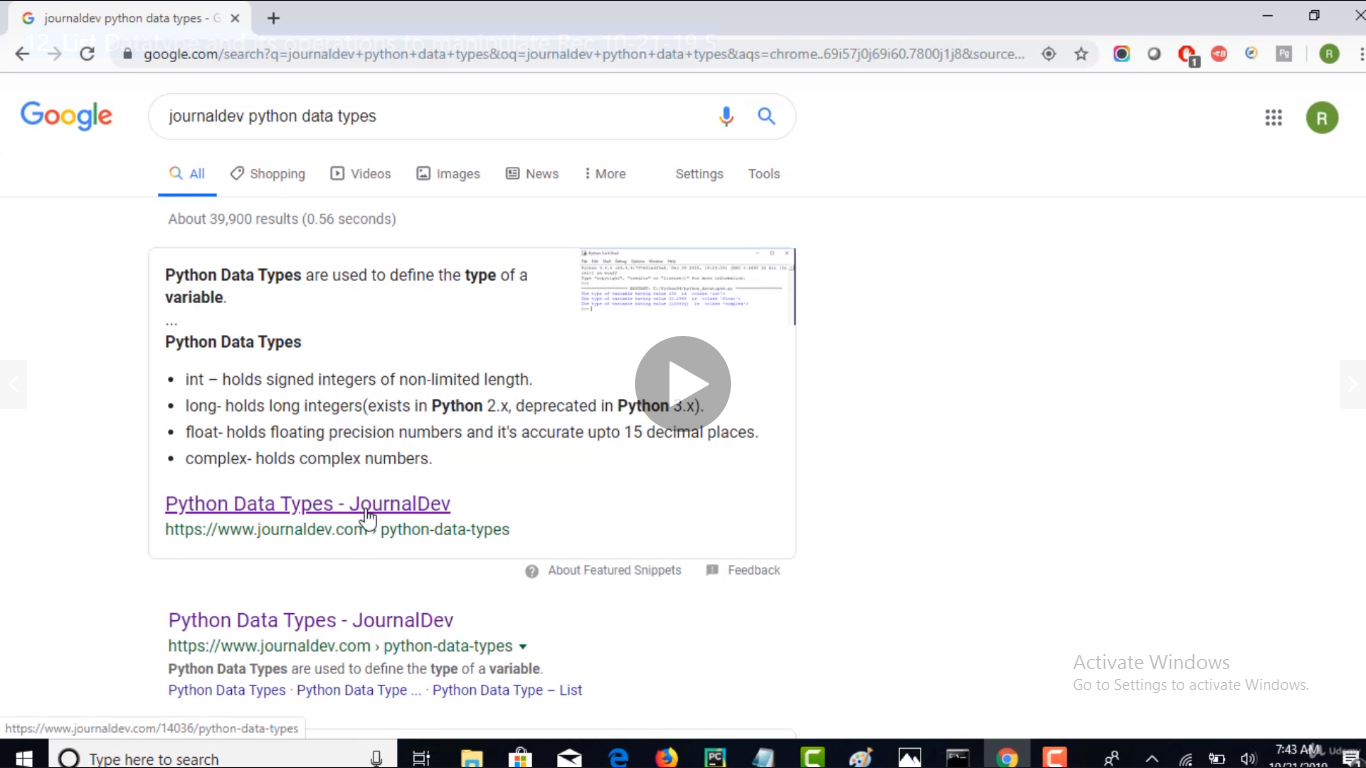
* Refer this <https://www.journaldev.com/14036/python-data-types>



* In Python we need not to declare datatype while declaring a variable like C or C++. We can simply just assign values in a variable. But if we want to see what type of numerical value is it holding right now, we can use type().
* Python data types:

1. Numeric
2. int – holds signed integers of non-limited length.
3. long- holds long integers(exists in Python 2.x, deprecated in Python 3.x).
4. float- holds floating precision numbers and it’s accurate upto 15 decimal places.

complex- holds complex numbers.

Eg:

#create a variable with integer value.

a=100

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

#create a variable with float value.

b=10.2345

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

#create a variable with complex value.

c=100+3j

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

1. [String](https://www.journaldev.com/14036/python-data-types#python-data-type-8211-string)
2. The string is a sequence of characters. Python supports Unicode characters. Generally, strings are represented by either single or double quotes.

Eg:

a = "string in a double quote"

b= 'string in a single quote'

print(a)

print(b)

# using ',' to concatenate the two or several strings

print(a,"concatenated with",b)

#using '+' to concate the two or several strings

print(a+" concated with "+b)

Note: we can concatenate the same datatype using ‘+’. But not two different datatypes.

1. [List](https://www.journaldev.com/14036/python-data-types#python-data-type-8211-list)
2. [Tuple](https://www.journaldev.com/14036/python-data-types#python-data-type-8211-tuple)
3. [Dictionary](https://www.journaldev.com/14036/python-data-types#dictionary)