**PMT: Vehicle Leases**

Arctic Bank leases vehicles for some of its senior employees. These vehicles come in three categories: **Royalty**, **Luxury**, and **Deluxe**. At the end of each year, the accountant needs to generate a report on all the leased vehicles that shows the amounts spent on vehicle leasing **annually**. The charges on the different categories of vehicles are shown below.

**Royalty**: a **monthly** rental fee of $1,000, a **monthly** insurance charge of $50, and a maintenance fee of 20% of total mileage (i.e. 20% of total miles covered in the year).

**Luxury**: a **monthly** rental fee of $800, a **monthly** insurance charge of $35, and a maintenance fee of 15% of total mileage (i.e. 15% of total miles covered in the year).

**Deluxe**: a **monthly** rental fee of $500, a **monthly** insurance charge of $20, and a maintenance fee of 10% of total mileage (i.e. 10% of total miles covered in the year).

All vehicles were leased at the beginning of the year and so were leased for 12 full months of the year.

Write a Python application to assist the accountant in computing the annual expenditure on vehicle leases for the year under review. Your application should have the following classes:

* **Vehicle**
* Classes **Royalty**, **Luxury**, and **Deluxe,** which are subclasses of **Vehicle**
* Class **Customer**

**Note:**

* Each class (including subclasses) should have an appropriate initializer method that initializes any attributes you think are necessary with their appropriate values.
* Superclass method(s) should be overridden in the subclasses where appropriate.

**Note**: You should not make any changes to the method names or parameter lists for the methods shown for each class.

The test function **main()** is also provided and **should not be modified**.

The methods in class **Customer** are shown below:

**class** Customer:  
**def** \_\_init\_\_ ...   
  
 *# Add this vehicle to list* **def** add\_vehicle(self, vehicle):  
  
 *# Return list of vehicles* **def** get\_vehicles(self):  
  
 *# Calculate total lease amount of all vehicles in list* **def** calculate\_total\_lease(self):

The methods in the Vehicle class are shown below:

**class** Vehicle:  
**def** \_\_init\_\_ ...  
  
 *# Return the brand of this vehicle* **def** get\_brand(self):  
  
 *# Return the vin (vehicle ID number) of this vehicle* **def** get\_vin(self):  
  
 *# Return the mileage of this vehicle* **def** get\_mileage(self):  
  
 *# Calculate and return the annual lease amount for this vehicle.* **def** get\_lease\_amount(self):

Also, you should test your classes using the main function below (**which should not be modified**):

**def** main():  
 cust = Customer()  
 vehicle1 = Royalty(**"Limousine"**, **'VZT2002357'**, 150) *# (brand, vin, mileage)* vehicle2 = Luxury(**"Lamborghini"**, **'VGH2005649'**, 200) *# (brand, vin, mileage)* vehicle3 = Deluxe(**"Lexus"**, **'VQP2005963'**, 200) *# (brand, vin, mileage)* cust.add\_vehicle(vehicle1)  
 cust.add\_vehicle(vehicle2)  
 cust.add\_vehicle(vehicle3)  
  
 vehicles = cust.get\_vehicles()  
  
 print(**"Vehicle Lease Expenses"**)  
 print(**"======================\n"**)  
  
 print(f'{"Vehicle Type":<15}{"Brand":<15}'

f'{"VIN":<12}'

f'{"Lease Amount":>10}')

print(f'{"------------":<15}{"-----":<15}'

f'{"---":<12}'

f'{"------------":>10}')  
  
 **for** i **in** range(len(vehicles)):  
 print(  
 **f'{**vehicles[i].get\_type()**:<15}{**vehicles[i].get\_brand()**:<15}'  
 f'{**vehicles[i].get\_vin()**:<12}'  
 f'{"$"}{**vehicles[i].get\_lease\_amount()**:>10,.2f}'**)  
  
 print()  
 print(  
 **f'{"Total vehicle lease amount: $"}{**cust.calculate\_total\_lease()**:>10,.2f}'**)  
  
 vehicle4 = Royalty(**"Rolls-Royce"**, **'VPL2006637'**, 200) *# (brand, vin, mileage)* vehicle5 = Deluxe(**"Camry"**, **'VVV2006415'**, 250) *# (brand, vin, mileage)* cust.add\_vehicle(vehicle4)  
 cust.add\_vehicle(vehicle5)  
  
 print()  
 **for** i **in** range(len(vehicles)):  
 print(  
 **f'{**vehicles[i].get\_type()**:<15}{**vehicles[i].get\_brand()**:<15}'  
 f'{**vehicles[i].get\_vin()**:<12}'  
 f'{"$"}{**vehicles[i].get\_lease\_amount()**:>10,.2f}'**)  
  
 print()  
 print(  
 **f'{"Total vehicle lease amount: $"}{**cust.calculate\_total\_lease()**:>10,.2f}'**)

**Expected output:**

Vehicle Lease Expenses

======================

Vehicle Type Brand VIN Lease Amount

------------ ----- --- ------------

Royalty Limousine VZT2002357 $ 12,630.00

Luxury Lamborghini VGH2005649 $ 10,050.00

Deluxe Lexus VQP2005963 $ 6,260.00

Total vehicle lease amount: $ 28,940.00

Royalty Limousine VZT2002357 $ 12,630.00

Luxury Lamborghini VGH2005649 $ 10,050.00

Deluxe Lexus VQP2005963 $ 6,260.00

Royalty Rolls-Royce VPL2006637 $ 12,640.00

Deluxe Camry VVV2006415 $ 6,265.00

Total vehicle lease amount: $ 47,845.00

Process finished with exit code 0

**PMT Submission**

Submit your completed program to the dropbox before it closes.