Answers for Debugging Exercises: Chapter 5

Find the Output

1.

```
num = 10
def show():
        var = 20
        print "In Function var is - ", num

show()
print "Outside function, var is - ", num
```

Ans.

```
In Function var is - 20

Outside function, var is - 10
```

2.

```
def f():
    s = "Hello World!"
    print(s)

s = "Welcome to Python Programming"
f()
print(s)
```

Ans. Hello World!
Welcome to Python Programming

3.

```
def f():
        global var
        print(varr)
        var = 10
        print(var)
var = 100
f()
```

Ans.
100
10

4.

```
def display (str):

      print(str+"!")


display ("Hello World")
```

Ans. Hello World!

5.

```
def sqr(x):
```

```
      print(x*x)
sqr(10)
```

Ans. 100

6.

```
def mul_twice(x,y):
    print(x*y)
    print(x*y)
mul_twice(5, 10)
```

Ans.

```
50
50
```

7.

```
def func():
    global x
    print("x =", x)
    x = 100
    print('x is now = ', x)

x = 10
func()
print('x =', x)
```

Ans.

```
x = 10
x is now =  100
x = 100
```

8.

```
def func1():
    var = 3
    func2(var)

def func2(var):
    print(var)

func1()
```

Ans. 3

9.

```
def func(x):
      print('x = ', x)
      x = 100
      print('In Function, x after modification = ', x)

x = 50
func(x)
print('Outside Function, x = ', x)
```

Ans.

```
x =  50
In Function, x after modification =  100
Outside Function, x =  50
```

10.

```
def display( str ):
   print str
   return
```

```
display("Hello World !!")
display("Welcome to Python Programming")
```

Ans.

```
Hello World !!
Welcome to Python Programming
```

11.

```
def sum(num1, num2):
        total = num1 + num2
        print("Inside function, Total = ", total)
        return total

total = sum(10, 20)
print("Outside the function, Total = ", total)
```

Ans.

```
Inside function, Total =  30
Outside the function, Total =  30
```

12.

```
def min(x,y):
    if x<y:
        return x
    else:
        return y
print(min(4, 7))
```

Ans. 4

13.

```
def add(x, y):
    sum = x + y
    return sum
    print("This won't be printed")
print(add(10,20))
```

Ans. 30

14.

```
def display(str):
        "This prints a passed string into this function"
        print(str)
        return

display(str = "Welcome")
```

Ans. Welcome

15.

```
def say(message, repeat_it = 2):
      print(message * repeat_it)
say('Hello')
say('Hello', 5)
```

Ans.

```
Hello Hello
Hello Hello Hello Hello Hello
```

16.

```
def func(x, y = 100, z = 1000):
    print('x = ', x, ' y = ', y, 'and z = ', z)

func(5, 15, 25)
func(35, z = 55)
func(y = 70, x = 200)
```

Ans.

```
x =  5  y =  15 and z =  25
x =  35  y =  100 and z =  55
x =  200  y =  70 and z =  1000
```

17.

```
def greet(*names):

      for name in names:

               print("Hello",name)
greet("Aryan","Nikita","Cahitanya")
```

Ans.

```
('Hello', 'Aryan')

('Hello', 'Nikita')

('Hello', 'Cahitanya')
```

18.

```
def func( arg1, *var):

      "This prints arbitrary arguments"

      print(arg1,end=' ')
      for i in var:
            print(i, end=' ')
      return


func("Score is : ", 10, 20, 30)

func( "\n Average Score = ", 20)
```

Ans.

```
Score is :  10 20 30
 Average Score =  20
```

19.

```
expo_3 = lambda x: x ** 3
print(expo_3(5))
```

Ans. 125

20.

```
add_five = lambda n: n + 5
mult_add_five = lambda n: add_five(n * 10)
print(mult_add_five(9))
```

Ans. 95

21.

```
def func():
    """Do nothing.

     Nothing doing.
     """
    pass

print(func.__doc__)
```

Ans.

```
Do nothing.

   Nothing doing.
```

22.

```
def C_to_F(c):
          return c * 9/5 + 32

print(C_to_F(37))
```

Ans.

```
98
```

23.

```
def pow(x, y=3):
    r = 1
   for i in range(y):
      r = r * x
      return r

print(pow(5))
print(pow(2, 5))
```

Ans. 125

```
      32
```

24.

```
def display(name, deptt, sal):
      print "Name: ", name
      print "Department: ", deptt
      print "Salary: ", sal

display(sal = 100000, name="Tavisha", deptt = "IT")
display(deptt = "HR", name="Dev", sal = 50000)
```

Ans.

```
Name:  Tavisha
Department:  IT
Salary:  100000
Name:  Dev
Department:  HR
Salary:  50000
```

25.

```
def display(mesg):
        return mesg + "!"
print_str = display
str  = print_str("Hello")
print(str)
```

Ans. Hello!

26.

```
from random import randint as r
for i in range(10):
        value = r(1,100)
        print value,
```

Ans. 10 Random numbers

27.

```
print((lambda x:x**2+5*x+6)(-3))
```

Ans. 0

28.

```
double =lambda x:x*2
sub=lambda x,y:x-y
print(sub(double(5),9))
```

Ans. 1

29.

```
def is_even(x):
    if x==0:
        return True
    else:
        return is_odd(x-1)

def is_odd(x):
    return not is_even(x)

print(is_even(22))
```

Ans. True

30.

```
def display(x):
    for i in range(x):
        print(i)
        return
display(5)
```

Ans. 0

Find the Error

1.

```
def func():
        print("Hello World")
```

Ans. NameError: name func is not defined

2.

```
var1 = "Good"
def show():
    var2 = "Morning"
    print(var1)
    print(var2)
show()
print(var1)
print(var2)
```

Ans. NameError

3.

```
def f():
```

```
        print var

        var = 10

        print(var)

var = 100

f()
```

Ans. UnboundLocalError: local variable 'str' referenced before assignment

4.
```
def f():
          var = 100
          print(var)

      f()
      print(var)
```

Ans. NameError: name 'var' is not defined

5.
```
def func(var):

      var+=1

      var *= 2

      print(var)

func(9)

print(var)
```

Ans. NameError: name 'variable' is not defines

6.
```
def func1():
    var = 3
    func2()

def func2():
    print(var)

func1()
```

Ans. NameError: global name 'var' is not defined

7.
```
def display(x,y):

   print(x+y)
```

```
display(10)
```

Ans. TypeError: display(x,y) takes exactly 2 argument (1 given)

8.

```
def func(a, b):
   print(a)
   print(b)
func(b=10, 20)
```

Ans. SyntaxError: non-keyword arg after keyword arg

9.

```
def func1():
      print("func1()")
func1()
func2()
def func2():
      print("func2()")
```

Ans. NameError: name 'func2' is not defined

10.

```
import math as m
   print(math.sqrt(25))
```

a. Error b. 25 c. 5 d. 625

Ans. (a) Error

11.

```
def factorial(x):
      return x*factorial(x-1)
print(factorial(6))
```

Ans. RuntimeError: maximum recursion depth exceeded

12.

```
   def sum_to(x):
         return x+sum_to(x-1)
      print(sum_to(5))
```

Ans. RuntimeError: maximum recursion depth exceeded