

# Structure and Interpretation of Computer Programs

with Python 

Lesson 10

-Presented By: Sean Li

# Object Oriented Programing

# Class

```
class Dog:  
    #define what dog are here
```



## `__init__` method

- Defines the properties of the class
- Can take unlimited number of parameters
- “Self” is always the first parameter
- “.” represents attributes
- Attributes created in `__init__()` are called instance attributes
- Class attributes are defined elsewhere

# `__init__` method

```
class Dog:
    #define what dog are here

    num_legs = 4
    num_heads = 1
    can_bark = True

    def __init__(self, name, kind):
        self.name = name
        self.kind = kind
```

# Object instantiation

```
[3] class Dog:
    #define what dog are here

    num_legs = 4
    num_heads = 1
    can_bark = True

    def __init__(self, name, kind):
        self.name = name
        self.kind = kind
```

```
[4] Dog("potato", "shiba")
```

```
[>] <__main__.Dog at 0x7fbf9a72acf8>
```



# Attribute Access

## Dot Notation

```
[3] class Dog:
      #define what dog are here

      num_legs = 4
      num_heads = 1
      can_bark = True

      def __init__(self, name, kind):
          self.name = name
          self.kind = kind
```

```
[6] potato = Dog("potato", "shiba")
      potato
```

```
> <__main__.Dog at 0x7fbf9a72aa58>
```

```
[7] potato.name
```

```
> 'potato'
```

```
[8] potato.kind
```

```
> 'shiba'
```

```
[9] potato.num_legs
```

```
> 4
```

Method/Function



```
class Dog:
    #define what dog are here

    num_legs = 4
    num_heads = 1
    can_bark = True

    def __init__(self, name, kind):
        self.name = name
        self.kind = kind

    def bark():
        print("woof")
```

```
class Dog:
    #define what dog are here

    num_legs = 4
    num_heads = 1
    can_bark = True

    def __init__(self, name, kind):
        self.name = name
        self.kind = kind

    def bark(self):
        print(self.name)
```

```
Dog.bark()
```

```
potato = Dog("potato", "shiba")
potato.bark()
```

## Exercise:

Create a car class. A car should have

- Four wheels
- Brand, Year, Price, Passengers (a list)
- Function: Drive
- Function: Sound the horn (Each car should have a difference sound)
- Function: add or delete passengers



Create a new car: Honda Civic, made in 2018, with price \$40,000. It currently has passengers: Sean the dude, Trump who knows it all, Biden the old guy.



Access each of the attributes of the car, drive, sound  
the horn, then delete passenger Trump who knows it all  
and add a new passenger Obama the terminator