# 标识符 & 保留字

## 标识符：

### 1、定义：

Java所有的组成部分都需要名字。类名、变量名以及方法名都被称为标识符。

### 标识符命名规范：

1）所有的标识符都应该以字母（a-z或者A-Z），美元符号（$），或者下划线（\_）开头。

2）首字符之后可以是字母（a-z或者A-Z），美元符号（$），下划线（\_）或数字（0-9）的任意字符组合。

3）关键字不能用作标识符。

4）标识符是大小写敏感的。

5）合法标识符举例：age、$salary、\_value、\_1\_value

6）非法标识符举例：123abc，-salary

## （关键字）：

### 定义：

具有专门的意义和用途，不能当做一般的标识符使用，这些标识符被称为保留字，也称为关键字

### 2、关键字列表：

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **关键字** | **描述** | | abstract | 抽象方法，抽象类的修饰符 | | assert | 断言条件是否满足 | | boolean | 布尔数据类型 | | break | 跳出循环或者label代码段 | | byte | 8-bit 有符号数据类型 | | case | switch语句的一个条件 | | catch | 和try搭配捕捉异常信息 | | char | 16-bit Unicode字符数据类型 | | class | 定义类 | | const | 未使用 | | continue | 不执行循环体剩余部分 | | default | switch语句中的默认分支 | | do | 循环语句，循环体至少会执行一次 | | double | 64-bit双精度浮点数 | | else | if条件不成立时执行的分支 | | enum | 枚举类型 | | extends | 表示一个类是另一个类的子类 | | final | 表示一个值在初始化之后就不能再改变了 表示方法不能被重写，或者一个类不能有子类 | | finally | 为了完成执行的代码而设计的，主要是为了程序的健壮性和完整性，无论有没有异常发生都执行代码。 | | float | 32-bit单精度浮点数 | | for | for循环语句 | | goto | 未使用 | | if | 条件语句 | | implements | 表示一个类实现了接口 | | import | 导入类 | | instanceof | 测试一个对象是否是某个类的实例 | | int | 32位整型数 | | interface | 接口，一种抽象的类型，仅有方法和常量的定义 | | long | 64位整型数 | | native | 表示方法用非java代码实现 | | new | 分配新的类实例 | | package | 一系列相关类组成一个包 | | private | 表示私有字段，或者方法等，只能从类内部访问 | | protected | 表示字段只能通过类或者其子类访问 子类或者在同一个包内的其他类 | | public | 表示共有属性或者方法 | | return | 方法返回值 | | short | 16位数字 | | static | 表示在类级别定义，所有实例共享的 | | strictfp | 浮点数比较使用严格的规则 | | super | 表示基类 | | switch | 选择语句 | | synchronized | 表示同一时间只能由一个线程访问的代码块 | | this | 表示调用当前实例 或者调用另一个构造函数 | | throw | 抛出异常 | | throws | 定义方法可能抛出的异常 | | transient | 修饰不要序列化的字段 | | try | 表示代码块要做异常处理或者和finally配合表示是否抛出异常都执行finally中的代码 | | void | 标记方法不返回任何值 | | volatile | 标记字段可能会被多个线程同时访问，而不做同步 | | while | while循环 | |

## 保留字：

现在还没有使用，但是以后可能使用的关键字。