**Feature of FP**

1. Treats [computation](https://en.wikipedia.org/wiki/Computation) as the evaluation of [mathematical functions](https://en.wikipedia.org/wiki/Function_(mathematics)).
2. It avoids changing-[state](https://en.wikipedia.org/wiki/Program_state) and works on [mutable](https://en.wikipedia.org/wiki/Immutable_object) data.
3. It follows declarative programming style i.e program is done with expressions.
4. In functional code, the output value of a function depends only on the arguments that are input to the function, so calling a function twice with the same value produces the same result each time.

Java 8 supports functional programming through Lambda Expressions and Stream APIs. Java has “java.util.function” package which expose so many FP computation class and methods. But FP is more (Closure, Currying).

We can write Object Oriented Programming in C. But it is not best.

So we can use Java 8 as small subset of the power of functional languages along with OOPs, but if you are looking for core FP features then Java8 is not the right choice.