



# **Programação Funcional com Elixir**



# **Underscore e Pin Operator**

# Pattern Matching

- Correspondência de Padrões



# Underscore e Pin Operator

- Ainda sobre Pattern Matching, imagine que temos a seguinte situação...
  - `{x, y} = {32, 25}`
- Até aí tudo bem, mas e se não quisermos o valor do `y`? Neste caso seremos obrigado a informar uma variável?
- A resposta para isso é o **underscore** “\_”

# Underscore e Pin Operator

- Sempre usaremos o underscore “\_” quando não nos importarmos com o valor. Veja o exemplo:
  - `{x, _} = {32, 25}`
- Assim, apenas o **x** vai apontar para o 32 enquanto ou outro valor pra gente não importa.
- Em resumo o underscore age como uma variável que descarta o valor logo depois de “atribuída”.

# Underscore e Pin Operator

- Agora vamos falar sobre o pin operator.
- Veja esse exemplo...
  - `iex(1) > x = 21`
  - `21`
  - `iex(2) > x = 43`
  - `43`
- Notadamente o x começou apontando para 21 e em seguida foi reassociado (rebind) para o 43.
- Mas, e se a gente não quisesse permitir essa reassociação?

# Underscore e Pin Operator

- É aí que entra o pin operator “^”.
- O uso do pin operator é justamente impedir a reassociação de variáveis. Veja:
  - `iex(1)> x = 34`
  - `34`
  - `iex(2)> ^x = 45`
  - `** (MatchError) no match of right hand side value: 45`
- Perceba que colocando o “^” antes da variável o Elixir levantou um erro informando que o 45 não “casa” com o valor atual do **x**, ou seja, o pin operator evitou o rebind.

# Underscore e Pin Operator

- O legal do pin operator é que podemos usá-lo em conjunto com o pattern matching, fazendo com que seja evitado novos rebinds que não desejamos. Veja:

- `iex(1)> {x,y} = {76, 89}`
- `{76, 89}`
- 
- `iex(2)> x`
- `76`
- 
- `iex(3)> y`
- `89`
- 
- `iex(4)> {x, ^y} = {12, 67}`
- `** (MatchError) no match of right hand side value: {12, 67}`



# Underscore e Pin Operator

- Perceba que ao usar o pin operator o Elixir vai “criticar” e verificar se os valores casam. Ou seja, o pin operator evita o rebind ao tempo que faz a correspondência de padrões.

Veja:

- 
- `iex(4) > {x, ^y} = {12, 89}`
- `{12, 89}`
- 
- `iex(5) > x`
- `12`
- 
- `iex(6) > y`
- `89`