**Worksheet No.2[Python]**

**Q1 to Q7 have only one correct answer. Choose the correct option to answer your question.**

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

**ANS. B) struct**

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

**ANS. C) 1\_no**

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

**ANS. A) in**

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

**ANS. A) Left to right**

Q.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**

Q.6:- (28//6)\*\*3/3%3 = ?

**ANS. C) 0.3333....**

Q.7:- a = input(“Enter an integer”). What will be the data type of a?

**ANS. B) str**

Q8 and Q10 have multiple correct answers. Choose all the correct options to answer your question.

Q.8:-Which of the following statements are correct?

A) Division and multiplication have same precedence in python

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

C) Python’s operators’ precedence is based on VBODMAS

D) In case of operators’ having the same precedence, the one on the left side is executed first.

**ANS. A) Division and multiplication have same precedence in python**

**B) Python’s operators’ precedence is based on PEDMAS**

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

A) abc = 1,000,000 B) a b c = 1000 2000 3000

C) a,b,c = 1000, 2000, 3000 D) a\_b\_c = 1,000,000

**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?

A) x\*\*4\*\*4 B) x\*\*16 C) x^16 D) (x\*\*4)\*\*4

**ANS. A) x\*\*4\*\*4**

**Q11 to Q13 are subjective questions, answer them briefly**

**Q.11**:- Differentiate between a list, tuple, set and dictionary.

**ANS. List:** Lists are just like [dynamic sized arrays](https://www.geeksforgeeks.org/how-do-dynamic-arrays-work/), declared in other languages ([vector in C++](https://www.geeksforgeeks.org/vector-in-cpp-stl/) and [Array List in Java](https://www.geeksforgeeks.org/arraylist-in-java/)). Lists need not be homogeneous always which makes it the most powerful tool in [Python](https://www.geeksforgeeks.org/python-programming-language/).

[**Tuple:**](https://www.geeksforgeeks.org/tuples-in-python/) 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.

[**Set:**](https://www.geeksforgeeks.org/sets-in-python/) 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.

[**Dictionary:**](https://www.geeksforgeeks.org/python-dictionary/)in Python is an ordered collection of data values, used to store data values like a map, which, unlike other Data Types that hold only a single value as an element, Dictionary holds **key:value** pair. Key-value is provided in the dictionary to make it more optimized.

List, Tuple, Set, and Dictionary are the data structures in python that are used to store and organize the data in an efficient manner.

|  |  |  |  |
| --- | --- | --- | --- |
| List | Tuple | Set | Dictionary |
| List is a non-homogeneous data structure that stores the elements in single row and multiple rows and columns | Tuple is also a non-homogeneous data structure that stores single row and multiple rows and columns | Set data structure is also non-homogeneous data structure but stores in single row | Dictionary is also a non-homogeneous data structure which stores key value pairs |
| List can be represented by [ ] | Tuple can be represented by ( ) | Set can be represented by set( ) | Dictionary can be represented by { } |
| List allows duplicate elements | Tuple allows duplicate elements | Set will not allow duplicate elements | Dictionary doesn’t allow duplicate keys. |
| List can use nested among all | Tuple can use nested among all | Set can use nested among all | Dictionary can use nested among all |
| List is mutable i.e we can make any changes in list. | Tuple  is immutable i.e we can not make any changes in tuple | Set is mutable i.e we can make any changes in set. But elements are not duplicated. | Dictionary is mutable. But Keys are not duplicated. |
| List is ordered | Tuple is ordered | Set is unordered | Dictionary is ordered |
| Example:[10,20, 30] | Tuple = (10, 20, 30 ) | set([“Hello”,”Python”]) | car={1:"Hello". 2:"Python"} |

**Q.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. Strings in Python are immutable which means that once a string variable is assigned to a string (For eg a ='Hello' ) the contents of the string cannot be changed unlike the list object. In the code above you are in a way transforming your string and not changing the contents of your string variable.

Strings are array in python, any character in python is string. Strings are immutable just like an array & tuples. Immutability is a clean and efficient solution to concurrent access. Having immutable variables means that no matter how many times the method is called with the same variable / value, the output will always be the same.

The elements of the strings can be accessed by slicing and indexing, elements can be replaced using replace ( ) method, it generates a copy of the string with new replaced items.

We also knowing that a string is immutable mean we can allocate space for it at creation time, and the storage requirements are fixed and unchanging.

Example:- ln [1]: a = "I+Love+Python" # define the a

new\_a = a.replace("+"," ") # replace + with space(" ")

new\_a # print the final output

Q.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. The ord() function returns the number representing the unicode code of a specified character. **Python ord() function**returns the Unicode code from a given character. This function accepts a string of unit length 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.

For example, ord(‘a’) returns the integer 97, ord(‘€’) (Euro sign) returns 8364. This is the inverse of [chr()](https://www.geeksforgeeks.org/chr-in-python/) for 8-bit strings and of unichr() for Unicode objects. If a Unicode argument is given and Python is built with UCS2 Unicode, then the character’s code point must be in the range [0..65535] inclusive.

Type( ) Function :- It is very simple syntax and can be used to find the type of any variable in python be it a collection type variable, a class object variable or a simple string or integer.

syntax:- type(variable name) Example : - a=20 # define variable a

type(a) # print the type variable

Q14 and Q15 are programming questions. Answer them in Jupyter Notebook.

Q.14:-Write a python program to solve a quadratic equation of the form ax^2+bx+c=0. Where a, b and c are to be taken by user input. Handle the erroneous input, such as ‘a’ should not be equal to 0.

ANS.SOLVED IN Jupyter Notebook