



“Codigo”

Maria Eugenia Herrera Sanchez

Ingeniería en sistemas computacionales y diseño de software, Instituto Universitario de Yucatán

46220517343724M: Sistemas operativos

Perla Alejandra Landero Hreredia

10 de agosto del 2025

BankAccount.java

1- import java.util.concurrent.locks.Lock;

2 import java.util.concurrent.locks.ReentrantLock;

3

4- public class BankAccount {

5 private double balance;

6 private Lock lock;

7

8- public BankAccount() {

9 balance = 0.0;

10 lock = new ReentrantLock();

11 }

12

13- public void deposit(double amount) {

14 lock.lock();

15- try {

16 balance += amount;

17 System.out.println("Deposito: " + amount);

18 System.out.println("Saldo despues del deposito : " + balance);

19 }

20- finally {

21 lock.unlock() ;

22 }

23 }

Run

Share

Output

Clear

Deposito: 1000.0
Saldo despues del deposito : 1000.0
Intento de retiro: 1200.0
Saldo Insuficiente, retiro cancelado.
Deposito: 300.0
Saldo despues del deposito : 1300.0
Retiro: 150.0
Saldo despues del retiro: 1150.0

=== Code Execution Successful ===

BankAccount.java

25- public void withdraw(double amount) {

26 lock.lock();

27- try {

28- if (balance >= amount) {

29 balance -= amount;

30 System.out.println("Retiro: " + amount);

31 System.out.println("Saldo despues del retiro: " + balance);

32 }

33- else {

34 System.out.println("Intento de retiro: " + amount);

35 System.out.println("Saldo Insuficiente, retiro cancelado.");

36 }

37 }

38- finally {

39 lock.unlock();

40 }

41 }

42- public static void main(String[] args) {

43 BankAccount account = new BankAccount();

44

45 Thread depositThread1 = new Thread() -> account.deposit(1000));

46 Thread depositThread2 = new Thread() -> account.deposit(300));

47 Thread withdrawalThread1 = new Thread() -> account.withdraw(150));

48 Thread withdrawalThread2 = new Thread() -> account.withdraw(1200));

Run

Share

Output

Clear

Deposito: 1000.0
Saldo despues del deposito : 1000.0
Intento de retiro: 1200.0
Saldo Insuficiente, retiro cancelado.
Deposito: 300.0
Saldo despues del deposito : 1300.0
Retiro: 150.0
Saldo despues del retiro: 1150.0

=== Code Execution Successful ===

BankAccount.java

40 }

41 }

42- public static void main(String[] args) {

43 BankAccount account = new BankAccount();

44

45 Thread depositThread1 = new Thread() -> account.deposit(1000));

46 Thread depositThread2 = new Thread() -> account.deposit(300));

47 Thread withdrawalThread1 = new Thread() -> account.withdraw(150));

48 Thread withdrawalThread2 = new Thread() -> account.withdraw(1200));

49

50 depositThread1.start();

51 depositThread2.start();

52 withdrawalThread1.start();

53 withdrawalThread2.start();

54 }

55 }

56 }

57 }

58 }

59 }

Run

Share

Output

Clear

Deposito: 1000.0
Saldo despues del deposito : 1000.0
Intento de retiro: 1200.0
Saldo Insuficiente, retiro cancelado.
Deposito: 300.0
Saldo despues del deposito : 1300.0
Retiro: 150.0
Saldo despues del retiro: 1150.0

=== Code Execution Successful ===