

变量

aborn

2018-05-18

Contents

1 变量	1
2 全局变量	1
2.1 不变量	1
2.2 定义全局变量	2
3 变量是无效的	2
4 局部变量	2
5 Buffer 本地变量 (Buffer-Local Variables)	3
6 文件本地变量 (File-Local Variables)	3

1 变量

变量在程序中是一种标识符，其指向某个值，这是一个很通用的编辑概念，不需要做过多解析。在 Lisp 中，每个变量都通过符号类型来表达。变量名就是对应的符号类型名，变量值是保存在符号类型的值单元 (value cell) 里。注意在前一章节里，我们介绍到符号类型既可用于变量名也可用于函数名，它们是相互独立不冲突的。

2 全局变量

全局变量在任意时刻都只有一个值，并且这个变量可使用于整个 lisp 运行环境。我们经常用 `setq` 这个特殊表达式将一个值绑定到具体某个符号变量中，如下：

```
(setq x '(a b))
```

这里 `setq` 是一个特殊表达式，所以它不会对第一个参数 `x` 进行求值，它会对第二个参数进行求值，然后将求得的值绑定到第一个参数对应的变量中。

2.1 不变量

在 Emacs Lisp 中，某些符号类型的求值是其本身。最常见的如 `nil`、`t`，以及以: 开头的符号类型（这些符号类型称之为关键字 `keywords`）。这些特殊的变量不能再进行绑定，同时其值也无法进行修改。

```
(keywordp object)
```

用来判断一个对象是否为关键字类型 (`keywords`)，即以: 开头的符号类型。

2.2 定义全局变量

变量的定义主要有三个目的：首先，它提示阅读代码人的，该符号变量用于一种特殊用途（这里用于变量）；其次，它提供给 Lisp 系统，有时候还会赋予初始值和文档；最后，它为编程工具提供信息，如 `etags`，提示它去哪里找到变量定义。定义全局变量采用 **`defvar`** 关键字，它定义一个符号类型为变量。

```
defvar symbol [value [doc-string]]
```

还有一种方式，采用 **`defconst`** 关键字

```
defconst symbol value [doc-string]
```

3 变量是无效的

当一个符号类型对应的值单元没有被赋值 (`unassigned`) 时，称对应的变量为无效的 (`void`)。对于个无效变量进行求值，会抛出 **`void-variable error`**。注意变量为无效的 (`void`) 与变量值为 `nil` 本质上是不一样的，`nil` 为一种对象，它可以赋值给变量。

```
(makunbound symbol)
```

`makunbound` 清空符号类型里值单元，使得一个变量成为无效的 (`void`)。它返回符号类型。

```
(boundp variable)
```

`boundp` 当 `variable` 不是无效的 (`nil`) 时返回 `t`，否则返回 `nil`。

4 局部变量

局部变量一般只用于一段程序，最常用的声明方式是采用 **let** 关键字。它的定义格式如下：

```
let (bindings. . . ) forms. . .
```

这里的 `let` 是一个特殊表达式 (Special Forms)，它按顺序绑定局部变量。下面是一个例子：

```
(let ((y 1)
      (z y))
  (list y z)) ;; (1 2)
```

还有一种局部变量，即函数的调用参数，因为这些参数只用于函数调用阶段。

5 Buffer 本地变量 (Buffer-Local Variables)

Buffer 本地变量，从字面意思可以看出，这种类型的变量只应用于 Buffer 中。这种机制可以满足对于同一个变量在不同的 Buffer 中的值不一样。

6 文件本地变量 (File-Local Variables)