Getting Started with Elixir

João Gonçalves

Video 1.3

Functional Programming

In this video, we are going to take a look at...

- Defining functional programming
- Discussing motivation
- Exploring characteristics and traits

What is Functional Programming?

- Evaluation of expressions
- Promotes immutable state

What is Functional Programming?

- Behaves more closely to their mathematical counterpart
- Is a first-class citizen (i.e. can be returned from functions and passed as argument)

Function

$$f(x) \rightarrow y$$

Motivation

- Better structuring discipline
- No side-effects reduces number of bugs
- Suitable for parallelism

Characteristics

- Higher order functions:
  - Functions as return values and as parameters to other functions

$$f(g(x)) \rightarrow z(x)$$

- Recursion:
  - Allowing a function to call itself or looping algorithms

$$\begin{cases} f(x) \rightarrow x + f(x-1) \\ f(1) = 1 \end{cases}$$

Characteristics

- Referential Transparency:
  - Same evaluation = same outcome

$$y = f(x)$$

$$g(y, y) \Leftrightarrow g(f(x), f(x))$$

Summary

- Explored history of Elixir and its heritage
- Defined Elixir as a language
- Discussed the impact of Elixir and when to use it
- Understood functional programming mindset