

Próximo:[Macros](#), Anterior:[Variáveis](#), Acima:[Topo](#) [[Conteúdo](#)][[Índice](#)]

## 13 Funções

Um programa Lisp é composto principalmente de funções Lisp. Este capítulo explica o que são funções, como elas aceitam argumentos e como defini-las.

- [O que é uma função](#) Funções Lisp versus primitivas; terminologia.
- [Expressões lambda](#) Como as funções são expressas como objetos Lisp.
- [Nomes de Função](#) Um símbolo pode servir como o nome de uma função.
- [Definindo Funções](#) Expressões Lisp para definir funções.
- [Funções de Chamada](#) Como usar uma função existente.
- [Funções de mapeamento](#) Aplicando uma função a cada elemento de uma lista, etc.
- [Funções Anônimas](#) As expressões lambda são funções sem nomes.
- [Funções genéricas](#) Polimorfismo, estilo Emacs.
- [Células de Função](#) Acessando ou configurando a definição de função de um símbolo.
- [Fechamentos](#) Funções que encerram um ambiente léxico.
- [Funções de Aconselhamento](#) Adicionando à definição de uma função.
- [Funções Obsoletas](#) Declarando funções obsoletas.
- [Funções Inline](#) Funções que o compilador expandirá inline.
- [Formulário de Declaração](#) Adicionando informações adicionais sobre uma função.
- [Declarando Funções](#) Dizendo ao compilador que uma função está definida.
- [Segurança da Função](#) Determinando se uma função é segura para chamar.
- [Tópicos Relacionados](#) Referências cruzadas para primitivos Lisp específicos que têm uma influência especial sobre como as funções funcionam.

Próximo:[Macros](#), Anterior:[Variáveis](#), Acima:[Topo](#) [[Conteúdo](#)][[Índice](#)]

