**Elixir – Paradigma funcional**

**Conceito**

Inicial da história do conceito de programação funcional começou em lambda.

A programação funcional no elixir não é um código matemático absoluto com apenas funções recursivas ou cheio de caracteres desconhecidos e aglomerados.

O principal objetivo da programação funcional é compor funções reduzidas e imutáveis e funções que mantêm seu estado e não se alteram. Um modelo de programação no elixir é utilizar o pipe operador, que ajuda muito na hora de construir uma função com imutabilidade.

**Diferenciação e comparação**

A diferença principal entre programação funcional e orientação a objetos é que na orientação a objetos, temos na chamada um objeto e respectivamente seu método, já na programação funcional temos um modulo e respectivamente suas funções.

**Imutabilidade**

Ao passar um valor para uma variável é necessário instanciar novamente pois os valores são armazenados em posições de memória diferentes, fazendo com que não seja armazenado o estado da função. Isso serve para que quando for realizar procedimentos assíncronos não seja pesado ou dê algum problema pois geralmente nesses casos não há garantia da integridade do valor salvo em memória.

**Aprendizado**

Programação funcional aparece em outras linguagens de uma forma diferente, mas existe, como no caso de um map, filter ou reduce em javascript, onde se passa uma função dentro de outra função, recebendo o nome de função de alta ordem.

Referência de video

https://youtu.be/53Lv3efp7Rk