15. 定义语法

1. 简介

本章中，我会讲解如何自定义语法。用户定义语法称作宏（Macro）。Lisp/Scheme中的宏比C语言中的宏更加强大。宏可以使你的程序优美而紧凑。

宏的代码的转换。代码在被求值或编译前进行转换，and the procedure continues as if the transformed codes are written from the beginning.

你可以在Scheme中通过使用符合R5RS规范的syntax-rules轻易地定义简单宏，相比之下，在Common Lisp中自定义语法就复杂多了。使用syntax-rules可以直接定义宏而不用担心变量的捕获（Variable Capture）。On the other hand, defining complicated macros that cannot be defined using the syntax-rules is more difficult than that of the Common Lisp.

1. 实例：简单宏

我将以一个简单的宏作为例子。

[代码片段 1] 一个将变量赋值为’()的宏

[代码片段 1]

(define-syntax nil!

(syntax-rules ()

((\_ x)

(set! x '()))))

syntax-reuls的第二个参数