```
# Car Class example
#   Info on classes:
#   https://docs.python.org/3/tutorial/classes.html

class Car:
    """Common base class for all cars"""
    car_count = 0    # class variable for keeping a cars created counter

    # The __init__method accepts arguments for the
    # make, model, and mileage. It initializes the data
    # attributes with these values and increments the car counter.
    def __init__(self, make, model, mileage):
        self.__make = make
        self.__model = model
        self.__mileage = mileage
        Car.car_count += 1

    # The __str__method returns a string representation of the object.
    def __str__(self):
        return '{0:s} {1:s} with {2:,d} miles'.format(self.__make, self.__model, self.__mileage)


    # The following methods are mutators (setters)
    # for the class's data attributes.
    def set_make(self, make):
        self.__make = make

    def set_model(self, model):
        self.__model = model

    def set_mileage(self, mileage):
        self.__mileage = mileage

    # The following methods are the accessors (getters)
    # for the class's data attributes.
    def get_make(self):
       return self.__make

    def get_model(self):
        return self.__model

    def get_mileage(self):
        return self.__mileage

    # A method to increase the car mileage by miles driven
    def add_mileage(self, miles):
        self.__mileage += miles
```

```
############################################################
## Test the Car class
############################################################
## The main function.
def main():
    print('Create car1 ...')
    car1 = Car('Ford', 'Mustang', 1000)
    print( 'Make:{:s}'.format(car1.get_make()) )
    print('Model: {:s}'.format(car1.get_model()) )
    print('Mileage: {:,d}'.format(car1.get_mileage()) )
    print(car1)

    print('\nAdd 500 to car1 mileage ...')
    car1.add_mileage(500)
    print(car1)

    print('\nCreate car2 ...')
    car2 = Car('Chevrolet', 'Cruze', 5000)
    print(car2)

    print('\nCreate car3 ...')
    car3 = Car('BMW', '3 Series', 8000)
    print(car3)

    print()
    print('Total Car Count: {:,d}'.format(Car.car_count))
    print()


# Call the main function.
if __name__ == '__main__':
    main()


------------------------------------------------------------
                      Sample Output

Create car1 ...
Make:Ford
Model: Mustang
Mileage: 1,000
Ford Mustang with 1,000 miles

Add 500 to car1 mileage ...
Ford Mustang with 1,500 miles

Create car2 ...
Chevrolet Cruze with 5,000 miles

Create car3 ...
BMW 3 Series with 8,000 miles

Total Car Count: 3
```