

语言按语法分类

- 非语言语法
 - 中文，英文
- 形式语言（乔姆斯基谱系）
 - 0型 无限制文法
 - 1型 上下文相关文法 词、句组合与上下文相关
 - 2型 上下文无关文法 词、句组合与上下文无关，放在任何位置都可以
 - 3型 正则文法

0-3 上包含下

产生式（BNF）

- 用尖括号括起来的名词用来表示语法结构名
- 语法接口分成基础结构和需要用其他语法结构定义的复合结构
 - 基础结构称终结符
 - 复合结构称非终结符
- 引号和中间的字符表示终结符
- 可以有括号
- *表示重复多次
- |表示或
- +表示至少一次

语言的分类

- 形式语言——用途
 - 数据描述语言
 - 编程语言

例： JSON, HTML, XAML, SQL, CSS //数据描述语言

例2： C C++ Java C# Python Ruby Perl Lisp T-SQL Clojure Haskell
JavaScript //编程语言

- 形式语言——表达方法
 - 声明式语言
 - 命令式语言

例：JSON, HTML, XAML, SQL, CSS Lisp Clojure Haskell //声明式语言

例2：C C++ Java C# Python Ruby Perl T-SQL JavaScript //命令式语言

动态与静态

- 动态：
 - 在用户设备/在线服务器上
 - 产品实际运行时
 - Runtime
- 静态：
 - 在程序员的设备上
 - 产品开发时
 - Compiletime

乔姆斯基谱系：是计算机科学中刻画形式文法表达能力的一个分类谱系，是由诺姆·乔姆斯基于 1956 年提出的。它包括四个层次：

0- 型文法（无限制文法或短语结构文法）包括所有的文法。

1- 型文法（上下文相关文法）生成上下文相关语言。

2- 型文法（上下文无关文法）生成上下文无关语言。

3- 型文法（正规文法）生成正则语言。

巴科斯诺尔范式：即巴科斯范式（英语：Backus Normal Form，缩写为 BNF）是一种用于表示上下文无关文法的语言，上下文无关文法描述了一类形式语言。它是由约翰·巴科斯（John Backus）和彼得·诺尔（Peter Naur）首先引入的用来描述计算机语言语法的符号集。

图灵机（Turing machine）：又称确定型图灵机，是英国数学家艾伦·图灵于 1936 年提出的一种将人的计算行为抽象掉的数学逻辑机，其更抽象的意义为一种计算模型，可以看作等价于任何有限逻辑数学过程的终极强大逻辑机器。

图灵完备性：在可计算性理论里，如果一系列操作数据的规则（如指令集、编程语言、细胞自动机）可以用来模拟单带图灵机，那么它是图灵完全的。这个词源于引入图灵机概念的数学家艾伦·图灵。虽然图灵机会受到储存能力的物理限制，图灵完全性通常指“具有无限存储能力的通用物理机器或编程语言”。

终结符：最终在代码中出现的字符（<https://zh.wikipedia.org/wiki/終結符與非終結符>）

产生式：在计算机中指 Tiger 编译器将源程序经过词法分析（Lexical Analysis）和语法分析（Syntax Analysis）后得到的一系列符合语法规则（Backus-Naur Form, BNF）的语句

静态和动态语言：<https://www.cnblogs.com/raind/p/8551791.html>

强类型：无隐式转换

弱类型：有隐式转换

协变与逆变：<https://jkchao.github.io/typescript-book-chinese/tips/covarianceAndContravariance.html>