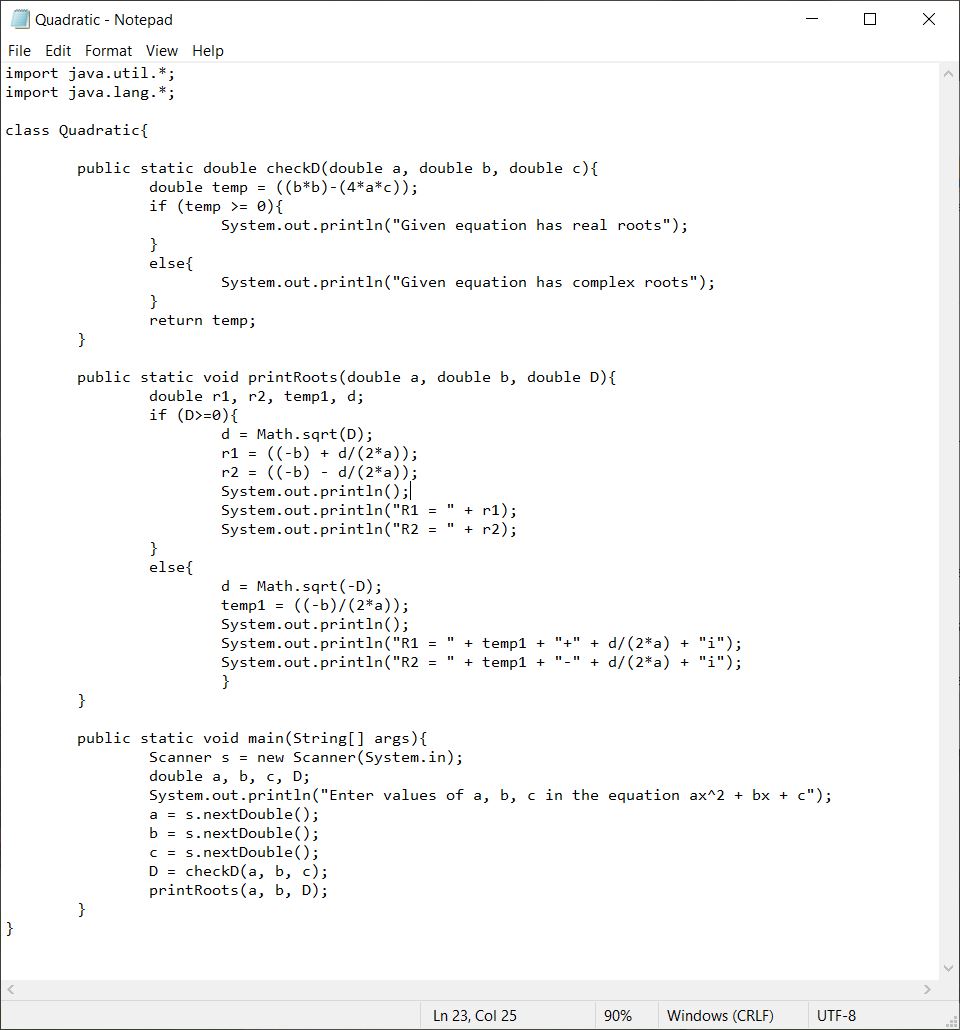
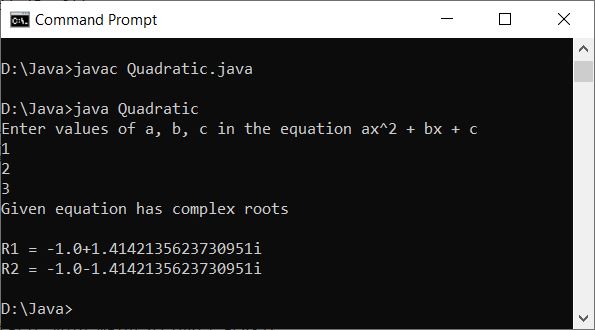
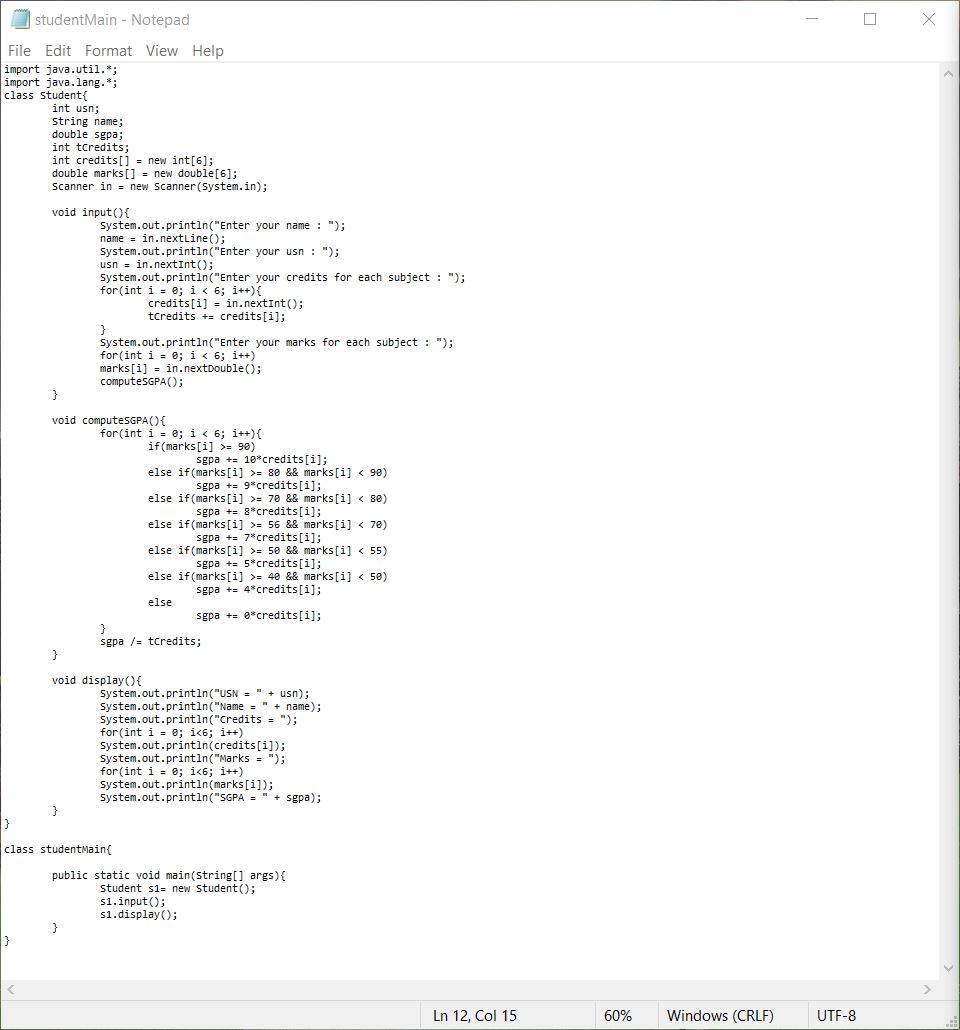
1. **Develop a Java program that prints all real solutions to the quadratic equation ax2 +bx+c = 0. Read in a, b, c and use the quadratic formula. If the discriminate b2-4ac is negative, display a message stating that there are no real solutions.**



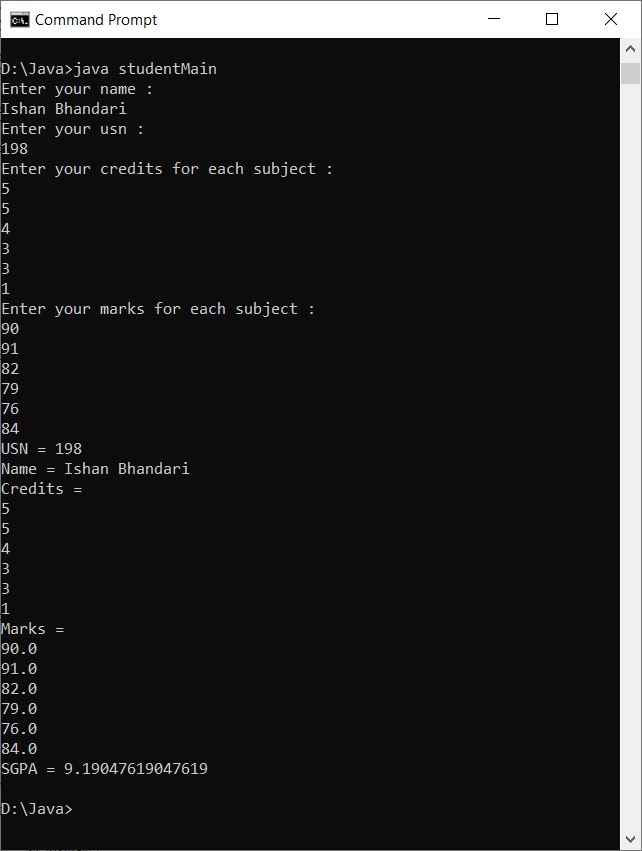
**Output:**



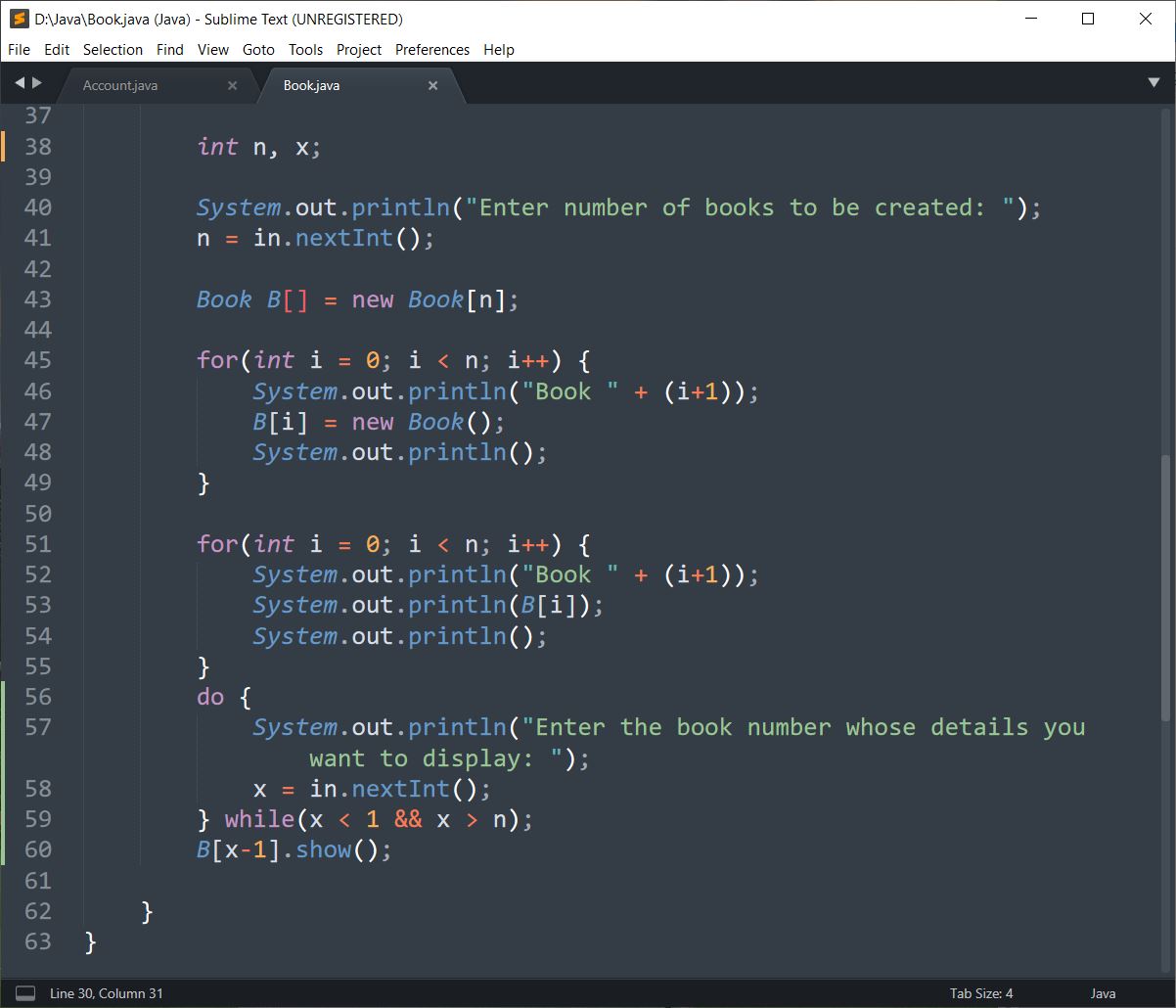
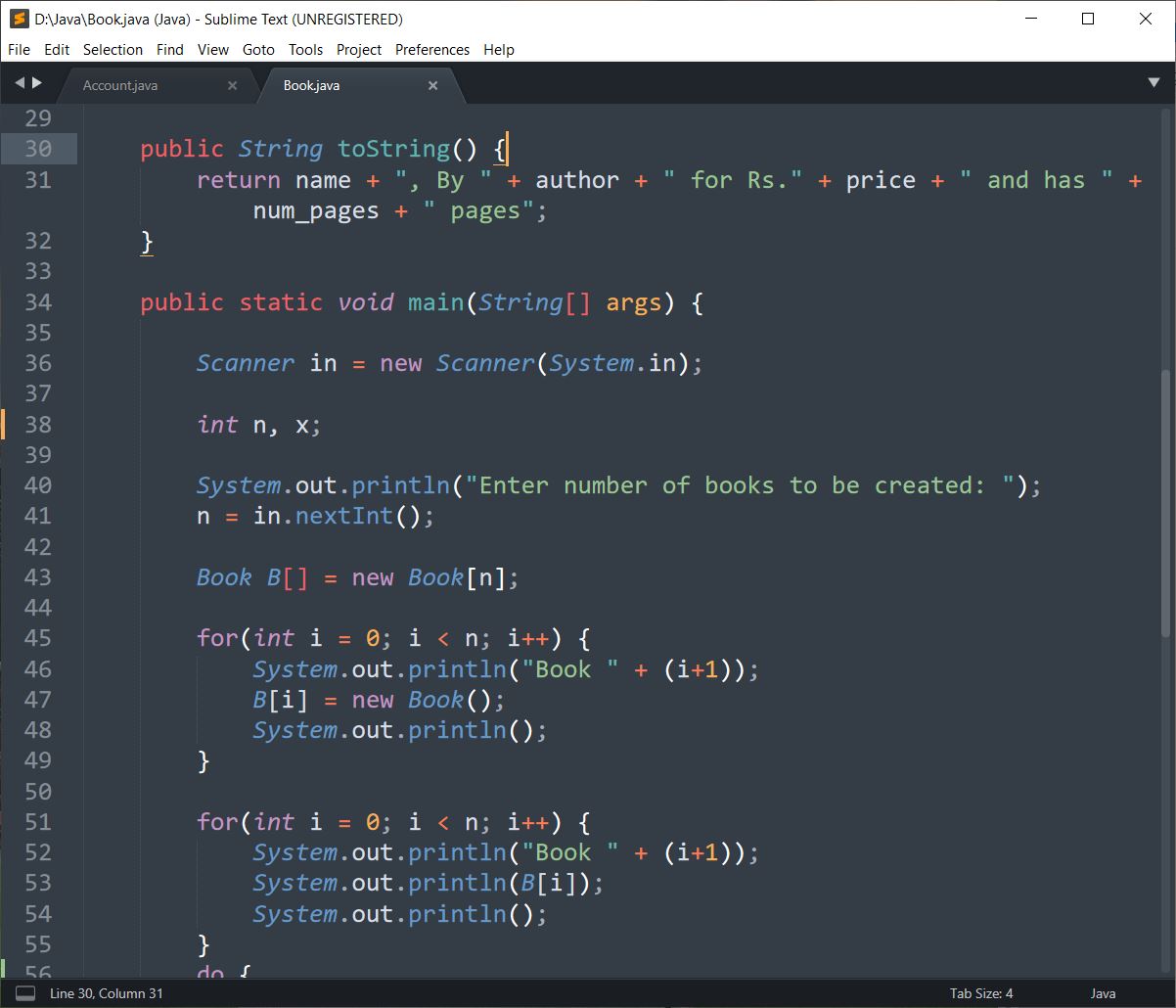
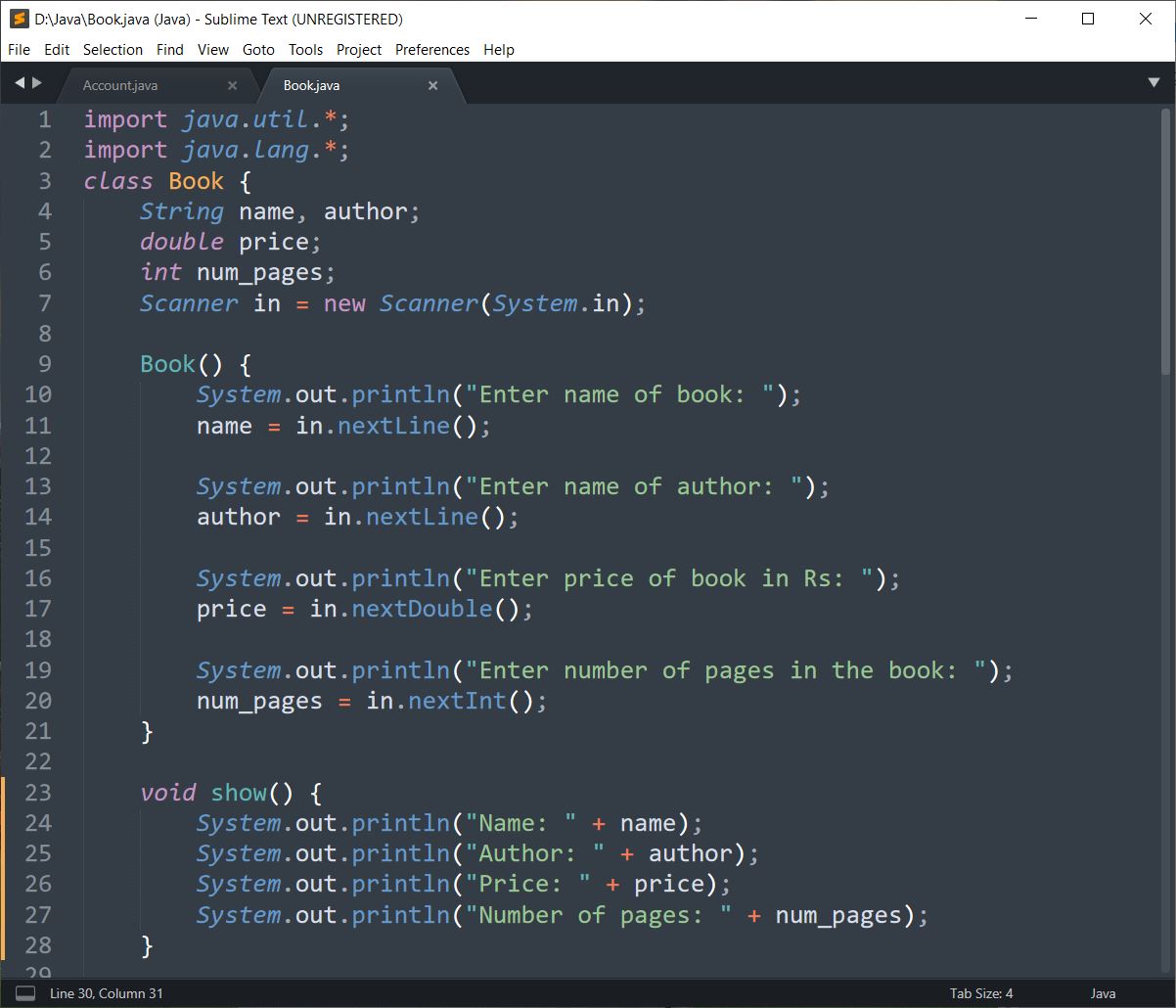
1. **Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.**



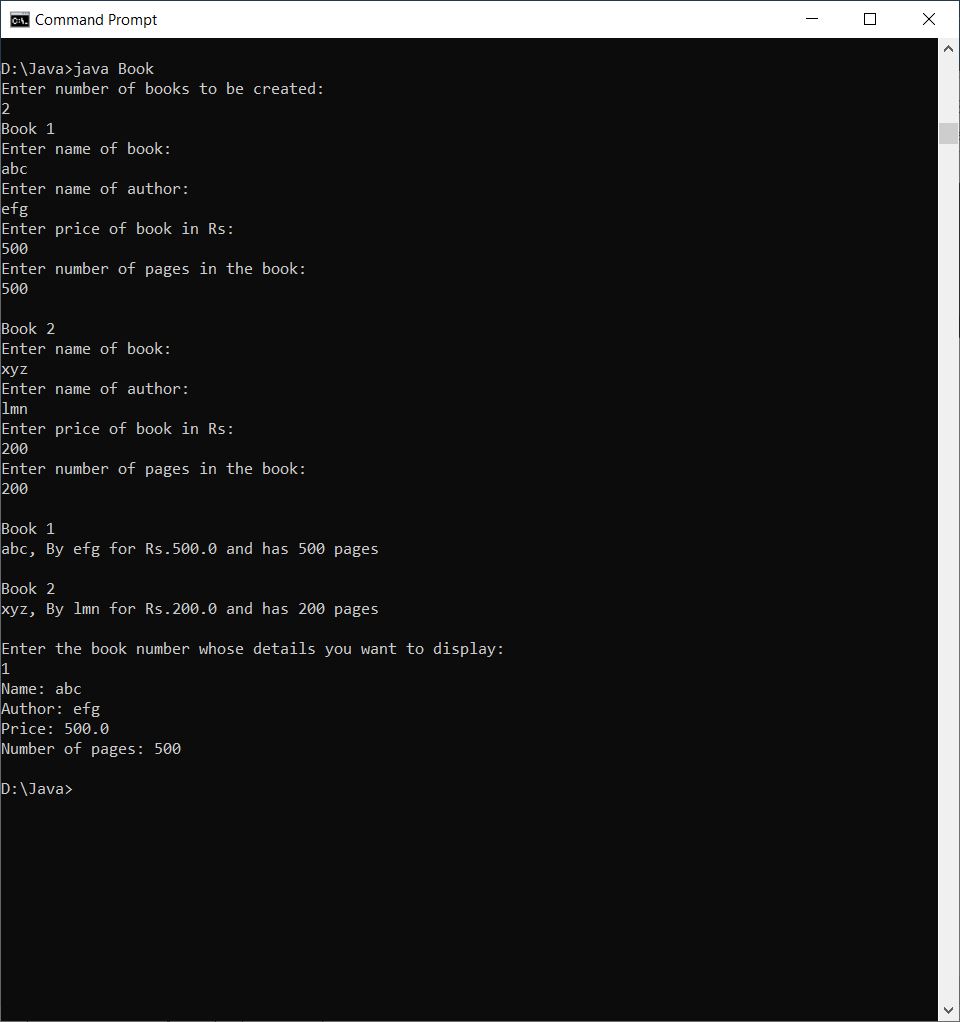
**Output:**



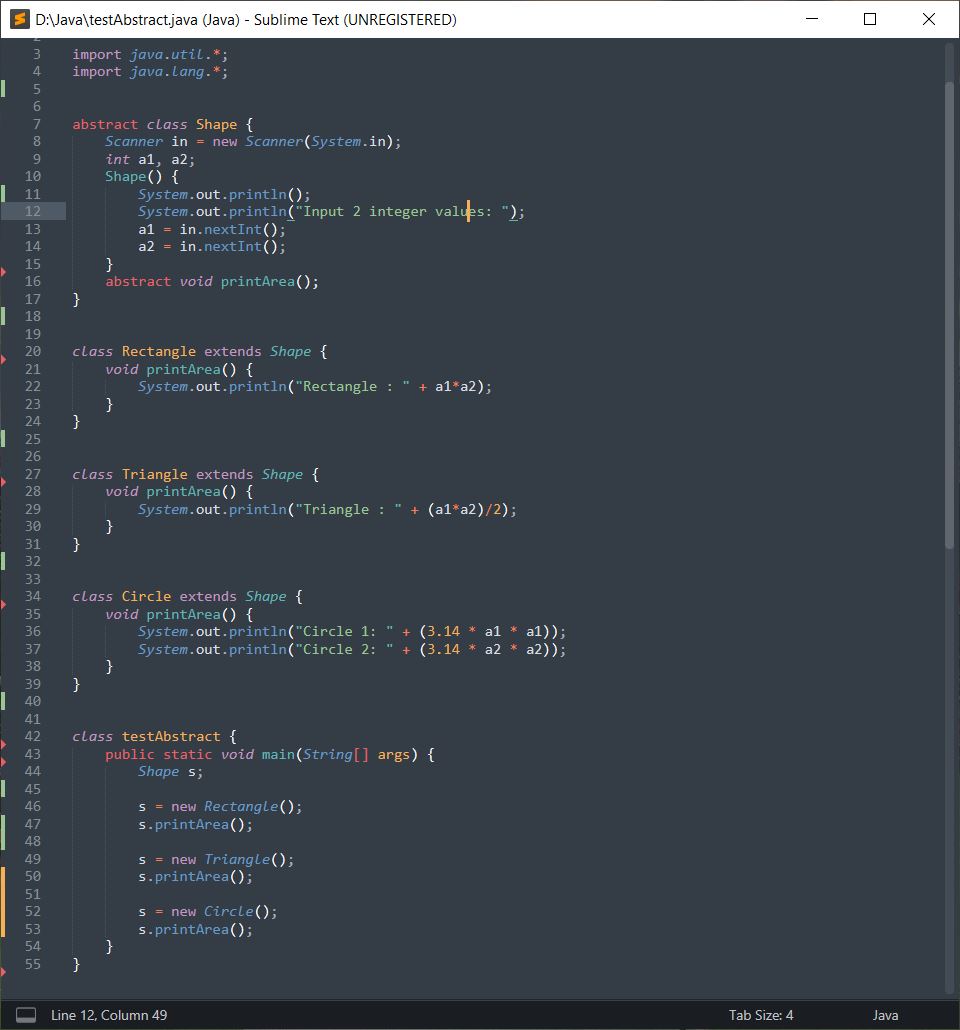
1. **Create a class Book which contains four members: name, author, price, num\_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString( ) method that could display the complete details of the book. Develop a Java program to create n book objects.**



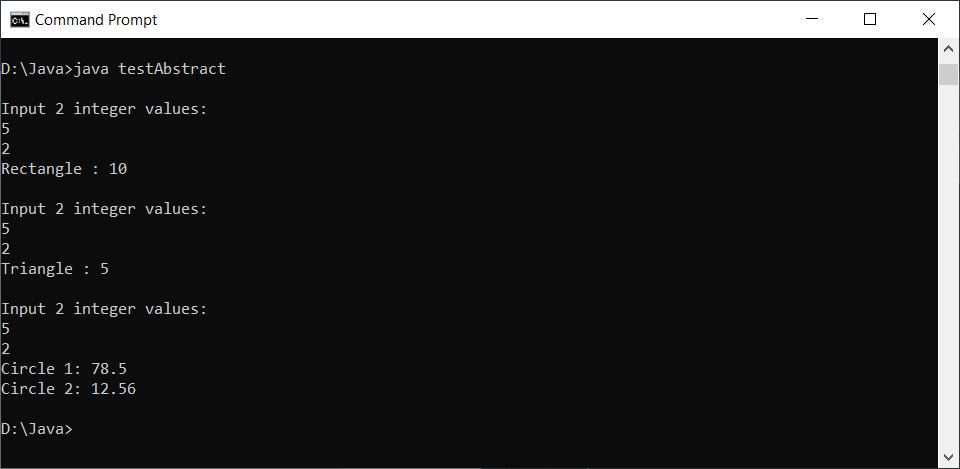
**Output:**



1. **Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea( ). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea( ) that prints the area of the given shape.**



**Output:**



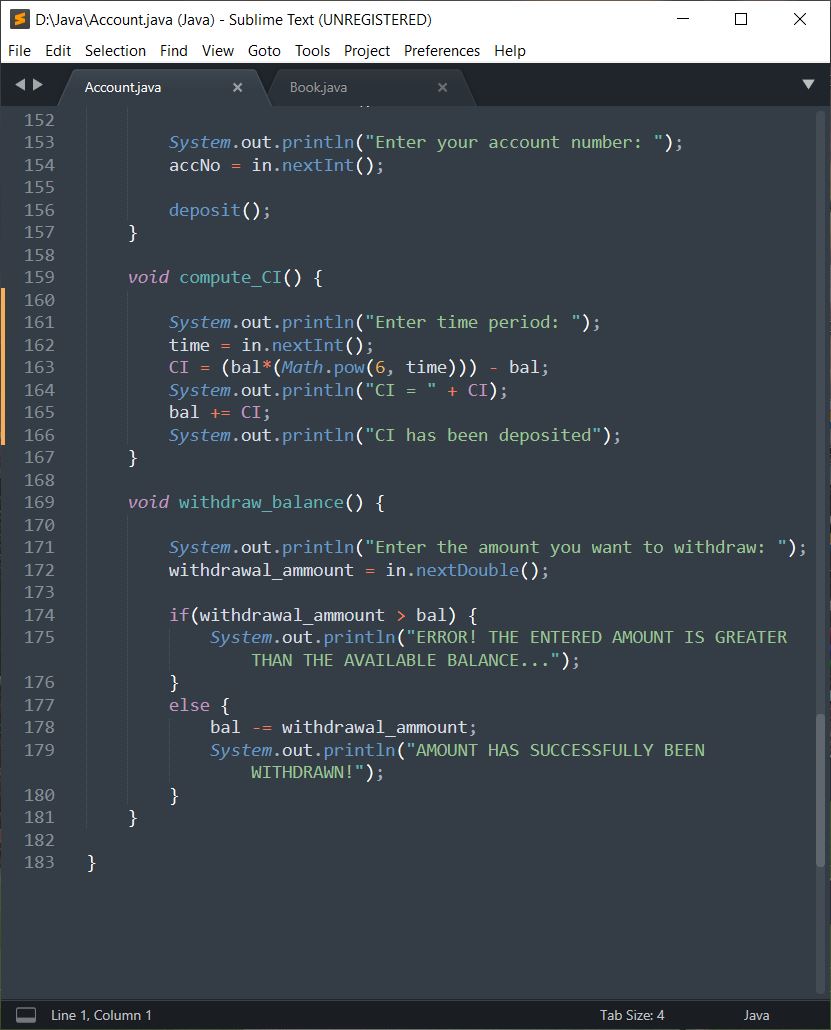
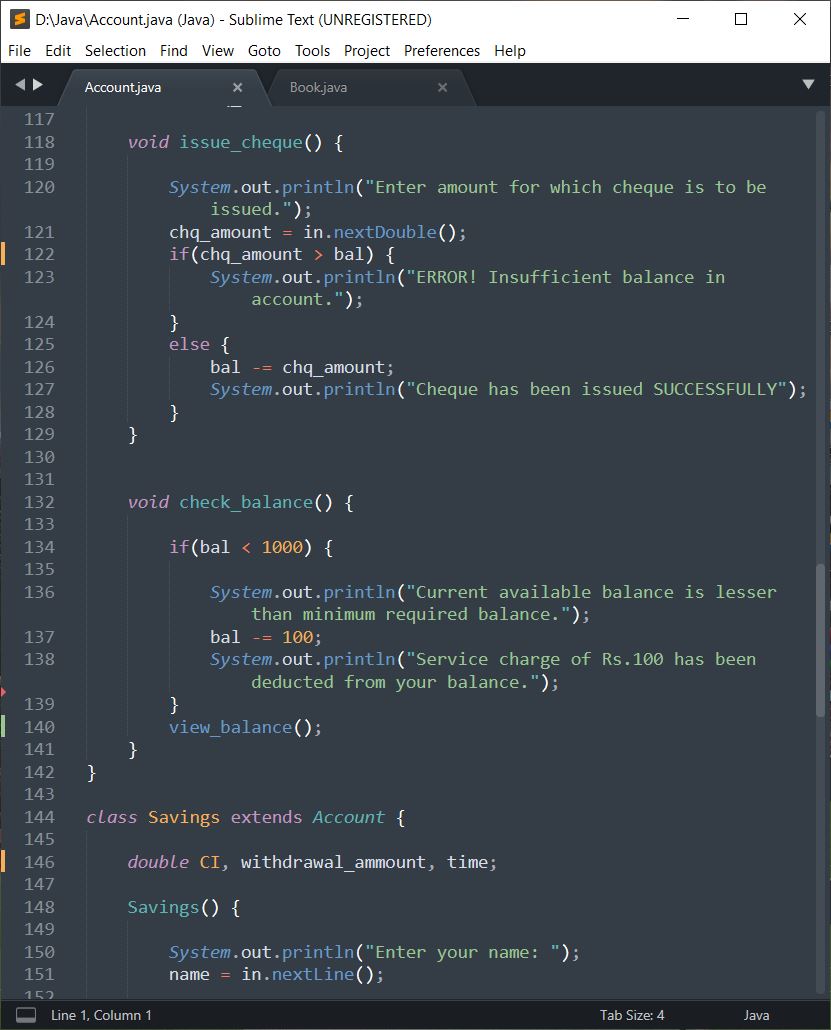
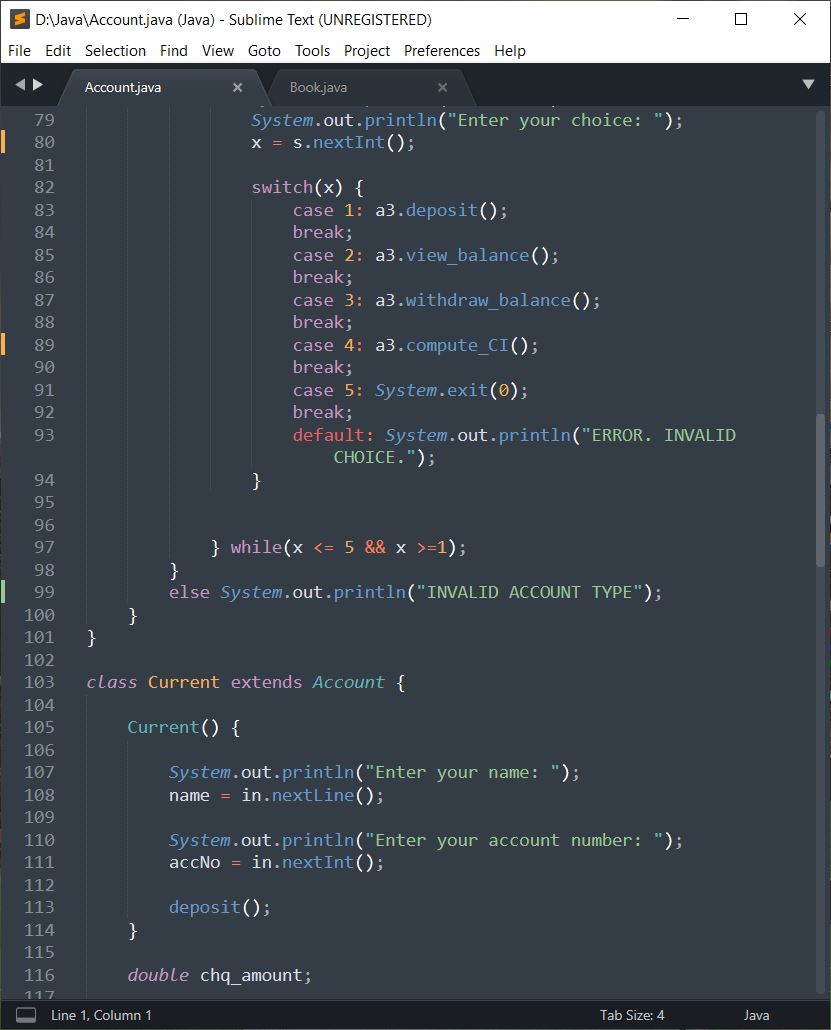
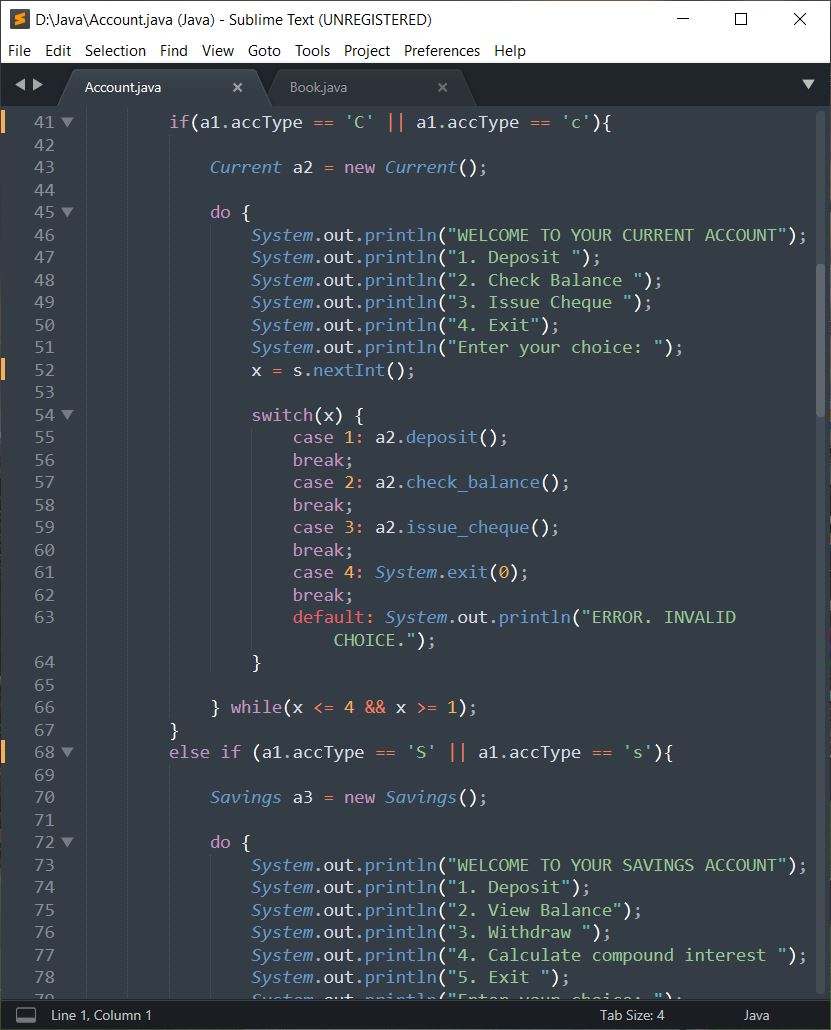
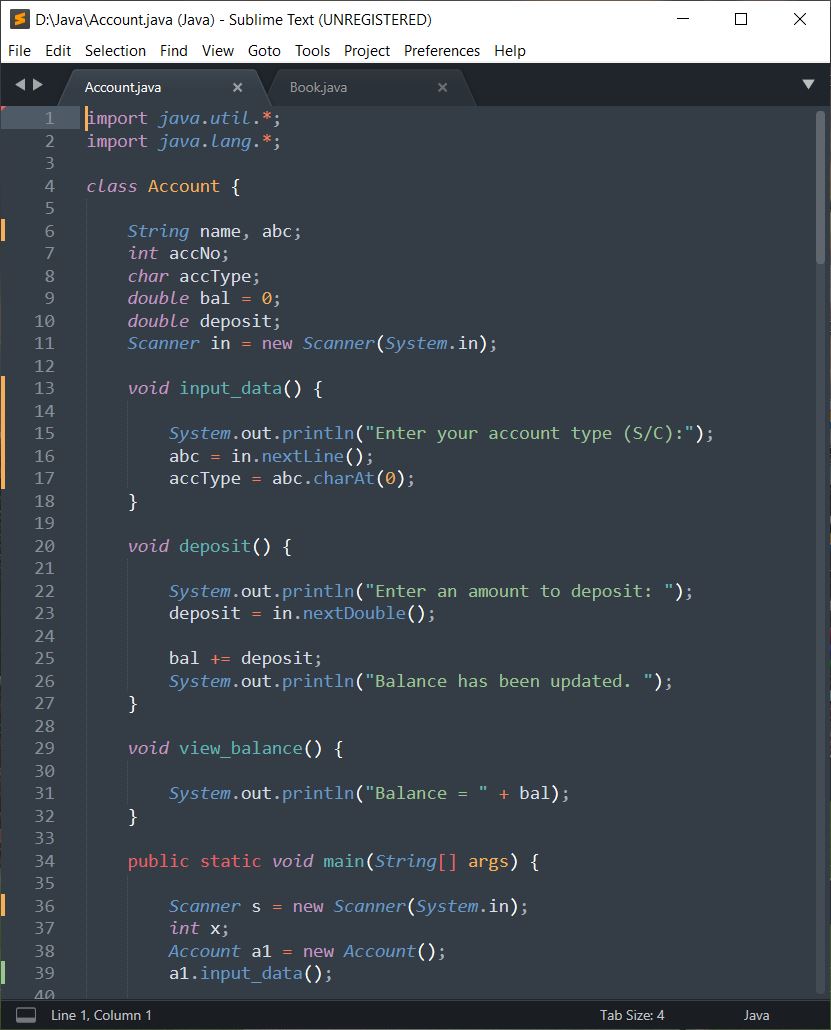
1. **Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of account. From this derive the classes Curr-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:**

**a) Accept deposit from customer and update the balance.**

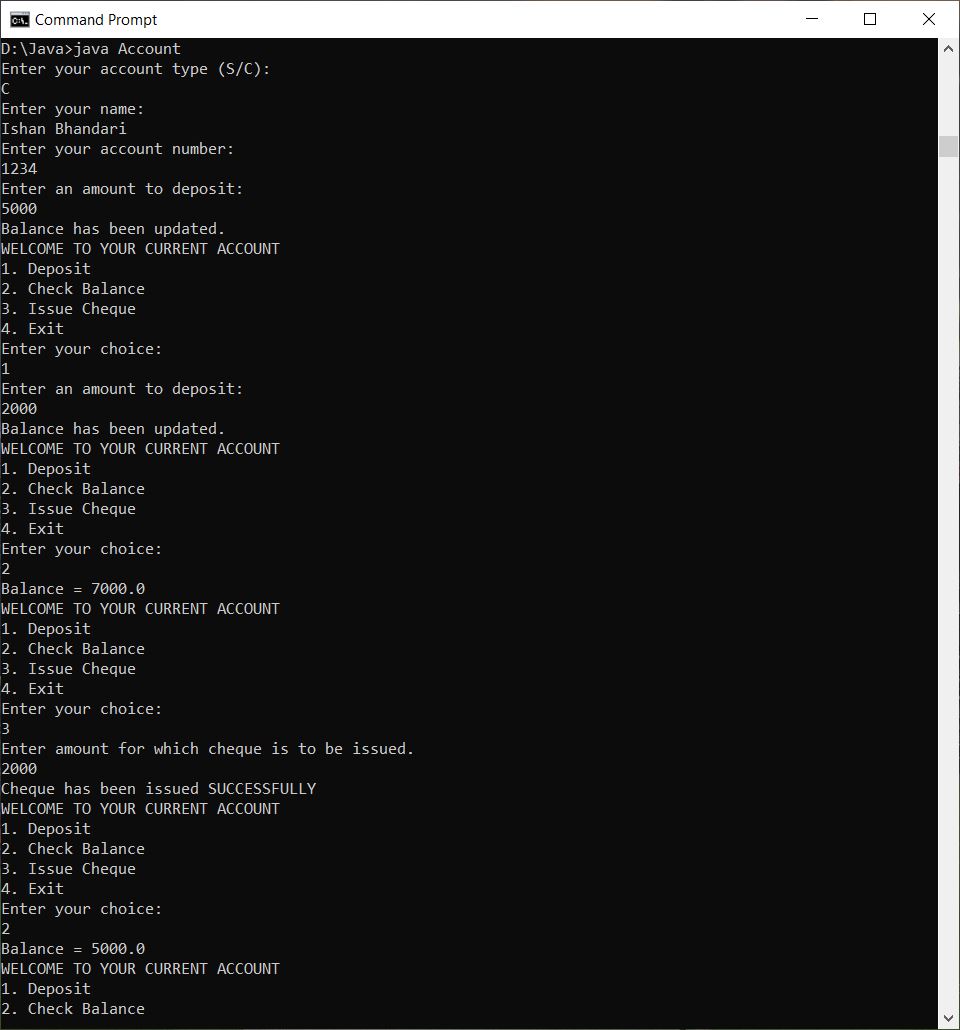
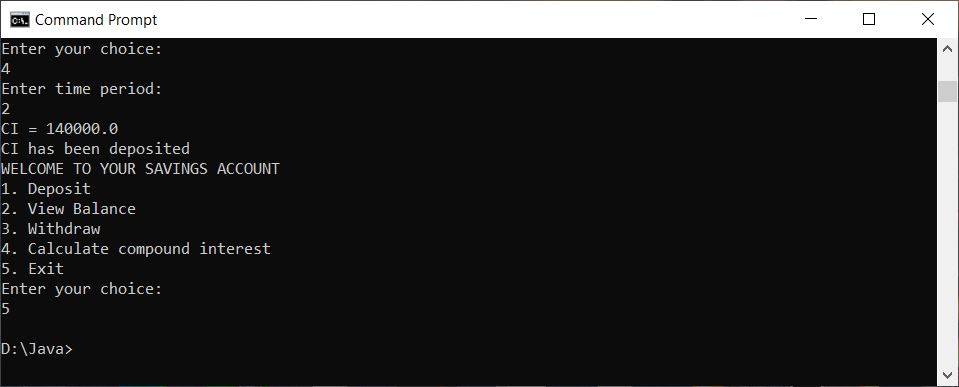
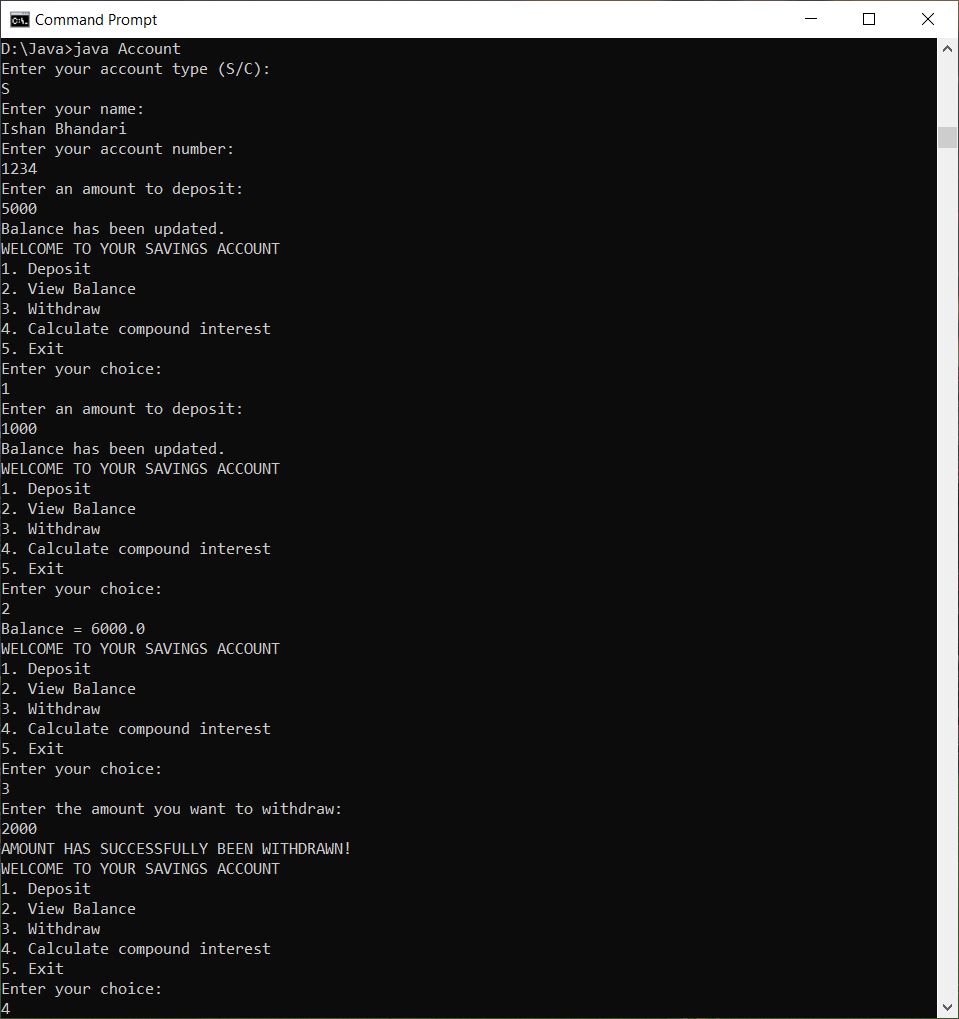
**b) Display the balance.**

**c) Compute and deposit interest**

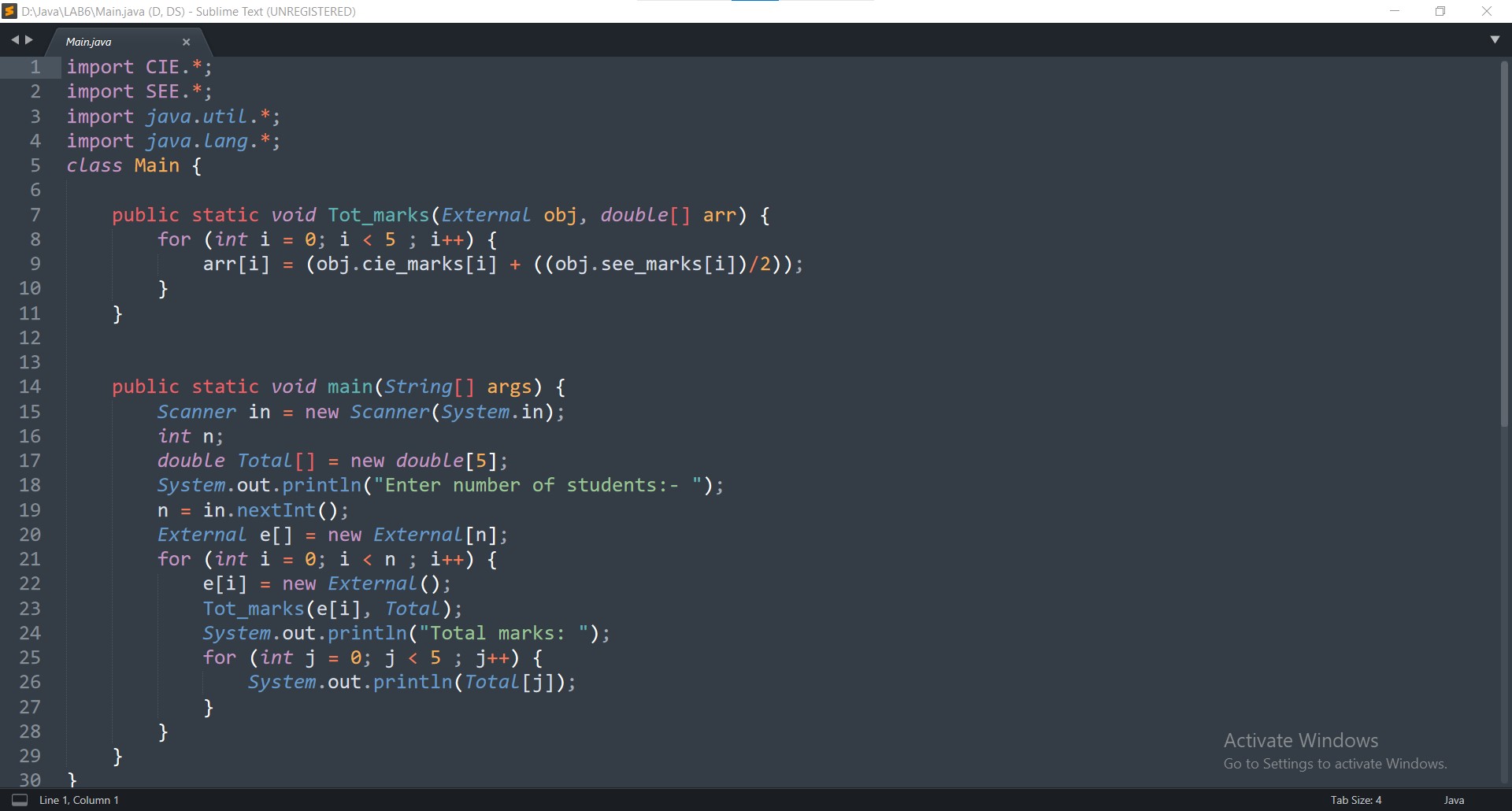
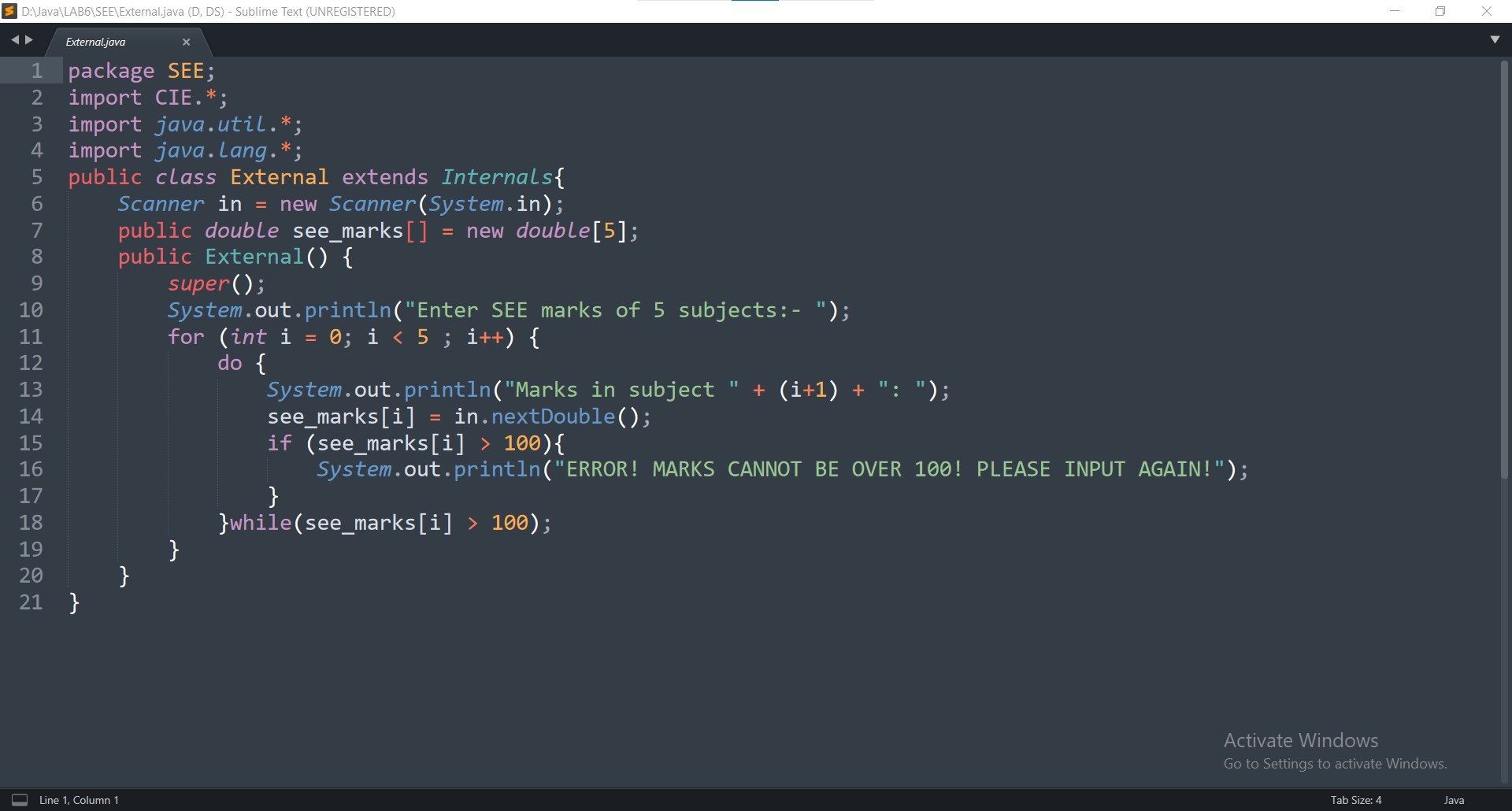
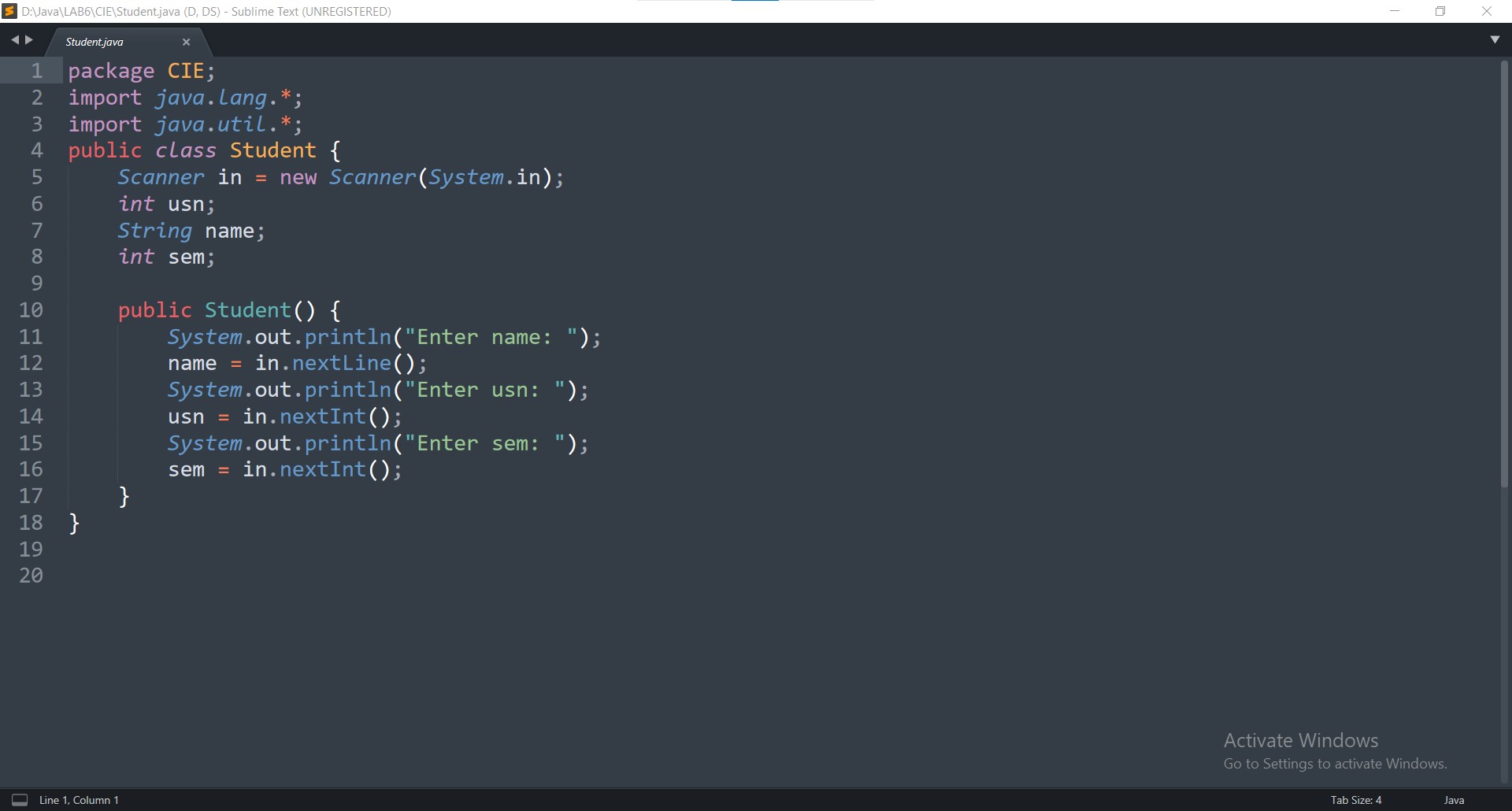
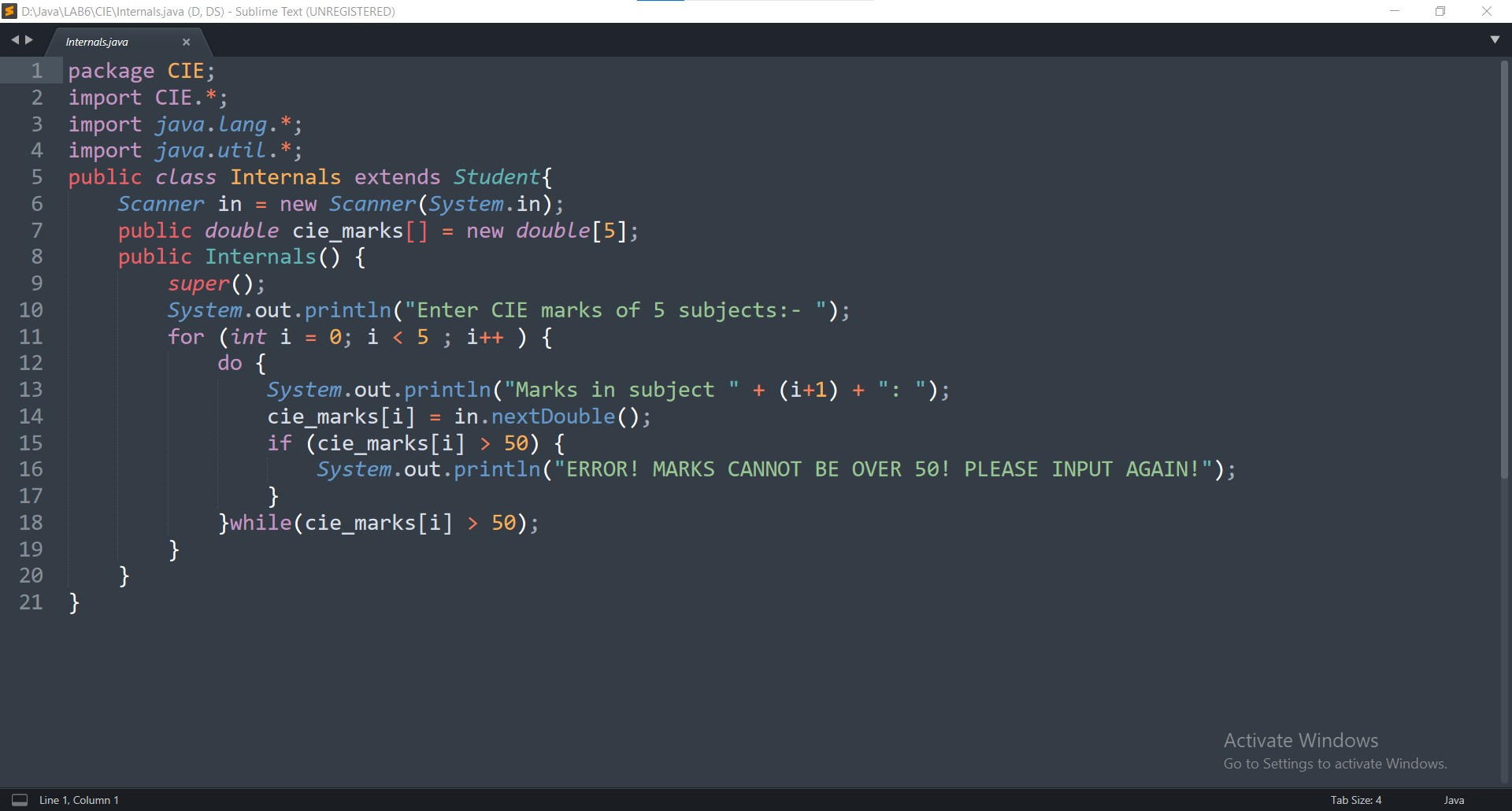
**d) Permit withdrawal and update the balance Check for the minimum balance, impose penalty if necessary and update the balance.**



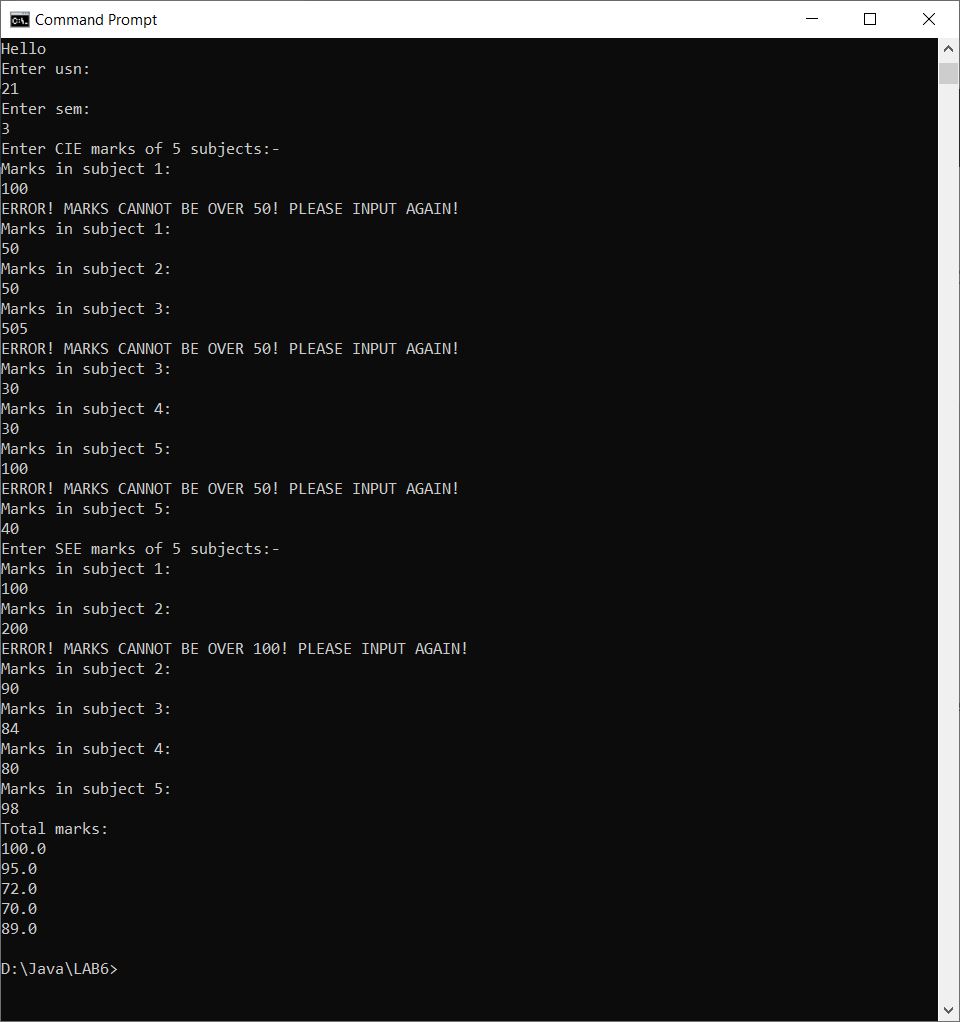
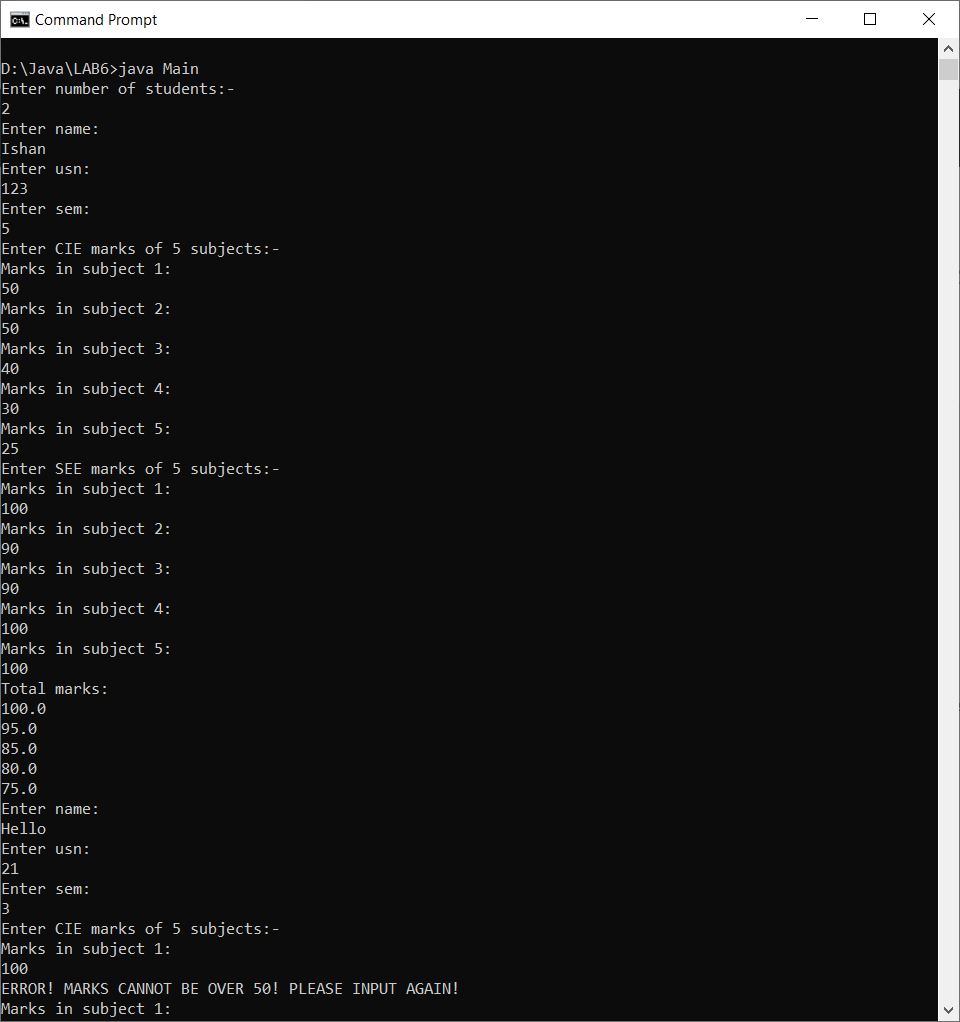
**Output:**



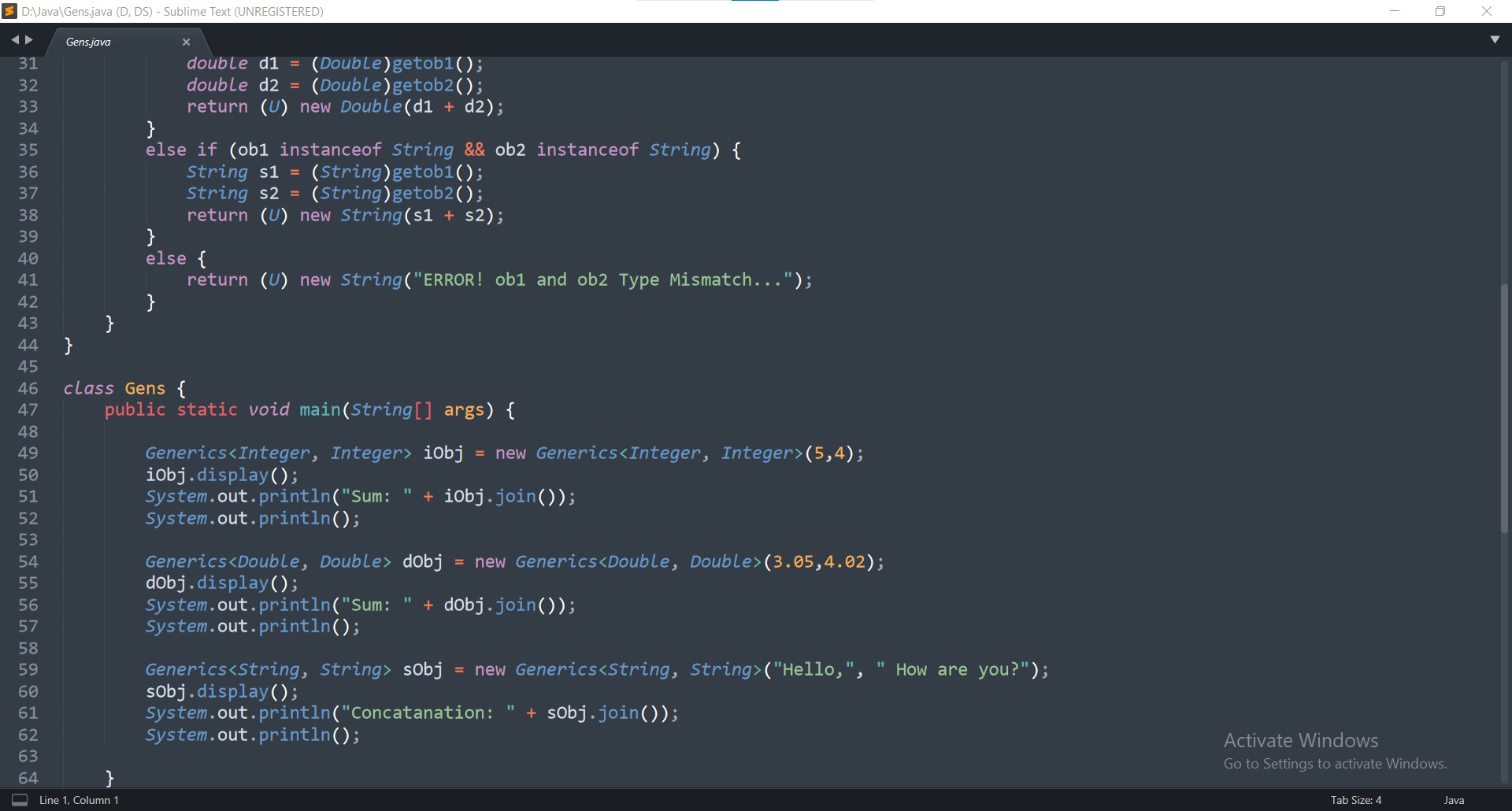
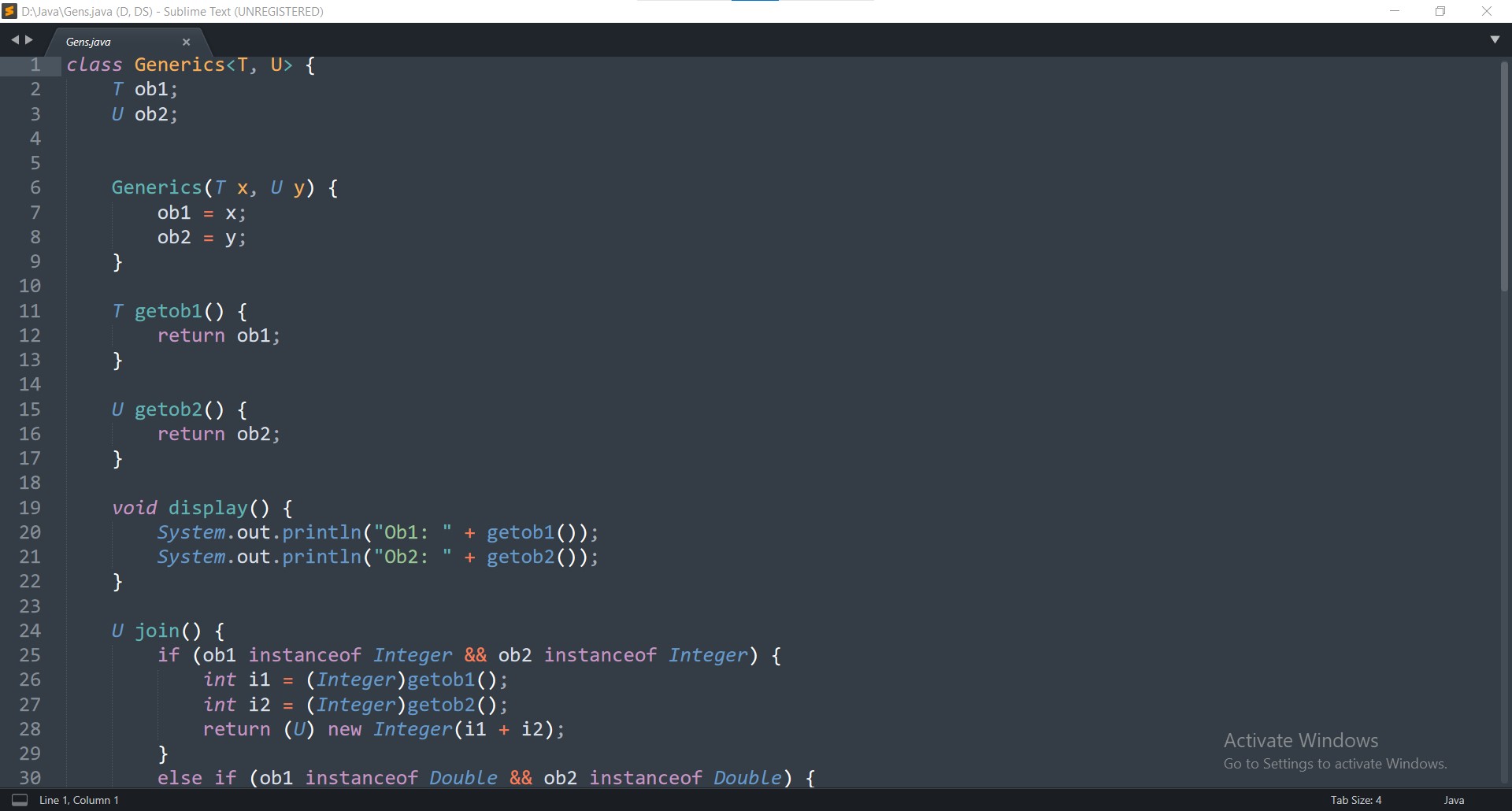
1. **Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class Internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of Student. This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses.**



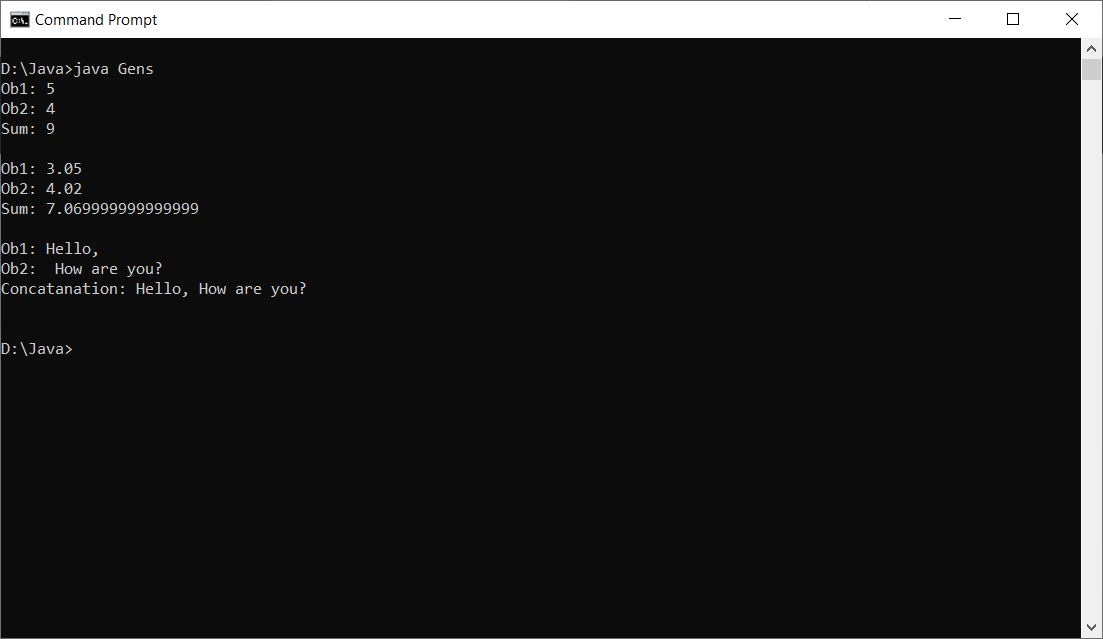
**Output:**



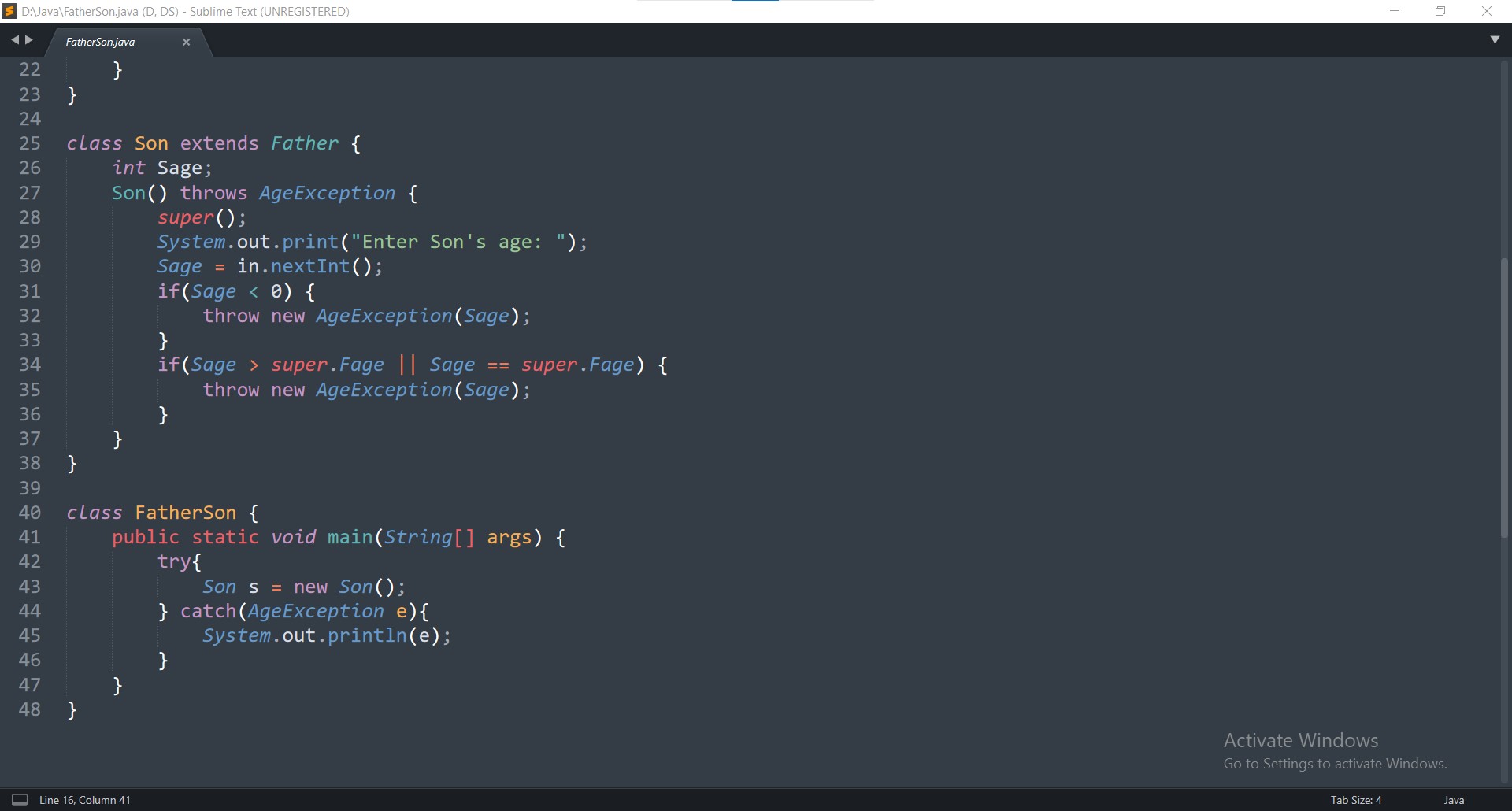
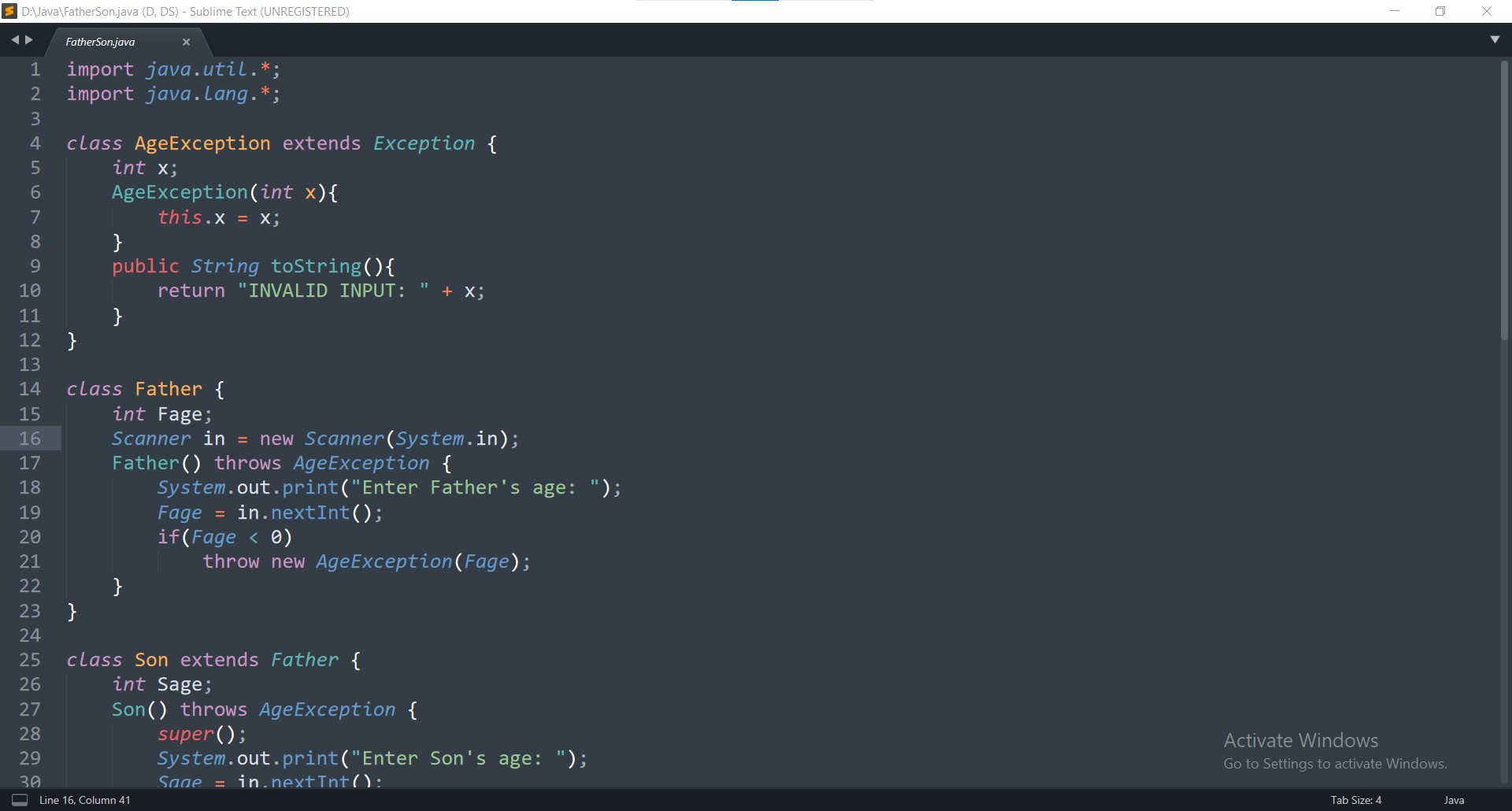
1. **Write a program to demonstrate generics with multiple object parameters.**



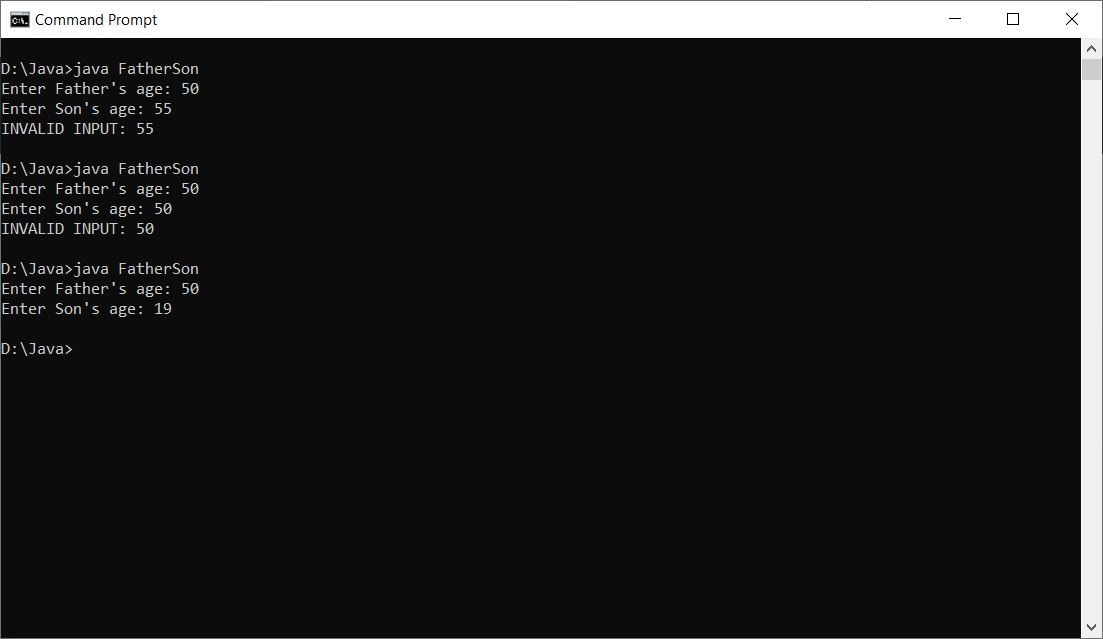
**Output:**



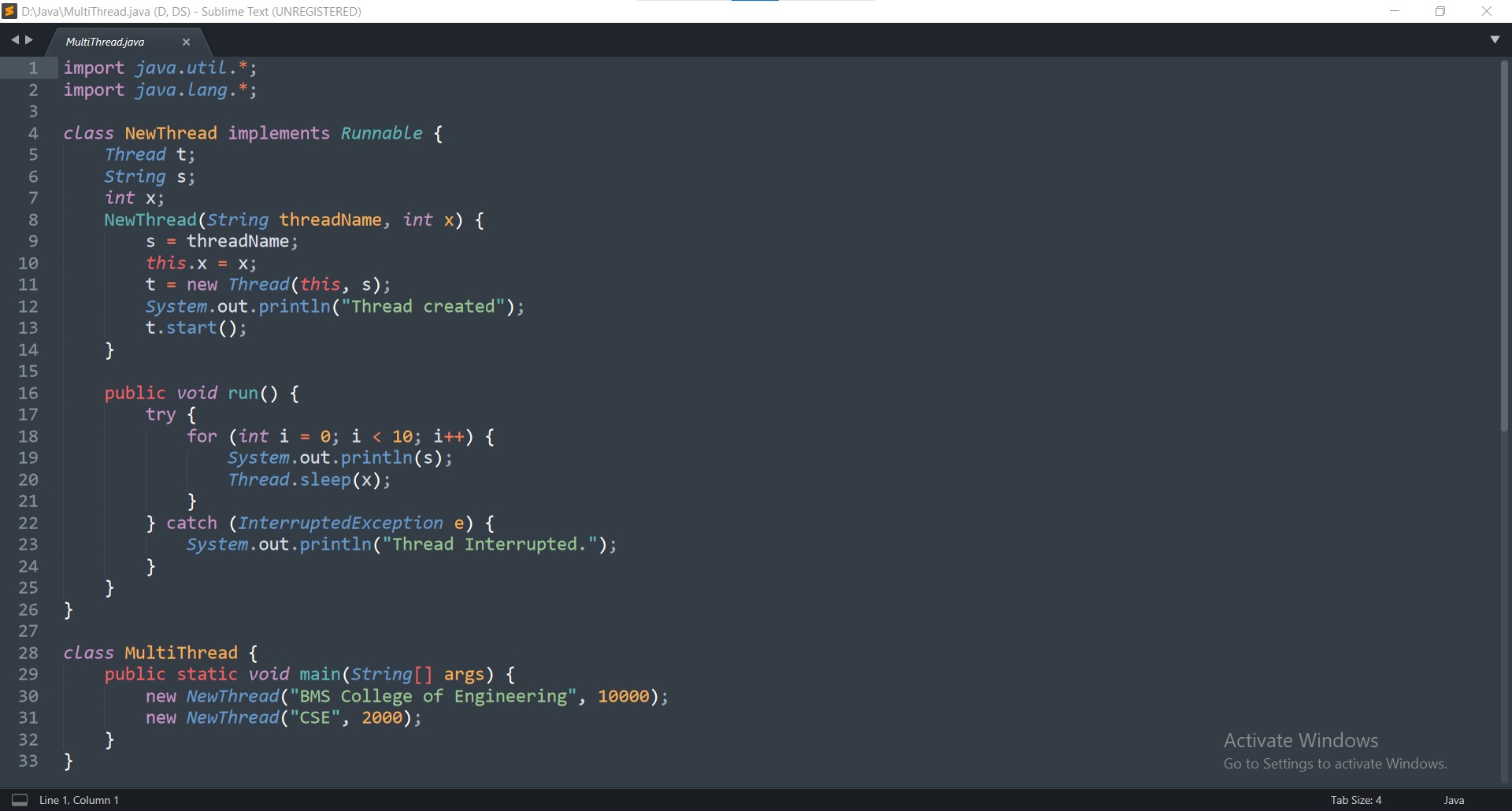
1. **Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age<0. In Son class, implement a constructor that cases both father and son’s age and throws an exception if son’s age is >=father’s age.**



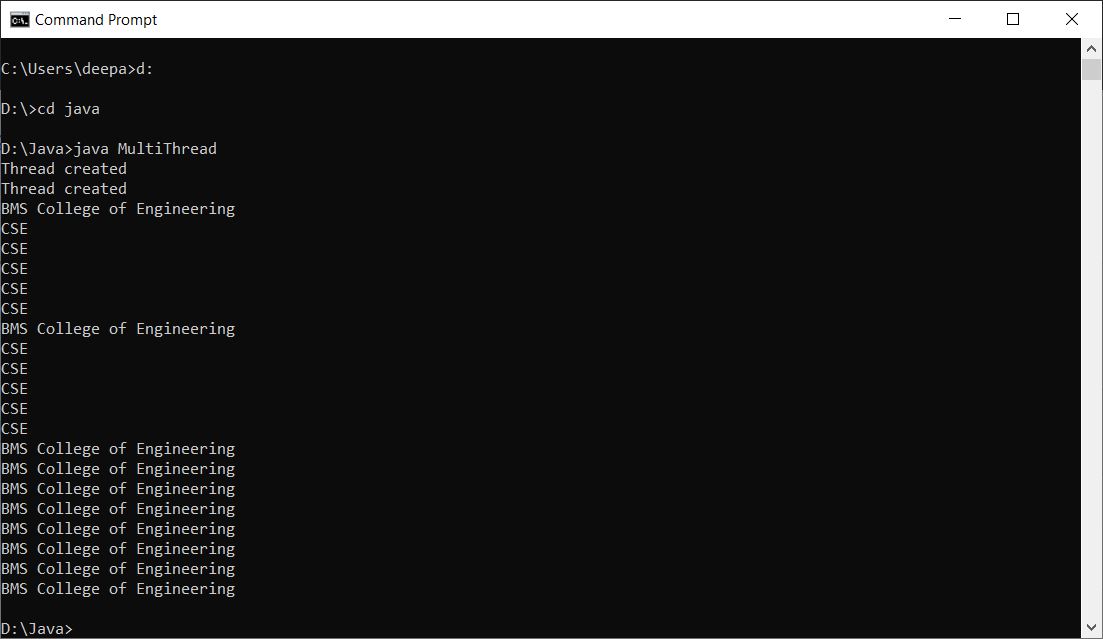
**Output:**



1. **Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.**



**Output:**

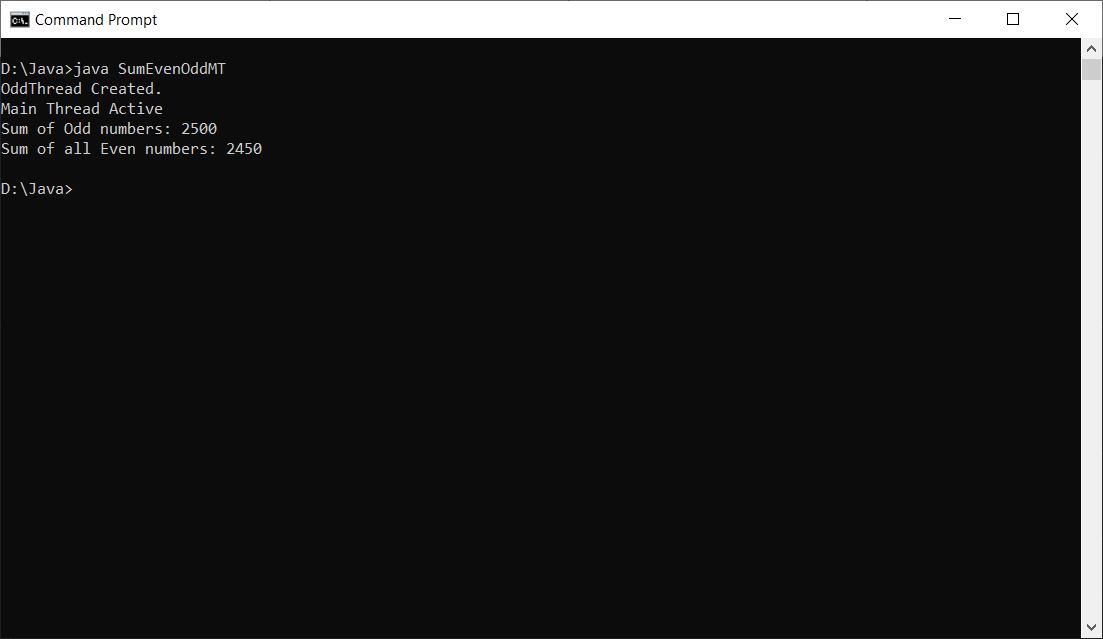


**10.1.**

**Write a program to create a thread and find the sum of odd numbers from 1 to 100 in this thread. Find the sum of even numbers for the same range in the main thread.**



**Output:**



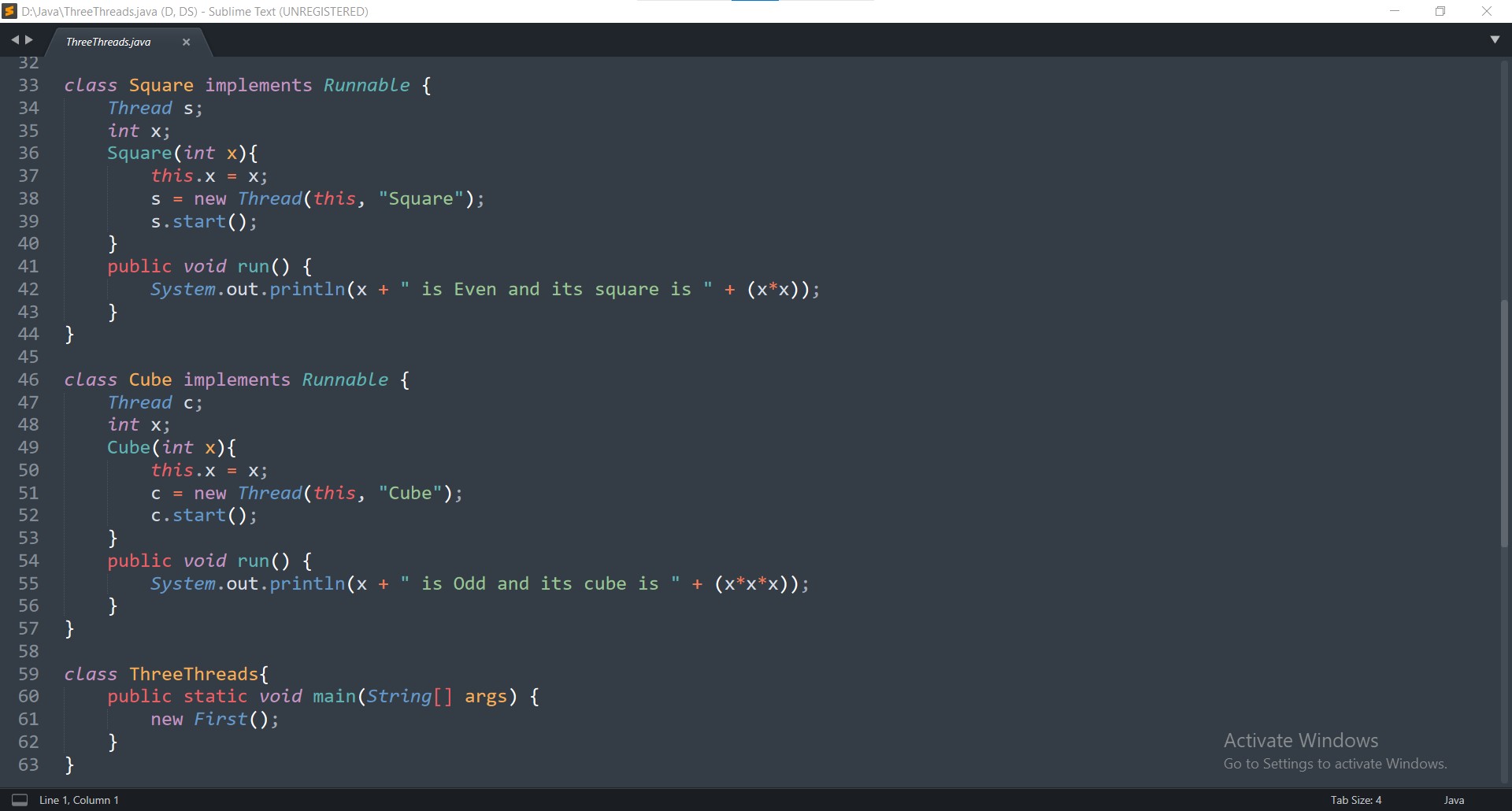
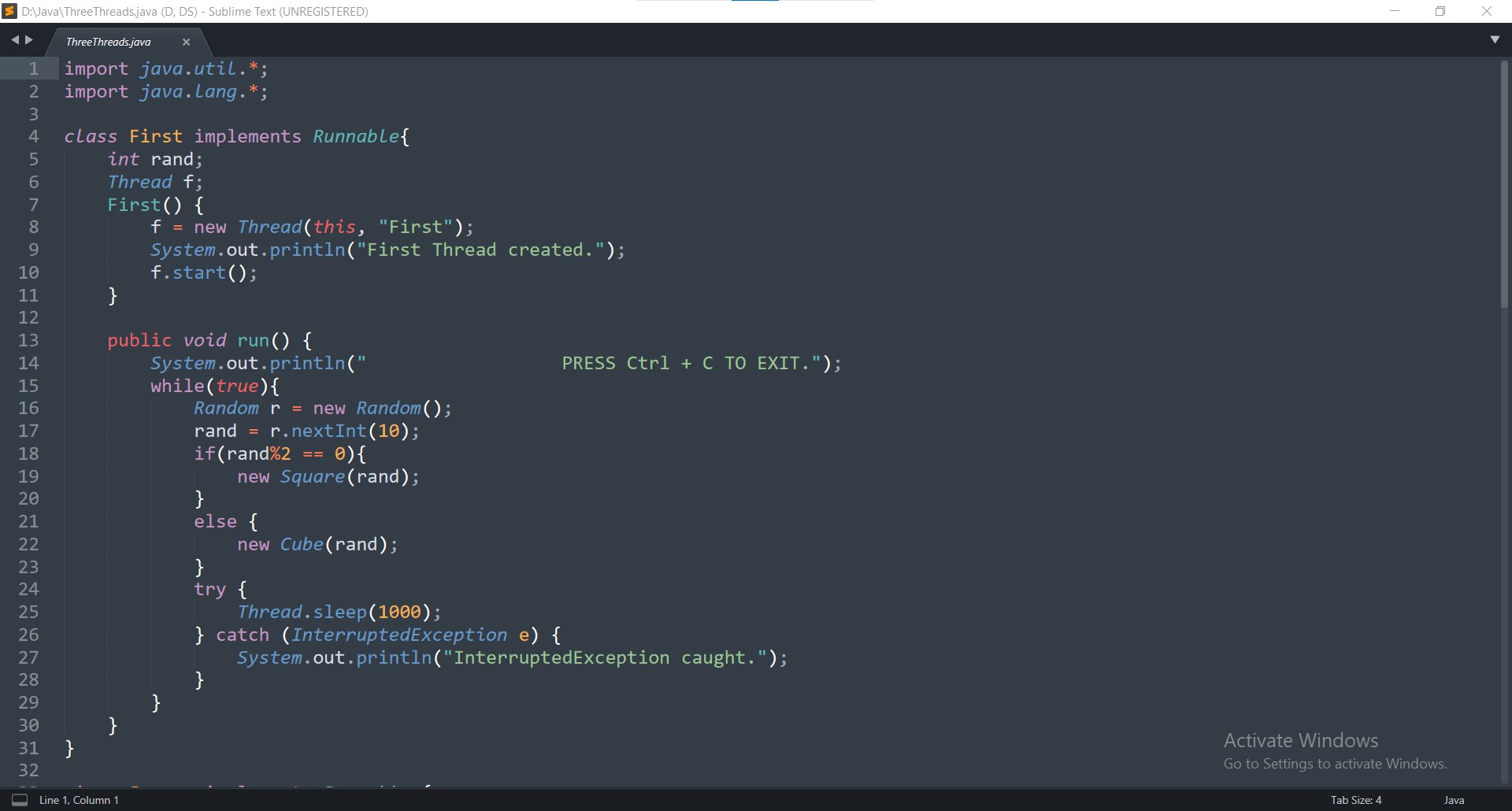
**10.2.**

**Develop a multithreaded Java program to create three threads. First thread generates**

**random integer for every second and if the value is even, second thread computes the**

**square of number and prints. If the value is odd, the third thread will print the value of**

**cube of number.**



**Output:**

