



5. Variáveis e Constantes

5.1 Conceito no Algoritmo

Vamos conhecer agora sobre as variáveis. Na verdade, já conhecemos, utilizamos as variáveis nos exemplos de algoritmos que usamos até agora, mas chamamos de nome de valor. Vamos rever nosso último exemplo de algoritmo:

```
[1] Salário = 1000
[2] Tem_Filhos = FALSE
[3] SE Tem_Filhos
[4]     ENTÃO
[5]         Imposto = 50
[6]     SENÃO
[7]         Imposto = 100
[8] FIM SE
[9] Salário_Final = Salário - Imposto
```

As duas primeiras linhas do algoritmo são variáveis. As *variáveis* são locais onde podemos armazenar valores e, como o nome diz, podem ser alterados ao longo do algoritmo, ou seja, podem ter seus valores variados, podem ter variação. Como exemplo, temos as linhas 5 e 7, que definem o valor do imposto como 50 e 100, respectivamente. Finalizamos o algoritmo com uma última variável, que armazena os valores da subtração entre Salário e Imposto, na linha 9.

As variáveis têm este nome pois há um outro tipo de armazenamento de valores que não permitem a mudança de seu valor ao longo do código, são os chamados *constantes*. A constante não permite alterar seu valor depois que definido, mantendo os valores constantes, conforme seu nome. No algoritmo acima, na linha 1, o local de armazenamento de “Salário” poderia ser uma constante.

Como podemos ver no código acima, temos 2 variáveis nas linhas 1 e 2, a primeira com o valor numérico **1000** e a segunda com o valor lógico **FALSE**. Isto mostra que as variáveis podem armazenar vários tipos de dados. Vamos conhecer agora os tipos de dados com o qual podemos trabalhar:



- **Integer (inteiro):** são os valores numéricos inteiros, sem vírgula decimal, tais como 10, 99, 1000, 9990 etc, assim como valores negativos, como -10, -999 etc.
- **String (texto):** são os valores compostos por letras ou números, escrito entre aspas, tais como “Roberto”, “Vôo 172”, “Hello World!” etc.
- **Boolean (booleano ou lógico):** podem assumir apenas 2 valores: TRUE ou FALSE, como já conhecemos.
- **Float (real ou ponto flutuante):** são os valores numéricos não inteiros, com casas decimais, tais como 10.98, 99.9, 1.0000098 etc, assim como seus valores negativos.

Existem tipos de dados são chamados de **tipos primitivos**, isto significa que seu funcionamento e uso será exatamente o mesmo em todas as linguagens de programação.

Os valores armazenados nas variáveis podem ser inicializados, atribuídos ou requisitados. Vamos repetir o código e, em seguida, entender a diferença entre estes tipos:

```
[1] Salário = 1000
[2] Tem_Filhos = FALSE
[3] SE Tem_Filhos
[4]   ENTÃO
[5]     Imposto = 50
[6]   SENÃO
[7]     Imposto = 100
[8] FIM SE
[9] Salário_Final = Salário – Imposto
```



- **Inicialização (*inicialize*):** inicializações são atribuições que ficam no início do código, assim como as linhas 1 e 2 de nosso exemplo. São pré-requisito para o funcionamento do restante do algoritmo.
- **Atribuição (*set*):** as atribuições acontecem sempre que um valor é atribuído à uma variável. Em nosso exemplo acima, as atribuições ocorrem nas linhas 1, 2, 5, 7 e 9.
- **Requisição (*get*):** a requisição acontece sempre que um valor armazenado dentro de uma variável é necessário em outro trecho do código. Em nosso exemplo, houve requisições na linha 3 (para verificar se “Tem_Filhos” é TRUE) e na linha 9 (“Salário” e “Imposto” foram requisitados para calcular o valor do salário final do funcionário).

Na próxima aula vamos entender como estes conceitos são aplicados no App Inventor.