>>> from guidoSingleton import Singleton

>>> dir()

['A', 'Doctype', 'Document', 'EndHead', 'EndHtml', 'Singleton', 'StartHead', 'StartHtml', 'Tag', '\_\_builtins\_\_', '\_\_doc\_\_', '\_\_name\_\_', '\_\_package\_\_', 'a1', 'a2', 'doc1', 'pywin']

>>> class B(Singleton):pass

...

>>> b1 = B()

>>> b2 = B()

>>> b1

<\_\_main\_\_.B object at 0x00000000046BF1D0>

>>> b2

<\_\_main\_\_.B object at 0x00000000046BF1D0>

>>> b1.value = 42

>>> b2.value

42

>>> b2.value = 666

>>> b1.value

666

>>> class BB(B):pass

...

>>> b3 = BB()

>>> b3

<\_\_main\_\_.BB object at 0x00000000046BF048>

>>> b3.value = 42

>>> b1.value

666

>>> b4 = BB()

>>> b4.value

42

>>> b1.value = 13

>>> b4.value

42

>>> del( Singleton)

>>> dir()

['A', 'B', 'BB', 'Doctype', 'Document', 'EndHead', 'EndHtml', 'Singleton', 'StartHead', 'StartHtml', 'Tag', '\_\_builtins\_\_', '\_\_doc\_\_', '\_\_name\_\_', '\_\_package\_\_', 'a1', 'a2', 'b1', 'b2', 'b3', 'b4', 'doc1', 'pywin']

>>> class C(Singleton):pass

...

>>> c1 = C()

>>> c2 = C()

>>> c1

<\_\_main\_\_.C object at 0x00000000046B94E0>

>>> c2

<\_\_main\_\_.C object at 0x00000000046B94E0>

>>> c1.value = 42

>>> c2.value

42

>>> class CC(C):pass

...

>>> c3 =- CC()

Traceback (most recent call last):

File "<interactive input>", line 1, in <module>

TypeError: bad operand type for unary -: 'C'

>>> c3 = CC()

>>> c3.value

42

>>> c3

<\_\_main\_\_.C object at 0x00000000046B94E0>

>>> dir()

['A', 'B', 'BB', 'Borg', 'C', 'CC', 'Doctype', 'Document', 'EndHead', 'EndHtml', 'Singleton', 'StartHead', 'StartHtml', 'Tag', '\_\_builtins\_\_', '\_\_doc\_\_', '\_\_name\_\_', '\_\_package\_\_', 'a1', 'a2', 'b1', 'b2', 'b3', 'b4', 'c1', 'c2', 'c3', 'doc1', 'pywin']

>>> class D(Borg):pass

...

>>> d1 = D()

>>> d2 = D()

>>> d1

<\_\_main\_\_.D object at 0x00000000045F8048>

>>> d2

<\_\_main\_\_.D object at 0x00000000045F87F0>

>>> d1.value = 42

>>> d2.value

42

>>>

from borg import Borg as Singleton

class A(Singleton):

pass

a1 = A()

a2 = A()

print(a1, a2)

a1.name = 'freda'

print(a2.name)

class B(A):

pass

b1 = B()

b2 = B()

print(b1, b2)

print(b2.name)