

Industrial strength Java/JEE Career Companion to open more doors


[Home](#)
[Java FAQs](#)
[600+ Java Q&As](#)
[Career](#)
[Tutorials](#)
[Member](#)
[Why?](#)
[Can u Debug?](#)
[Java 8 ready?](#)
[Top X](#)
[Productivity Tools](#)
[Judging Experience?](#)

[Home](#) › [Interview](#) › [Core Java Interview Q&A](#) › [Multithreading](#) › Producer and Consumer Java Multi-threading code

# Producer and Consumer Java Multi-threading code

Posted on [June 15, 2015](#) by [Arulkumaran Kumaraswamipillai](#)

0  
Like  
Share

Tweet

0  
G+1  
Share

The following code is used for [demonstrating thread-safety issues on eclipse](#).

## Step 1: ProducerConsumerTest

```
1 package com.debug.multithread;
2
3 public class ProducerConsumerTest {
4
5     public static void main(String[] args) throw
6         ProducerConsumer pc = new ProducerConsum
7
8         //spawn a new producer thread and start
9         Thread producer = new Thread(new Produce
```

[9 tips to earn more](#) | [What can u do to go places?](#) | **945+** paid members. [LinkedIn Group](#). [Reviews](#)

**600+ Full Stack Java/JEE Interview Q&As ♥Free ♦FAQs**

[open all](#) | [close all](#)

[Ice Breaker Interview](#)

[Core Java Interview C](#)

[Java Overview \(4\)](#)

[Data types \(6\)](#)

[constructors-methc](#)

[Reserved Key Wor](#)

[Classes \(3\)](#)

[Objects \(8\)](#)

[OOP \(10\)](#)

[GC \(2\)](#)

[Generics \(5\)](#)

[FP \(8\)](#)

[IO \(7\)](#)

```

12     producer.start();
13
14     Thread.sleep(10); //main thread sleeps for 10ms
15     //spawn a new consumer thread and start it
16     Thread consumer = new Thread(new Consumer());
17     consumer.start();
18
19 }
20
21 }

```

### Step 2: Producer thread.

```

1 package com.debug.multithread;
2
3 public class ProducerThread implements Runnable {
4     private ProducerConsumer pc;
5
6     public ProducerThread(ProducerConsumer pc) {
7         this.pc = pc;
8     }
9
10    @Override
11    public void run() {
12        pc.produce();
13    }
14
15 }
16 }

```

### Step 3: Consumer thread.

```

1 package com.debug.multithread;
2
3 public class ConsumerThread implements Runnable {
4     private ProducerConsumer pc;
5
6     public ConsumerThread(ProducerConsumer pc) {
7         this.pc = pc;
8     }
9
10    @Override
11    public void run() {
12        pc.consume();
13    }
14
15 }
16 }

```

### Step 4: ProducerConsumer logic.

```

1 package com.debug.multithread;
2
3 import java.util.concurrent.ArrayBlockingQueue;
4

```

#### Multithreading (12)

01: ♥♦ 15 Beginner

02: ♥♦ 10+ Java

03: ♦ More Java

04: ♦ 6 popular J

05: ♦ How a thre

06: ♦ 10+ Atomic

07: 5 Basic multi

08: ♦ ThreadLoc

09: Java FutureT

10: ♦ ExecutorSe

Java ExecutorSe

Producer and Co

#### Algorithms (5)

#### Annotations (2)

#### Collection and Data

#### Differences Between

#### Event Driven Progr

#### Exceptions (2)

#### Java 7 (2)

#### Java 8 (24)

#### JVM (6)

#### Reactive Programn

#### Swing & AWT (2)

#### JEE Interview Q&A (3)

#### Pressed for time? Jav

#### SQL, XML, UML, JSC

#### Hadoop & BigData Int

#### Java Architecture Inte

#### Scala Interview Q&As

#### Spring, Hibernate, & I

#### Testing & Profiling/Sa

#### Other Interview Q&A I

#### Free Java Interview

## As a Java Architect

[Java architecture & design concepts](#)

```

5 //only one thread can access either produce or c
6 public class ProducerConsumer {
7
8     private int count = 0;
9     private ArrayBlockingQueue<Integer> queue =
10
11     public synchronized void produce() {
12         while (true) {
13             try {
14                 wait(100);
15             } catch (InterruptedException ie) {}
16
17             if (queue.isEmpty()) {
18                 count++;
19                 try {
20                     Thread.sleep(4000); // takes
21                 } catch (InterruptedException e)
22                 queue.add(count);
23                 System.out.println(Thread.curren
24                 notifyAll();
25             }
26         } //end while
27     }
28
29     public synchronized void consume() {
30         while (true) {
31             try {
32                 wait(100);
33             } catch (InterruptedException ie) {}
34
35             if (!queue.isEmpty()) {
36                 Integer consumed = queue.remove(
37                 System.out.println(Thread.curren
38                 notifyAll();
39             }
40         } //end while
41     }
42 }

```

[interview Q&As with diagrams](#) | [What should be a typical Java EE architecture?](#)

## Senior Java developers must have a good handle on

[open all](#) | [close all](#)

- [Best Practice \(6\)](#)
- [Coding \(26\)](#)
- [Concurrency \(6\)](#)
- [Design Concepts \(7\)](#)
- [Design Patterns \(11\)](#)
- [Exception Handling \(3\)](#)
- [Java Debugging \(21\)](#)
- [Judging Experience \(1\)](#)
- [Low Latency \(7\)](#)
- [Memory Management \(1\)](#)
- [Performance \(13\)](#)
- [QoS \(8\)](#)
- [Scalability \(4\)](#)
- [SDLC \(6\)](#)
- [Security \(13\)](#)
- [Transaction Management \(1\)](#)

## Popular Posts

♦ 11 Spring boot interview questions & answers

893 views

♦ Q11-Q23: Top 50+ Core on Java OOP Interview Questions & Answers

851 views

18 Java scenarios based interview Questions and Answers

456 views

001A: ♦ 7+ Java integration styles & patterns interview questions & answers

413 views

♦ 7 Java debugging interview questions & answers

316 views

## 80+ step by step Java Tutorials

[open all](#) | [close all](#)

- [Setting up Tutorial \(6\)](#)
- [Tutorial - Diagnosis \(2\)](#)
- [Akka Tutorial \(9\)](#)
- [Core Java Tutorials \(2\)](#)
- [Hadoop & Spark Tuto](#)

## 01b: ♦ 13 Spring basics Q8 – Q13 interview questions & answers

303 views

## ♦ 10 ERD (Entity-Relationship Diagrams) Interview Questions and Answers

301 views

## 01: ♦ 15 Ice breaker questions asked 90% of the time in Java job interviews with hints

290 views

## ♦ Q24-Q36: Top 50+ Core on Java classes, interfaces and generics interview questions & answers

259 views

## 8 Git Source control system interview questions & answers

216 views

- ✚ JEE Tutorials (19)
- ✚ Scala Tutorials (1)
- ✚ Spring & Hibernate Tutorials (1)
- ✚ Tools Tutorials (19)
- ✚ Other Tutorials (45)

## 100+ Preparing for pre-interview Java written home assignments & coding tests

open all | close all

- ✚ Can you write code? (1)
- ✚ ♦ Complete the given code (1)
- ✚ Converting from A to B (1)
- ✚ Designing your class (1)
- ✚ Java Data Structures (1)
- ✚ Passing the unit tests (1)
- ✚ What is wrong with this code? (1)
- ✚ Writing Code Home Assignment (1)
- ✚ Written Test Core Java (1)
- ✚ Written Test JEE (1)

Bio

Latest Posts



### Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via [Amazon.com](http://Amazon.com) in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site.



### About Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via [Amazon.com](http://Amazon.com) in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site.

## How good are your...to go places?

open all | close all

- ✚ Career Making Knowledge (1)
- ✚ Job Hunting & Resumes (1)

◀ Part 1: Creating an empty multi module maven project structure

Part 2- Adding dependency jars to multi-module MVN project ▶

**Posted in** Multithreading

## Empowers you to open more doors, and fast-track

### Technical Know Hows

☀ [Java generics in no time](#) ☀ [Top 6 tips to transforming your thinking from OOP to FP](#) ☀ [How does a HashMap internally work? What is a hashing function?](#)

☀ [10+ Java String class interview Q&As](#) ☀ [Java auto un/boxing benefits & caveats](#) ☀ [Top 11 slacknesses that can come back and bite you as an experienced Java developer or architect](#)

### Non-Technical Know Hows

☀ [6 Aspects that can motivate you to fast-track your career & go places](#) ☀ [Are you reinventing yourself as a Java developer?](#) ☀ [8 tips to safeguard your Java career against offshoring](#) ☀ [My top 5 career mistakes](#)

## Prepare to succeed

☀ [Turn readers of your Java CV go from “Blah blah” to “Wow”? ☀ \[How to prepare for Java job interviews?\]\(#\) ☀ \[16 Technical Key Areas\]\(#\) ☀ \[How to choose from multiple Java job offers?\]\(#\)](#)

Select Category ▼

## © Disclaimer

The contents in this Java-Success are copy righted. The author has the right to correct or enhance the current content without any prior notice.

These are general advice only, and one needs to take his/her own circumstances into consideration. The author will not be held liable for any damages caused or alleged to be caused either directly or indirectly by these materials and resources. Any trademarked names or labels used in this blog remain the property of their respective trademark owners. No guarantees are made regarding the accuracy or usefulness of content, though I do make an effort to be accurate. Links to external sites do not imply endorsement of the linked-to sites.

