# **Python - Data Types**

Python Data Types are used to define the type of a variable. It defines what type of data we are going to store in a variable. The data stored in memory can be of many types. For example, a person's age is stored as a numeric value and his or her address is stored as alphanumeric characters.

Python has various built-in data types which we will discuss with in this tutorial:

* **Numeric - int, float, complex**
* **String - str**
* **Sequence - list, tuple, range**
* **Mapping - dict**
* **Boolean - bool**

Python Numeric Data Type

Python numeric data types store numeric values. Number objects are created when you assign a value to them. For example −

var1 = 1

var2 = 10

var3 = 10.023

Python supports four different numerical types −

* int (signed integers)
* long (long integers, they can also be represented in octal and hexadecimal)
* float (floating point real values)
* complex (complex numbers)

Example

Following is an example to show the usage of Integer, Float and Complex numbers:

# integer variable.

a=100

print("The type of variable having value", a, " is ", type(a))

# float variable.

b=20.345

print("The type of variable having value", b, " is ", type(b))

# complex variable.

c=10+3j

print("The type of variable having value", c, " is ", type(c))

**Python String Data Type**

Python Strings are identified as a contiguous set of characters represented in the quotation marks. Python allows for either pairs of single or double quotes. Subsets of strings can be taken using the slice operator ([ ] and [:] ) with indexes starting at 0 in the beginning of the string and working their way from -1 at the end.

The plus (+) sign is the string concatenation operator and the asterisk (\*) is the repetition operator in Python. For example −

str = 'Hello World!'

print (str) # Prints complete string

print (str[0]) # Prints first character of the string

print (str[2:5]) # Prints characters starting from 3rd to 5th

print (str[2:]) # Prints string starting from 3rd character

print (str \* 2) # Prints string two times

print (str + "TEST") # Prints concatenated string

This will produce the following result −

Hello World!

H

llo

llo World!

Hello World!Hello World!

Hello World!TEST

**Python Boolean Data Types**

Python **boolean** type is one of built-in data types which represents one of the two values either **True** or **False**. Python **bool()** function allows you to evaluate the value of any expression and returns either True or False based on the expression.

Examples

Following is a program which prints the value of boolean variables a and b −

a = True

# display the value of a

print(a)

# display the data type of a

print(type(a))

This produce the following result −

true

<class 'bool'>

Following is another program which evaluates the expressions and prints the return values:

# Returns false as a is not equal to b

a = 2

b = 4

print(bool(a==b))

# Following also prints the same

print(a==b)

# Returns False as a is None

a = None

print(bool(a))

# Returns false as a is an empty sequence

a = ()

print(bool(a))

# Returns false as a is 0

a = 0.0

print(bool(a))

# Returns false as a is 10

a = 10

print(bool(a))

This produce the following result −

False

False

False

False

False

True

**Python Data Type Conversion**

Sometimes, you may need to perform conversions between the built-in data types. To convert data between different Python data types, you simply use the type name as a function.

**Conversion to int**

Following is an example to convert number, float and string into integer data type:

a = int(1) # a will be 1

b = int(2.2) # b will be 2

c = int("3") # c will be 3

print (a)

print (b)

print (c)

This produce the following result −

1

2

3

**Conversion to float**

Following is an example to convert number, float and string into float data type:

a = float(1) # a will be 1.0

b = float(2.2) # b will be 2.2

c = float("3.3") # c will be 3.3

print (a)

print (b)

print (c)

This produce the following result −

1.0

2.2

3.3

**Conversion to string**

Following is an example to convert number, float and string into string data type:

a = str(1) # a will be "1"

b = str(2.2) # b will be "2.2"

c = str("3.3") # c will be "3.3"

print (a)

print (b)

print (c)

This produce the following result −

1

2.2

3.3