Chapter 2 – Variables and Data Types

A variable is a name given to a memory location in a program. For example

a=30

b=”Harry”

c=71.22

Copy

**Variable –**Container to store a value

**Keywords –**Reserved words in Python

**Identifiers –**class/function/variable name

**Data Types:**

Primarily there are the following data types in Python:

1. Integers
2. Floating point numbers
3. Strings
4. Booleans
5. None

Python is a fantastic language that automatically identifies the type of data for us.

a = 71 #Identifies a as class<int>

b = 88.44 #Identifies b as class<float>

name = “Harry” #Identifies name as class<Str>

Copy

**Rules for defining a variable name:**(Also applicable to other identifiers)

* A variable name can contain alphabets, digits, and underscore.
* A variable name can only start with an alphabet and underscore.
* A variable can’t start with a digit.
* No white space is allowed to be used inside a variable name.

Examples of few valid variable names,

Harry, harry, one8, \_akki, aakash, harry\_bro, etc.

**Operators in Python**

The following are some common operators in Python:

1. Arithmetic Operators (+, -, \*, /, etc.)
2. Assignment Operators (=, +=, -=, etc.)
3. Comparison Operators (==, >=, <=, >, <, !=, etc.)
4. Logical Operators (and, or, not)

**type() function and Typecasting**

type function is used to find the data type of a given variable in Python.

a = 31

type(a) #class<int>

b = “31”

type(b) #class<str>

Copy

A number can be converted into a string and vice versa (if possible)

There are many functions to convert one data type into another.

Str(31) # ”31” Integer to string conversion

int(“32”) # 32 String to int conversion

float(32) #32.0 Integer to float conversion

Copy

… and so on

Here “31” is a string literal, and 31 is a numeric literal.

**input() function**

This function allows the user to take input from the keyboard as a string.

a = input(“Enter name”) #if a is “harry”, the user entered harry

Copy

**Note:**The output of the input function is always a string even if the number is entered by the user.

Suppose if a user enters 34, then this 34 will automatically convert to “34” string literal.

Chapter 2 – Practice Set

1. Write a Python program to add two numbers.
2. Write a Python program to find the remainder when a number is divided by Z(Integer).
3. Check the type of the variable assigned using the input() function.
4. Use a comparison operator to find out whether a given variable a is greater than b or not. (Take a=34 and b=80)
5. Write a Python program to find the average of two numbers entered by the user.
6. Write a Python program to calculate the square of a number entered by the user.