Javarevisited

Blog about Java, Programming, Spring, Hibernate, Interview Questions, Books and Online Course Recommendations from Udemy, Pluarlsight etc

Home core java spring hibernate collections multithreading design patterns interview questions coding data structure OOP java 8 books About Me
Java Certifications JDBC jsp-servlet JSON SQL Linux Courses online resources jvm-internals REST Eclipse jQuery Java IO Java XML

TUESDAY, NOVEMBER 20, 2018

Top 15 Java Multithreading, Concurrency Interview Questions Answers asked in Investment banks

Multi-threading and concurrency questions are an essential part of any Java interview. If you are going for any Java interview on any Investment bank e.g. Barclays, Citibank, Morgan Stanley etc for Cash Equities Front Office Java Developer position, you can expect a lot of multi-threading interview questions on your way. Multi-threading and concurrency are favorite topics on Investment banking interviews, especially on electronic trading development jobs and they grill candidate on many tricky java thread interview questions. They just want to ensure that the guy has solid knowledge of multi-threading and concurrent programming in Java because most of them are in the business of performance which provides them a competitive advantage and it's hard to write correct and robust concurrent code.

For example, high volume and low latency Electronic Trading Systems which are used for Direct to Market (DMA) trading are usually concurrent in nature. Most of the time the focus on microsecond latency, that's why a good knowledge of how to use effectively to minimize latency and improve throughput is important.

These are some of my favorite *thread interview questions* on Java asked on different on a different time. I am not providing an answer to these thread interview questions but I will give you hint whenever possible, some time hint is also enough to answer. I will update the post further with detailed answers just like I did for 10 Singleton interview questions in Java recently.

After the introduction of concurrency package in JDK 1.5 questions on concurrent utility and concurrent collections were increased e.g. ThreadLocal, BlockingQueue, Counting Semaphore and ConcurrentHashMap become popular.

Same is true for Java 8 and Java 9, where questions on lambda expression and parallel streams, new fork-join pool, `CompletableFuture` is also on the rise in 2018 and will remain in 2019. Hence you should prepare well for those topics.

15 Java Thread Interview Questions and answers

Anyway, without any further ado, here is my list of some of the frequently asked Java multithreading and concurrency questions from Java developer Interviews on Investment banks e.g. Barclays, Morgan Stanley, Citibank etc.

1) You have thread T1, T2, and T3, how will you ensure that thread T2 run after T1 and thread T3 run after T2?
This thread interview question is mostly asked in the first round or phone screening round of interview and purpose of this multi-threading question is to check whether the candidate is familiar with the concept of "join" method or not. The answer to this multi-threading questions is simple it can be achieved by using the join method of Thread class. If you are not familiar with join method, please see my post how to join threads in Java.

2) What is the advantage of new Lock interface over a synchronized block in Java? You

Follow by Email

Email address... Submit

Interview Questions

core java interview question (169)
Coding Interview Question (72)
data structure and algorithm (69)
interview questions (47)
object oriented programming (31)
SQL Interview Questions (30)
design patterns (30)
thread interview questions (30)
collections interview questions (25)
spring interview questions (19)
database interview questions (16)
servlet interview questions (15)
Programming interview question (6)
hibernate interview questions (6)

Best of Javarevisited

How Spring MVC works internally?
How to design a vending machine in Java?
How HashMap works in Java?
Why String is Immutable in Java?
10 Articles Every Programmer Must Read
How to convert lambda expression to method reference in Java 8?
10 Tips to improve Programming Skill
10 OOP design principles programmer should know
How Synchronization works in Java?
10 tips to work fast in Linux
5 Books to improve Coding Skills

Java Tutorials

date and time tutorial (21)
FIX protocol tutorial (15)
Java Certification OCPJP SCJP (24)
java collection tutorial (73)
java IO tutorial (28)
Java JSON tutorial (12)
Java multithreading Tutorials (56)
Java Programming Tutorials (18)
Java xml tutorial (16)
JDBC (29)
jsp-servlet (37)
online resources (113)

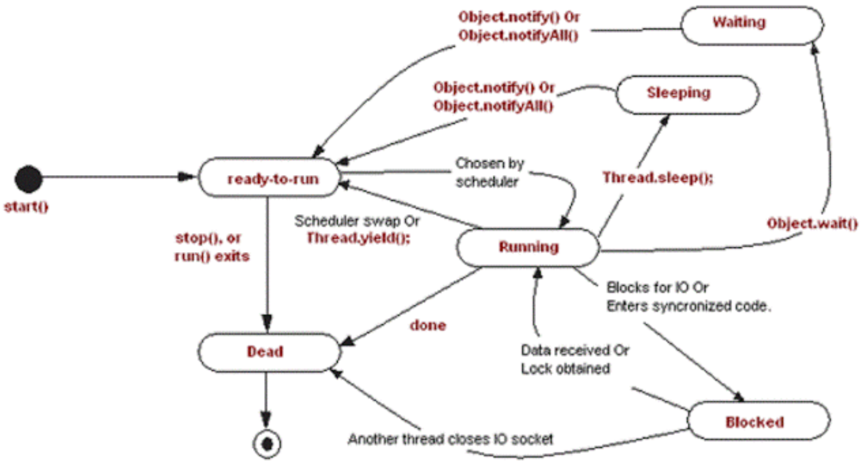
Followers

need to implement a high-performance cache which allows multiple readers but the single writer to keep the integrity how will you implement it?
The major advantage of lock interfaces on multi-threaded and concurrent programming is they provide two separate lock for reading and writing which enables you to write high-performance data structure like ConcurrentHashMap and conditional blocking.

This java threads interview question is getting increasingly popular and more and more follow-up questions come based on the answer of the interviewee.

I would strongly suggest reading about Locks before appearing for any Java multi-threading interview because nowadays It's heavily used to build cache for an electronic trading system on client and exchange connectivity space.

3) What are differences between wait and sleep method in Java?
Another frequently asked thread interview question in Java mostly appear in a phone interview. The only major difference is that wait releases the lock or monitor while sleep doesn't release any lock or monitor while waiting. The wait is used for inter-thread communication while sleep is used to introduce pause on execution. See my post wait vs sleep in Java for more differences



4) Write code to implement blocking queue in Java?
This is relatively tough Java multi-threading interview question which serves many purposes, it checks whether a candidate can actually write Java code using thread or not, it sees how good candidate is on understanding concurrent scenarios and you can ask a lot of follow-up question based upon his code. If he uses wait() and notify() method to implement blocking queue, Once candidate successfully writes it you can ask him to write it again using new Java 5 concurrent classes etc.

5) Write code to solve the Produce consumer problem in Java? (solution)
Similar to above questions on the thread but more classic in nature, some time interviewer ask follow up questions How do you solve producer consumer problem in Java, well it can be solved in a multiple ways, I have shared one way to solve producer consumer problem using BlockingQueue in Java, so be prepared for surprises. Sometimes they even ask to implement a solution of dining philosopher problem as well.

6) Write a program which will result in a deadlock? How will you fix deadlock in Java?

Followers (4670) Next

Follow

Categories

courses (56)
SQL (55)
linux (37)
database (33)
Eclipse (28)
Java Certification OCPJP SCJP (24)
JVM Internals (22)
JQuery (17)
REST (16)
Maven (11)
Testing (11)
general (10)

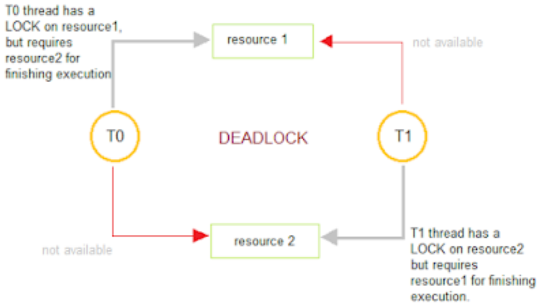
Blog Archive

► 2019 (67)
▼ 2018 (132)
► December (22)
▼ November (36)
Top 5 Websites to Learn Coding in Java Online for ...
Top 5 Free Git Courses for Programmers to Learn On...
Spring Framework Certification FAQ - Frequently as...
Top 10 Programming and Coding Interview Courses fo...
Top 5 books to learn Agile and Scrum for Programme...
Top 5 IntelliJ IDEA and Android Studio Courses for...
How to Check/Uncheck CheckBoxes in a Page using jQ...
How to delete a key value pair from a HashMap duri...
Top 8 Python Libraries for Data Science, Machine L...
How Http Basic Authentication works in Spring Secu...

This is my favorite java thread interview question because even though deadlock is quite common while writing multi-threaded concurrent program many candidates not able to write deadlock free code and they simply struggle.

Just ask them you have N resources and N threads and to complete an operation you require all resources.

Here N can be replaced with 2 for the simplest case and higher number to make the question more intimidating. You can further check [Applying Concurrency and Multi-threading to Common Java Patterns](#) course on Pluarlsight to learn more about popular concurrency patterns and the right way to use them.


T0 thread has a LOCK on resource1, but requires resource2 for finishing execution
resource 1
not available
T0
DEADLOCK
T1
not available
resource 2
T1 thread has a LOCK on resource2 but requires resource1 for finishing execution.

7) What is an atomic operation? What are atomic operations in Java?
Simple java thread interview questions, another follow-up is do you need to synchronize an atomic operation? :) You can read more about [java synchronization](#) here.

8) What is a volatile keyword in Java? How to use it? How is it different from the synchronized method in Java?
Thread questions based on a [volatile keyword in Java](#) has become more popular after changes made on it on Java 5 and Java memory model. It's good to prepare well about how volatile variables ensures visibility, ordering, and consistency in a concurrent environment.

9) What is a race condition? How will you find and solve race condition?
Another multi-threading question in Java which appear mostly on senior-level interviews. Most interviewer grill on recent race condition you have faced and how did you solve it and sometime they will write sample code and ask you detect race condition. See my post on the [Race condition in Java](#) for more information.

In my opinion, this is one of the best java thread interview questions and can really test the candidate's experience on solving race condition or writing code which is free of data race or any other race condition.

One of the best book to get the mastery of this topic is ["Java Concurrency in Practice by Brian Goetz"](#). If you are thinking that this book is old then think again, it's still relevant in the era of Java 11.

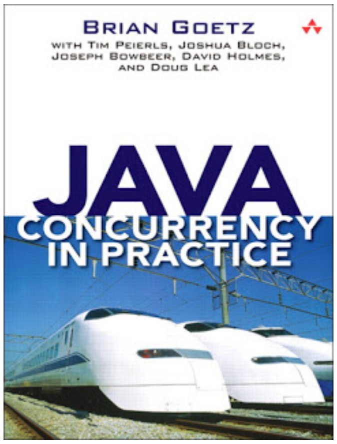
Entuware or Whizlabs? Which Exam Simulator Should ...
Top 30 Eclipse Keyboard Shortcuts for Java Program...
Does Spring Certification helps in Job and Career?...
Java vs Python - Which Programming Language Should...
Top 5 Data Science and Machine Courses to Learn On...
How to avoid deadlock in Java Threads?
How to convert float to long or int data type in J...
Top 15 Java Multithreading, Concurrency Interview ...
How to find a Process Listening on a Specific Port...
The @SpringBootApplication Annotation Example in J...
10 Differences between StringBuffer and StringBuil...
Top 10 Angular Tutorials and Courses for Web Devel...
How to Schedule Spring Professional Certification ...
Do you Need to Pass OCAJP before taking OCPJP - Ja...
Top 10 ConcurrentHashMap Questions from Java Inter...
Top 5 Courses to Learn Shell Scripting in Linux an...
Top 5 Soft Skill and Career Development books and ...
Top 5 Java Multithreading and Concurrency Courses ...
5 Must Read Books to become Software Architect or ...
Top 5 RESTful Web Services with Spring and Java Co...
3 Ways to Learn Spring Core, Spring MVC, Spring Se...
3 Ways to convert String to Boolean in Java? Examp...
How to Remove Leading/Trailing White Space from a ...
The best Course to Learn Spring Security 5 for Exp...
Top 5 Node.js and Express JS online courses for We...
Top 5 Free Agile Courses for Programmers and Softw...

► October (24)
► September (13)
► August (4)
► July (12)
► June (8)
► May (1)
► March (5)
► February (5)
► January (2)

► 2017 (794)
► 2016 (127)

References

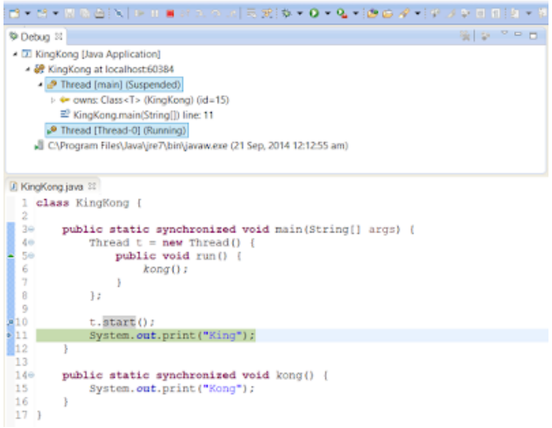
1. [Oracle's Java Tech Network](#)
2. [jQuery Documentation](#)
3. [Microsoft SQL Server Documentation](#)


BRIAN GOETZ
WITH TIM PEIERLS, JOSHUA BLOCH, JOSEPH BOWBEER, DAVID HOLMES, AND DOUG LEA
JAVA
CONCURRENCY
IN PRACTICE

10) How will you take thread dump in Java? How will you analyze Thread dump?
In UNIX you can use `kill -3` and then thread dump will print on log on windows you can use `"CTRL+Break"`. Rather simple and focus thread interview question but can get tricky if he asks how you analyze it. A thread dump can be useful to analyze deadlock situations as well.

11) Why we call start() method which in turns calls run() method, why not we directly call run() method?
Another classic java multi-threading interview question This was my original doubt when I started programming in the thread. Now days mostly asked in a phone interview or first round of interview at mid and junior level java interviews.

The answer to this question is that when you call `start()` method it creates new Thread and executes code declared in the `run()` while directly calling run() method doesn’t create any new thread and execute code on the same calling thread. Read my post Difference between start and run method in Thread for more details.



```
KingKong [Java Application]
    KingKong at localhost:60384
        Thread [main] (Suspended)
            owns: Class<T> (KingKong) (id=15)
            KingKong.main(String[]) line: 11
        Thread [Thread-0] (Running)
    C:\Program Files\Java\jre7\bin\javaw.exe (21 Sep, 2014 12:12:55 am)
```

```java
class KingKong {

    public static synchronized void main(String[] args) {
        Thread t = new Thread() {
            public void run() {
                kong();
            }
        };

        t.start();
        System.out.print("King");
    }

    public static synchronized void kong() {
        System.out.print("Kong");
    }
}
```

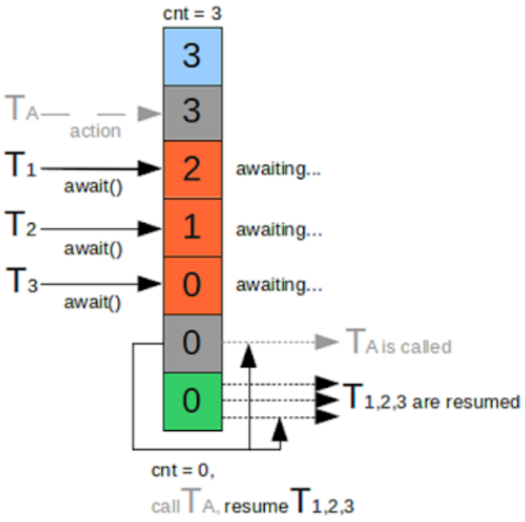
12) How will you awake a blocked thread in Java?
This is a tricky question on threading, blocking can result in many ways, if thread is blocked on IO then I don't think there is a way to interrupt the thread, let me know if there is any, on the other hand, if thread is blocked due to result of calling `wait()`, `sleep()`, or `join()` method you can interrupt the thread and it will awake by throwing InterruptedException. See my post How to deal with blocking methods in Java for more information on handling blocked thread.

4. Java SE 8 API Documentation
5. Spring Documentation
6. Oracle's JAva Certification
7. Spring Security 5 Documentation

Pages

Privacy Policy

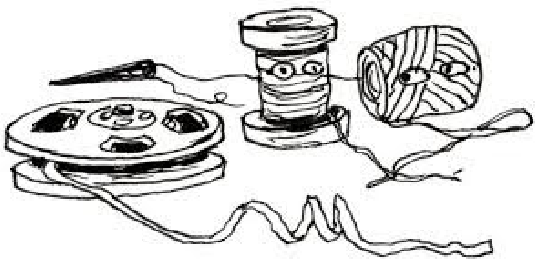
13) What is the difference between CyclicBarriar and CountdownLatch in Java? (answer)
New Java thread interview questions mostly to check familiarity with JDK 5 concurrent packages. One difference is that you can reuse CyclicBarrier once the barrier is broken but you can not reuse CountDownLatch. If you want to learn more see Multithreading and Parallel Computing in Java course on Udemy.



14) What is an immutable object? How does it help in writing a concurrent application?
Another classic interview questions on multi-threading, not directly related to the thread but indirectly helps a lot. This java interview question can become more tricky if ask you to write an immutable class or ask you Why String is immutable in Java as a follow-up.

15) What are some common problems you have faced in multi-threading environment? How did you resolve it?
Memory-interference, race conditions, deadlock, livelock, and starvation are an example of some problems comes in multi-threading and concurrent programming. There is no end to a problem if you get it wrong and they will be hard to detect and debug.

This is mostly experienced based interview question on java thread instead of fact-based. You can further see Java Concurrency in Practice Course by Heinz Kabutz for some real-world problems faced in actual high-performance multi-threaded applications.



These were some of my favorite J*ava thread interview questions and mostly asked on Investment banks*. This list is by no means complete so please contribute some of interesting java thread questions you have faced during the interview. Purpose of this article is to collect and share great interview questions on multi-threading concept which not only helps in the interview but opens the door for learning a new threading concept.

Update:

One of the Javarevisited readers, Hemant has contributed some more thread interview questions in Java, though he hasn’t provided answer sand left that job for me, I will certainly do when time allows, just like I have recently updated 10 Singleton interview question in Java with answers.

If you guys know answers to this Java concurrency questions than please post a comment.

Here are his comment “Good questions on multi-threading though you may need to prepare more in order to clear any multi-threading interview, you need to be familiar with the concept of immutability, thread-safety, race condition and many more.

10 or 15 question is good for the quick recap but you at least need to prepare more than 50 questions on threading and concurrency to perform better on Java interview.

You can find some interesting thread question below which is no doubt highly popular –

1) Difference between green thread and native thread in Java?

2) Difference between thread and process? (answer)

3) What is context switching in multi-threading?

4) Difference between deadlock and livelock, deadlock and starvation?

5) What thread-scheduling algorithm is used in Java?

6) What is thread-scheduler in Java?

7) How do you handle an unhandled exception in the thread?

8) What is thread-group, why its advised not to use thread-group in Java?

9) Why Executor framework is better than creating and managing thread by the application?

10) Difference between Executor and Executors in Java? (answer)

11) How to find which thread is taking maximum CPU in windows and Linux server?

Apart from practicing these question answers, more important is to understand the concept behind these multi-threading questions simply mugging the answers of these thread interview questions is not going to help because there would be a lot of follow-ups based on your answer and if you haven't mastered the particular thread topic it would be difficult to answer them.

Other **Java Interview Questions** on different topics you may like to Explore
The Complete Java Master Class (learn more)
Top 10 Spring Framework Interview Questions with Answers (see here)
20 Java Design Pattern Questions asked in Interviews (see here)
10 Frequently asked Servlet Interview Questions with Answers (see here)
10 JSP Questions from J2EE Interviews (read here)
12 Good RESTful Web Services Questions from Interviews (read here)
10 JDBC Interview Questions for Java Programmers (questions)
15 Java NIO and Networking Interview Questions with Answers (see here)
15 Data Structure and Algorithm Questions from Java Interviews (read here

Thanks for reading this article so far. If you like these multi-threading and Concurrency Interview questions then please share with your friends and colleagues. If you have any questions or feedback then please drop a note.

Difference between @RestController and @Controller Annotation in Spring MVC and REST

The @RestController annotation in Spring MVC is nothing but a combination of @Controller and @ResponseBody annotation. It was added into...

Java Revisited

By Javin Paul at November 20, 2018

Labels: core java , core java interview question , Java multithreading Tutorials , thread interview questions
Location: United States

30 comments :

Anonymous said...

Indeed a great collection of java multi-threading interview questions and I agree with you. wall street firms really need people who are very strong in multi-threading and concurrent programming. you could probably include more interview questions from new Thread classes added on JDK5

July 28, 2011 at 9:50 PM

Anonymous said...

answers of your thread interview questions are not proper. you should work to make your interview questions and answers more relevant, accurate and better. Though I agree that multi-threading is not an easy thing to achieve and multi-threading interview is real tough.

August 4, 2011 at 11:33 PM

Anonymous said...

Thanks for another great post! You're very smart, and you're writing really important articles. That being said, you have to get an editor! Your spelling, grammar, and punctuation distract from otherwise amazing content. Writing is a great way to get better at writing, but you can get better even faster with feedback from writers with more skill than yourself. In any case, you rock and have amazing potential.

September 23, 2011 at 9:03 AM

Javin@ OutOfMemoryError in Java said...

@Anonymous, thanks for your feedback , English not being my first language, it has some characteristic or effect of Hindi in grammar and work flow.I love to make my English flawless and increase reading experience of my readers but more importantly I like to share my experience and until I connect to my readers or my fellow developers I am happy and it motivates me to improve. Once again thanks for your feedback appreciate.

Javin

September 23, 2011 at 6:26 PM

Javin @ OutOfMemoryError in Perm Gen Space said...

Hi Vineet, its indeed a interview question guide, you can easily find detailed answer for these multi-threading question by doing google, Though I have also provided pointers and main point which some one will look. would be great if you could contribute some good answer and I will include in post.

October 14, 2011 at 6:12 PM

Anonymous said...

Usage of Singleton Pattern in Java API

java.lang.Runtime#getRuntime() java.awt.Desktop#getDesktop()

November 9, 2011 at 11:16 AM

Garima said...

I am looking for some more thread questions and answers in Java, your thread questions are good but I need more tough thread interview questions something which is new and not everybody know about those interview questions.Please help

November 23, 2011 at 2:27 AM

Anonymous said...

Question: what are the benfits we will get when we extend a java.lang.Thread class instead of implimenting Runnable interface?

January 11, 2012 at 5:12 AM

Arulkumaran.K said...

Good list of questions. I have listed frequently asked multi-threading interview questions and answers at http://java-success.blogspot.com.au/2011/09/java-multi-threading-interview.html that will be very handy as well.

February 26, 2012 at 10:07 PM

Javin @ spring interview question answer said...

Thanks Arul. you like these thread interview questions. glad to here that.

February 27, 2012 at 5:36 AM

Javin @ Servelt Interview Questions said...

Thanks Kamal. I have shared many of those in my blog, just use search box on top left. by the way here are few :
10 questions on Singleton Pattern in Java
Why String is immutable in Java

March 23, 2012 at 7:10 AM

Peter Lawrey said...

1. I would call run() instead of start() ;)

2. I would make it clear there is a difference between ReentrantLock and ReentrantReadWriteLock.

4. While I started programming Java before version 5.0, many people started programming after 2004, so I wouldn't consider the concurrency library that new as there when many people started. BTW It was available as a separate library from 1998

6. My favourite answer to this is to create a Thread in a static block which access the class and does a join. This creates a deadlock with just one new thread and no explicit locking. ;)

8. Another question I get is the difference in volatile between Java and C++

10. You might add, how do you take a thread dump programmatically.

Excellent posts, keep them up.

April 8, 2012 at 12:50 AM

Javin @ thread safe class in java said...

Hi Peter, Good to see you back here :) difference between volatile in Java and C++ is a good question if you mention C++ in your resume and also taking a thread dump programmatically and with command prompt in case of deadlock is a good skill to have. What is current trend, have you spotted any new trend related to concurrency in Java interviews ?

April 9, 2012 at 6:10 AM

Anonymous said...

Can you please share some multi-threading questions asked in Google Interviews. I am very interested on what these top companies like Google, Amazon, Facebook or Investment banks like Nomura or Goldman asked in Java, JVM and Threading. Thanks

April 10, 2012 at 8:02 PM

Jon said...

Is it possible that interviewers do not always know the answers, and are taking advantage of job interviews to get free software consultancy?

July 21, 2012 at 5:15 AM

Javin @ transient vs volatile said...

@Jon, You bring the very interesting point about Java interviews,Many do and which I guess is not bad because if some one can solve the problem you are facing than it's good signal that guy can be useful.

July 21, 2012 at 5:21 AM

Anonymous said...

Please try to answer question dont leave reader in dark room

August 28, 2012 at 12:10 AM

Anonymous said...

Javin, really good questions. Just another query which core java book you recommned to understand concepts.

December 4, 2012 at 11:53 PM

mayank tripathi said...

@Javin : The main advantage of new Locks over Synchronized block is to handle Deadlock better way. In blocks once deadlock happens you have no way to fix it unless restart the jvm but Lock gives you tools to handle deadlock situations. Other benefits are bonus :)

February 11, 2013 at 4:09 PM

Rashi said...

Hi Guys, I would like to share one of the Interview Question related to ThreadPool implementation. I was asked in JP Morgan and Chase that , What should you take care, if you have to make your own Thread Pool? I thought about Executors framework of Java 5 but couldn't answer much. I am interested in points, what are the important points one should take care while creating there own thread pool in Java?

Thx
Rashi

April 16, 2013 at 8:52 PM

Anonymous said...

I was asked to implement thread pool in java as well and they were interested in knowing key things you would consider to design pool

April 26, 2013 at 10:01 PM

Andrew said...

Why knowledge of multithreading in Java is so important to get a Job? What makes it so desirable and same time tough to get master?

August 12, 2013 at 9:04 PM

Aravind Datta said...

Javin, I am a Java developer with 10+ years of experience and accidentally bumped into your blog around 15 days ago. Since then, I am stuck with it :-). I tried to read thru so many articles that you wrote and they are extremely useful and extremely well written with a lot of clarity on the fundamental concepts! Its a great job! Keep it up!

I have a couple of questions on threads:
1. Is "a thread of execution" same as "thread object" ?
2. Lets say, there is a java application (in real production scenario) and it can accept concurrent requests.. implies, multiple users can get data from this application while it is running in production.. and I can see the logs that multiple "threads" are running in the log.. like say, Thread-1, Thread-2, Thread-23, Thread-50.. etc.. based on the number of "concurrent" requests it is processing.
My question is, is this concurrency same as "multi threading within an application"? Or is it different? Can a java application handle concurrent requests (from multiple users/sessions) simultaneously by default? How does it work? Please let us know.

Best wishes..

September 28, 2013 at 10:28 AM

Sekhar babu Karri said...

```
package com.learning.material;

public class MyThread {

/**
* @param args
*/
public static void main(String[] args) throws InterruptedException {

System.out.println("i am the Main thread...starting......");

final Thread t1 = new Thread(){
```

```
public void run(){
System.out.println("i am thread 1 starting......");
try{
Thread.sleep(5000);
System.out.println("I am Thread 1 finished ");
}catch(InterruptedException ie){
System.out.println("Exception:"+ie.getMessage());
}
}
};

final Thread t2 = new Thread(){
public void run(){
try{
t1.join();
System.out.println("I am Thread 2 finished ");
}catch(InterruptedException ie){
System.out.println("Exception:"+ie.getMessage());
}
}
};

Thread t3 = new Thread(){
public void run(){
try{
t2.join();
System.out.println("I am Thread 3 finished ");
}catch(InterruptedException ie){
System.out.println("Exception:"+ie.getMessage());
}
}
};
t1.start();
t2.start();
t3.start();
t3.join();

System.out.println("i am Main thread finished");

}

}
```

November 12, 2013 at 10:27 AM

Gopal said...

I think you could do better by adding few more multithreading and concurrency questions from java.util.concurrent package e.g.

1) Difference between Callable and Runnable interface in Java?
2) What is FutureTask in Java? What is benefit of using it?
3) Difference between Thread.isInterrupted() and Thread.interrupt() method?
4) How ConcurrentHashMap achieves scalability?
5) What is Semaphore in Java?
6) Difference between ReentrantLock and synchronized keyword in Java?
7) What is ReadWriteLock in Java?
8) What is busy spinning?
9) Difference between notify() and notifyAll() in Java? When do you use notifyAll() over notify()?
10) Difference between wait, sleep and yield in Java?
11) Difference between Runnable vs Thread in Java?
12) Why wait and notify methods are called from synchronized block or method in Java?
13) What is doubled checked locking of Singleton?
14) Difference between LiveLock and Starvation in Java?
15) What is daemon thread in Java? What is use of daemon threads?
16) Why wait, notify and notifyAll are in Object class instead of Thread?
17) Why wait method is called from loop in Java
18) Difference between wait method and Condition object in Java?
19) Write code for getInstance() method of thread-safe Singleton in Java?
20) What is Java Memory Model?
21) Can we make static method synchronized in Java?
22) What is lock stripping?
23) How many thread is created by Timer class?
24) What is ThreadLocal in Java?
25) What is ThreadLocal memory leak in Java? How to fix that?
26) What is race condition in Java? Have you ever faced race conditions?
27) What is thread-safety? What are few ways to achieve thread-safety in multithreading code?
28) Difference between synchronized method and block in Java? Which one is better?
29) Difference between volatile and transient variable in Java?
30) How to check if a Thread holds a Lock in Java?

December 25, 2013 at 7:46 PM

cbp said...

Thanks for the list of questions and all the great comments. I wanted to point out an error in #11 though.

When you call the start() method in the Thread class, it does not directly call the run() method. It calls a native code method start0(). This native code will only schedule the thread to be executed with the ThreadScheduler. The JVM will then grab the item from the ThreadScheduler and execute the run() function. It is very similar to a Strategy Pattern from GoF where the JVM knows how to execute a strategy and a Thread class defines what the strategy is that is to be executed.

March 27, 2014 at 5:44 AM

Nihar Ranjan Khatua said...

Hi Guys,
This is Nihar.I have one question from threading concept i.e in an application 0-10 thread is good or 10-100 thread..which one & why????

July 15, 2014 at 11:42 PM

Anonymous said...

This was asked in my recent interview .
Object ob1 has two methods A and B.
You have two Threads (Thread 1 and Thread 2)

Question is I should be access Method A of Ob1 from Thread 1 and at the same time Thread 2 should be able to access Method B of Ob1.

However if Thread 1 is using Method A , then Thread 2 should not be able to use Method A of Ob1.
How is this possible ?? One hint given was Monitors :-)

August 9, 2014 at 4:53 AM

Anonymous said...

answer of question 1:

```
MyRunnable.java
public class MyRunnable implements Runnable {
private String name;
public MyRunnable(){

}
public MyRunnable(String name){
this.name=name;
}
@Override
public void run() {
// TODO Auto-generated method stub
for(int i=0;i<5;i++){
System.out.println("Thread name - "+Thread.currentThread().getName()+" i = "+i);
}
}

}

Main.java
public class Main {

public static void main(String[] args) {
// TODO Auto-generated method stub
Runnable runnable=new MyRunnable();
Thread t1=new Thread(runnable,"Thread1");
Thread t2=new Thread(runnable,"Thread2");
Thread t3=new Thread(runnable,"Thread3");
try{
t1.start();
t1.join();
t2.start();
t2.join();
t3.start();
t3.join();
}catch(InterruptedException ie){
ie.printStackTrace();
}
}

}
```

July 11, 2015 at 10:23 PM

Anonymous said...

While execution , a thread in ExecutorFramework throws some exception.How can we notify this exception to the calling application.

July 20, 2015 at 4:56 AM

Post a Comment

Enter your comment...

Comment as: lemon_leave (Go ◆ Sign out

Publish Preview □ Notify me

How to solve java.sql.SQLException: No suitable driver found for jdbc:mysql://localhost:3306/test

The error "java.sql.SQLException: No suitable driver found for jdbc:mysql://localhost:3306/test" occurs when you try to connect MySQL data...

Java Revisited

Newer Post Home Older Post

Subscribe to: Post Comments (Atom)

Search This Blog

Search