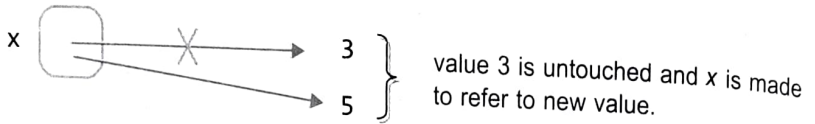


5. What do you understand by term 'immutable' ?

Solution. Immutable means unchangeable. In Python, immutable types are those whose value cannot be changed in place. Whenever one assigns a new value to a variable referring to immutable type, variable's reference is changed and the previous value is left unchanged. e.g.,

```
x = 3
x = 5
```



6. What will be the output of the following ?

```
print (len(str(17//4)) )
print (len(str(17/4)) )
```

Solution. 1
3

because

```
len(str(17//4))
= len(str(4))
= len('4')
= 1
```

and

```
len(str(17/4))
= len(str(4.0))
= len('4.0')
= 3
```

7. What will be the output produced by these ?

- (a) 12/4 (b) 14//14 (c) 14%4 (d) 14.0/4 (e) 14.0//4 (f) 14.0%4

Solution. (a) 3.0 (b) 1 (c) 2 (d) 3.5 (e) 3.0 (f) 2.0

8. Given that variable CK is bound to string "Raman" (i.e., CK = "Raman"). What will be the output produced by following two statements if the input given in "Raman" ? Why ?

```
DK = input("Enter name:")
Enter name : Raman
```

- (a) DK == CK (b) DK is CK

Solution. The output produced will be as : (a) True (b) False

The reason being that both DK and CK variable are bound to identical strings 'Raman'. But input strings are always bound to fresh memory even if they have value identical to some other existing string in memory.

Thus DK == CK produces True as strings are identical.

But DK is CK produces False as they are bound to different memory addresses.

9. What will be the output of following code ? Explain reason behind output of every line :

```
5 < 5 or 10
5 < 10 or 5
5 < (10 or 5)
5 < (5 or 10)
```

Solution.

- 10
True
True
False

Explanation

| | | |
|--------|---|---|
| Line 1 | 5 < 5 or 10 = False or 10 = 10 | precedence of < is higher than or because or would evaluate the second argument if first argument is False or false _{total} |
| Line 2 | 5 < 10 or 5 = True or 5 = True | precedence of < is higher than or or would return first argument if it is True or true _{total} |
| Line 3 | 5 < (10 or 5) = 5 < 10 = True | 10 or 5 returns 10 since 10 is true _{total} |
| Line 4 | 5 < (5 or 10) = 5 < 5 = False | 5 or 10 returns 5 since 5 is true _{total} |

10. What will be output produced by the three expressions of the following code ?

```
a = 5
b = -3
c = 25
d = -10
a + b + c > a + c - b * d
str(a + b + c > a + c - b * d) == 'true'
len(str(a + b + c > a + c - b * d)) == len(str(bool(1)))
```

Solution. True
False
True

11. What would Python produce if for the following code, the input given is

- (i) 11 (ii) hello (iii) just return key pressed, no input given (iv) 0 (v) 5-5

Code

```
bool(input("Input:")) and 10 < 13 - 2
```

Solution.

- (i) Input : 11 would yield
bool ('11') and 10 < 11
True and 10 < 11
= **True**
- (ii) **True** [For the same reason as in (i)]
- (iii) **False** because when just return key is pressed, input is " i.e., empty string, hence expression becomes
bool(' ') and 10 < 11
False and True
= **False**

- (iv) bool ('0') and 10 < 11
True and True
= **True**

(Please note '0' is a non-empty string and hence has truth value as true_{total})
- (v) bool ('5-5') and 10 < 11
= True and 10 < 11
= **True**

12. What would be the output of the following code ? Explain reason(s).

```

a = 3 + 5/8
b = int(3 + 5/8)
c = 3 + float(5/8)
d = 3 + float(5)/8
e = 3 + 5.0/8
f = int(3 + 5/8.0)
print (a, b, c, d, e, f)

```

Solution. The output would be

3.625 3 3.625 3.625 3.625 3

Explanation

Line 1 $a = 3 + 5/8$
 $= 3 + 0.625$
 $\therefore a = 3.625$

Line 2 $b = \text{int}(3 + 5/8)$
 $= \text{int}(3 + 0.625)$ (int() will drop the fractional part)
 $= \text{int}(3.625)$
 $b = 3$

Line 3 $c = 3 + \text{float}(5/8)$
 $c = 3 + \text{float}(0.625)$
 $= 3 + 0.625$
 $c = 3.625$

Line 4 $d = 3 + \text{float}(5)/8$
 $= 3 + 5.0/8$ $5.0/8 = 0.625$ because one operand is floating point, the
 $= 3 + 0.625$ integer operand will be internally converted to floating-pt
 $d = 3.625$

Line 5 $e = 3 + 5.0/8$
 $= 3 + 0.625$ (same reason as above)
 $e = 3.625$

Line 6 $f = \text{int}(3 + 5/8.0)$ (same reason as above)
 $f = \text{int}(3 + 0.625)$
 $f = \text{int}(3.0)$ (int() will drop the fractional part)
 $f = 3$

13. What will be the output produced by following code statements ?

(a) $87 // 5$

(b) $87 // 5.0$

(c) $(87 // 5.0) == (87 // 5)$

(d) $(87 // 5.0) == \text{int}(87 / 5.0)$

(e) $(87 // \text{int}(5.0)) == (87 // 5.0)$

Solution. (a) 17 (b) 17.0 (c) True (d) True (e) True

14. What will be the output produced by following code statement ? State reasons(s).

- (a) `17 % 5` (b) `17 % 5.0`
 (c) `(17 % 5) == (17 % 5)` (d) `(17 % 5) is (17 % 5)`
 (e) `(17 % 5.0) == (17 % 5.0)` (f) `(17 % 5.0) is (17 % 5.0)`

Solution. (a) 2 (b) 2.0 (c) True (d) True (e) True (f) False

Both (c) and (e) evaluate to **True** because both the operands of `==` operator are same values (`2 == 2` in (c) and `2.0 == 2.0` in (e)).

(d) evaluates to **True** as both the operands of `is` operator are same integer objects (2).

Since both operand expressions evaluate to same integer value 2 and 2 is a small integer value, both are bound to same memory address, hence `is` operator returns **True**.

In (f), even though both operands evaluate to same floating point value 2.0, the `is` operator returns **False** because Python assigns different memory address to floating point values even if their exists a same value in the memory.

15. What will be the output produced by the following code statements ? State reasons.

- (a) `bool(0)` (b) `bool(1)` (c) `bool('0')` (d) `bool('1')` (e) `bool()`
 (f) `bool(0.0)` (g) `bool('0.0')` (h) `bool(0j)` (i) `bool('0j')`

Solution.

- (a) **False**. Integer value 0 has false truth value hence `bool()` converts it to **False**.
 (b) **True**. Integer value 1 has true truth value hence `bool()` converts it to **True**.
 (c) **True**. '0' is string value, which is a non-empty string and has a true truth value, hence `bool()` converts it to *True*.
 (d) **True**. Same reason as above.
 (e) **False**. " is an empty string, thus has false truth value, hence `bool()` converts it to *False*.
 (f) **False**. 0.0 is zero floating point number and has false truth value, hence `bool()` converts it to *False*.
 (g) **True**. '0.0' is a non-empty string hence it has true truth value and thus `bool()` converted it to *True*.
 (h) **False**. 0j is zero complex number and has false truth value and thus `bool()` converted it to *False*.
 (i) **True**. '0j' is non-empty string, hence it has true truth value and thus `bool()` converted it to *True*.

16. What will be the output produced by these code statement ?

- (a) `bool(int('0'))` (b) `bool(str(0))` (c) `bool(float('0.0'))` (d) `bool(str(0.0))`

Solution. (a) False (b) True (c) False (d) True

17. What will be the output of following code ? Why ?

- (i) `13 or len(13)`
 (ii) `len(13) or 13`

Solution.

- (i) **13** because `or` evaluates first argument 13's truth value, which is *true* and hence returns the result as 13 without evaluating second argument.
 (ii) **error** because when `or` evaluates first argument **len(13)**, Python gives error as **len()** works on strings only.

18. Mr Sharma has invested Rs 27500 in 200 shares of XYZ co. Ltd and 52500 in 250 shares of ABC co. Ltd. Write a program to calculate average investment cost per share.

Solution.

```

amountXYZ = float(input("Amount invested in company XYZ : "))
sharesXYZ = int(input("Number of shares bought: "))
amountABC = float(input("Amount invested in company ABC : "))
sharesABC = int(input("Number of shares bought: "))
totalAmount = amountXYZ + amountABC
totalShares = sharesXYZ + sharesABC
avgCostperShare = totalAmount / totalShares
print("_____")
print("Total amount invested :", totalAmount)
print("Total shares bought :", totalShares)
print("Average investment per share :", avgCostperShare)

```

19. An individual would like to determine their ending balance (Future Value) after one year on an account that earns .5% per month and is compounded monthly. The original balance on the account is Rs.1,00,000 (present value). Write a program to calculate Future Value from this given information.

Hint : Formula for future value : $FV = PV \times (1+r)^n$ $PV = \text{Present Value}$; $r = \text{rate of return}$

Solution.

computing FV from given data

 $n = \text{number of periods}$

PV = 100000

r = 5/100 # 5 % means 5/100

n = 12

FV = PV * (1+r)**n

print("Present Value : ", PV)

print("Rate of Interest : ", (r *100), " % per month")

print("Number of periods : ", n)

print("Future Value : ", FV)

20. Given the following Python code, which is repeated four times. What could be the possible set of outputs given four sets (dddd represent any combination of digits) ?

import random

print(15 + random.random() * 5)

(i) 17.dddd, 19.dddd, 20.dddd, 15.dddd

(ii) 15.dddd, 17.dddd, 19.dddd, 18.dddd

(iii) 14.dddd, 16.dddd, 18.dddd, 20.dddd

(iv) 15.dddd, 15.dddd, 15.dddd, 15.dddd

Solution. Option (ii) and (iv) are the correct possible outputs because :

(a) `random()` generates number N between range $0.0 \leq N < 1.0$.(b) when it is multiplied with 5, the range becomes 0.0 to < 5 (c) when 15 is added to it, the range becomes 15 to < 20

Only option (ii) and (iv) fulfill the condition of range 15 to < 20 .

21. In a school fest, three randomly chosen students out of 100 students (having roll numbers 1-100) have to present bouquets to the guests. Help the school authorities choose three students randomly.

Solution. import random

student1 = random.randint(1, 100)

student2 = random.randint(1, 100)

student3 = random.randint(1, 100)

print("3 chosen students are",)

print(student1, student2, student3)

GLOSSARY

| | |
|---------------------------------|---|
| Atom | Something that has a value. |
| Coercion | Implicit Type Conversion |
| Expression | Valid combination of operators and atoms. |
| Explicit Type Conversion | Forced data type conversion by the user. |
| Immutable Type | A type whose value is not changeable in place. |
| Implicit Type Conversion | Automatic Internal Conversion of data type (lower to higher type) by Python |
| Mutable Type | A type whose value is changeable in place. |
| Operator | Symbol/word that triggers an action or operation. |
| Type Casting | Explicit Type Conversion |

Assignments

For
Solutions for
Selected Questions



Scan
QR Code

Type A : Short Answer Questions/Conceptual Questions

1. What are data types ? How are they important ?
2. How many integer types are supported by Python ? Name them.
3. How are these numbers different from one another ? 33, 33.0, 33j, 33 + j
4. The complex numbers have two parts : real and imaginary. In which data type are real and imaginary parts represented ?
5. How many string types does Python support ? How are they different from one another ?
6. What will following code print ?


```
str1 = '''Hell ↵
o'''
str2 = '''Hell\
o'''
print(len(str1) > len(str2))
```
7. What are immutable and mutable types ? List immutable and mutable types of Python.
8. What are three internal key-attributes of a value-variable in Python ? Explain with example.
9. Is it true that if two objects return True for **is** operator, they will also return True for **==** operator ?
10. Are these values equal ? Why/why not ?

(i) 20 and 20.0 (ii) 20 and int(20) (iii) str(20) and str(20.0) (iv) 'a' and "a"
11. What is an atom ? What is an expression ?
12. What is the difference between implicit type conversion and explicit type conversion ?
13. Two objects (say *a* and *b*) when compared using **==**, return True. But Python gives False when compared using **is** operator. Why ?

(i.e., $a == b$ is True but why is a is b False ?)
14. Given `str1 = "Hello"`, what will be the values of ?

(a) `str1 [0]` (b) `str1 [1]` (c) `str [-5]` (d) `str [-4]` (e) `str [5]`
15. If you give the following for `str1 = "Hello"`, why does Python report error ?


```
str1[2] = 'p'
```

16. What will the result given by the following ?
 (a) type (6 + 3) (b) type (6 - 3) (c) type (6 * 3) (d) type (6/3) (e) type (6//3) (f) type (6 % 3)
17. What are augmented assignment operators ? How are they useful ?
18. Differentiate between (555/222)**2 and (555.0/222)**2.
19. Given three Boolean variables a, b, c as : $a = \text{False}, b = \text{True}, c = \text{False}$.
 Evaluate the following Boolean expressions :
- (a) b and c (b) b or c (c) not a and b (d) (a and b) or not c
 (e) not b and not (a or c) (f) not ((not b or not a) and c) or a
20. What would following code fragments result in ? Given $x = 3$.
 (a) $1 < x$ (b) $x >= 4$ (c) $x == 3$ (d) $x == 3.0$ (e) "Hello" == "Hello"
 (f) "Hello" > "hello" (g) $4/2 == 2.0$ (h) $4/2 == 2$ (i) $x < 7$ and $4 > 5$.
21. Write following expressions in Python
 (a) $\frac{1}{3}b^2h$ (b) πr^2h (c) $\frac{1}{3}\pi r^2h$ (d) $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$ [Hint. for sqrt() use math.sqrt()
 (e) $(x-h)^2 + (y-k)^2 = r^2$ (f) $x = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$ (g) $a^n \times a^m = a^{n+m}$
 (h) $(a^n)^m = a^{nm}$ (i) $\frac{a^n}{a^m} = a^{n-m}$ (j) $a^{-n} = \frac{1}{a^n}$
22. `int('a')` produces error. Why ?
23. `int('a')` produces error but following expression having `int('a')` in it, does not return error. Why ?
`len('a') + 2` or `int('a')`
24. Write expression to convert the values 17, `len('ab')` to (i) integer (ii) str (iii) Boolean values
25. Evaluate and Justify : (i) $22 / 17 = 37 / 47 + 88 / 83$ (ii) `len('375')**2`.
26. Evaluate : (i) $22.0/7.0 - 22/7$ (ii) $22.0/7.0 - \text{int}(22.0/7.0)$ (iii) $22/7 - \text{int}(22.0)/7$ and justify.
27. Evaluate and justify : (i) false and None (ii) 0 and None (iii) True and None (iv) None and None.
28. Evaluate and justify :
 (a) \emptyset or None and "or" (b) 1 or None and 'a' or 'b'
 (c) False and 23 (d) 23 and False
 (e) not (1 == 1 and $\emptyset != 1$) (f) "abc" == "Abc" and not (2 == 3 or 3 == 4)
 (g) False and 1 == 1 or not True or 1 == 1 and False or $\emptyset == \emptyset$
29. Evaluate the following for each expression that is successfully evaluated, determine its value and type for unsuccessful expression, state the reason.
 (a) `len("hello") == 25/5` or `20/10` (b) $3 < 5$ or $50/(5 - (3 + 2))$
 (c) $50/(5 - (3 + 2))$ or $3 < 5$ (d) $2 * (2 * (\text{len}("01")))$
30. Write an expression that uses exactly 3 arithmetic operators with integer literals and produces result as 99.
31. Add parentheses to the following expression to make the order of evaluation more clear.
 $y \% 4 == \emptyset$ and $y \% 100 != \emptyset$ or $y \% 400 == \emptyset$

Type B : Application Based Questions

- What is the result produced by (i) `bool(0)` (ii) `bool(str(0))` ? Justify the outcome.
- What will be the output, if input for both the statements is $5 + 4/2$.
 $6 == \text{input}(\text{"Value 1:"})$
 $6 == \text{int}(\text{input}(\text{"value 2:"}))$

3. Following code has an expression with all integer values. Why is the result in floating point form ?

```
a, b, c = 2, 3, 6
d = a + b * c/b
print(d)
```

4. What will following code print ?

```
(a) a = va = 3          (b) a = 3
    b = va = 3          b = 3.0
    print (a, b)        print (a == b)
                        print (a is b)
```

5. What will be output produced by following code ? State reason for this output.

```
(a) a, b, c = 1, 1, 2          (b) a = 5 - 4 - 3          (c) a, b, c = 1, 1, 1
    d = a + b                  b = 3**2**3              d = 0.3
    e = 1.0                    print(a)                  e = a + b + c - d
    f = 1.0                    print(b)                  f = a + b + c == d
    g = 2.0                    print(f)                  print(e)
    h = e + f                  print(f)                  print(f)
    print(c == d)
    print(c is d)
    print(g == h)
    print(g is h)
```

6. What will be the output of following Python code ?

```
(a) a = 12                    (b) x, y = 4, 8
    b = 7.4                    z = x/y*y
    c = 1                      print(z)
    a - = b
    print(a, b)
    a *= 2 + c
    print(a)
    b += a * c
    print(b)
```

7. Make change in the expression for z of previous question so that the output produced is zero. You cannot change the operators and order of variables.

[Hint. Use a function around a sub-expression]

8. Following expression does not report an error even if it has a sub-expression with 'divide by zero' problem :

3 or 10/0

What changes can you make to above expression so that Python reports this error ?

9. What is the output produced by following code ?

```
a, b = bool (0), bool (0.0)
c, d = str (0), str (0.0)
print (len(a), len(b))
print (len(c), len(d))
```

10. Given a string s = "12345". Can you write an expression that gives sum of all the digits shown inside the string s i.e., the program should be able to produce the result as 15 (1 + 2 + 3 + 4 + 5).

[Hint. Uses indexes and convert to integer]

11. Predict the output if e is given input as 'True'

```
a = True
b = 0 < 5
print (a == b)
print (a is b)
c = str (a)
d = str (b)
print (c == d)
print (c is d)
e = input ("Enter :")
print (c == e)
print (c is e)
```

12. Find the errors(s).

(a) name = "Harit"
print (name)
name[2] = 'R'
print (name)

(b) a = bool (0)
b = bool (1)
print (a == false)
print (b == true)

(c) print (type (int("123")))
print (type (int("Hello")))
print (type (str("123.0")))

(d) pi = 3.14
print (type (pi))
print (type ("3.14"))
print (type (float ("3.14")))
print (type (float ("three point fourteen")))

(e) print ("Hello" + 2)
print ("Hello" + "2")
print ("Hello" * 2)

(f) print ("Hello"/2)
print ("Hello" / 2)

Type C : Programming Practice/Knowledge based Questions

- Write a program to obtain principal amount, rate of interest and time from user and compute simple interest.
- Write a program to obtain temperatures of 7 days (Monday, Tuesday ... Sunday) and then display average temperature of the week.
- Write a program that reads a number of seconds and prints it in form : mins and seconds, e.g., 200 seconds are printed as 3 mins and 20 seconds.
[Hint. use // and % to get minutes and seconds]
- Write a program that takes sales for four quarters and then displays total sales and average sales made.
- Write a program to calculate working capital for company A that has cash of about ₹ 2,50,000, Accounts received of about ₹ 1,50,000 and inventories of about ₹ 5,00,000. Company A also has accounts payable of ₹ 3,00,000, short term borrowings of ₹ 75,000 and accrued liabilities of about ₹ 1,25,000.
[Hint. Working Capital = Current assets - Current liabilities]
- Write a program to calculate EMI as per formula : $E = PR(1+R)^n / ((1+R)^n - 1)$
where E = EMI ; P = Principal Loan Amount ; R = Rate of interest per month i.e., (Annual rate of interest/100/12) ; n = tenure of loan repayment in months.
Get the input from user (e.g., calculate EMI for loan amount of 2,00,000 at 10% p.a. rate of interest (10/100/12) for a period of 2 years i.e., 24 months.