

# 编程语言

## 动态/静态类型

### 动态类型语言

- 定义 运行期间才做数据类型检查的语言，即动态类型语言编程时，永远不用给任何变量指定数据类型。该语言会在第一次赋值给变量时，在内部将数据类型记录下来。
- 典型 Python、Ruby、PHP、ECMAScript(JavaScript)、VBScript

### 静态类型语言

- 定义 编译期间做检查数据类型的语言，即写程序时要声明所有变量的数据类型，是固定的。使用数据之前，必须先声明数据类型（int ,float,double等）。
- 典型 C、C++、C#、JAVA

### 区别

变量类型是否是在编译时确定的

## 强定义/弱定义

### 强定义类型定义语言

- 是否安全 类型安全
- 定义 强制数据类型定义的语言。也就是说，一旦一个变量被指定了某个数据类型，如果不经强制转换，那么它就永远是这个数据类型了。
- 典型 C/C++/Java/C#
- 举例说明 若定义了一个整型变量a，若不进行显示转换，不能将a当作字符串类型处理。强类型语言是指需要进行变量/对象类型声明的语言，一般情况下需要编译执行。

### 弱定义类型定义语言

- 是否安全 类型不安全
- 定义 数据类型可以被忽略的语言。它与强类型定义语言相反，一个变量可以赋不同数据类型的值。
- 典型 PHP/ASP/Ruby/Python/Perl/ABAP/SQL/JavaScript/Unix Shell
- 举例说明 在VBScript中，可以将字符串 '12' 和整数 3 进行连接得到字符串 '123'，然后可以把它看成整数 123，而不需要显示转换

### 区别

能不能将一种类型当作另一种类型（已经存在转换关系的不算）

### 利弊

【强类型定义语言】在速度上可能略逊色于【弱类型定义语言】  
但是，【强类型定义语言】带来的严谨性能够有效的避免许多错误。

## 总结

### 举例

Python是动态语言，是强类型定义语言（类型安全的语言）；  
VBScript是动态语言，是弱类型定义语言（类型不安全的语言）；  
JAVA是静态语言，是强类型定义语言（类型安全的语言）。

### 联系

“这门语言是不是动态语言”与“这门语言是否类型安全”之间是完全没有联系的！