

## There are techniques to print output of a variable:

Ex1:

```
a=10
print(a)
print("The value of a is: ", a)
```

Ex2:

```
a=10
b=20
print(a+b)
c=a+b
print(c)
print("the sum of ", a, "and ", b, "is: ", c )
```

Ex3:

```
a=10
b="A"
c=a+b
print(c)
```

this will show some error. But if we want to add two or more string-

```
a="A"
```

```
b="B"
```

```
print(a+b)
```

this is valid.

## Scope of a variable in python:

The value of a function created inside any block has the scope only to that block. otherwise the global value is appeared.

```
a=10
def func():
    print(a)
```

```
func()
```

it will print the given value of a

lets have another example-

```
a=10
def func():
    a=100
    print("Inside value of a is: "a)
print("Outside value of a is: ",a)
func()
```

-----

Outside value of a is: 10

Inside value of a is: 100

**\*global keyword:** if we want to create a global variable from inside of a block

```
a=10
```

```
b=20
```

```
def func():
```

```
a=100
global b
b=200
print("Inside value of a is: ", a, "and b is: ",b)
func()
print("Outside value of a is: ",a, "and b is: ",b)
```

---

Inside value of a is: 100 and b is: 200

Outside value of a is: 10 and b is: 200

### **Python has the following data types**

built-in by default, in these categories:

Text Type:str

Numeric Types:int, float, complex

Sequence Types:list, tuple, range

Mapping Type:dict

Set Types:set, frozenset

Boolean Type:bool

Binary Types:bytes, bytearray, memoryview

-we check the type of any data by **type()** function

```
a=10
print(type(a))
```

The data types are automatically be assigned when we put some value on it.

### **Setting the Specific Data Type**

```
a=int(input("Enter some value: "))
```

If you want to specify the data type, you can use the following constructor functions:

Example

```
x = str("Hello World")
»x = int(20)
»x = float(20.5)
»x = complex(1j)
»x = list(("apple", "banana", "cherry"))
»x = tuple(("apple", "banana", "cherry"))
»x = range(6)
»x = dict(name="John", age=36)
»x = set(("apple", "banana", "cherry"))
»x = frozenset(("apple", "banana", "cherry"))
»x = bool(5)
»x = bytes(5)
»x = bytearray(5)
»x = memoryview(bytes(5))
```

### **Type Casting:**

## Specify a Variable Type

There may be times when you want to specify a type on to a variable. This can be done with casting. Python is an object-orientated language, and as such it uses classes to define data types, including its primitive types.

Casting in python is therefore done using **constructor functions**:

- **int()** - constructs an integer number from an integer literal, a float literal (by removing all decimals), or a string literal (providing the string represents a whole number)
- **float()** - constructs a float number from an integer literal, a float literal or a string literal (providing the string represents a float or an integer)
- **str()** - constructs a string from a wide variety of data types, including strings, integer literals and float literals