Answers for Debugging Exercises: Chapter 8

Find the Output

1.

```
from math import pi
[str(round(pi, val)) for val in range(1, 5)]
print(list)
```

Ans. Rounds the value of PI to 1, 2, 3, 4 and 5 places of decimal, also makes a list of the values thus obtained

```
['3.1', '3.14', '3.142', '3.1416', '3.14159']
```

2.

```
colors = ['red', 'blue', 'green']
print(colors[2])
print(len(colors))
```

Ans. green, 3

3.

```
list = ['abc', 'def', 'ghi', 'jkl']
print(list[1:-1])
list[0:2] = 'xyz'
print(list)
```

Ans.

```
['def', 'ghi']
['xyz', 'ghi', 'jkl']
```

4.

```
list = ['abc', 'def', 'ghi', 'jkl', [1,2,3,4,5]]
print(list[4][2])
```

Ans. 3

5.

```
list = ['p','r','o','g','r','a','m','m','i','n','g']
print(list[2:5])
print(list[:-5])
```

```
print(list[5:])
print(list[:])
```

Ans.

```
['o', 'g', 'r']
['p', 'r', 'o', 'g', 'r', 'a']
['a', 'm', 'm', 'i', 'n', 'g']
['p', 'r', 'o', 'g', 'r', 'a', 'm', 'm', 'i', 'n', 'g']
```

6.

```
even = [2,4,6]
print(even + [10, 12, 14])
print(even*2)
even.insert(1,0)
print(even)
del even[2]
print(even)
```

Ans.

```
[2, 4, 6, 10, 12, 14]
[2, 4, 6, 2, 4, 6]
[2, 0, 4, 6]
[2, 0, 6]
```

7.

```
list = ['p','r','o','g','r','a','m']
list.remove('p')
print(list)
print(list.pop(1))
print(list)
print(list.pop())
print(list)
```

Ans.

```
['r', 'o', 'g', 'r', 'a', 'm']
o
```

```
['r', 'g', 'r', 'a', 'm']
m
['r', 'g', 'r', 'a']
```

8.

```
list = [9,4,3,8,0,2,3,6]
print(list.index(3))
print(list.count(8))
list.sort()
print(list)
list.reverse()
print(list)
print(0 in list)
```

Ans.

```
2
1
[0, 2, 3, 3, 4, 6, 8, 9]
[9, 8, 6, 4, 3, 3, 2, 0]
True
```

9.

```
list = [2 ** x for x in range(5)]
print(list)
```

Ans. `[1, 2, 4, 8, 16]`

10.

```
countries = ['India', 'Sri Lanka', 'New Zealand', 'Japan',
'Russia']
for index, country in enumerate(countries):
    print("The country, " + country + ", is at position " +
str(index) + ".")
```

Ans.

```
The country, India, is at position 0.
The country, Sri Lanka, is at position 1.
```

```
The country, New Zealand, is at position 2.
The country, Japan, is at position 3.
The country, Russia, is at position 4.
```

11.

```
list = [(1, 2), [3, 4], '56', 78, 9.0]
print(list[0], type(list[0]))
print(list[2:3], type(list[0:1]))
print(list[2], type(list[2]))
```

Ans.

```
((1, 2), <type 'tuple'>)
(['56'], <type 'list'>)
('56', <type 'str'>)
```

12.

```
words = 'Welcome to the world of Programming'.split()
msg = [[word.upper(), word.lower(), len(word)] for word in
words]
for i in msg:
   print(i)
```

Ans.

```
['WELCOME', 'welcome', 7]
['TO', 'to', 2]
['THE', 'the', 3]
['WORLD', 'world', 5]
['OF', 'of', 2]
['PROGRAMMING', 'programming', 11]
```

13.

```
item = [x+y for x in 'cup' for y in 'pen']
print(item)
```

Ans.

```
['cp', 'ce', 'cn', 'up', 'ue', 'un', 'pp', 'pe', 'pn']
```

14.

```
print([x+y for x in 'cup' for y in 'pen' if x != 't' and y !=
'o' ])
```

Ans.

```
['cp', 'ce', 'cn', 'up', 'ue', 'un', 'pp', 'pe', 'pn']
```

15.

```
list =[ [1,2]*3 ] *4

print(list)
```

Ans.

```
[[1, 2, 1, 2, 1, 2], [1, 2, 1, 2, 1, 2], [1, 2, 1, 2, 1, 2], [1, 2,
1, 2, 1, 2]]
```

16.

```
list = [10, 20, 30, 40, 50, 60, 70, 80, 90]

print(list[-4:-1])

print(list[-1:-4])

print(list[-5:])

print(list[-6:-2:2])

print(list[::-1])
```

Ans.

```
[60, 70, 80]

[]

[50, 60, 70, 80, 90]

[40, 60]

[90, 80, 70, 60, 50, 40, 30, 20, 10]
```

17.

```
list = [[10, 20, [30, 40, [50, 60]]]]

print(list[0])

print(list[0][2])

print(list[0][2][2])

print(list[0][0])

print(list[0][2][1])

print(list[0][2][2][0])
```

Ans.

```
[10, 20, [30, 40, [50, 60]]]

[30, 40, [50, 60]]
```

```
[50, 60]

10

40

50
```

18.
```
List = [100, 90, 80, 70, 60, 50]
List[2] = List[1] - 20
if 30 in List:
  print(List[3])
else:
  print(List[4])
```

Ans. 60

19.

```
List = list(range(2, 20, 3))

print(List[5])
```

Ans. 17

20.

```
List = [-5, -3, 0, 3, 6]

print([x*2 for x in List])

print([x for x in List if x >= 0])
```

Ans.

```
[-10, -6, 0, 6, 12]

[0, 3, 6]

[5, 3, 0, 3, 6]
```

21.
```
print([(x, x*2) for x in range(5)])
```
Ans. `[(0, 0), (1, 2), (2, 4), (3, 6), (4, 8)]`

22.

```
List = [[1,2,3], [4,5,6], [7,8,9]]

print([val for x in List for val in x])
```

Ans. `[1, 2, 3, 4, 5, 6, 7, 8, 9]`

23.

```
DC = [-100, 0, 32, 40, 100]

DF = map(lambda temp: (9.0/5)*temp + 32, DC)
```

```
print(DF)
```

Ans. `[-148.0, 32.0, 89.6, 104.0, 212.0]`

24.

```
List = [1,2,3,4,5,6,7,8,9,10]
print(list(filter(lambda x: x % 4 == 0, List)))
print(list( map(lambda x: x * 2 + 5, List))
print(reduce(lambda x, y: x + y, List))
```

Ans.

```
[4, 8]
[7, 9, 11, 13, 15, 17, 19, 21, 23, 25]
55
```

25.

```
Tup = ("abc", "def")
(key, value) = Tup
print(key, value)
```

Ans. `abc def`

26.

```
Tup = (1,2,3)
Add_Tup = Tup + Tup
print(Add_Tup)
Mul_Tup = Tup * 3
print(Mul_Tup)
```

Ans.

```
(1, 2, 3, 1, 2, 3)
(1, 2, 3, 1, 2, 3, 1, 2, 3)
```

27.

```
msg = "HelloWorld"
pairs = []
for i in range(1, len(msg), 2):
    first = msg[i - 1]
    second = msg[i]
```

```
        pairs.append((first, second))
    for item in pairs:
        print(item)
```

Ans.

```
('H', 'e')
('l', 'l')
('o', 'W')
('o', 'r')
('l', 'd')
```

28.

```
Tup  = (1, 'abc')
List = [1, 'abc']
print(Tup == List)
print(Tup == tuple(List))
print(list(Tup) == List)
print((1, 2) + (3, 4))
```

Ans.

```
False
True
True
(1, 2, 3, 4)
```

29.

```
list = ['Good', 'Morning']
y, x = list
print(x, y)
```

Ans. Morning Good

30.

```
A = ('Chinu', 30, 'Female')
B = ('Varun', 32, 'Male')
for i in [A, B]:
    print('%s is a %d year old %s' %i)
```

Ans.

```
Chinu is a 30 year old Female
Varun is a 32 year old Male
```

31.

```
Tup = ('Good',)
for i in range(4):
    Tup = (Tup,)
    print(Tup)
```

Ans.

```
(('Good',),)
(('Good',),),)
((('Good',),),),)
(((('Good',),),),),)
```

32.

```
Tup1='a','bcd',12.34
Tup2=Tup1,(5,6,7,8)
print(Tup2)
```

Ans. `(('a', 'bcd', 12.34), (5, 6, 7, 8))`

33.

```
Tup = (1, 2, [3, 4])
Tup[2][0] = 5
print(Tup)
```

Ans. `(1, 2, [5, 4])`

```
34. Tup = ("Good Morning")
print(Tup.index('M'), end = ' ')
print(Tup.index('n', 5))
print(Tup.index('r',4,8))
```

Ans. `5, 8, 7`

35.

```
 IT_studs = set(['Dev', 'Era', 'Francis', 'Geet'])
```

```
Elec_studs = set(['Geet', 'Harman', 'Susan', 'Janak'])

CS_studs = set(['Era', 'Francis', 'Susan', 'Krishnav'])

students = IT_studs | Elec_studs | CS_studs

print("Students : ", students)

It_Elec_studs = IT_studs & Elec_studs

CS_studs.add('Loveya')

print("Is Students Superset of IT : ",
students.issuperset(IT_studs))

CS_studs.update(Elec_studs)

print("CS Students : ", CS_studs)
```

Ans.

```
Students :  set(['Janak', 'Susan', 'Krishnav', 'Dev', 'Geet',
'Harman', 'Era', 'Francis'])

Is Students Superset of IT :  True

CS Students :  set(['Janak', 'Susan', 'Krishnav', 'Loveya', 'Geet',
'Harman', 'Era', 'Francis'])
```

36.

```
x = {1, 2, 3, 4, 5}

y = {4, 5, 6, 7, 8}

print(x.difference(y))

print(y.symmetric_difference(x))

x.difference_update(y)

print(x)
```

Ans.

```
set([1, 2, 3])

set([1, 2, 3, 6, 7, 8])

set([1, 2, 3])
```

37.

```
x = set()

x.add("abc")

x.add("def")

x.update(["ghi","jkl"])
```

```
print(x)
```

Ans. `set(['jkl', 'abc', 'ghi', 'def'])`

38.

```
Dict = {"India":"New Delhi", "Nepal":"Kathmandu"}

Dict1 = {"USA":"Washington DC"}

Dict.update(Dict1)

print(Dict)
```

Ans. `{'Nepal': 'Kathmandu', 'India': 'New Delhi', 'USA': 'Washington DC'}`

39.

```
Dict = {"India":"New Delhi", "Nepal":"Kathmandu",
"USA":"Washington DC"}

del Dict["Nepal"]

for key,val in Dict.items():

    print(key,val)
```

Ans.

```
India New Delhi

USA Washington DC
```

40.

```
Dict = {"India":"New Delhi", "Nepal":"Kathmandu",
"USA":"Washington DC"}

print(Dict.get("Russia"))

print(Dict.get("Pakistan", "No Idea"))
```

Ans.

```
None

No Idea
```

41.

```
Studs = {'Mitanshi', 'Harshita', 'Pritika'}

Toppers = {}.fromkeys(Studs, 0)

print(Toppers)

Toppers['Mitanshi'] = 97
```

```
Toppers['Harshita'] = 92

Toppers['Pritika'] = 89

Toppers.setdefault('Nisha', -1)

print(Toppers)
```

Ans.

```
{'Pritika': 0, 'Harshita': 0, 'Mitanshi': 0}

{'Pritika': 89, 'Harshita': 92, 'Nisha': -1, 'Mitanshi': 97}
```

42.

```
Toppers = {}

Toppers['Mitanshi'] = 97

Toppers['Harshita'] = 92

Toppers['Pritika'] = 89

print('Harshita got ' + str(Toppers.get('Harshita')) + '
marks.')
```

Ans. Harshita got 92 marks.

43.

```
rec = {'Name': {'First': 'Chaitanya', 'Last': 'Raj'},

                'Marks': [80, 76, 84],

                'Course': 'BTech'}

print(rec['Name'])

print(rec['Name']['Last'])

print(rec['Marks'])

rec['Marks'].append(72)

print(rec)
```

Ans.

```
{'Last': 'Raj', 'First': 'Chaitanya'}

Raj

[80, 76, 84]

{'Course': 'BTech', 'Name': {'Last': 'Raj', 'First': 'Chaitanya'},
'Marks': [80, 76, 84, 72]}
```

44.

```
List = [-10,20,-30,40,-50]
```

```
if all([abs(i)<30 for i in List]):
    print("Hi")
else:
    print("Bye")
```

Ans. Bye

45.

```
def add_two(x):
      return x+2
List = [10,20,30,40,50]
result = list(map(add_two,List))
print(result)
```

Ans. [12, 22, 32, 42, 52]

46.
```
List = [13,26,39,52,64]
print(list(filter(lambda x:x%2==1,List)))
```

Ans. [13,39]

47.

```
str = "abcdefghijklmno"
for i in range(0, len(str), 2):
     print(str[i], end = " ")
```

Ans. a c e g i k m o

48.
```
print([ord(ch) for ch in 'PYTHON'])
```

Ans. [80, 89, 84, 72, 79, 78]

Find the Error

1.
```
list = ['abc', 'def', 'ghi', 'jkl']
print(list[2.0])
```

Ans. TypeError: list indices must be integers, not float

2.

```
even = [2,4,6]
del even
```

```
print(even)
```

Ans. NameError: name 'even' is not defined

3.
```
list = [(1, 2), [3, 4], '56', 78, 9.0]
list.remove('abc')
```

Ans. ValueError: list.remove(x): x not in list

4.
```
msg = "Hello"
msg.append("World")
print(msg)
```

Ans. AttributeError: 'str' object has no attribute 'append'

5.
```
tup = ("abc", "def", "ghi", "jkl")
tup.append("mno")
```

Ans. AttributeError: 'tuple' object has no attribute 'append'

6.
```
tup.remove("abc")
```

Ans. AttributeError: 'tuple' object has no attribute 'remove'

7.
```
Tup = ('abc', 'def', 'ghi','jkl')
Tup[2] = 'xyz'
```

Ans. TypeError: 'tuple' object does not support item assignment

8.
```
x, y = 10, 20, 30
```

Ans. ValueError: too many values to unpack

9.
```
x = {1, 2, 3, 4, 5}
   x.add([6,7,8])
   print(x)
```

Ans. TypeError: unhashable type: 'list'

10.
```
Dict = {[02,89, 85]:"PCM"}
print(Dict)
```

Ans. TypeError: unhashable type: 'list'

11.
```
Dict = {"India":"New Delhi", "Nepal":"Kathmandu"}
print(Dict["USA"])
```

Ans. KeyError: 'USA'

12.
```
Dict = {}
print(Dict[0])
```

Ans. KeyError

13.
```
Tup1 = (9,8,7,6,5)
Tup2 = (1,2,3,4,5)
print(Tup1 - Tup2)
```

Ans. TypeError: unsupported operand type(s) for -: 'tuple' and 'tuple'