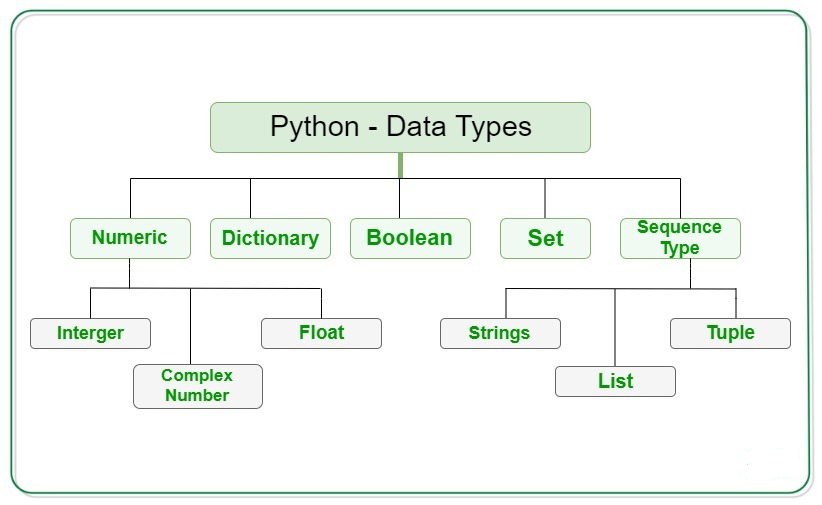
# Experiment Number – 1

**Title - Program to find largest and smallest of three integer numbers without using decision making statements**

# Python Data Types

Data types are the classification or categorization of data items. It represents the kind of value that tells what operations can be performed on a particular data. Since everything is an object in [Python programming](https://www.geeksforgeeks.org/python-programming-language/), data types are actually classes and variables are instances (object) of these classes. The following are the standard or built-in data types in Python:

* **Numeric**
* **Sequence Type**
* **Boolean**
* **Set**
* **Dictionary**
* **Binary Types([memoryview](https://www.geeksforgeeks.org/memoryview-in-python/),** [**bytearray**](https://www.geeksforgeeks.org/python-bytearray-function/)**,** [**bytes**](https://www.geeksforgeeks.org/python-bytes-method/)**)**



## Numeric Data Type in Python

The numeric data type in Python represents the data that has a numeric value. A numeric value can be an integer, a floating number, or even a complex number. These values are defined as [Python int](https://www.geeksforgeeks.org/python-int-function/), [Python float](https://www.geeksforgeeks.org/float-in-python/), and [Python complex](https://www.geeksforgeeks.org/python-complex-function/) classes in [Python](https://www.geeksforgeeks.org/python-programming-language/).

* **Integers** – This value is represented by int class. It contains positive or negative whole numbers (without fractions or decimals). In Python, there is no limit to how long an integer value can be.
* **Float** – This value is represented by the float class. It is a real number with a floating-point representation. It is specified by a decimal point. Optionally, the character e or E followed by a positive or negative integer may be appended to specify scientific notation.
* **Complex Numbers** – Complex number is represented by a complex class. It is specified as *(real part) + (imaginary part)j*. For example – 2+3j

## Sequence Data Type in Python

The sequence Data Type in Python is the ordered collection of similar or different data types. Sequences allow storing of multiple values in an organized and efficient fashion. There are several sequence types in Python –

* [Python String](https://www.geeksforgeeks.org/python-data-types/#string)
* [Python List](https://www.geeksforgeeks.org/python-data-types/#list)
* [Python Tuple](https://www.geeksforgeeks.org/python-data-types/#tuple)

### String Data Type

[Strings](https://www.geeksforgeeks.org/python-strings/) in Python are arrays of bytes representing Unicode characters. A string is a collection of one or more characters put in a single quote, double-quote, or triple-quote. In python there is no character data type, a character is a string of length one. It is represented by str class.

### List Data Type

[Lists](https://www.geeksforgeeks.org/python-list/) are just like arrays, declared in other languages which is an ordered collection of data. It is very flexible as the items in a list do not need to be of the same type.  Lists in Python can be created by just placing the sequence inside the square brackets[].

### Tuple Data Type

Just like a list, a [tuple](https://www.geeksforgeeks.org/python-tuples/) is also an ordered collection of Python objects. The only difference between a tuple and a list is that tuples are immutable i.e. tuples cannot be modified after it is created. It is represented by a tuple class. In Python, [tuples](https://www.geeksforgeeks.org/python-tuples/) are created by placing a sequence of values separated by a ‘comma’ with or without the use of parentheses for grouping the data sequence. Tuples can contain any number of elements and of any datatype (like strings, integers, lists, etc.). **Note:** Tuples can also be created with a single element, but it is a bit tricky. Having one element in the parentheses is not sufficient, there must be a trailing ‘comma’ to make it a tuple.

## Boolean Data Type in Python

Data type with one of the two built-in values, True or False. Boolean objects that are equal to True are truthy (true), and those equal to False are falsy (false). But non-Boolean objects can be evaluated in a Boolean context as well and determined to be true or false. It is denoted by the class bool.

**Note** – True and False with capital ‘T’ and ‘F’ are valid booleans otherwise python will throw an error.

## Set Data Type in Python

In Python, a [Set](https://www.geeksforgeeks.org/python-sets/) is an unordered collection of data types that is iterable, mutable and has no duplicate elements. The order of elements in a set is undefined though it may consist of various elements. Sets can be created by using the built-in set() function with an iterable object or a sequence by placing the sequence inside curly braces, separated by a ‘comma’. The type of elements in a set need not be the same, various mixed-up data type values can also be passed to the set.

## Dictionary Data Type in Python

A dictionary in Python is an unordered collection of data values, used to store data values like a map, unlike other Data Types that hold only a single value as an element, a Dictionary holds a key: value pair. Key-value is provided in the dictionary to make it more optimized. Each key-value pair in a Dictionary is separated by a colon : , whereas each key is separated by a ‘comma’. In Python, a Dictionary can be created by placing a sequence of elements within curly {} braces, separated by ‘comma’. Values in a dictionary can be of any datatype and can be duplicated, whereas keys can’t be repeated and must be immutable. The dictionary can also be created by the built-in function dict(). An empty dictionary can be created by just placing it in curly braces{}.

# Python - Variables

Python variables are the reserved memory locations used to store values with in a Python Program. This means that when you create a variable you reserve some space in the memory. Based on the data type of a variable, Python interpreter allocates memory and decides what can be stored in the reserved memory. Therefore, by assigning different data types to Python variables, you can store integers, decimals or characters in these variables.

## Creating Python Variables

Python variables do not need explicit declaration to reserve memory space or you can say to create a variable. A Python variable is created automatically when you assign a value to it. The equal sign (=) is used to assign values to variables.

The operand to the left of the = operator is the name of the variable and the operand to the right of the = operator is the value stored in the variable. For example −

counter = 100 # Creates an integer variable

miles = 1000.0 # Creates a floating point variable

name = "Zara Ali" # Creates a string variable

## Identifier Naming

Variables are the example of identifiers. An Identifier is used to identify the literals used in the program. The rules to name an identifier are given below.

* The first character of the variable must be an alphabet or underscore ( \_ ).
* All the characters except the first character may be an alphabet of lower-case(a-z), upper-case (A-Z), underscore, or digit (0-9).
* Identifier name must not contain any white-space, or special character (!, @, #, %, ^, &, \*).
* Identifier name must not be similar to any keyword defined in the language.
* Identifier names are case sensitive; for example, my name, and MyName is not the same.
* Examples of valid identifiers: a123, \_n, n\_9, etc.
* Examples of invalid identifiers: 1a, n%4, n 9, etc.

## Python Constants

A constant is a special type of variable whose value cannot be changed.

## Rules to be followed while declaring a Constant

1. Python Constants and variable names should contain a combination of lowercase (a-z) or capital (A-Z) characters, numbers (0-9), or an underscore ( ).
2. When using a Constant name, always use UPPERCASE, For example, CONSTANT = 50.
3. The Constant names should not begin with digits.
4. Except for underscore(\_), no additional special character (!, #, ^, @, $) is utilized when declaring a constant.
5. We should come up with a catchy name for the python constants. VALUE, for example, makes more sense than V. It simplifies the coding process.

## Python Literals

* The data which is being assigned to the variables are called as **Literal**.
* In Python, **Literals** are defined as raw data which is being assigned to the variables or constants.

For e.g. strng = ‘How are you?’

Here, we have declared a variable ‘strng’, and the value assigned to it ‘How are you?’ is a literal of type string.

## Numeric Literals

Numeric Literals are values assigned to the Variables or Constants which cannot be changed i.e., they are immutable. There are a total of 3 categories in Numeric Literals. They are –  Integer, Float, and Complex.

Example

# Int Numeric Literal

a = 30

# Float Numeric Literal

b = 40.67

# Complex Numeric Literal

c = 10+4j

## String Literals

A string literal is a series of characters surrounded by quotation marks. For a string, we can use single, double, or triple quotations. We can write multi-line strings or display them in the desired format by using triple quotes. A single character surrounded by single or double quotations is also known as a character literal.

Example

string = 'Hello Guys'

multi\_line = '''Hey

There!!'''

char = 'Z'

## Boolean Literals

A Boolean Literal has either of the 2 values – True or False. Where True is considered as 1 and False is considered as 0.

a= True