```
class myclass:
    x = 10
    def f1(): # class method or function
        print('myclass:f1( )')
        print('myclass.x =',myclass.x)

class yourclass:
    x = 20
    def f1():
        print('yourclass:f1( )')
        print('yourclass.x =',myclass.x)
```

```
myclass.f1()
```

```
myclass:f1( )
myclass.x = 10
```

```
yourclass.f1()
```

```
yourclass:f1( )
yourclass.x = 10
```

```
class myclass:
    x = 10
    @classmethod
    def f1(cls): # class method or function
        print('myclass class method:f1( )')
        print('myclass.x =',cls.x)
    def f2(self):
        print('myclass instance method:f2( )')
    @staticmethod
    def f3():
        print('myclass static method:f3( )')

class yourclass:
    x = 20
    @classmethod
    def f1(cls): # class method or function
        print('yourclass:f1( )')
        print('yourclass.x =',cls.x)
    def f2(self):
        print('yourclass instance method:f2( )')
    @staticmethod
    def f3():
        print('yourclass static method:f3( )')
```

```
myclass.f1()
```

```
myclass class method:f1( )
myclass.x = 10
```

```
yourclass.f1()
```

```
yourclass:f1( )
yourclass.x = 20
```

```
m1 = myclass()
y1 = yourclass()
```

```
m1.f2()
```

```
myclass instance method:f2( )
```

```
y1.f2()
```

```
y1.f2()
```

```
yourclass instance method:f2( )
```

```
myclass.f3()
```

```
myclass static method:f3( )
```

```
myclass.f1()
```

```
myclass class method:f1( )
myclass.x = 10
```

```
m1.f1()
```

```
myclass class method:f1( )
myclass.x = 10
```

```
myclass.f3()
```

```
myclass static method:f3( )
```

```
m1.f3()
```

```
myclass static method:f3( )
```

```
class emp:
    def __init__(self,name,company):
        self.name = name
        self.company = company
    def __call__(self):
        print('*'*40)
        print('Name :',self.name)
        print('Company :',self.company)
        print('*'*40)
```

```
e1 = emp('prasanth','infosys')
e2 = emp('venkat','microsoft')
e3 = emp('prudhviraj','tcs')
```

```
e1()
```

```
****************************************
Name : prasanth
Company : infosys
****************************************
```

```
e2()
```

```
****************************************
Name : venkat
Company : microsoft
****************************************
```

```
e3()
```

```
****************************************
Name : prudhviraj
Company : tcs
****************************************
```

```
class emp:
    employees = [] # class variable employees
    def __init__(self,name,company):
        self.name = name
        self.company = company
        emp.employees.append(self)
    def __call__(self):
        print('*'*40)
        print('Name :',self.name)
        print('Company :',self.company)
        print('*'*40)
    def __str__(self):
        return self.name + ' , ' + self.company
    @classmethod
    def search(cls,name):
        for e in cls.employees:
            if e.name.lower() == name.lower():
                print('Name =',e.name)
                print('Company =',e.company)
                break
        else:
            print(name,'details not found')
```

```
print(emp.employees)
```

```
[]
```

```
e1 = emp('prasanth','infosys')
```

```
for e in emp.employees:
    print(e)
```

```
prasanth , infosys
```

```
e2 = emp('venkat','microsoft')
```

```
for e in emp.employees:
    print(e)
```

```
prasanth , infosys
venkat , microsoft
```

```
e3 = emp('prudhviraj','tcs')
```

```
for e in emp.employees:
    print(e)
```

```
prasanth , infosys
venkat , microsoft
prudhviraj , tcs
```

```
emp.search('venkat')
```

```
Name = venkat
Company = microsoft
```

```
emp.search('praSANTH')
```

```
Name = prasanth
Company = infosys
```

```
emp.search('sai')
```

```
sai details not found
```

✓ 0s completed at 2:48 PM ● ×