PYTHON WORKSHEET 2

1. Which of the following is not a core datatype in python?

Ans-B) struct

2. Which of the following is an invalid variable name in python?

Ans-C) 1_no

3. Which one of the following is a keyword in python?

Ans-A) in

4. In which of the following manner are the operators of the same precedence executed in python?

Ans- A) Left to Right

5. Arrange the following in decreasing order of the precedence when they appear in an expression in python?

i) Multiplication ii) Division iii) Exponential iv) Parentheses

Ans- C) iv – iii – ii – i

6. (28//6)**3/3%3 = ?

Ans- C) 0.3333…

7. a = input("Enter an integer"). What will be the data type of a?

Ans- B) str

8. Which of the following statements are correct?

Ans- A) Division and multiplication have same precedence in python

B) Python's operators' precedence is based on PEDMAS

D) In case of operators' having the same precedence, the one on the left side is executed First

9. Which of the following is(are) valid statement(s) in python?

Ans- A) abc = 1,000,000

C) a,b,c = 1000, 2000, 3000

D) a_b_c = 1,000,000

10. Which of the following is not equal to x16 in python?

Ans- C) x^16

11. Differentiate between a list, tuple, set and dictionary

Ans-
Lists:

- Lists are just like dynamic sized arrays, declared in other languages (vector in C++ and ArrayList in Java). Lists need not be homogeneous always which makes it the most powerful tool in Python.
- List can be represented by [ ], Example: [1, 2, 3, 4, 5]
- List allows duplicate elements
- List is mutable i.e we can make any changes in list.
- List is ordered

Tuple:

- A Tuple is a collection of Python objects separated by commas. In some ways, a tuple is similar to a list in terms of indexing, nested objects, and repetition but a tuple is immutable, unlike lists that are mutable.
- Tuple can be represented by ( ), Example: (1, 2, 3, 4, 5)
- Tuple allows duplicate elements
- Tuple is immutable i.e we can not make any changes in tuple
- Tuple is ordered

Set:

- A Set is an unordered collection data type that is iterable, mutable, and has no duplicate elements. Python's set class represents the mathematical notion of a set.
- Set can be represented by { }, Example: {1, 2, 3, 4, 5}
- Set will not allow duplicate elements
- Set is mutable i.e we can make any changes in set. But elements are not duplicated.
- Set is unordered

12. Are strings mutable in python? Suppose you have a string “I+Love+Python”, write a small code to replace ‘+’ with space in python.

Ans- No Strings are not mutable in python. One potential arrangement to make changes in string is to make another string object with vital alterations:

Prog-

str= "I+Love+Python"

print(str)

O/p- I+Love+Python

str.replace('+',' ') #This will only store the results in dummy variable, original string is unaffected

O/p- 'I Love Python'

print(str)

O/p- I+Love+Python

To make permanent changes , storing result in a new string object

str2=str.replace(‘+’,’ ‘)

print(str2)

O/p- 'I Love Python'

13. What does the function ord() do in python? Explain with an example. Also, write down the function for getting the data type of a variable in python.

Ans- Python ord() function returns the Unicode code from a given character. This function accepts a string of <u>unit length</u> as an argument and returns the Unicode equivalence of the passed argument. In other words, given a string of length 1, the ord() function returns an integer representing the Unicode code point of the character when an argument is a Unicode object, or the value of the byte when the argument is an 8-bit string. If the string length is more than one, a TypeError will be raised.

Syntax: ord(ch)

For example, ord(‘a’) returns the integer 97, ord(‘€’) (Euro sign) returns 8364.

<u>Function for getting the data type of a variable in python:</u>

Type() is a python built function that returns the type of objects

Syntax- type(object)