

形式化语法

此处显示的形式语法和随附文本描述了 Scheme 数据值或基准面的书面语法。语法还有效地涵盖了方案句法形式的书面语法，因为每个方案语法形式都有一个表示为方案基准。特别是，括号内的句法形式写成列表，标识符（例如，关键字和变量）写成符号。每个句法形式的高级结构由第4章至第11章中标记为“语法”的条目详细描述，句法形式在形式摘要中总结。

基准的书面表示形式包括标记、空格和注释。标记是表示原子基准或用作标点符号的一个或多个字符的序列。表示原子基准的标记是符号、数字、字符串、布尔值和字符，而用作标点符号的标记是左括号和右括号、左括号和右括号、左向量括号 # ()、开向量括号 #vu8 (、虚线对标记。(点)、引号 ' 和 '、非引号 , 以及 , @, 语法引号 #' 和 #'，语法取消引号将标记 #和 #, @。

空格由空格、制表符、换行符、换行符、回车符和下一行字符以及 Unicode 标准 [30] 分类为 Z、Zl 或 Zp 的任何其他字符组成。换行符也称为换行符。某些空格字符或字符序列用作行尾，这些行尾被识别为行注释和字符串语法的一部分。行尾是换行符、下一行字符、行分隔符、后跟换行符的回车符、后跟下一行字符的回车符或不后跟换行符或下行字符的回车符。一组不同的 whitespace 字符用作行内空白，这些空格被识别为字符串语法的一部分。行内空白区域包括空格、制表符以及常规类别为 Zs 的任何其他 Unicode 字符。行内空白字符和行尾的集合是不相交的，并且还有其他空格字符（如表单源）不在两个集中。

注释有三种形式：行注释、块注释和基准注释。行注释由分号 (;) 后跟任意数量的字符组成，直到输入的下一行结束或结束。块注释由 #| 前缀、任意数量的字符和嵌套块注释以及|# 后缀组成。基准注释由 #; 前缀后跟任何基准组成。

符号、数字、字符、布尔值和虚线对标记 (.) 必须由输入、空格、注释开头、左括号或右括号、左括号或右括号、字符串引号 (“) 或哈希标记 (#) 分隔。任何标记都可以在前面或后面加上任意数量的空格字符和注释。

大小写在字符、字符串和符号的语法中很重要，但在十六进制标量值中除外，其中十六进制数字“a”到“f”可以用大写或小写形式书写。（十六进制标量值是十六进制数，表示 Unicode 标量值。大小写在布尔值和数字的语法中无关紧要。例如，Hello 与 hello 不同，#\A 与 #\a 不同，“String”与“string”不同，而 #T 等效于 #t，#E1E3等效于 #e1e3，#X2aBc等效于 #x2abc，#\x3BA 等效于 #\x3ba。

不允许对修订的⁶报告的一致性实现扩展基准面的语法，但有一个例外：允许将任何以前缀 #! 开头的标记识别为标志，该标志在标志后面的文本中指示某些扩展是有效的。因此，例如，实现可能会识别标志 #! 大括号，并切换到一种模式，在这种模式下，列表可以括在大括号以及括号和括号中。

`#! braces ' {a b c} ⇒ (a b c)`

标志 #! r6rs 可用于声明后续文本是用 R6RS 语法编写的。最好在包含可移植库或顶级程序的任何文件的开头包含 #! r6rs，以指定正在使用 R6RS 语法，以防将来的报告以与库或程序文本不兼容的方式扩展语法。#! r6rs 否则被视为注释。

在下面显示的语法中，<空>代表空字符序列。后跟星号（*）的项表示该项的零个或多个匹配项，后跟加号（+）的项表示一个或多个匹配项。生产中物料之间的间距仅出于可读性考虑，应将其视为不存在。

基准面。基准面是布尔值、字符、符号、字符串、数字、列表、矢量或字节向量。

<基准> → <布利安>

- | <特征>
- | <同义词>
- | <弦>
- | <编号>
- | <列表>
- | <向量>
- | <字节向量>

列表、向量和字节向量是由可能由空格和注释分隔的标记组组成的复合基准面。其他的都是单个令牌。

布尔值。布尔值 `false` 写 `#f`。虽然所有其他值都算作 `true`，但规范 `true` 值（以及只有被布尔值？谓词视为布尔值的其他值）`#t` 写入。

〈布利安〉 $\rightarrow \#t \mid \#f$

大小写在布尔值的语法中并不重要，因此这些大小写也可以写成 `#T` 和 `#F`。

字符。字符对象被编写为前缀 `#\`，后跟单个字符、字符名称或指定 Unicode 标量值的字符序列。

〈特征〉 $\rightarrow \#\backslash \langle \text{一个角色} \rangle \mid \#\backslash \langle \text{字符名称} \rangle \mid \#\backslash x \langle \text{十六进制} \rangle$

〈角色名称〉 $\rightarrow \text{报警} \mid \text{退格} \mid \text{删除} \mid \text{esc} \mid \text{换行} \mid \text{换行符} \mid \text{第} \mid \text{页返回} \mid \text{空间} \mid \text{选项卡} \mid \text{vtab}$

〈十六进制标量值〉 $\rightarrow \langle \text{数字 } 16 \rangle^+$

命名字符对应于 Unicode 字符警报（Unicode 标量值 7，即 `U+0007`）、退格（`U+0008`）、删除（`U+007F`）、`esc`（`U+001B`）、换行符（`U+000A`；与换行符相同）、换行符（`U+000A`）、页面（`U+000C`）、返回（`U+000D`）、空格（`U+0020`）、制表符（`U+0009`）和垂直制表符（`U+000B`）。

十六进制标量值表示 Unicode 标量值 n 或 $0 \leq n \leq D800_{16}$ $E000_{16} \leq n \leq 10FFF_{16}$ 。〈数字 16〉非终端定义在下面的数字下。

`#\` 前缀后跟字符名称始终被解释为命名字符，例如，`#\page` 被视为 `#\page` 而不是 `#\p`，后跟符号 `age`。如上所述，还必须对字符进行分隔，以便将 `#\pager` 视为语法错误，而不是字符 `#\p` 后跟符号 `ager` 或字符 `#\page` 后跟符号 `r`。

大小写在字符对象的语法中很重要，十六进制标量值中除外。

字符串。字符串被编写为括在字符串引号（双引号）中的字符串元素序列。字符串引号或反斜杠以外的任何字符都可以显示为字符串

元素。字符串元素还可以由反斜杠后跟单个字符、反斜杠后跟指定 Unicode 标量值的字符序列或反斜杠后跟包含单行结尾的行内空白字符序列组成。

⟨弦⟩ → “⟨字符串⟩”

⟨弦元素⟩ → ⟨除 “ 或 \⟩

| \” | \\ | \a | \b | \f | \n | \r | \t | \v
 | \x <十六进制标量值>;
 | \<内线空格>* <内线结尾> <内空>*

由单个字符组成的字符串元素表示该字符，但表示行尾的任何单个字符或一对字符表示单个换行符。反斜杠后跟双引号表示双引号，而反斜杠后跟反斜杠表示反斜杠。反斜杠后跟 a 表示警报字符（U+0007）；由 b，退格（U+0008）；通过 f， 形成进给（U+000C）；由 n，换行符（U+000A）；按 r，回车符（U+000D）；按 t， tab（U+0009）；和 v，垂直选项卡（U+000B）。后跟 x 的反斜杠、十六进制标量值和分号（；）表示由标量值指定的 Unicode 字符。⟨十六进制标量值⟩非终端在上述“字符”下定义。最后，由反斜杠后跟行内空白（包括单行结尾）组成的字符序列不表示任何字符。

大小写在字符串的语法中很重要，十六进制标量值除外。

符号。符号要么写成“初始”字符，后跟一系列“后续”字符，要么写成“奇特符号”。初始字符是字母、某些特殊字符、一组附加的 Unicode 字符或由 Unicode 标量值指定的任意字符。后续字符包括初始字符、数字、某些附加特殊字符和一组附加 Unicode 字符。特殊符号是 +、-、...，以及任何以 -> 为前缀的后续字符序列。

⟨同义词⟩ → ⟨初始⟩ <附属>*

⟨初始⟩ → ⟨信⟩ | ! | \$ | % | | * | / | : | < | = | > | ? | ~ | _ | ^
 | <Unicode Lu, Ll, Lt, Lm, Lo, Mn, Nl, No,
 | \x <十六进制标量值>;

⟨实质⟩ → ⟨初始⟩ | <数字 10> | 。 | + | - | @ | <Unicode Nd, Mc或Me>

⟨信⟩ → — | b | ... | z | | B | ... | 兹

⟨Unicode Lu, Ll, Lt, Lm, Lo, Mn, Nl, No, Pd, Pc, Po, Sc, Sm, Sk, So, or Co⟩ 表示 Unicode 标量值大于 127 且其 Unicode 类别是所列类别之一的任何字符。⟨Unicode Nd、Mc 或 Me⟩ 表示 Unicode 类别是所列类别之一的任何字符。⟨十六进制标量值⟩非终端定义在上面的“字符”下定义，⟨数字 10⟩在下面的“数字”下定义。

大小写在符号中很重要。

数字。数字可以出现在以下四个基数之一中：2、8、10 和 16，默认值为 10。下面的几个作品由基数 r 参数化，每个作品代表四个作品，四个可能的基数各一个。包含基数点或指数的数字被限制为以基数 10 显示，因此⟨十进制 r ⟩仅在 r 为 10 时才有效。

⟨编号⟩	→⟨2⟩ ⟨8⟩ ⟨10⟩ ⟨16⟩
⟨编号 r ⟩	→⟨前缀 r ⟩ ⟨复合体 r ⟩
⟨前缀 r ⟩	→⟨半径 r ⟩ ⟨精确性⟩ ⟨精确性⟩ ⟨半径 r ⟩
⟨半径 2⟩	→#b
⟨半径 8⟩	→#o
⟨半径 10⟩	→⟨空⟩ #d
⟨16⟩	→#x
⟨精确性⟩	→⟨空⟩ #i #e
⟨复式 r ⟩	→⟨现实 r ⟩ ⟨现实 r ⟩ @ ⟨现实 r ⟩ ⟨现实 r ⟩ + ⟨imag r ⟩ ⟨现实 r ⟩ - ⟨imag r ⟩ +⟨imag r ⟩ - ⟨imag r ⟩
⟨现实 r ⟩	→⟨标志⟩ ⟨脉⟩ +nan.0 -南.0 +inf.0 -inf.0
⟨imag r ⟩	→i ⟨脉⟩ i inf.0 i nan.0 i
⟨脉⟩	→⟨uinteger r ⟩ ⟨uinteger r ⟩ / ⟨uinteger r ⟩ <+>
⟨uinteger r ⟩	→⟨数字 r ⟩ ⁺
⟨十进制 10⟩	→⟨第10⟩ ⟨后缀⟩ . ⟨数字 10⟩ ⁺ ⟨后缀⟩ ⟨数字 10⟩ ⁺ . ⟨数字 10⟩* ⟨后缀⟩
⟨后缀⟩	→⟨外⟩ ⟨曼蒂萨宽度⟩
⟨爆炸⟩	→⟨空⟩ ⟨外部标记⟩ ⟨符号⟩ ⟨数字 10⟩ ⁺ →

<爆炸标记> e | s | f | d | l
 <曼蒂萨宽度> → <空> || <数字 10>⁺
 <签名> → <空> | + | -
 <数字2> → 0 | 1 个
 <数字 8> → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 6
 <数字 10> → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 10
 <数字 16> → <数字 10> | - | b | c | d | e | f

如果按 #i 为前缀，或者没有以 #e 为前缀，并且包含小数点、非空指数或非空尾数宽度，则如上所述编写的数字是不准确的。否则，它是准确的。

大小写在数字的语法中并不重要。

列表。列表是由标记组组成的复合基准面，可能涉及其他基准面，包括其他列表。列表写成括号或括号内的基准序列；作为非空基准序列、虚线对标记（ . ）和括在括号或括号内的单个基准；或作为缩写。

<列表> → (<基准>*) | [<基准>*]
 | (<基准>⁺ <基准>) | [<基准>⁺ <基准>]
 | <缩写>
 <缩写> → ' <基准> | ' <基准> | 、 <基准> | , @ <基准>
 | # ' <基准> | # ' <基准> | #, <datum> | #, @ <基准>

如果用括号或方括号括起来的列表中没有出现虚线对标记，则该列表是正确的列表，基准是列表中的元素，按给定的顺序排列。如果出现虚线对标记，则列表的初始元素是标记之前的元素，标记后面的基准是列表的尾部。通常，仅当标记后面的基准本身不是列表时，才使用虚线对标记。虽然任何正确的列表都可以在没有虚线对标记的情况下编写，但可以通过在点对标记之后放置列表来用点对符号编写正确的列表。

缩写等效于下面显示的相应双元素列表。一旦阅读了缩写，结果就与其非缩写形式无法区分。

```

'<datum> → (引用 <datum>)
'<datum> → (quasiquote <datum>)
,<datum> → (unquote <datum>)
,@<datum> → (unquote-splicing <datum>)
#'<datum> → (syntax <datum>)
#'<datum> → (quasisyntax <datum>)
#,<datum> → (unsyntax <datum>)
#,@<datum> → (非语法拼接<基准>)

```

矢量。向量是由标记组形成的复合基准面，可能涉及其他基准面，包括其他向量。向量写为开向量括号，后跟一系列基准面和一个右括号。

<向量> → #(<基准>*)

字节向量。字节向量是由标记组组成的复合基准面，但语法不允许它们包含任意嵌套基准。字节向量写为开字节向量括号，后跟一系列八位字节（无符号的 8 位精确整数）和一个右括号。

<字节向量> → #vu8 (<八月>*)

<八月> → <任何<数字>表示一个精确的整数 n , $0 \leq n \leq 255$ >

R. Kent Dybvig / The Scheme Programming Language, Fourth Edition

Copyright © 2009 The MIT Press. 经许可可以电子方式复制。

插图 © 2009 让-皮埃尔·赫伯特

ISBN 978-0-262-51298-5 / LOC QA76.73.S34D93

订购本书 / 关于这本书

<http://www.scheme.com>