

16th March, 2022

Object Functions

→ When one defines a class it's usually class + class name.

"Class Car:"

→ It's possible to include the object for.

Eg. Class car(object):
""" Class object for a car """

↓ Then the code continues normally

I still don't know the purpose of this yet.

Inheritance

This is a feature in oop, that allows one to define a class based on an existing class

→ Main class is base or super class

→ Inherited class: subclass or derived class

→ Create the instance

```
1 my_cat = Cat("Fluffy")
2 my_cat.eat() # prints "Fluffy is eating"
3 my_cat.meow()
✓ 0.0s
Fluffy is eating.
Meow!
```

Call the method eat from (Animal) base class

Call method meow from Cat class

attributes

```
1 ## Base class/ Super class
2 class Animal:
3     def __init__(self, name, species):
4         self.name = name
5         self.species = species
6
7     def eat(self):
8         print(f"{self.name} is eating.")
9
```

→ Method 1

→ Method 2

```
10 # subclass or derived class
11 #
12 class Cat(Animal):
13     def __init__(self, name):
14         super().__init__(name, "cat") # Animal class has 2 attributes ,name and species that has to be present
15
16     def meow(self):
17         print("Meow!")
18
```

Base class

→ Method 1

→ Method 2

Initializing the base class properly

Unique to only the class Cat.