

5 functional programming languages ??you should know

If you take the time to learn programming trends on the Internet, you've probably heard of functional programming. Here are 5 functional programming languages ??you should know.

If you take the time to learn programming trends on the Internet, you've probably heard of functional programming. So what is functional programming? Even when understanding functional programming, do you know which programming language is best suited for it? Not all programming languages ??are the same, created for the same purpose. Although you can apply the functional programming model in many languages, there will be more appropriate languages. Here are 5 functional programming languages ??you should know.

1. Top 5 programming languages ??to develop AI
2. Top 5 languages ??for blockchain programming
3. The world's 5 most annoying 'programming languages'

What is functional programming?

If you have a math background, you will easily use functional programming because the functional programming model handles computation as mathematical functions. If you don't know anything about math, you will feel confused when programming functions.

Basically, functional functions treat functions and data as immutable objects. When using data in a function, it usually returns that data to be converted or another data type. In functional programming, the function will never change the original data or program state.

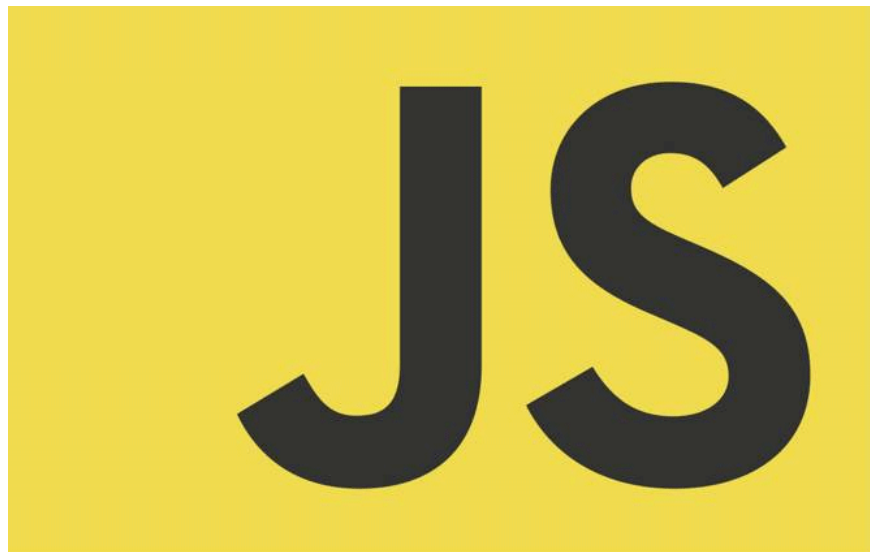
There is a Unix philosophy (Unix philosophy) that each program does a good job of it. So is the function, it will not touch other parts of your program. Instead, it will take its input and give you an output. Ideally in functional programming, when given the same input, it will produce the same output.

Functional and object-oriented programming comparisons

In object-oriented programming (Object-Oriented Programming), you often have a base object with many different methods to change data or status (which is part of the object). This method may change data or status if not specified.

This makes program maintenance more difficult because it is unclear whether the status and data have been changed. But with functional programming you will no longer worry about this problem.

1. JavaScript



JavaScript allows functional programming, you can use functional programming model that can use object-oriented method. There are many functional programming models built into JavaScript like the higher-order function. This is a function that can take another function as an argument.

In addition, JavaScript has some functions that work with arrays such as `map ()`, `reduce ()`, `filter ()`, etc., all of which are higher-order functions. This means you can string them together to quickly implement the methods in an array.

Although JavaScript initially had some problems with transformability, newer versions of the ECMAScript standard provided fixes. Instead of using the **var** keyword to define variables, you can use **const** and **let** . The first keyword allows you to define constants as implied names. The second keyword `let`, limits the scope of a variable in the function it declares.

2. Python



Like JavaScript, Python is a general language that you can use any programming model. Any language has pros and cons, but functional programming is not one of Python's disadvantages.

You can find many built-in functions in Python language such as `map ()`, `filter ()`, `reduce ()`, etc. . Just like JavaScript, these are all higher-order functions that can take other functions as an argument. In Python,

programming functions in the form of lambda keywords, this is an advantage over other languages.

You can use lambda expressions in a number of ways such as using it as shorthand (shorthand method) for simple functions. When assigned to a variable, you can call the exact lambda expression exactly like a standard Python function. The real advantage of lambda expressions is when using them as anonymous functions (anonymous functions).

Anonymous functions also work with JavaScript and other languages ??on this list. They are especially useful when used with higher-order functions because you can identify them on the spot. Without an anonymous function, you must specify before addition to make bespoke function.

3. Clojure



Clojure is a 'dialect' of Lisp programming language that appeared in the late 1950s. This is the perfect language for functional programming.

Like variants of Lisp language, Clojure handles code like data. This means that the code can change itself effectively. However, Clojure can run on the Java platform and is compiled into JVM bytecode. Therefore it can work with Java libraries whether or not they are written in Clojure.

Unlike other languages ??in the list, Clojure is a functional programming language, ensuring immutability in data structures.

4. Elm



One of the new languages on this list, Elm is the pure functional programming language designed by Evan Czaplicki in 2012. This language has become popular among web developers, especially for the purpose. create user interface.

Unlike the languages on this list, Elm uses static type checking, ensuring that no exception runtime error occurs, catching an error while compiling. Users will see fewer errors, this is a big plus for this language.

The compiler Elm uses HTML, CSS and JavaScript. Just like the way Clojure writes programs running on Java, you can write applications that use the JavaScript library with Elm.

One key difference between Elm and other languages is that you won't find filter (), map () and similar functions. Instead, they are identified by data types like List.map or Dict.map.

5. Haskell



Haskell is a functional programming language, using static testing. Unlike Elm, Haskell has been around for a while. The first version of the language was designed in 1990. The latest standard is Haskell 2010 and the next version is expected to be released in 2020.

Haskell is a language designed for functional programming, so it is suitable for solving real-world problems even though the source of functional programming is in academia.

Haskell is used in widely used projects such as the Xmonad window manager written entirely in Haskell and Pandoc that convert different markup types into other formats that also use this language. You will find some standard functions like `map ()`, `filter ()`, `reduce ()` and other higher-order functions.

Some of the above terms and languages ??seem a bit confusing if you are not an experienced programmer. That's good, because discovering new things is one of the first steps to becoming a good learner. You can refer to the article [Beginners of computer programming need to focus on what?](#)

You finished reading the article "**5 functional programming languages ??you should know**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.