

Javarevisited

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

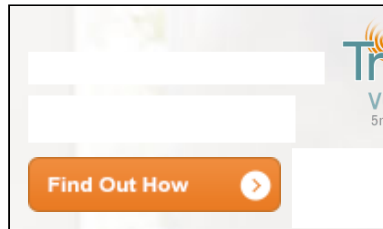
Home core java spring hibernate collections multithreading design patterns interview questions coding data structure OOP java 8 books About Me

Java Certifications JDBC jsp-servlet JSON SQL Linux Courses online resources jvm-internals REST Eclipse jQuery Java IO Java XML

SATURDAY, JUNE 17, 2017

How to use Callable and Future in Java? Example

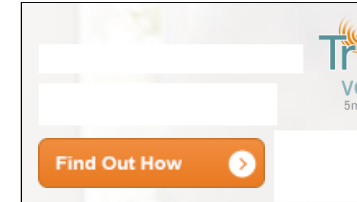
Callable [interface](#) was added in Java 5 to complement existing Runnable interface, which is used to wrap a task and pass it to a Thread or thread pool for asynchronous execution. Callable actually represent an asynchronous computation, whose value is available via Future object. All the [code](#) which needs to be executed asynchronously goes into `call()` method. Callable is also a [single](#) abstract method type (SAM type), so it can be used along with [lambda expression on Java 8](#). Both Callable and Future are parametric type and can be used to wrap [classes](#) like Integer, String or anything else. When you pass a Callable to thread pool, it choose one thread and execute the Callable. It immediately return a Future object which promises to hold result of computation once done. You can then call `get()` method of Future, which will return result of computation or block if Computation is not complete. If you don't like indefinite blocking then you can also use overloaded `get()` method with timeout. Future also allows you to cancel the task if its not started or interrupt if its started. We will see, how we can calculate factorial of large number using Callable and Future in Java. BTW, if you are serious about mastering concurrency API of Java, I suggest you to also take a look at one of the best book on the subject, [Java Concurrency in Practice](#) by Brian Goetz. It is one of the [book](#) I keep refer whenever I have a doubt or want to refresh my knowledge.



Important Safety Information

Suicidal Thoughts and Actions and Anti Drugs

Antidepressants may increase suicidal actions in some children, teens or your within the first few months of treatment the dose is changed. Depression or other mental illnesses are the most important



Important Safety Information

Suicidal Thoughts and Actions and Anti Drugs

Antidepressants may increase suicidal actions in some children, teens or your within the first few months of treatment the dose is changed. Depression or other mental illnesses are the most important

43

Like

43

Share

Share

Tweet



43

Like

43

Share

Share

Tweet

Callable vs Runnable

Many of you would be familiar with Runnable interface, one of the most popular way to use thread in Java, but you would be happy to know that Runnable is not the only way to create a task which can be executed by parallel threads. You can also use Callable interface to do the same. Main *difference between Runnable and Callable* is that Runnable cannot return any value back to caller but Callable can return value. Another difference is that `call()` method from Callable can also throw checked exception which was not possible by `run()` method of Runnable interface. See [here](#) to learn more about difference between Runnable and Callable in Java.

What's the Future

• Callable – Runnable on steroids

```
Callable<V> {
    V call() throws Exception;
}
```

• Future

```
Future<V> {
    V get();
    boolean cancel();
    boolean isCancelled();
    boolean isDone();
}
```

GoLang Developer | 170495

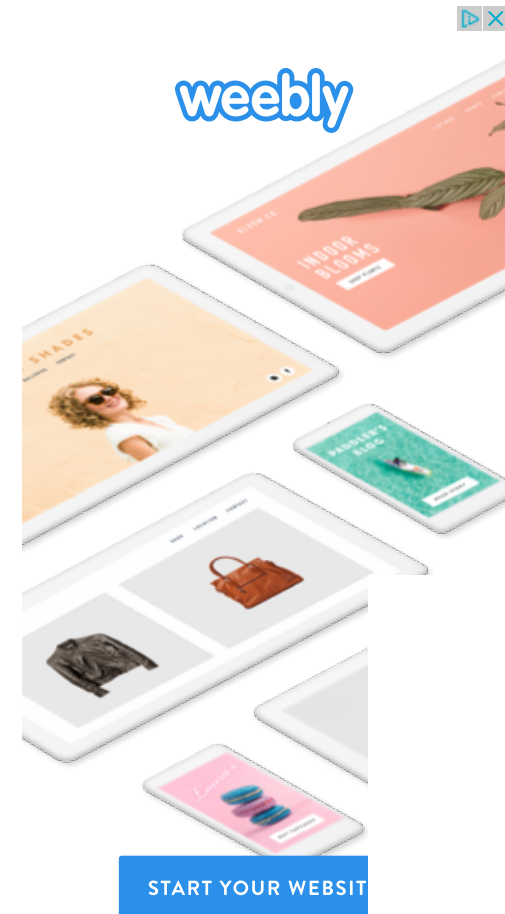
GoLang Developer

Apply

Apply

Army Veteran Discovers Skin Tightening

<https://javarevisited.blogspot.com/2015/06/how-to-use-callable-and-future-in-java.html>



Follow by Email

Email address...

Submit

Interview Questions

core java interview question (169)

Coding Interview Question (72)

data structure and algorithm (70)

interview questions (48)

object oriented programming (31)

SQL Interview Questions (30)

design patterns (30)

thread interview questions (30)

Technique

DermalMedix | Sponsored

[Read Next Story >](#)

Callable in Java

Here is our complete Java program to demonstrate *how you can use Callable and Future together* to implement asynchronous processing in Java. Once you started using thread pool, Callable and Future, you don't need to wait for task to be completed, you can move on with other task and comeback to check whether task is completed or not. If task is finished then just get the result by calling `get()` method, but remember its a [blocking call](#), so it will block if task is not finished.

```
import java.util.concurrent.Callable;
import java.util.concurrent.ExecutionException;
import java.util.concurrent.ExecutorService;
import java.util.concurrent.Executors;
import java.util.concurrent.Future;

/**
 * Simple Java program to demonstrate how to use Callable and Future class in
 * Java. You can use FutureTask for asynchronous processing.
 *
 * @author WINDOWS 8
 */
public class HelloWorldApp {

    public static void main(String... args) throws InterruptedException, ExecutionException {

        // creating thread pool to execute task which implements Callable
        ExecutorService es = Executors.newSingleThreadExecutor();

        System.out.println("submitted callable task to calculate factorial of 10");
        Future result10 = es.submit(new FactorialCalculator(10));

        System.out.println("submitted callable task to calculate factorial of 15");
        Future result15 = es.submit(new FactorialCalculator(15));
    }
}
```

[collections interview questions \(25\)](#)[spring interview questions \(19\)](#)[database interview questions \(16\)](#)[servlet interview questions \(15\)](#)[Programming interview question \(6\)](#)[hibernate interview questions \(6\)](#)**nextiva**

**Over 60% of U.S.
businesses are moving
to cloud phone system**

LEARN WHY**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](#)**Army Veteran Discovers Skin Tightening**

Technique

DermalMedix | Sponsored

[Read Next Story >](#)

```
callable task to calculate factorial of 20");
new FactorialCalculator(20));
```

```
get method of Future to fetch result of factorial 1
```

```
long factorialof10 = result10.get();
System.out.println("factorial of 10 is : " + factorialof10);

System.out.println("Calling get method of Future to get result of factorial 15"
long factorialof15 = result15.get();
System.out.println("factorial of 15 is : " + factorialof15);

System.out.println("Calling get method of Future to get result of factorial 20"
long factorialof20 = result20.get();
System.out.println("factorial of 20 is : " + factorialof20);
```

}

}

```
class FactorialCalculator implements Callable<Long> {
    private int number;

    public FactorialCalculator(int number){
        this.number = number;
    }

    @Override
    public Long call() throws Exception {
        return factorial(number);
    }

    private long factorial(int n) throws InterruptedException {
        long result = 1;
        while (n != 0) {
            result = n * result;
            n = n - 1;
            Thread.sleep(100);
        }

        return result;
    }
}
```

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 \(123\)](#)**Army Veteran Discovers Skin Tightening**

Technique

DermalMedix | Sponsored

[Read Next Story >](#)

```
submitted callable task to calculate factorial of 10
submitted callable task to calculate factorial of 15
submitted callable task to calculate factorial of 20
Calling get method of Future to fetch result of factorial 10
factorial of 10 is : 3628800
Calling get method of Future to get result of factorial 15
factorial of 15 is : 1307674368000
Calling get method of Future to get result of factorial 20
factorial of 20 is : 2432902008176640000
```

43

Like

43

Share

Share

Tweet

When you run this program, you will see that first 4 lines will be printed immediately because `submit()` is a non blocking method, it just takes a task and [returns a Future object](#), it doesn't wait until task is completed. That's why you see that all three tasks to calculate factorials are submitted immediately, but they are not done yet. When our code calls `get()` on first Future object its blocked until the calculation is done, that's why you will see the fifth line printing after sometime. For next two lines also same story because when you call `get()` method it will block until result is available. BTW, you don't need to block, you can even use `isDone()` method to check if calculation is completed or not before calling get method.

Important points about Callable and Future

- 1) Callable is a SAM type interface, so it can be used in lambda expression.
- 2) Callable has just one method `call()` which holds all the code needs to executed asynchronously.
- 3) In Runnable interface, there was no way to return the result of computation or throw checked exception but with Callable you can both return a value and can throw [checked exception](#).
- 4) You can use `get()` method of Future to retrieve result once computation is done. You can check if computation is finished or not by using `isDone()` method.

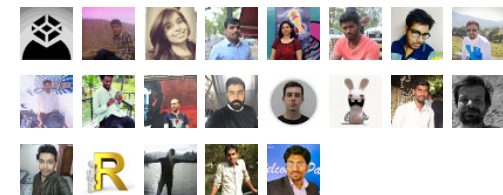
Army Veteran Discovers Skin Tightening<https://javarevisited.blogspot.com/2015/06/how-to-use-callable-and-future-in-java.html>**The Stock of the Century — Buy This RIGHT NOW!**

This tech is going to "hit the reset button" on all human activity...

Promote

[GoLang Developer |...](#)[Agile Transformatio...](#)[Watch The Vi](#)

Followers

Followers (4712) [Next](#)[Follow](#)[Subscribe to Download the E-book](#)

Technique

DermalMedix | Sponsored

[Read Next Story >](#)

Future.cancel() method.

until computation is completed.

CALLABLE & FUTURE

- In this thread, this...
- Callable expects a main entry point called 'call' method and can return a value. Also it can throw an Exception.
- Future represents the lifecycle of a task and provides methods to test whether a task is completed or been canceled, retrieve its result and cancel the task.

Agile Transformation Program Man...
Development Engineer - Java with ...

43

Like

43

Share

Share

Tweet

That's all about **how to use Callable and Future object in Java**. You can wrap asynchronous computation inside `call()` method and pass it to a single thread or thread pool for execution. you don't need to wait until execution complete, your thread can carry on with future object returned by call method. Once computation is done you can query the future object and get the result back.

Further Learning[Multithreading and Parallel Computing in Java](#)[Java Concurrency in Practice - The Book](#)[Applying Concurrency and Multi-threading to Common Java Patterns](#)[Java Concurrency in Practice Course by Heinz Kabutz](#)**Army Veteran Discovers Skin Tightening**<https://javarevisited.blogspot.com/2015/06/how-to-use-callable-and-future-in-java.html>Download
The E-bookBuilding a REST API with
Spring 4?

Email address...

Submit

Categories

courses (66)

SQL (55)

linux (38)

database (34)

Eclipse (28)

Java Certification OCPJP SCJP (24)

JVM Internals (23)

jQuery (17)

REST (16)

Maven (12)

Testing (11)

general (11)

Blog Archive

▶ 2019 (136)

▶ 2018 (99)

▼ 2017 (784)

▶ December (90)

▶ November (31)

▶ October (44)

▶ September (40)

Technique

DermalMedix | Sponsored

Ad ▾

[Read Next Story >](#)

Weebly Customer Service

Email, Phone, and Live
Chat Available. Start Your
Store!

www.Weebly.com

43

Like

43

Share

Share

Tweet

3 Dangerous Foods People Feed Their Dogs (Without Realizing It)

Dr. Marty ProPower Plus Supplement | Sponsored

Plano, Texas Drivers Are Stunned By This New Rule

EverQuote Insurance Quotes | Sponsored

New Rule in Plano, Texas Leaves Drivers Fuming

Insured Nation - Auto Insurance Quotes | Sponsored

Drink This Before Bed, Watch Your Body Fat Melt Like Crazy

Health MD | Sponsored

By javin paul at [June 17, 2017](#) Labels: [Java multithreading](#) [Tutorials](#)

2 comments :

Army Veteran Discovers Skin Tightening

► August (51)

► July (55)

▼ June (56)

JSP - How to check if ArrayList is Empty using JST...

How to create Custom Exception in Java - Tutorial ...

Builder Design pattern in Java - Example Tutorial
Is Joshua Bloch's Effective Java Still Valid in th...10 xargs command example in Linux - Unix
tutorial2 Ways to check If String is Palindrome in Java?
R...10 Java Exception and Error Interview Questions
An...java.lang.ClassNotFoundException:
org.postgresql.D...How to use CopyOnWriteArraySet in Java with
Exempl...How to Escape JSON String in Java- Eclipse IDE
Tip...List of special bash parameter used in Unix or
Li...5 Books to Learn Object Oriented Programming
and D...What is -XX:+UseCompressedOops in 64 bit JVM
HashSet in Java - 10 Examples Programs TutorialHow to create HTTP Server in Java - ServerSocket
E...3 Exampels to Convert an Array to ArrayList in
Java...

JUnit4 Annotations : Test Examples and Tutorial

2 Ways to Parse CSV Files in Java -
BufferedReader...3 Ways to Find Duplicate Elements in an Array -
Ja...10 example of using Vim or VI editor in UNIX and
L...Java Lock and Condition Example using Producer
Con...3 ways to solve java.lang.NoClassDefFoundError
in ...How to Add Two Integer Numbers without using
Plus ...

Technique

DermalMedix | Sponsored

[Read Next Story >](#)

Otherwise get() would return an Object type and so the compilation errors will be shown at
 long factorialof10 = result10.get();

-Arun

June 19, 2015 at 2:07 AM[Shiva](#) said...

Thanks!!

Very helpful for me :)

January 11, 2018 at 10:07 PM**Post a Comment**

Enter your comment...



Comment as:

jeffrey.x.xu@gr ▼

[Sign out](#)[Publish](#)[Preview](#)☐ [Notify me](#)**Feeling Old? Do This Once A Day And Watch What Happens**

LCR Health | Sponsored

If Your Indoor Cat Vomits (Do This Every Day)

Ultimate Pet Nutrition | Sponsored

3 Ways Your Dog Asks For Help

Dr. Marty | Sponsored

Army Veteran Discovers Skin Tightening[How to Close Java Program or Swing Application wit...](#)[How to use Comparator and Comparable in Java? With...](#)[Top 5 Java And Android Game Programming Books for ...](#)[How to use Callable and Future in Java? Example](#)[How to Count number of Set bits or 1's of Integer ...](#)[JDBC Database Connection Pool in Spring Framework ...](#)[Maven - 10 Things Java developer should know](#)[How to Connect to MySQL database in Java with Exam...](#)[How to converts Java Object to XML - JAXB Example](#)[Is Java Compiled or Interpreted Programming langua...](#)[How to Generate MD5 checksum for Files in Java](#)[Jersey Web Service Hello World Example in Java](#)[How to compare two lists of values in Microsoft Ex...](#)[How to Read Write Excel file in Java - POI Example...](#)[Top 30 Programming questions asked in Interview - ...](#)[10 Interview Questions on Java Generics for Progra...](#)[How Spring MVC Framework works? How HTTP Request i...](#)[SynchronousQueue Example in Java - Producer Consume...](#)[java.lang.NullPointerException - Common Cause of N...](#)[Top 30 Array Interview Questions and Answers for P...](#)[10 Examples of Date Command in Linux and UNIX Exception in thread "main" java.lang.ExceptionInIn...](#)[How to get current URL, parameters and Hash tag us...](#)[How to get JAR files of Jackson Library for JSON Pr...](#)

Technique

DermalMedix | Sponsored

[Read Next Story >](#)**Don't Do This if You're Trying to Lose Weight****U.S. Doctor: High Blood Sugar? Do This Immediately**

Blood Sugar Stabilizer | Supplement | Sponsored

43

Like

43

Share

Share

Not Ready for Wrinkles? Put This On and Watch What Happens

Beverly Hills MD | Sponsored

Tweet

If You Have Gutters You Have To Try This To Avoid Cleaning Gutters For Life!

LeafFilter Partner | Sponsored

Yearbooks That Were Printed And Handed To The Whole School Before Teachers Saw The Photo Error

Livestly | Sponsored

Top 5 Difference Between Callable and Runnable Interface in Java

Java Revisited

How to use Counting Semaphore in Concurrent Java Application? Example Tutorial

Java Revisited

[Newer Post](#)[Home](#)[Older Post](#)Subscribe to: [Post Comments \(Atom\)](#)**Army Veteran Discovers Skin Tightening**

5 Reasons to Use Composition over Inheritance in J...

What is the cost of Spring Framework Certification...

How Volatile in Java works? Example of volatile ke...

Google Protocol Buffers (ProtoBuf) - Java Serializ...

2 Examples to read Zip Files in Java, ZipFile vs Z...

How to find CPU and Memory used by Java process in...

20 Design Patterns and Software Design Interview Q...

2 Books to Prepare for Spring Professional Certifi...

Difference between Inheritance and Composition in ...

► May (48)

► April (79)

► March (99)

► February (87)

► January (104)

► 2016 (117)

References

1. [Oracle's Java Tech Network](#)
2. [jQuery Documentation](#)
3. [Microsoft SQL Server Documentation](#)
4. [Java SE 8 API Documentation](#)
5. [Spring Documentation](#)
6. [Oracle's JAVa Certification](#)
7. [Spring Security 5 Documentation](#)

Pages[Privacy Policy](#)Copyright by Javin Paul 2010-2018. Powered by [Blogger](#).

Technique

DermalMedix | Sponsored

[Read Next Story >](#)

Search This Blog

43

Like

43

Share

Share

Search

Tweet

Army Veteran Discovers Skin Tightening