a. Error
b. This
c. That
d. None
e. No output

Question 2: What will be the output of below program?

```
print (None == True)
```

a. True
b. False
c. None
d. Error
e. No output

Question 3: Which statement of variable name is not correct in python?

a. 12_username
b. user_name
c. user_name_12
d. user_45_name

Question 4: What will be the output of the following code?

```
numbers = [1, 2, 3, 4, 5]
squared = map(lambda x: x ** 2, numbers)
print(list(squared))
```

a. [1, 4, 9, 16, 25]
b. [2, 4, 6, 8, 10]
c. [1, 3, 5, 7, 9]
d. [1, 2, 3, 4, 5]

Question 5. Which of the following statements is true about Python?

a) Python is a compiled language

b) Python is a low-level language

c) Python is dynamically typed

d) Python is primarily used for numeric computation

Question 6. What does the 'pass' statement do in Python?

a) Terminates the loop

b) Continues to the next statement inside

c) Skips the current iteration without doing anything

d) Breaks out of the loop immediately

Question 7. Which of the following statements is true about Python dictionaries?

a) Dictionaries are ordered collections of items

b) Dictionary keys must be immutable

c) Dictionary values must be unique

d) Dictionaries are accessed by index

Question 8. What will be the output of the following code?

```
x = 10

y = 3

print (x // y)
```

a) 1

b) 3

c) 0

d) 2

Question 9 . Which of the following is used to open a file in Python?

a) open()

b) read()

c) file_open()

d) fopen()

Question 10. What is the output of the following code?

```
x = 5

y = 2

print(x / y)
```

a) 2.5

b) 2

c) 2.0

d) 2.5 (depending on Python version)

Question 11. What will be the output of below code?

```
a = [1, 2, 3]
b = a
b.append(4)
print(a)
```

a) [1, 2, 3]

b) [1, 2, 3, 4]

c) [1, 2, 3, 4, 4]

d) Error

Question 12. What is the output of the following code?

```
def my_function(x=[]):
   x.append(1)
   return x
print(my_function() , end = "")
print(my_function())
```

a) [1] [1]

b) [1] [1, 1]

c) [1,1]

d) [1, 1] [1, 1]