



7. Listas

7.1 Conceito no algoritmo

Vamos iniciar voltando ao exemplo dos descontos de imposto do salário usamos os exemplos com apenas 2 funcionários com as variáveis “Salário_Final_1” e “Salário_Final_2”. Para apenas 2 funcionários não foi nenhum problema, certo?

Agora, imagine que na próxima empresa quer iremos executar este cálculo existam 400 funcionários. Não daria para criarmos variáveis de “Salário_Final_1”, “Salário_Final_2” até “Salário_Final_400”... Seria impossível!

Para resolver este problema, temos um tipo de variável que pode armazenar vários valores: são as **listas**, também conhecidas como **array** ou **vetores**. Todos os valores são separados por vírgula e ficam entre colchetes ou parêntesis. Elas funcionam da seguinte forma:

```
Lista_Funcionários = ["Bruno", "Luiz", "Ana", "João", "Moisés"]
```

Para armazenar vários valores numa única variável, cada item da lista recebe um índice, um número de posição na lista, sempre iniciando em zero. Por exemplo:

Lista_Funcionários =	0	1	2	3	4
	Bruno	Luiz	Ana	João	Moisés

Para resgatar um valor, usamos estes índices que identificam sua posição. Por exemplo, para o nome “João” usamos da seguinte forma:

```
Nome = Lista_Funcionários[3]
```

As listas podem ser de um único tipo, como no exemplo acima que é uma lista de nomes (strings), ou pode ser uma lista de valores numéricos ou qualquer outro tipo. Uma outra possibilidade é uma lista de valores mistos, armazenando textos, números, valores booleanos, todos misturados.



Vejamos um exemplo:

Lista_dados =	0	1	2	3	4
	Bruno	3,14	False	João	863

Quando estávamos aprendendo sobre variáveis, vimos que os tipos comuns (texto, números e lógicos) são chamados de ***tipos primitivos***, lembra? As listas são de um tipo mais recente e complexo, que são controlados pela própria linguagem de programação. Por isso, a ***manipulação*** de seus dados pode ser realizada de forma diferenciada de uma linguagem para outra, mas todas seguem os mesmos conceitos.

Vamos conhecer agora os principais conceitos de manipulação de listas, como a ***criação*** (que já vimos), a ***adição*** e a ***remoção***.

[1] **//criação e atribuição**

[2] Lista_Funcionários = ["Bruno", "Luiz", "Ana"]

[3]

[4] **//adição de novo valor após a criação da lista**

[5] Lista_Funcionários.Add("Moisés")

[6]

[7] **//removendo o segundo item da lista, "Luiz" (2º porque inicia em zero ;-)**

[8] Lista_Funcionários.Remove[1]

[9]

[10] **//imprime os nomes**

[11] print (Lista_Funcionários[0]) //imprime "Bruno"

[12] print (Lista_Funcionários[1]) //imprime "Ana"

[13] print (Lista_Funcionários[2]) //imprime "Moisés"

Agora vamos trabalhar com listas na prática.