CLASS OR STATIC VARIABLES IN PYTHON

Class or static variables are shared by all objects. Instance or non-static variables are different for different objects (every object has a copy of it).

For example, let a Computer Science Student be represented by class **CS Student**. The class may have a static variable whose value is “cse” for all objects. And class may also have non-static members like **name** and **roll**.
In C++ and Java, we can use static keyword to make a variable as class variable. The variables which don’t have preceding static keyword are instance variables. See this for Java example and this for C++ example.
The **Python** approach is simple, it doesn’t require a static keyword. *All variables* which are assigned a value in class declaration are class variables. And variables which are assigned values inside class methods are instance variables.

```
# Python program to show that the variables with a value

# assigned in class declaration, are class variables


# Class for Computer Science Student

class CSStudent:

        stream = 'cse'                          # Class Variable

        def __init__(self,name,roll):

        self.name = name                        # Instance Variable

        self.roll = roll                        # Instance Variable


# Objects of CSStudent class

a = CSStudent('Geek', 1)
```

```
b = CSStudent('Nerd', 2)


print(a.stream) # prints "cse"

print(b.stream) # prints "cse"

print(a.name) # prints "Geek"

print(b.name) # prints "Nerd"

print(a.roll) # prints "1"

print(b.roll) # prints "2"


# Class variables can be accessed using class

# name also

print(CSStudent.stream) # prints "cse"
```

Output:

```
cse
cse
Geek
Nerd
1
2
cse
```