Class Store {

Private boolean orderReady = false;

Public synchronized void waitForOrder() {

While (!orderReady) {

Try {

System.out.println(“Khách hàng: Tôi đang chờ đơn...”);

Wait(); // khách hàng chờ nhân viên thông báo

} catch (InterruptedException e) {

e.printStackTrace();

}

}

System.out.println(“Khách hàng: Tôi đã nhận được đơn hàng!”);

}

Public synchronized void prepareOrder() {

System.out.println(“Nhân viên: Đang chuẩn bị đơn hàng...”);

Try {

Thread.sleep(3000); // giả lập chuẩn bị mất 3 giây

} catch (InterruptedException e) {

e.printStackTrace();

}

orderReady = true;

notify(); // thông báo cho khách hàng

System.out.println(“Nhân viên: Đơn hàng đã sẵn sàng!”);

}

}

Public class WaitNotifyOrderExample {

Public static void main(String[] args) {

Store store = new Store();

Thread customer = new Thread(store::waitForOrder);

Thread staff = new Thread(store::prepareOrder);

Customer.start();

Try { Thread.sleep(1000); } catch (InterruptedException e) {}

Staff.start();

}

}