

Key Terms

- Object/Instance
- Class
- State/Fields
- Behavior/Methods
- Reference

Example Class

```
class Dog:
    def __init__(self, name):
        self.name = name

    def bark(self):
        print(self.name + ': Woof')
```

1. Define a class named **Cat** that meets the description on the slide. Watch for the syntax!

2. Write a main program (that use the main-method pattern) that constructs 2 **Cats**. The first should be named “Mocha” and should have 9 lives. The second should be named “Kitty” and should have 7 lives. Your code should then add 4 lives to Kitty before printing out the information for both cats (first Mocha, then Kitty).

3. In the space below, draw the memory model after the following code has been run.

```
cats = []
cats.append(Cat('Mimi', 5))
cats.append(Cat('Leslie', 9))
cats.append(Cat('Mimi', 5))
cats.append(Cat('Tyler', 3))
cats.append(cats[0])
```



4. Can you answer the following questions?

- What is the difference between a class and an object?
- Why does every method defined in a class need to take a `self` parameter? How do you access the fields/methods of an instance from inside the class definition.
- Why don't you need to pass `self` in when calling the method on an instance?