----------X----------X----------X
There exist a lot of data types in python. However, teaching all of them in the beginning of the course may only confuse the student. Hence, in the first lecture only the basic data types like the numeric data types, the basic sequence data types and the boolean data types should be explained.
*Complex data types like set, tuple and dictionary should be explained as the course progresses and the student gets a hold of what the programming language is actually about.
----------X----------X
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, data types are actually classes and variables are instance (object) of these classes.

Following are the standard or built-in data type of Python:

- Numeric
- Sequence Type
- Boolean
- Set
- Dictionary

*It must be mentioned that while this is the entire list of the data types, not all of them will be covered right now but only a handful.

Numeric

In Python, numeric data type represent the data which has numeric value. Numeric value can be integer, floating number or even complex numbers. These values are defined as int, float and complex class in Python.

- **Integers** – This value is represented by int class. It contains positive or negative whole numbers (without fraction or decimal). In Python there is no limit to how long an integer value can be.
- **Float** – This value is represented by float class. It is a real number with 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 complex class. It is specified as *(real part) + (imaginary part)j*. For example – 2+3j

Note – type() function is used to determine the type of data type.

- Python3

```
# Python program to

# demonstrate numeric value



a = 5
```

```
print("Type of a: ",
type(a))
```

```
b = 5.0
```

```
print("\nType of b: ",
type(b))
```

```
c = 2 + 4j
```

```
print("\nType of c: ",
type(c))
```

Output:

```
Type of a:  <class 'int'>
```

```
Type of b:  <class 'float'>
```

```
Type of c:  <class 'complex'>
```

Sequence Type

In Python, sequence is the ordered collection of similar or different data types. Sequences allows to store multiple values in an organized and efficient fashion. There are several sequence types in Python –

- String
- List
- Tuple

1) String

In Python, Strings 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.

Creating String

Strings in Python can be created using single quotes or double quotes or even triple quotes.

- Python3

```
# Python Program for

# Creation of String



# Creating a String

# with single Quotes

String1 = 'Welcome to the Geeks World'

print("String with the use of Single Quotes: ")

print(String1)
```

```
# Creating a String

# with double Quotes

String1 = "I'm a Geek"

print("\nString with the use of Double Quotes: ")

print(String1)

print(type(String1))



# Creating a String

# with triple Quotes
```

```
String1 = '''I'm a Geek and I live in a world of
"Geeks"'''

print("\nString with the use of Triple Quotes: ")

print(String1)

print(type(String1))



# Creating String with triple

# Quotes allows multiple lines

String1 = '''Geeks

            For

            Life'''
```

```
print("\nCreating a multiline String: ")

print(String1)
```

Output:

```
String with the use of Single Quotes:

Welcome to the Geeks World




String with the use of Double Quotes:

I'm a Geek

<class 'str'>




String with the use of Triple Quotes:

I'm a Geek and I live in a world of "Geeks"
```

```
<class 'str'>
```

```
Creating a multiline String:

Geeks

            For

            Life
```

Boolean

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 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.

- Python3

```
# Python program to

# demonstrate boolean
type


print(type(True))

print(type(False))


print(type(true))
```

Output:

```
<class 'bool'>

<class 'bool'>


Traceback (most recent call last):

  File "/home/7e8862763fb66153d70824099d4f5fb7.py", line 8, in

    print(type(true))

NameError: name 'true' is not defined
```

----------X----------X----------x

The following data types must be explained further in the course:

1. Dictionary
2. Tuples

3. List
4. Set

----------X----------X