class Car():

def \_\_init\_\_(self, make, model, year):

self.make = make

self.model = model

self.year = year

self.odometer\_reading = 0

def get\_descriptive\_name(self):

long\_name = str(self.year) + ' ' + self.make + ' ' + self.model

return long\_name.title()

def read\_odometer(self):

print("This car has " + str(self.odometer\_reading) + " miles on it.")

def update\_odometer(self, mileage):

if mileage >= self.odometer\_reading:

self.odometer\_reading = mileage

else:

print("You can't roll back an odometer!")

def increment\_odometer(self, miles):

self.odometer\_reading += miles

class Car():

class Battery():

def \_\_init\_\_(self, battery\_size=60):

self.battery\_size = battery\_size

def describe\_battery(self):

print("This car has a " + str(self.battery\_size) + "-kWh battery.")

def get\_range(self):

if self.battery\_size == 70:

range = 240

elif self.battery\_size == 85:

range = 270

message = "This car can go approximately " + str(range)

message += " miles on a full charge."

print(message)

class ElectricCar(Car):

def \_\_init\_\_(self, make, model, year):

super().\_\_init\_\_(make, model, year)

self.battery = Battery()