A variable is used to store a value. You can think of a variable as storage that has a name and stores some value.

In [1]:
```
a = 10
```

Here a is a variable and 10 is the value of variable a.

In [4]:
```
# declaring a variable count
count = 1
print(count)

# reassigning value to the variable count
count = "Hello World"
print(count)
```

```
1
Hello World
```

In [5]:
```
#Assigning Values to Multiple Variables
a=1
b=2
print(a,b)
print (a)
print(b)
```

```
1 2
1
2
```

In [6]:
```
#Swapping
x = 10
y = 20
x,y = y,x
print(x,y)
print(x)
print(y)
```

```
20 10
20
10
```

In [9]:
```
#Deleting Variables
color = "Blue"
del color
print (color) # It will give error as color variable has been deletd.
```

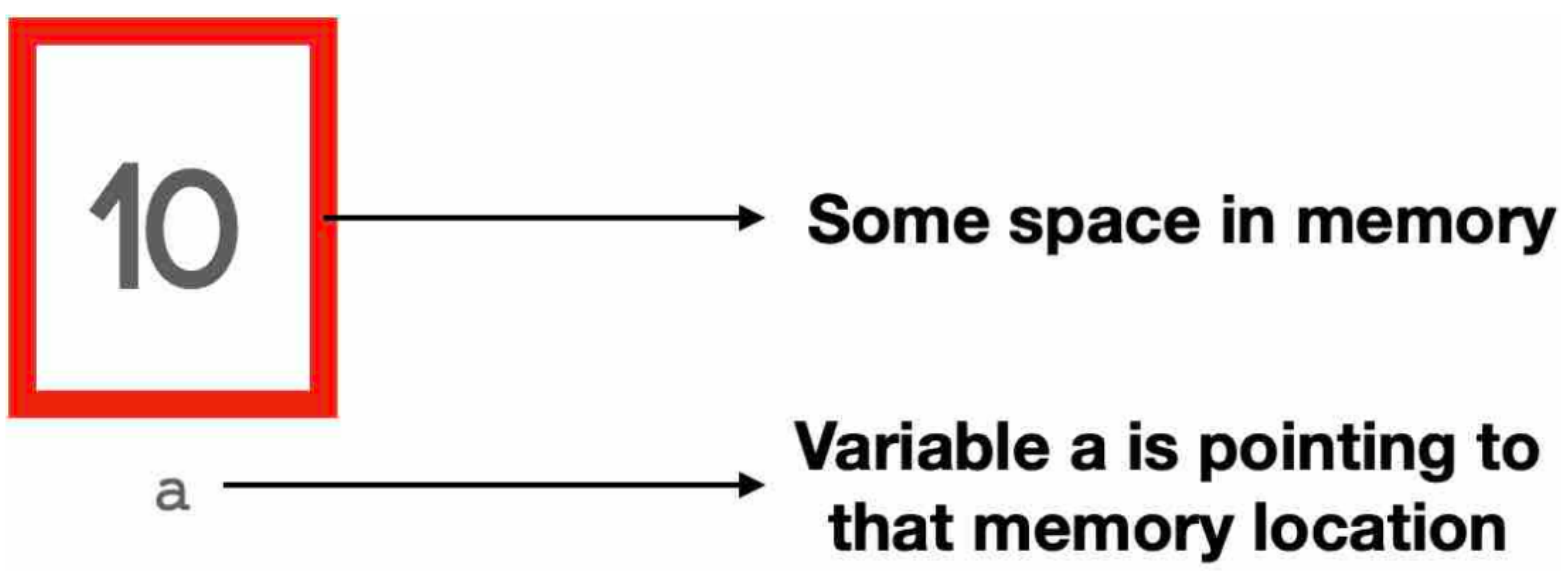
```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-9-d667fb088018> in <module>
      2 color = "Blue"
      3 del color
----> 4 print (color) # It will give error as color variable has been deletd.

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

Memory Management in Python

In Python, when we use a variable, it is stored in the memory of the computer. For example, two variables having values 5 and 10 will be stored in two different memory locations by Python.

In [10]:
```
a = 10
```



We can check the memory location of a value by using the id() function.

In [12]:
```
a = 10
b = 20
print(id(a))  # printing memory location pointed by variable a
print(id(b))  # printing memory location pointed by variable b
```

```
140703429765200
140703429765520
```

This id will be different every time you run your program because a different memory in the CPU will be allocated everytime.