

```

class Sample:
    pass

class Sample(object):
    pass

# %%
class A:
    pass
class B(A):
    pass
class C(B):
    pass
print(C.__mro__)
# (<class '__main__.C'>, <class '__main__.B'>, <class '__main__.A'>, <class 'object'>)
# %%
class A:
    pass
class B:
    pass
class C(A, B): # çoklu türetme
    pass

# %%
class A:
    def foo(self):
        print('foo')
class B:
    def bar(self):
        print('bar')
class C(A, B):
    def tar(self):
        print('tar')
c = C()
c.foo()
c.bar()
c.tar()
# %%
class A:
    def __init__(self):
        print('A __init__ called: {}'.format(type(self)))
        self.x = 10
    def dispA(self):
        print('A.Disp(): {}'.format(self.x))
class B(A):
    def __init__(self):

```

```

        print('B __init__ called: {}'.format(type(self)))
        super(B, self).__init__()
        self.x = 20
    def dispB(self):
        print('B.Disp: {}'.format(self.x))

b = B()
b.dispB()
b.dispA()
"""
B.Disp: 20
A.Disp(): 20
"""
# %%
class A:
    def foo(self):
        print('A.foo')
class B(A):
    def foo(self):
        print('B.foo')

a = A()
a.foo()  # A.foo çağrılır

b = B()
b.foo()  # B.foo çağrılır
# %%
class A:
    def foo(self):
        print('A.foo')
class B(A):
    def foo(self):
        print('B.foo')

b = B()
A.foo(b)  # A.foo çağrılır
super(B, b).foo()  # A.foo çağrılır
# %%
def foo():
    print('foo')
print(foo)  # <function foo at 0x107921ee0>
print(type(foo))  # <class 'function'>
# %%
def foo():
    print('foo')
foo()  # foo
def foo():
    print('other foo')
foo()  # other foo
# %%

```

```

x = 123
print(x)

x = 'Atakan'
print(x)
# %%
def foo():
    print('foo')
bar = foo
foo() # foo
bar() # foo

# %%
def foreach(l, f):
    for x in l:
        f(x)
def foo(x):
    print('foo: {}'.format(x))
foreach([1, 2, 3, 4, 5], foo)
"""
foo: 1
foo: 2
foo: 3
foo: 4
foo: 5
"""
foreach([1, 2, 3, 4, 5], print)
"""
1
2
3
4
5
"""
# %%
class Sample:
    def foo(self):
        print('parametresiz foo')
    def foo(self, a):
        print('parametreli foo')
s = Sample()
s.foo(10) # parametreli foo
s.foo() # error!!!
# %%
class Sample:
    def foo(self):
        print('foo')

```

```

def tar(self):
    print('bar')
Sample.bar = tar
s = Sample()
s.foo()
s.bar()
"""
foo
bar
"""
Sample.x = 100
print(Sample.x) # 100
# %%
class A:
    def foo(self):
        print("A.foo")
class B(A):
    def foo(self):
        print("B.foo")
class C(B):
    pass
c = C()
c.foo() # B.foo
"""
print(C.__mro__)
# (<class '__main__.C'>, <class '__main__.B'>, <class
'__main__.A'>, <class 'object'>)
"""
# %%
class A:
    pass
class B(A):
    pass
class C(A, B):
    pass
print(C.__mro__)
# %%
class A:
    pass
class B(A):
    pass
class C:
    pass
class D(C):
    pass
class E(B, D):
    pass

```

```
print(E.__mro__)
# (<class '__main__.E'>, <class '__main__.B'>, <class
'__main__.A'>, <class '__main__.D'>, <class '__main__.C'>,
<class 'object'>)

# %%
import tensorflow as tf
import numpy as np
```