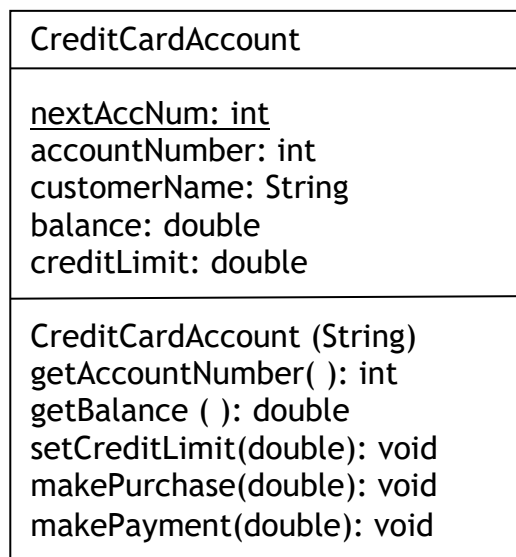


1. Modify the `BankAccount` class from earlier so that each instance will have a unique account number. Use a static variable to achieve this. Make the appropriate changes to your class. (Note: you will have to change the constructors)
2. A Credit Card works by allowing the user to spend money on credit. The user can spend money up to the value of their credit limit. The amount of money they spend is added to their balance, which represents the amount of money they owe. This works like a loan. So, making a purchase increases the balance, and making a payment to the credit card reduces the balance.

Implement a `CreditCardAccount` class as represented in the following UML diagram. The `makePurchase()` method should only increase the balance if the `CreditCardAccount` has a sufficient credit limit. The `makePayment()` method should allow a customer to pay off all or part of their balance.

(`nextAccNum` is a `static` variable, shown by its underlining.)



3. Write a program to test the `CreditCardAccount` class. The program should instantiate a `CreditCardAccount` object and set its limit to an appropriate value before displaying the following menu:
  1. Make Purchase
  2. Make Payment (on balanceDue)
  3. View Balance Due
  4. View account details
  5. Exit System

The user should be asked repeatedly to choose an option from the menu, until they choose to exit. Options 1 - 3 should result in calls to appropriate methods.

4. Write a program that allows the user to search an array of six double numbers for a particular value. The program should ask the user what value they wish to find, and should then output the position of that value, the value rounded to the nearest whole number, the square root, and the square of that value. Use static methods of the Math class to perform these operations. **Sample output:**

```
Contents of the array: 3.45 6.28 35.6 9.99 33.33
Enter number you wish to find: 9.99
```

```
9.99 number was found at position 3
Rounded to the nearest whole number: 10
Square root: 3.1606961258558215
Square: 99.8001
```