```
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 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()

c1 = car("Audi", "A4" , 2016)
C2 = Electric_car("Tesal", "Model S", 2016)

print(c1.get_full_name())
```

```
print(c2.get_full_name())


c1.increment_odometer(30)

c2.increment_odometer(60)


c1.print_odometer()

c2.print_odometer()c2.battery.print_range()
```