

Próximo: [Variáveis locais](#), Anterior: [Variáveis globais](#), Acima: [Variáveis](#) [[Conteúdo](#)][[Índice](#)]

12.2 Variáveis que nunca mudam

No Emacs Lisp, certos símbolos normalmente avaliam a si mesmos. Estes incluem `nil` e `t`, bem como qualquer símbolo cujo nome comece com `'` (estes são chamados de palavras-*chave*). Esses símbolos não podem ser recuperados, nem seus valores podem ser alterados. Qualquer tentativa de definir ou vincular `nil` ou `t` sinaliza um `setting-constant` erro. O mesmo vale para uma palavra-chave (um símbolo cujo nome começa com `'`), se estiver internado no obarray padrão, exceto que definir esse símbolo para si mesmo não é um erro.

```
nil => 'nil
      => nada
(configuração zero 500)
erro→ Tentativa de definir símbolo constante: nil
```

Função: *objeto* keywordp

função retorna `t` se o *objeto* é um símbolo cujo nome começa com `'`, internado no obarray padrão, e retorna `nil` outra forma.

Essas constantes são fundamentalmente diferentes das constantes definidas usando a `defconst` forma especial (consulte [Definindo Variáveis](#)). Um `defconst` formulário serve para informar aos leitores humanos que você não pretende alterar o valor de uma variável, mas o Emacs não gera um erro se você realmente alterá-lo.

Um pequeno número de símbolos adicionais torna-se somente leitura por vários motivos práticos. Estes incluem `enable-multibyte-characters`, `most-positive-fixnum`, `most-negative-fixnum`, e alguns outros. Qualquer tentativa de definir ou vincular isso também sinaliza um `setting-constant` erro.