

匈牙利命名法

匈牙利命名法 (Hungarian notation) 是电脑程序设计中的一种变量命名规则。在匈牙利命名法中，一个变量名由一个或多个小写字母开始，这些字母有助于记忆变量的类型和用途，紧跟着的就是程序设计师选择的任何名称。这个**后半部分的首字母可以大写**，以区别前面的类型指示字母此命名法又可细分为：**系统匈牙利命名法**和**匈牙利应用命名法**。

- **系统匈牙利命名法和匈牙利应用命名法的区别**

两者的区别在于**前缀的目的**

- **系统命名法前缀的目的**：前缀代表变量的实际数据类型

- `lAccountNum`：变量为长整数
- `arru8NumberList`：变量为无符号8位整型数组
- `szName`：变量为零结束字符串

- **应用命名法前缀的目的**：

- `rwPosition`：变量为一个行地址
- `usName`：变量为非安全字符串
- `strName`：变量为一个字符串（包含名字）

- **匈牙利命名法的优点**

- 在不同集成开发环境中，可以直接通过名字看出变量的类型和作用
- 可以使决定变量名的过程更加快捷
- 方便修改一系列同一类型的变量

- **匈牙利命名法的缺点**

- 变量前面的变量类型或许是多余的，若要查找变量名字，可以直接转到变量定义模块（Go to definition）
- 同时纵容相同主体名而不同类型的变量导致的歧义，开发人员无法从 `swidth`、`nwidth`、`fwidht` 中了解这几个Width的用法区别

思考：

很多时候我们的目的是“简化”，而看上去匈牙利命名法并没有做到这一点，`lAccountNum`、`arru8NumberList` 这些名字和 `a`、`b` 这些变量名比起来长太多了。但是当程序足够长，或者需要完成一个项目的时候，更长、更详细的命名能够增加代码的可阅读性，不至于忘记 `a`、`b` 代表的东西。

参 考 资 料

[1] https://en.wikipedia.org/wiki/Hungarian_notation