**Day 8 – Assignment**

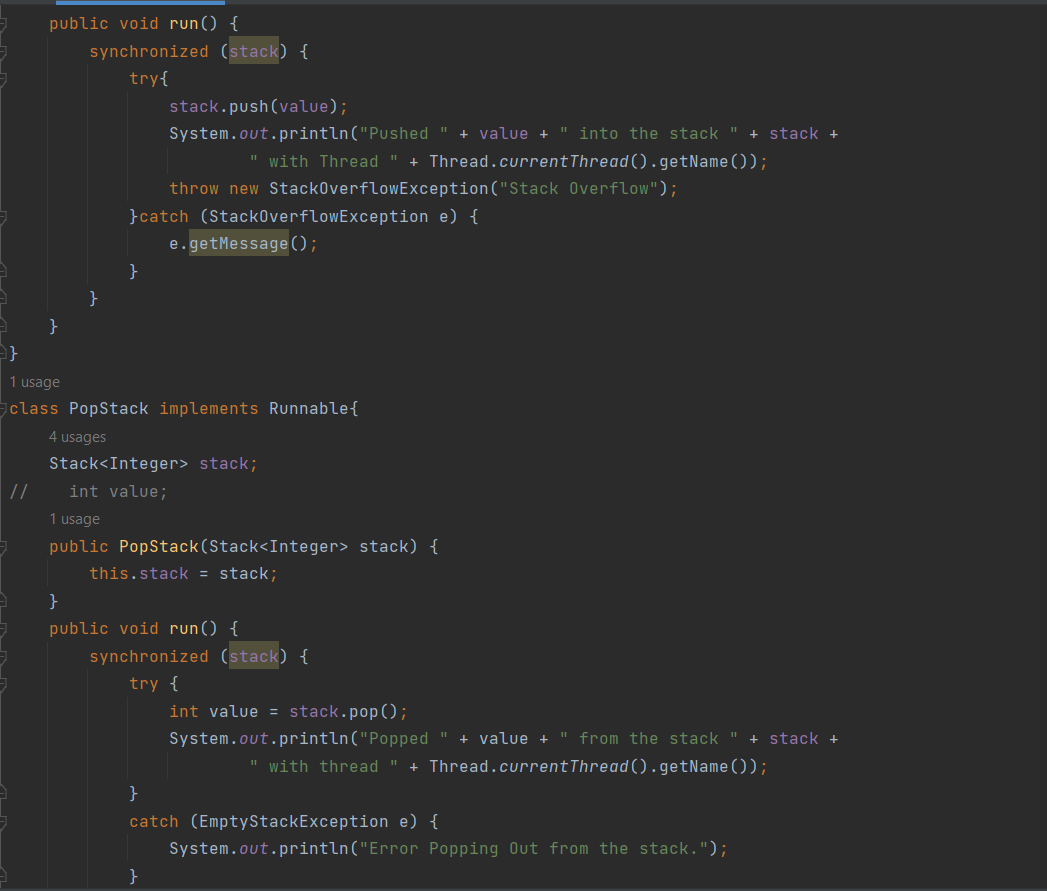
**Pratik K Kamble**

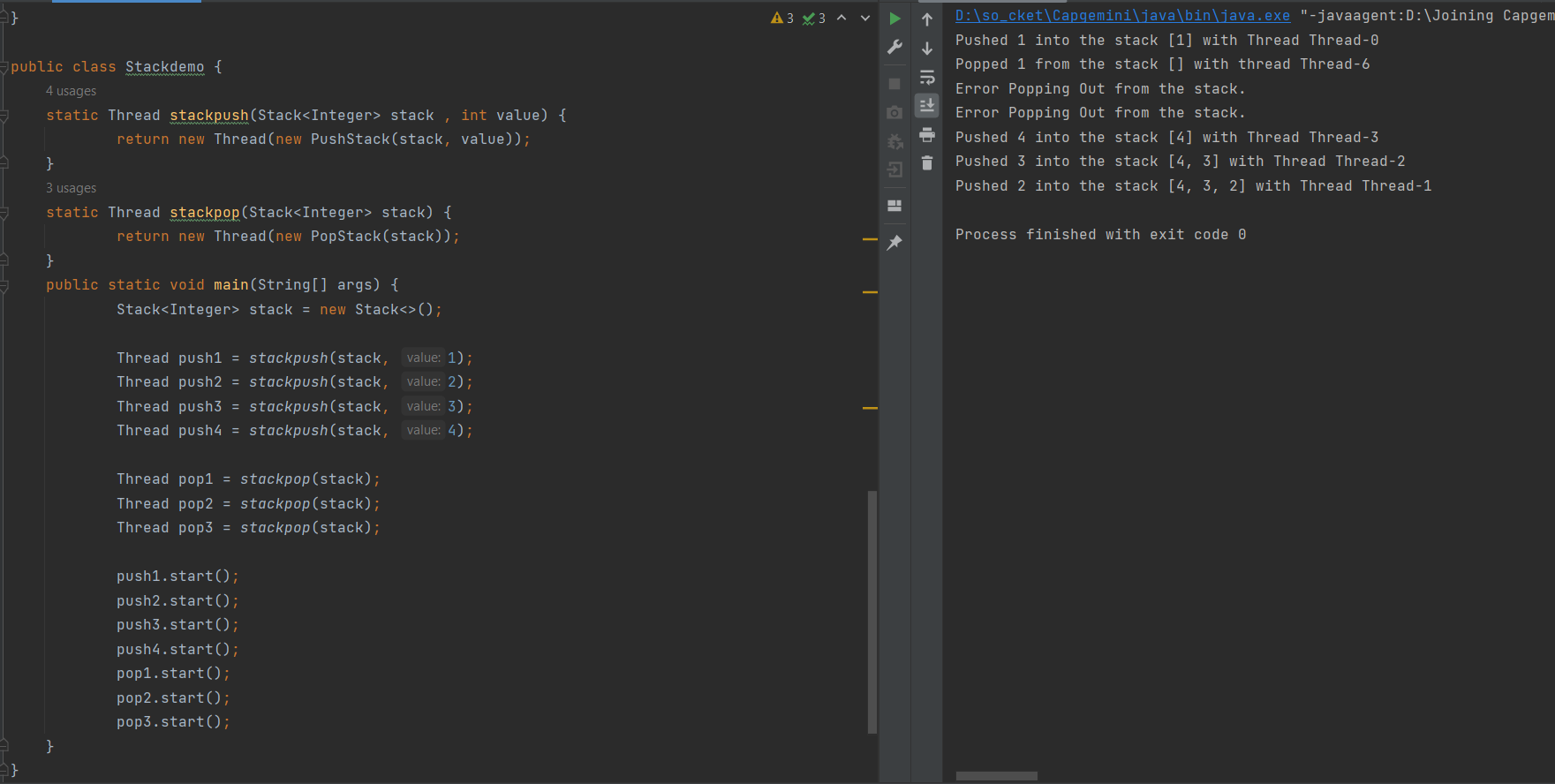
**Employee ID : 46263548**

**19 – 09 – 2022**

**Q1:** Write a Java program which handles Push operation and Pop operation on stack concurrently.

**Solution –**

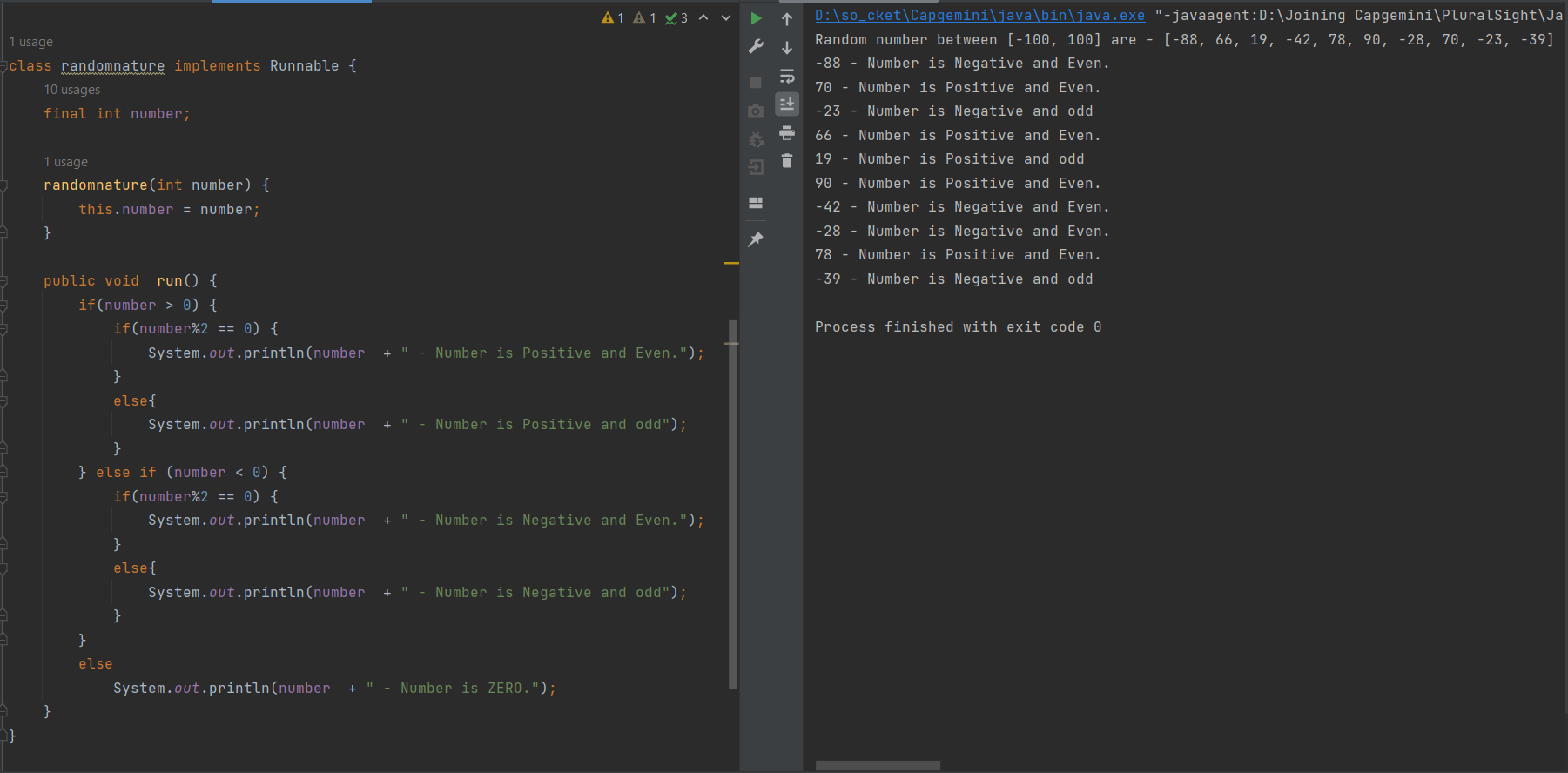
****

****

**Q2:** Write a Java program which first generates a set of random numbers and then determines negative, positive even, positive odd numbers concurrently.

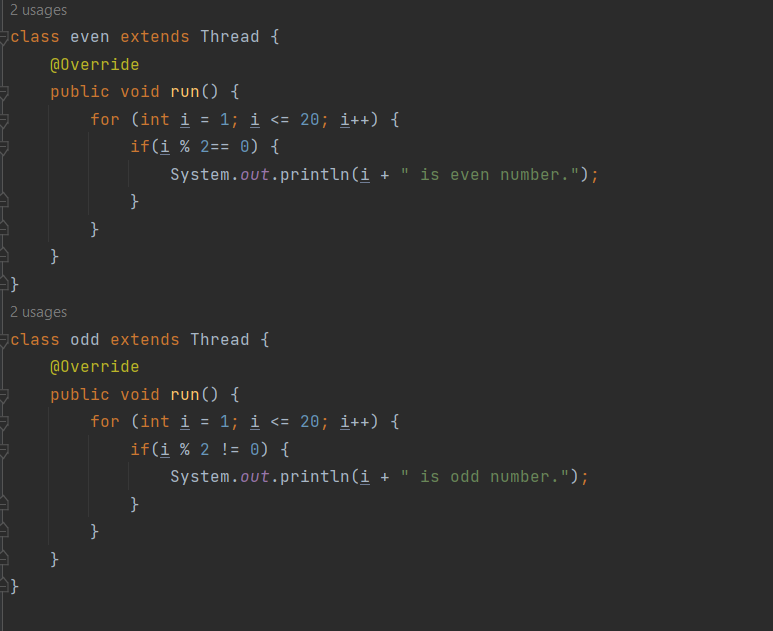
**Solution –**

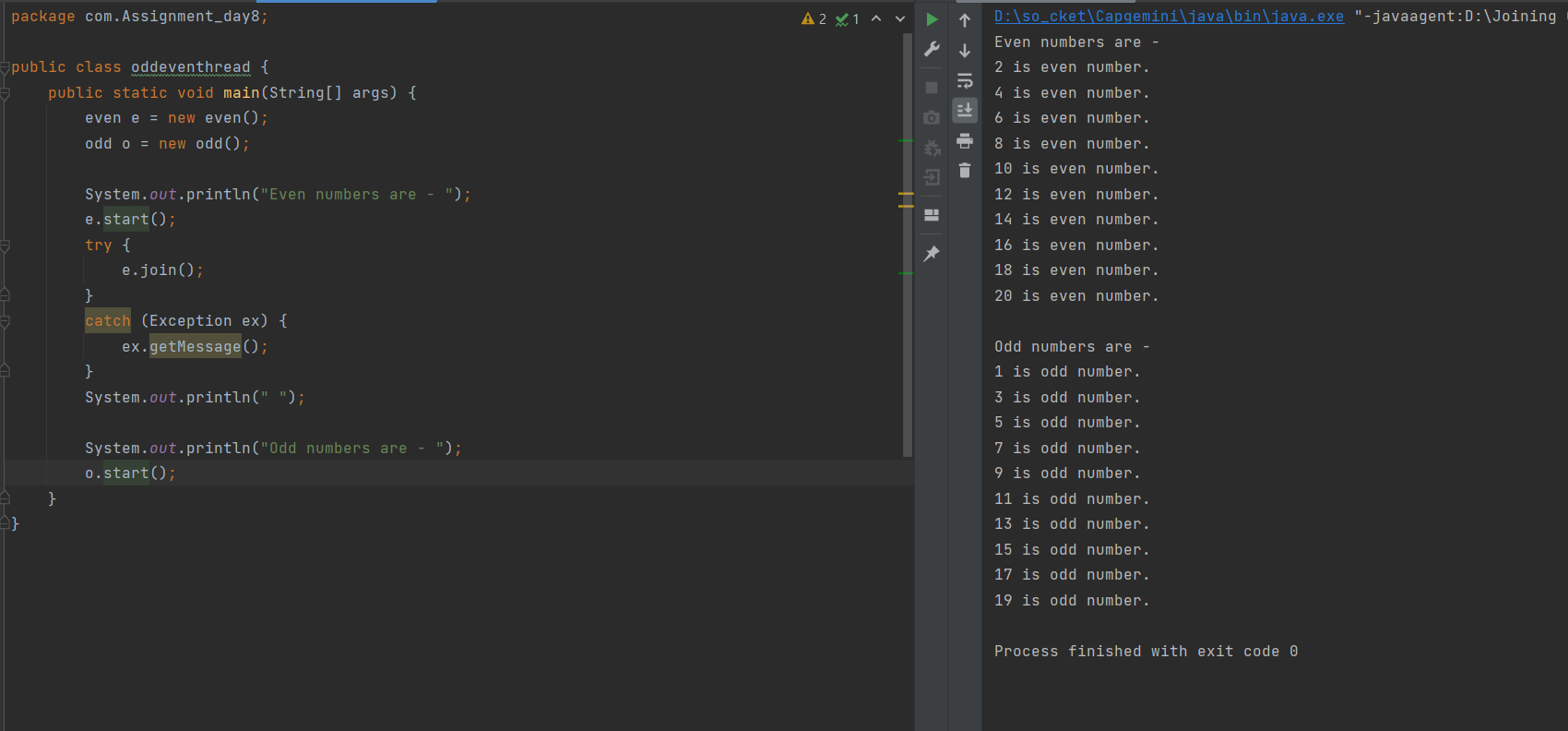
****

****

**Q3:** Create two threads, one thread to display all even numbers between 1 & 20, another to display odd numbers between 1 & 20. Note: Display all even numbers followed by odd numbers Hint: use join

**Solution –**

****

****

**Q4:** I am providing the below code snippet that you need to write in a simple java project. So follow below steps and then attempt to solve the problem statement.

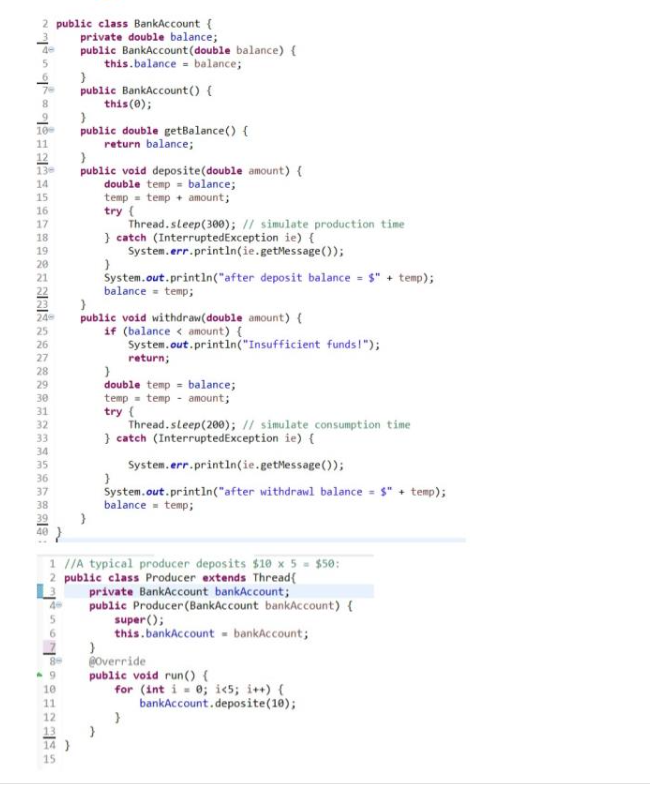
Steps

**Step1:** Create a core java project with the name as **AdvanceThreadDemo**

**Step2:** Create 4 classes separately in src directory. Code is as below.

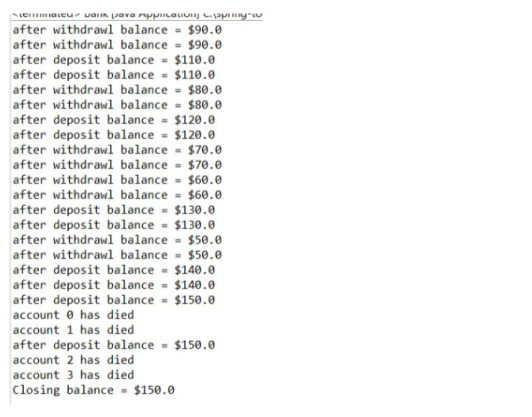
Classes

1. BankAccount
2. Producer extends Thread
3. Consumer extends Thread
4. Bank --------- with main method





Now run the main class and observe the output.



**Problem Statement:**

Notice that the “insufficient funds” message never appeared. That means both consumers successfully withdrew $10 five times each for a total of -$100. Similarly, both producers successfully deposited $10 five times each for a total of $100. The initial balance was $100, so the closing balance should have been:

**balance=$100+$100-$100 = $100** somehow, the bank lost $50.

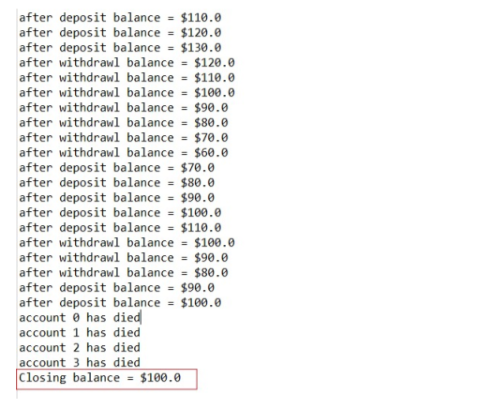
It seems pretty clear what went wrong. Look at the first three lines of output:

after withdrawl balance = $90.0 after withdrawl balance = $90.0

after deposit balance = $110.0

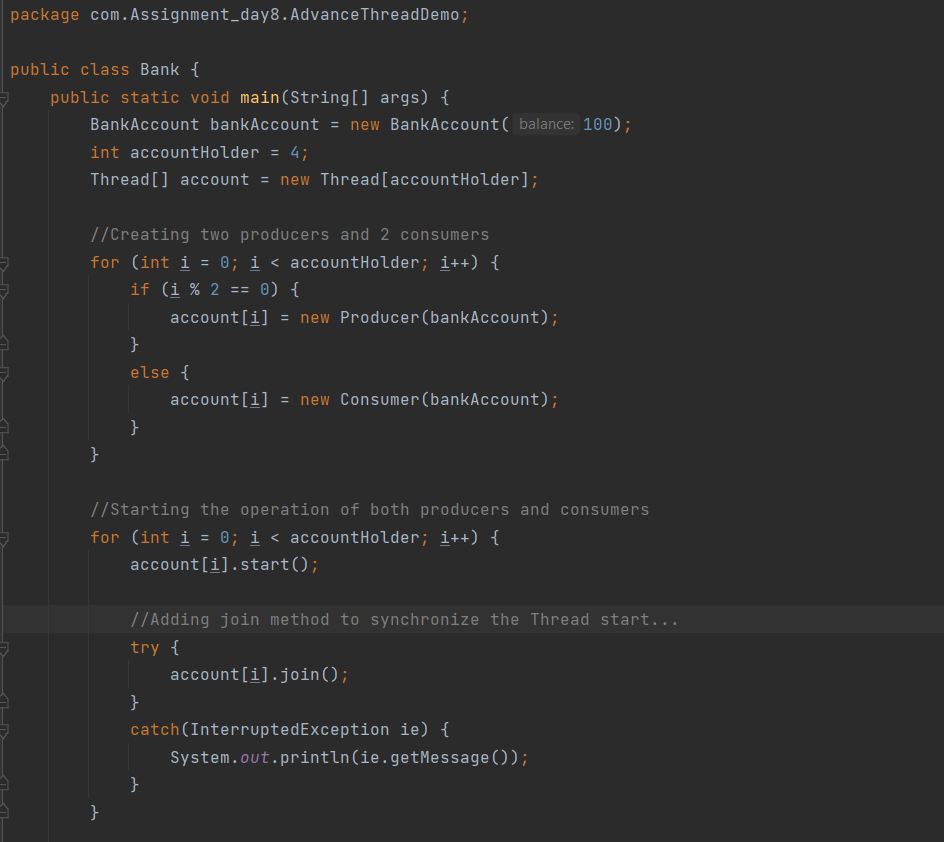
Apparently, when the first producer was in the middle of making a deposit, he was interrupted by a consumer who quicly withdrew $10, leaving the balance at $90. Unfortunately, the last time the producer checked, the balance was $100, so after adding an additional $10, the producer now believes the balance is $110. In other words, the bank lost track of the $10 withdrawn by the consumer.

Make the appropriate changes in your app so that you must get the output as below.

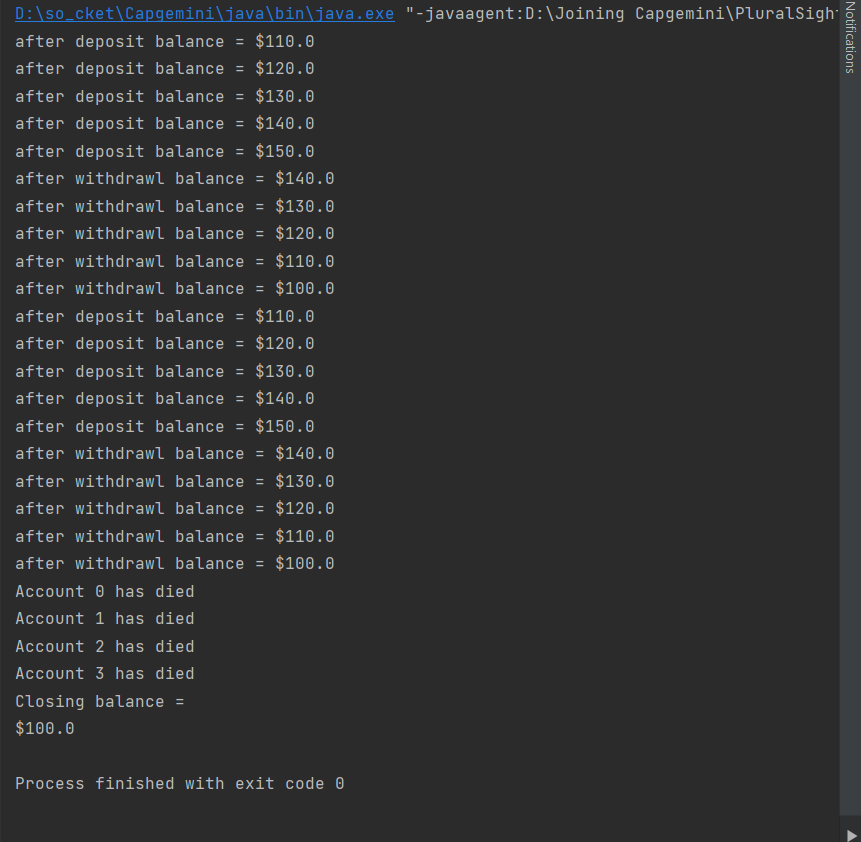


Note :  the sequence of deposit or withdraw can be changed. But finally the closing balance must be $100. Because same number of withdraw and deposits are happening with same amount in the system, so ideally closing balance must be initial balance only i.e. $100;

**Solution –**

****

****

****