Builtin attributes of a class:
----

whenever we are creating a class internally our python interpreter to add some attributes to that particular class,that type of attributes are called Builtin attributes of a class.

__name__ --> to return the name of the classs

__doc__ --> to return the documentation string of a class

__module__ --> to return the module name,in python by default every python file act as a one module,that module is called main module.

__bases__ --> to return the base class name,by default in python object class is the Base class of all other classes.

__dict__ --> to return the builtin attributes and user-defined attributes in the form of dictionary format.

__str__ --> to return the string format

__repr__ --> to return the original represenation of an object

__mro__ --> to return the method resolution order

__init__ --> constructor
s
__del__ --> destructor

.....
.....

ex1:

```
class test:
    """simple class example"""
    x=10
    y="siva"
    def m1(self):
        pass
    def m2(self):
        pass
print(test.__doc__)
print('*'*20)
print(test.__name__)
print('*'*20)
print(test.__module__)
print('*'*20)
print(test.__bases__)
```

```
print('*'*20)
print(test.__dict__)

output:
-------
simple class example
********************
test
********************
__main__
********************
(<class 'object'>,)
********************
{
'__module__': '__main__',
'__doc__': 'simple class example',
'x': 10,
'y': 'siva',
'm1': <function test.m1 at 0x0000021F1E0363B0>,
'm2': <function test.m2 at 0x0000021F1E036440>,
......
......
}

ex2:
---
import datetime
x=datetime.datetime.now()
print(x)
print(str(x))
print(repr(x))

output:
------
2021-11-28 16:05:49.219633
2021-11-28 16:05:49.219633
datetime.datetime(2021, 11, 28, 16, 5, 49, 219633)
```