

## **Atividade 006.2 - Exercício 19**

Crie 4 exercícios e suas respostas com o que aprendeu até agora sobre Haskell.

**Ex. 1) Faça um programa em Haskell que calcule a média de 3 números.**

```
Prelude> media x y z = (x + y + z) / 3.0  
Prelude> media 2 5 9  
5.333333333333333
```

```
Prelude> media x y z = (x + y + z) / 3.0  
Prelude> media 2 5 9  
5.333333333333333
```

**Ex. 2) Faça um programa em Haskell que dada uma entrada numérica do teclado x, calcule  $x*7+8$ ;**

```
Prelude> do { x <- readLn; print (x*7+8) }  
2  
22
```

```
Prelude> do { x <- readLn; print (x*7+8) }  
5  
43
```

**Ex. 3) Qual a diferença entre tuplas e listas em Haskell? Dê exemplos.**

Tuplas e Listas são mecanismos que o Haskell oferece para a construção de dados compostos.

Em tuplas podemos combinar os componentes de um dado em uma única estrutura, e os componentes podem ter tipos e propriedades distintas.

Exemplos de tuplas:

```
(True, 7, 5)  
("Haskell", False)  
(0, 15, "Oito", True, 'j')  
((6,'w'),  
[7,8,9])
```

A lista possibilita a reunião de vários elementos – todos do mesmo tipo em uma única estrutura.

Exemplos de listas:

```
Prelude> False : []  
[False]
```

```
Prelude> 'a':['b','c','d']  
"abcd"
```

**Ex. 4) Para quê serve a função “zip” do Haskell, e dê um exemplo.**

Zip é uma função da biblioteca do Haskell que transforma duas listas em uma lista de tuplas.

Exemplo:

```
Prelude> zip [4 .. 9] ['a' .. 'f']  
[(4,'a'),(5,'b'),(6,'c'),(7,'d'),(8,'e'),(9,'f')]
```