

语法

本章描述了本规范中用于定义程序词法和语法结构的上下文无关语法。

2.1 上下文无关语法

与上下文无关的语法由许多产品组成。每个产品的左手边都有一个抽象符号，称为非终止符，右手边是一个或多个非终止符和终止符的序列。对于每种语法，终止符号都是从指定的字母表中提取的。

从一个由单个可区分的非终止符(称为目标符号)组成的句子开始，给定的与上下文无关的语法指定了一种语言，即，可能的终止符序列集是从重复地将序列左手系产品的非终止符替换成右手系产品的非终止符产生的，

2.2 词法语法

Java 语言的词法语法在§3（词法结构）介绍。这种语法的终止符号是 Unicode 字符集的字符。它定义了一组产品，从目标符号 Input(§3.5)开始，描述 Unicode 字符序列(§3.1)如何被转换为输入元素(§3.5)。

这些输入元素去掉了空格(§3.6)和注释(§3.7)，成为 Java 编程语言语法的终止符号，被称为标记(§3.5)。这些标记就是 Java 编程语言的标识符(§3.8)、关键字(§3.9)、字面量(§3.10)、分隔符(§3.11)和操作符(§3.12)。

2.3 句法语法

Java 编程语言的句法语法在第 4、6-10、14 和 15 章中给出。这种语法的终止符号是由词法语法定义的标记。它定义了一组产品，从目标符号 CompilationUnit(§7.3)开始，描述了标记序列如何形成语法正确的程序。

为了方便起见，句法语法在第 19 章中介绍。

2.4 语法符号

在词法和语法的产品中，终止符号以固定宽度的字体显示，并且在整个规范中，只要文本直接引用这样的终止符号，就这样显示。这些将完全按照所写的出现在程序中。

非终止符以斜体显示。非终止符的定义由定义的非终止符的名称和冒号引入。一个或多个非终止符的可选定义紧随其后。

例如，语法产品：

IfThenStatement:
if (*Expression*) *Statement*

表明非终止符 *IfThenStatement* 代表标记 *if*，后跟左圆括号标记，然后是 *Expression*，然后是右圆括号标记，最后是 *Statement*。

产品右边的语法{x}表示 x 出现 0 次或多次。

例如，语法产品：

ArgumentList:
Argument {, *Argument*}

表明 *ArgumentList* 由一个 *Argument* 组成，后面跟着零次或多次出现的逗号和一个 *Argument*。结果是 *ArgumentList* 可以包含任何正数的参数。

产品右边的语法[x]表示 x 出现了零次或一次。也就是说，x 是一个可选的符号。包含可选符号的选项实际上定义了两个选项：一个省略可选符号，另一个包含可选符号。

这意味着：

BreakStatement: *break* [*Identifier*] ;

是以下语句的缩写：

BreakStatement: *break* ; *break Identifier* ;

另一个例子是这样的：

BasicForStatement:
for ([*ForInit*] ; [*Expression*] ; [*ForUpdate*]) *Statement*

是以下语句的缩写：

BasicForStatement:
for (; [*Expression*] ; [*ForUpdate*]) *Statement*
for (*ForInit* ; [*Expression*] ; [*ForUpdate*]) *Statement*

又是以下的缩写：

BasicForStatement:
for (; ; [*ForUpdate*]) *Statement*
for (; *Expression* ; [*ForUpdate*]) *Statement*
for (*ForInit* ; ; [*ForUpdate*]) *Statement*
for (*ForInit* ; *Expression* ; [*ForUpdate*]) *Statement*

也是以下的缩写:

```
BasicForStatement:
for ( ; ; ) Statement
    for ( ; ; ForUpdate ) Statement
    for ( ; Expression ; ) Statement
    for ( ; Expression ; ForUpdate ) Statement
    for ( ForInit ; ; ) Statement
    for ( ForInit ; ; ForUpdate ) Statement
    for ( ForInit ; Expression ; ) Statement
    for ( ForInit ; Expression ; ForUpdate ) Statement
```

所以非终止符 **BasicForStatement** 实际上有八个右边选项（可以展开成 8 种形式）。

很长的右边选项可以通过明显的缩进在第二行继续。

例如，语法包含这个产品:

```
NormalClassDeclaration:
{ClassModifier} class TypeIdentifier [TypeParameters] [Superclass] [ClassImplements]
[ClassPermits] ClassBody
```

它为非终止符 **NormalClassDeclaration** 定义了一个右边选项。

产品右手边的短语(one of)表示下面一行或几行上的每个符号都是一个可选定义。

例如，词法语法包含产品:

```
ZeroToThree:
(one of) 0 1 2 3
```

这仅仅是一个方便的缩写:

```
ZeroToThree:
0
1
2
3
```

当产品中的一个替代出现为一个标记时，它表示将组成该标记的字符序列。

因此，产品:

```
BooleanLiteral: (one of) true false
```

是以下的缩写:

```
BooleanLiteral: true false
```

产品的右手边可以使用短语“but not”来指定不允许某些扩展，然后指示要排除的扩展。

例如:

Identifier:

IdentifierChars but not a *ReservedKeyword* or *BooleanLiteral*
or *NullLiteral*

最后, 一些非终结符由一个罗马类型的叙述短语定义, 列出所有的选项是不现实的。

例如:

RawInputCharacter:

any Unicode character

qingliu