Próximo:Criando símbolos, Anterior:Componentes de Símbolos, Acima:Símbolos [Conteúdo][Índice]

9.2 Definindo Símbolos

Uma *definição* é um tipo especial de expressão Lisp que anuncia sua intenção de usar um símbolo de uma maneira particular. Normalmente, especifica um valor ou significado para o símbolo para um tipo de uso, além de documentação para seu significado quando usado dessa maneira. Assim, ao definir um símbolo como uma variável, você pode fornecer um valor inicial para a variável, além de documentação para a variável.

`defvar` `defconst`são formas especiais que definem um símbolo como uma *variável global* — uma variável que pode ser acessada em qualquer ponto de um programa Lisp. Consulte Variáveis , para obter detalhes sobre variáveis. Para definir uma variável personalizável, use a `defcustom`macro, que também chama `defvar`como sub-rotina (consulte Personalização).

Em princípio, você pode atribuir um valor de variável a qualquer símbolo com `setq`, independentemente de ter sido definido ou não como uma variável. No entanto, você deve escrever uma definição de variável para cada variável global que deseja usar; caso contrário, seu programa Lisp pode não agir corretamente se for avaliado com escopo léxico ativado (consulte Variable Scoping).

`defun`define um símbolo como uma função, criando uma expressão lambda e armazenando-a na célula de função do símbolo. Esta expressão lambda torna-se assim a definição de função do símbolo. (O termo “definição de função”, significando o conteúdo da célula de função, é derivado da ideia que `defun`dá ao símbolo sua definição como função.) `defsubst` `defalias`são duas outras formas de definir uma função. Consulte Funções .

`defmacro`define um símbolo como uma macro. Ele cria um objeto de macro e o armazena na célula de função do símbolo. Observe que um determinado símbolo pode ser uma macro ou uma função, mas não ambas ao mesmo tempo, porque as definições de macro e função são mantidas na célula de função e essa célula pode conter apenas um objeto Lisp em um determinado momento. Consulte Macros .

Como observado anteriormente, o Emacs Lisp permite que o mesmo símbolo seja definido como uma variável (por exemplo, com `defvar`) e como uma função ou macro (por exemplo, com `defun`). Tais definições não entram em conflito.

Essas definições também funcionam como guias para ferramentas de programação. Por exemplo, os comandos `C-h f` e `C-h v`criam buffers de ajuda contendo links para a variável, função ou definições de macro relevantes. Veja a Ajuda de Nome no Manual do GNU Emacs .

Próximo:Criando símbolos, Anterior:Componentes de Símbolos, Acima:Símbolos [Conteúdo][Índice]