# Synchronization

Synchronization is a mechanism that allows to access shared resources only one at a time.

How to Synchronize: There are two ways to synchronize.

* Synchronize a method
* Synchronize block of code

Synchronizing a method:

Synchronized ReturnType MethodName(arg1, arg2…)

{

//Code

}

Synchronizing a block of code:

ReturnType MethodName(arg1,arg2…)

{

//Code

Synchronized (Object)

{

//Code

}

}

Note: wait and notify methods must be used in the synchronization area otherwise illegalMonitorStateException will be thrown.

Refer code snippet: SynchronizationExample

|  |
| --- |
| **//Program to demonstrate Synchronization, wait and notify methods**  **package** synchronization;  **import** java.lang.Thread;  **public** **class** SynchronizationExample **extends** Thread  {  **public** **static** **void** main(String[] args)  {  Bank x = **new** Bank();  customer1 c1 = **new** customer1(x);  c1.start();    customer2 c2 = **new** customer2(x);  c2.start();  }  }  **class** Bank  {  **float** balance = 10000.00f;    **synchronized** **public** **void** withdraw(**int** amount) **throws** InterruptedException  {  **if** (balance < amount)  {  System.***out***.println("Wait - No Suffienct Balance to Withdraw.");  wait();  }  balance = balance - amount;  System.***out***.println("Amount Withdraw - "+amount);  }  **synchronized** **public** **void** deposit(**int** amount)  {  balance = balance + amount;  System.***out***.println("Amount Deposited - "+amount);  notify();    }  **synchronized** **public** **void** balance()  {  System.***out***.println("Account Balance - "+balance);  }  }  **class** customer1 **extends** Thread  {  Bank obj = **null**;  customer1(Bank x)  {  obj = x;  }    **public** **void** run()  {  **try** {  obj.withdraw(15000);  } **catch** (InterruptedException e) {  // **TODO** Auto-generated catch block  e.printStackTrace();  }  obj.balance();  }    }  **class** customer2 **extends** Thread  {  Bank obj = **null**;  customer2(Bank x)  {  obj = x;  }  **public** **void** run()  {  obj.deposit(15000);  obj.balance();  }  }  //OUTPUT  Wait - No Suffienct Balance to Withdraw.  Amount Deposited - 15000  Account Balance - 25000.0  Amount Withdraw - 15000  Account Balance - 10000.0 |