

# Funções como expressões

Uma característica importante da programação funcional é que a própria **definição de função** pode ser encarada como **um valor**, mais precisamente, **uma expressão**.

Como consequência, definição de **funções, valores e expressões** passam a ser **tratados de forma equânime** dentro da linguagem em questão, compartilhando de propriedades absolutamente semelhantes.

Isso "eleva" de fato o conceito de função para o mesmo nível de um valor ou expressão, ou seja, o mais importante nível de representatividade dentro da programação, os chamados CIDADÃOS DE PRIMEIRA CLASSE.

Isso é possível separando-se a *nomenclatura da função* da *expressão que representa o mapeamento domínio  $\rightarrow$  imagem*.



**[EXEMPLO]** Função que relaciona um valor ao seu quadrado.

$$f(x) = x^2$$

Nome da função:  $f$

Expressão de mapeamento:  $(x) \rightarrow x^2$



**[EXEMPLO]** Função que relaciona três valores à média ponderada.

$$h(x, y, z) = \frac{(2x+3y+5z)}{10}$$

Nome da função:  $h$

Expressão de mapeamento:  $(x, y, z) \rightarrow \frac{(2x+3y+5z)}{10}$



**[EXEMPLO]** Função que calcula a área de uma elipse.

$$f_{areaEli}(x, y) = \pi xy$$

Nome da função:  $f_{areaEli}$

Expressão de mapeamento:  $(x, y) \rightarrow \pi xy$

---