



class

def

0

0





#Makeaninstantobject!

$$-4.5 \quad 3.0 \quad -4.5$$

$$\# \Rightarrow (3.0, -4.5)$$









# Custom Constructor using `__init__`

```
class Complex:
```

```
    def __init__(self, realpart=0, imagpart=0):
```

```
        self.real = realpart
```

```
        self.imag = imagpart
```

Class instantiation calls the special method `__init__` if it exists, supplying a freshly-minted instance object as the first parameter.

```
# Make an instance object!
```

```
c = Complex(3.0, -4.5)
```

```
c.real, c.imag # => (3.0, -4.5)
```

You can't overload `__init__`!  
Use keyword arguments  
or factory methods

# Instance Objects

Only support attribute references