



Eg:

```
_name='durga'
```

But is just convention and in reality does not exist protected attributes.

private attributes can be accessed only within the class.i.e from outside of the class we cannot access. We can declare a variable as private explicitly by prefixing with 2 underscore symbols.

syntax: __variablename=value

Eg: __name='durga'

Demo Program:

```
1) class Test:
2)     x=10
3)     _y=20
4)     __z=30
5)     def m1(self):
6)         print(Test.x)
7)         print(Test._y)
8)         print(Test.__z)
9)
10) t=Test()
11) t.m1()
12) print(Test.x)
13) print(Test._y)
14) print(Test.__z)
```

Output:

```
10
20
30
10
20
```

Traceback (most recent call last):

```
File "test.py", line 14, in <module>
```

```
print(Test.__z)
```

AttributeError: type object 'Test' has no attribute '__z'