**NAME : MANGESH A. GHADWAJE**

**ROLL NO:24**

**BATCH : B2**

**COURSE: OOPs PRACTICAL**

**Assginment No. 11**

**Problem Statment : Write a program to create multiple threads and demonstrate how two threads communicate with each other.**

**CODE:**

import java.util.concurrent.ArrayBlockingQueue;

import java.util.concurrent.BlockingQueue;

class Producer extends Thread {

private BlockingQueue<Integer> queue;

public Producer(BlockingQueue<Integer> queue) {

this.queue = queue;

}

public void run() {

for (int i = 0; i < 5; i++) {

try {

System.out.println("Producing " + i);

queue.put(i);

Thread.sleep(1000);

} catch (InterruptedException e) {

e.printStackTrace();

}

}

}

}

class Consumer extends Thread {

private BlockingQueue<Integer> queue;

public Consumer(BlockingQueue<Integer> queue) {

this.queue = queue;

}

public void run() {

try {

while (true) {

int item = queue.take();

System.out.println("Consuming " + item);

Thread.sleep(2000);

}

} catch (InterruptedException e) {

e.printStackTrace();

}

}

}

public class Main {

public static void main(String[] args) {

BlockingQueue<Integer> sharedQueue = new ArrayBlockingQueue<>(10);

Producer producer = new Producer(sharedQueue);

Consumer consumer = new Consumer(sharedQueue);

producer.start();

consumer.start();

try {

producer.join();

sharedQueue.put(null); // Signal the consumer to stop

consumer.join();

} catch (InterruptedException e) {

e.printStackTrace();

}

System.out.println("Program terminated");

}

}

**OUTPUT:**

Producing 0

Consuming 0

Producing 1

Producing 2

Consuming 1

Producing 3

Consuming 2

Producing 4

Consuming 3

Consuming 4

Program terminated