

Forum:[Hack Forums](#)

Thread: [Learning Lambda expression and Stream API](#)

Tekpe - February 5, 2017, 6:23 am

Hello,

Does anyone have any suggestion of good guide to learn lambda expressions and Stream API? I'm quite good with object oriented part of java and I would like to get more into functional programming part of it.

I learned java in college but mostly with help of youtube tutorials(video lectures make suit me best and I'd prefer something similar) but Lambda and Stream content on youtube is poor(or maybe it's just my searching skills)

Thanks in advance.

g4143 - February 5, 2017, 7:24 am

I started with this video..

<https://www.youtube.com/watch?v=a450CqNXFgs>

Tip: If you are serious about programming with lambdas and streams then try not to drag OOP style thinking into that realm because it will only confused you. OOP/imperative programming concentrate on CRUD(create, read, update and delete) style programming while Lambda/streams/functional style programming concentrates on expressions returning values. In functional style of programming you tend to return new collections instead of modifying existing data... This is not as efficient but it is much easier to reason about and produces much safer code.

Basically in a functional style of programming you start with a collection and perform operations on that collection that produce a new collection, a value or some side-effects. i.e.(pseudo code)

```
Code ArrayList<SomeType> col = new ArrayList<>();
/* load collection col with data */
```

```
/*perform operations on data to produce new collection*/
col.stream().map(e -> /*do something with element e and produce new collection*/).forEach(s ->
/*perform side-effect with s on new collection*/);
```

If you start getting hardcore with lambdas and streams then you can proceed to the Optional type and maybe the holy grail of functional programming ----<Monads>----.

<https://docs.oracle.com/javase/8/docs/api/java/util/Optional.html>

Note: Optional types are not that difficult. Optionals are just a type safe way of passing data that is optional. You can pass data that has no value or is empty. Monads are not quite as simple as optional types but they have generated quite a bit of hype in other functional languages... I won't worry about monads too much because they are only essential in languages like Haskell but every functional programmer should be aware of this term so they can sound hip and Uber.

Tekpe - February 5, 2017, 11:19 am

Thank you very much, OOP style thinking is my main problem in here, I used it on daily base for last

year or 2 so it's hard to get out of that OOP mode in my brain.

kane2026 - February 13, 2017, 1:30 am

I was recently assigned to a project at work that made a lot of use of Lambda expressions and Streams. After several weeks I left for another project because I couldn't wrap my head around it. Sure, the basics are easy enough to understand but the complexity in this particular project was very high. I highly recommend you learn all you can about functional programming in Java because as of Java 8 this stuff is not going away. Over the coming period I will also study it more and make use of it as much as I can.

g4143 - February 13, 2017, 4:49 am

(Feb 13, 2017 02:30 AM)kane2026 Wrote: [\[CITE\]](#) I was recently assigned to a project at work that made a lot of use of Lambda expressions and Streams. After several weeks I left for another project because I couldn't wrap my head around it. Sure, the basics are easy enough to understand but the complexity in this particular project was very high. I highly recommend you learn all you can about functional programming in Java because as of Java 8 this stuff is not going away. Over the coming period I will also study it more and make use of it as much as I can.

I found many languages like Java 8 and Scala just assume the programmer is familiar with functional concepts. This naive approach either leaves the newly exposed programmer to explore with a 'try and see what happens approach' or ignoring the new functionality all together. The Java crowd is lucky because the documentation/tutorials will eventually catch up with the new features. Its just a matter of time before most Java programmers are embracing lambdas and streams.

Tekpe - February 14, 2017, 5:34 am

Well, it has been 2(or 3) years since java 8 release and tutorials are still not here.

I thnk the best way to learn Stream API is to learn haskel or some functional language first and then just countinue with java.

g4143 - February 14, 2017, 6:45 am

(Feb 14, 2017 06:34 AM)Tekpe Wrote: [\[CITE\]](#) Well, it has been 2(or 3) years since java 8 release and tutorials are still not here.

I thnk the best way to learn Stream API is to learn haskel or some functional language first and then just countinue with java.

Before you go jumping into the world of Haskell I would try Scala first. Haskell is a very daunting and strange language for the uninitiated.