Fundamentals of Python Programming

Richard L. Halterman

Chapter 3 exercises

1. Is the literal 4 a valid Python expression? Yes
2. Is the variable x a valid Python expression? Yes
3. Is x + 4 a valid Python expression? Yes
4. What affect does the unary + operator have when applied to a numeric expression?
   when unary+operator applied to a numeric value, variable, or expression, the resulting value is no different from the original value of its operand.
5. Sort the following binary operators in order of high to low precedence: +, -, *, //, %,/,=.
   Unary +,-
   Binary *,//,/,%
   Binary +,-
   Binary = .
6. Given the following assignment:

× = 2

Indicate what each of the following Python statements would print.

1. print ("x")
2. print ('x')
3. print (x)
4. print ("× + 1")
5. print('x' + 1)
6. print (x + 1)

1. x
2. x
3. 2
4. x+1
5. error
6. 3

7. Given the following assignments:

i1 = 2
i2 = 5
i3 =-3
d1=2.0
d2= 5.0
d3 = -0.5

Evaluate each of the following Python expressions.

(a) i1 + i2
(b) i1 / i2
(c) i1 // i2
(d) i2 / i1
(e) i2 // i1
(f) i1 * i3
(g) d1 + d2
(h) d1 / d2
(i) d2 / d1
(j) d3 * d1
(k) d1 + i2
(l) i1 / d2
(m) d2 / i1
(n) i2 / d1
(o) i1/i2*d1
(p) d1*i1/i2

(q) d1/d2*i1
(r) i1*d1/d2
(s) i2/i1*d1
(t) d1*i2/i1
(u) d2/d1*i1
(v) i1*d2/d1

(a) 5
(b) 0.4
(c) 0
(d) 2.5
(e) 2
(f) -6
(g) 7.0
(h) 0.4
(i) 2/5
(j) -1
(k) 7.0
(l) 4.0
(m)0.25
(n) 2.5
(o) 0.8
(p) 0.8
(q) 0.8
(r) 8.0
(s) 5.0
(t) 5.0
(u) 5.0
(v) 5.0

8. What is printed by the following statement:
#print(5/3)

=1.66

9. Given the following assignments:
i1 = 2
i2 = 5
i3 = -3
d1 = 2.0
d2 = 5.0
d3 = -0.5

_Evaluate each of the following Python expressions.64
(a) i1 + (i2 * i3)
(b) i1 * (i2 + i3)
(c) i1 / (i2 + i3)
(d) i1 // (i2 + i3)
(e) i1 / i2 + i3
(f) i1 // i2 + i3
(g) 3 + 4 + 5 / 3
(h) 3 + 4 + 5 // 3
(i) (3 + 4 + 5) / 3
(j) (3 + 4 + 5) // 3
(k) d1 + (d2 * d3)
(l) d1 + d2 * d3
(m) d1 / d2 - d3
(n) d1 / (d2 - d3)
(o) d1 + d2 + d3 / 3

(p) (d1 + d2 + d3) /3
(q) d1 + d2 + (d3 / 3)
(r) 3 * (d1 + d2) * (d1 - d3)

(a)-13
(b) 4
(c) 1.0
(d) 1
(e)-2.6
(f) -3
(g) 8
(h) 8
(i) 4
(j) 4
(k)-0.5
(l) -0.5
(m)0.9
(n) 0.36
(o) 6.84
(p) 2.16
(q) 6/84
(r) -31.5

10. What symbol signifies the beginning of a comment in Python?#
11. How do Python comments end?
The comment is in effect until the end of the line of code

12.Which is better, too many comments or too few comments?
In general, programmers are not prone to providing too many comments.
When in doubt, add a remark. The extra time it takes to write good comments is well worth the effort.

13. What is the purpose of comments?
Good programmers annotate their code by inserting remarks that explain the purpose of a section of code or why they chose to write a section of code the way they did.

14.Why is human readability such an important consideration?
Since people develop and read source code, human readability is important

15.What circumstances can cause each of the following run-time errors to arise?
• NameError
• ValueError
• ZeroDivisionError
• IndentationError
• OverflowError
• SyntaxError
• TypeError

• NameError: This error occurs if a variable is not assigned.
• ValueError: This error occurs if you pass an invalid value to the function. This error message usually occurs in math operations that require a specific type of value.
• ZeroDivisionError: Division by zero is undefined in mathematics, and division by zero in Python is illegal
• IndentationError: Incorrect indentation of Python commands causes an IndentationError.
• OverflowError: This error appears when the result of an arithmetic operation is greater than what is displayed
• SyntaxError: by the parser and occurs when a syntax error occurs
• TypeError: This error occurs if a function or operation is applied to a wrong variable

Hint: Try some of following activities in the interpreter or within a Python program:
○ print a variable that has not been assigned (NameError)
○ convert the string 'two' to an integer (NameError)
○ add an integer to a string (TypeError)
○ assign to a variable named end-point (SyntaxError)

- experiment adding spaces and tabs at various places in the code of an error-free Python program (IndentationError)
- compute raise a floating-point number to a large power, as in 1.510,000. (OverflowError)

16. Consider the following program which contains some errors. You may assume that the comments within the program accurately describe the program's intended behavior.

```
# Get two numbers from the user
n1 = float(input())                    #1
n2 = float(input())                    #2
# Compute sum of the two numbers
print(n1 + n2)                          #3
# Compute average of the two numbers
print(n1+n2/2)                          #4
# Assign some variables
d1=d2=0                                #5
# Compute a quotient
print(n1/d1)                           #6
# Compute a product
n1*n2 = d1                             #7
# Print result
print(d1)                              #8
```

For each line listed in the comments, indicate whether or not an interpreter error, run-time exception , or logic error is present. Not all lines contain an error.

Line 6 (ZeroDivisionError)   Line 8 (SyntaxError)

17. Write the shortest way to express each of the following statements.
(a) x = x + 1
(b) x = x / 2
(c) x = x – 1
(d) x = x + y
(e) x = x - (y + 7)
(f) x = 2*x
(g) number_of_closed_cases = number_of_closed_cases + 2*ncc

(a) x+=1
(b) x/=2
(c) x-=1
(d) x+=y
(e) x-=y+7
(f) x*=2
(g) number_of_closed_cases+=2*ncc

18. What is printed by the following code fragment?

```
x1 = 2
x2 = 2
x1 += 1
x2 -= 1
print(x1)
print(x2)
```

Why does the output appear as it does?

3
1

Python provides a more general way of simplifying a statement that modifies a variable through simple arithmetic.like this question.actually x+=1 == x=x+1

19. Consider the following program that attempts to compute the circumference of a circle given the radius entered by the user. Given a circle’s radius, r, the circle’s circumference, C is given by the formula:

C = 2πr

```
r=0
PI = 3.14159
# Formula for the area of a circle given its radius
C = 2*PI*r
# Get the radius from the user
r = float(input("Please enter the circle's radius: ")) # Print the circumference
print("Circumference is", C)
```

(a) The program does not produce the intended result. Why?
(b) How can it be repaired so that it works correctly

(a)Because in this program, the radius is always zero

(b)
```
r=float (input())
    PI = 3.14159
    C = 2*PI*r
    print("Circumference is", C)
```

20.write a python program that calculate the area of a circle.

```
r=float (input())
PI = 3.14159
C = PI*(r**2)
print(C)
```

21.write a python program that calculate the the volume of cube.

```
a = int(input())
b= int (input())
c=int(input())
v=a*b*c
Print(v)
```