

Introduction to Python

1) Which of the following is an assignment operator?

- a) =
- b) ==
- c) !=
- d) None of the above

EXPLANATION: Assignment operators are used in Python to assign values to variables. `a = 5` is a simple assignment operator that assigns the value 5 on the right to the variable a on the left i.e. `a=5`

2) Arrays contains

- a) Homogeneous elements
- b) Heterogeneous elements
- c) Both of them
- d) None of the above

EXPLANATION: Homogeneous data structures are those data structures that contain only similar type of data e.g. like a data structure containing only integer or float values. `["I", [3,6,9]]`

3) `List=['sam','ram','cam']`

`print (list [2])`

The output for above code is

- a) Cam
- b) Ram
- c) Sam
- d) **cam**

EXPLANATION: The indices of a list starts at 0, in our case `Print (list [0])` gives sam, `print (list [1])` gives ram and `print (list [2])` gives cam.

4) `a = ('12','14','15')` is an

Introduction to Python

- a) **Tuple**
- b) Dictionary
- c) List
- d) None of the above

4) what is the output of the following?

```
print(abc.islower( ))
```

- a. True
- b. False
- c. None
- d. **Error**

EXPLANATION: since 'abc' is not given in quotes it will throw error

5) Tuple can be

- a) Mutable
- b) **Immutable**

EXPLANATION: A tuple is a sequence of immutable Python objects.

6) A=['veer','raj','sud']

```
del(A[0])  
print(A[1])
```

- a) Veer
- b) Raj

Introduction to Python

c) Sud

d) None of the above

EXPLANATION: del operator stands for “delete”, and if we write del(A[0]) it will delete 0 index i.e. veer and now our list looks like A=['raj','sud'],and now if we print, print(A[1]) for the new list it will return 'sud'

7) Which will convert string into list

a) tuple(s)

b) list(s)

c) eval(str)

d) None of the above

7) what data type is the object below?

Y=[1, 23, 'hello', 1]

a) list

b) Dictionary

c) Tuple

d) Array

EXPLANATION: [] defines a list

8) Nameage= {"veer":20," neer":21," zeer":22} is an

a) Dictionary

b) Tuple

c) List

d) Set

EXPLANATION: Dictionaries are indexed by keys, which can be any immutable type; strings and numbers can always be keys.

9) Tuples are

a) Faster than list

b) Slower than list

Introduction to Python

- c) Equal to list
- d) None of the above

EXPLANATION: Lists are allocated in two blocks: the fixed one with all the Python object information and a variable sized block for the data. It is the reason creating a tuple is faster than List. It also explains the slight difference in indexing speed is faster than lists, because in tuples for indexing it follows fewer pointers.

10) What is the output of this $3*4**2$

- a) 48
- b) 14
- c) 10
- d) None of the above

10) Which is the correct operator for (X to the power Y)

- a) X^Y
- b) $X**Y$
- c) $X^^Y$
- d) X^*Y

EXPLANATION: In python power operator is $X**Y$ i.e. $2**3=8$ for better context run the code in python shell

11) Which of this won't get error

- a) `round(63.02)`
- b) `round(63222,52)`

Introduction to Python

- c) round(7856,56)
- d) round (965,23)

11) what does ~5 evaluate to?

- a) -3
- b) -4
- c) -6
- d) -5

EXPLANATION: ~x is equivalent to -(x+1), for better context run in python shell

12) name= "raju"

name [1] =" l"

print(name)

- a) Raju
- b) Riju
- c) Raiu
- d) None of the above

12) which of the following statements create a dictionary?

- a) d={}
- b) d={"john":40, "peter":45}
- c) d={40:"john",45:"peter"}
- d) All of the mentioned

EXPLANATION: Dictionaries are created by specifying keys and values

Introduction to Python

13) `==` is an

- a) **Comparison operator**
- b) Assignment operator
- c) Logical operator
- d) None of the above

EXPLANATION: You use `==` to compare two symbols or values and see if they are equal. Example: `b = 5`, `c = 10`, `d = 10`

`b == c` #false because 5 is not equal to 10

`c == d` #true because 10 is equal to 10

14) `print("hi"+1+2+3)`

- a) hi123
- b) hi 123
- c) hi12
- d) **None of the above**

14) what is the type of `inf`?

- a) Boolean
- b) Integer
- c) **Float**
- d) complex

EXPLANATION: Infinity is a special case of floating point numbers. It can be obtained by `float('inf')`

15) `th1.isalpha()` output for this is

- a) True
- b) **False**
- c) Error
- d) None

Introduction to Python

15) what is the output of the code

```
print(not(4>3) )
```

```
print(not(5&5) )
```

- a) True, False
- b) True, True
- c) False, True
- d) **False, False**

EXPLANATION: The not function returns true if the argument is false, and false if the argument is true. Hence the first line of above code returns false, and the second line will also return false

16) 7.8.isnumeric()

- a) True
- b) **False**
- c) Error
- d) None

16) what is the result of the snippet of the code shown below if X=1?

```
X<<2
```

- a) 8
- b) 1
- c) 2
- d) **4**

EXPLANATION: The binary form of 1 is 0001. The expression `x<<2` implies we are performing bitwise left shift on x. This shift yields the value: 0100, which is the binary form of the number 4. (bitwise operator)

Introduction to Python

17) Output for the code is int (88.8)

- a) Name error
- b) Value error**
- c) Type error
- d) Import error

17) suppose there is a list such that: l=[2,3,4].

If we want to print this list in reverse order, which of the following methods should be used?

- a) reverse(l)
- b) list(reverse(l))
- c) reversed(l)
- d) list(reversed(l))**

EXPLANATION: The built-in function reversed () can be used to reverse the elements of a list. This function accepts only an iterables as an argument. To print the output in the form of a list, we use: list(reversed(l)). The output will be: [4,3,2] for **better context run the code in python shell**

18) Output for this code is 18%3

- a) 0**
- b) 6
- c) 5
- d) 7

18) Which of the following statements is used to create an empty set?

- a. { }
- b. set()**
- c. []
- d. ()

EXPLANATION: { } creates a dictionary not a set. Only set() creates an empty set

Introduction to Python

19) & is an

- a) **Bitwise AND**
- b) Bitwise OR
- c) Bitwise NOT
- d) None

19) which of these about a “set” is not true?

- a) mutable data points
- b) Allows duplicate values
- c) Data type with unordered values

d) immutable data types

EXPLANATION: A set is a mutable data type with non-duplicate, unordered values, providing the usual mathematical set operations

20) x=4

Y=5

x==y

output for the above code is

- a) **True**
- b) False
- c) Error
- d) None of the above

20) What is the output of the snippet of code shown below?

```
Z=set('abc$de')
```

```
'a' in Z
```

- a) True
- b) false
- c) no output
- d) Error

Introduction to Python

EXPLANATION: The code shown above is used to check whether a particular item is a part of a given set or not. Since 'a' is a part of the set z, the output is true. Note that this code would result in an error in the absence of the quotes. for better context run the code in python shell