

# 面向对象的 允替语言程序设计

孟祥武摇张玉洁摇编著

北京邮电大学出版社  
· 北京 ·

## 内 容 提 要

本书介绍了 ~~Visual Basic~~ 语言的基本语法和程序设计方法,全书共 ~~15~~章,分别介绍了 ~~Visual Basic~~ 语言的背景、数据类型、运算符、控制流、数组、面向对象程序设计、异常处理、线程、~~Visual Basic~~ 图形用户界面、输入输出处理和网络通信。

本书可作为高等学校理工科学生教材或教学参考书,也可作为工程技术人员的培训教材或自学参考书。

### 图书在版编目(CIP)数据

面向对象的语言程序设计 蒋摇贵,张玉洁编著. —北京:北京邮电大学出版社, ~~2005~~

摇 ISBN ~~7-5306-2842-2~~

摇 I 摇 蒋摇贵 II 摇 孟摇爱 III 摇 蒋摇贵 IV 摇 计算机语言—程序设计—教材—中国

摇 中国版本图书馆 CIP 数据核字 (~~2005~~) 第 ~~0467~~ 号

---

书 名:面向对象的 ~~Visual Basic~~ 语言程序设计

责任编辑:蒋摇贵

出版者:北京邮电大学出版社(北京市海淀区西土城路 ~~10~~号)

邮编:~~100876~~ 电话:~~010-62080068~~

网址:~~http://www.bjptupress.com.cn~~

经 销:各地新华书店

印 刷:北京源海胶印厂

开 本:~~787mm×1092mm~~ 1/16 印张: ~~15.5~~ 字数: ~~380~~ 千字

版 次:~~2005~~ 年 ~~1~~ 月第 ~~1~~ 版 ~~2005~~ 年 ~~1~~ 月第 ~~1~~ 次印刷

书 号:~~ISBN 7-5306-2842-2~~ 源 苑

定 价:~~29.80~~ 元

---

# 前摇摇言

计算机的主要用途将从单机应用转向网络应用,这是一大趋势,不可逆转。网络就是计算机,网络是网络上的世界语。它已成为国际通用网络编程语言事实上的标准。它是一种跨平台的、适合于网络计算环境的面向对象程序设计语言。网络语言的基础是面向对象语言,但去除了其中易出错部分。

对于网络语言程序,首先由网络语言编译器将源程序编译为字节码,字节码是一种中间码,这种中间码由网络虚拟机解释和执行。

网络语言具有简单性、面向对象、分布性、编译和解释性、鲁棒性、安全性、中性的体系结构、可移植性、高性能、多线程和动态性等特点。

网络语言源于失败的消费电子类技术,然后在一个流产的系统中为计算机提供交互视频信号,最终在 1995 年再次革新,并于 1996 年 1 月推出。

网络语言自从 1996 年问世以来,很快就流行于全世界,并获得了极大地成功,甚至改变了计算模式,在世界范围内,引起了一股网络热。

网络语言是一种面向对象的程序设计语言,特别适合于国际通用网络上的应用软件开发。

目前,越来越多的软件开发项目选择网络语言作为编程语言,特别是有关计算机网络方面的软件,所以越来越多的人开始学习并使用网络语言,因此我们编写了这本教材。

本书共 10 章,第 1 章简要介绍了网络语言产生的背景、发展与现状、特点、开发环境。第 2 章讲述了数据类型、标识符、关键字、注释语句等。第 3 章讲述了运算符和优先级。第 4 章讲述了控制流,主要是分支语句和循环语句。第 5 章讲述了数组,包括一维数组和二维数组。第 6 章首先讲述了面向对象程序设计的主要概念,然后具体讲述了网络语言的类、继承、对象、接口、包、修饰符。第 7 章讲述了异常和异常处理过程。第 8 章讲述了线程,包括线程的创建、状态、调度、同步。第 9 章首先讲述了网络语言和网络语言的区别,然后讲述了网络语言和网络语言。第 10 章讲述了图形用户界面,主要是网络语言的事件处理机制和网络语言的组成。第 11 章讲述了输入输出处理,包括流、网络语言和网络语言和网络语言和网络语言。第 12 章讲述了网络通信,主要是网络语言和网络语言和网络语言和网络语言,并给出了一些实例。

本书的作者从 1995 年起一直从事有关网络语言的科研、开发、教学、培训等工作。

本书是作者在从事有关网络语言科研和教学的基础上编写出来的,本书作为一本初学者的入门教材,通俗易懂、简明扼要。

希望本书的编写能对广大初学者有所帮助。因作者水平有限,书中如有不妥之处,敬此为试读,需要完整PDF请访问: [www.ertongbook.com](http://www.ertongbook.com) · 1 ·

请广大读者批评指正。

感谢丁宇新博士、冀振燕博士为我们收集、提供了部分资料。感谢中国科学院软件研究所博士生导师程虎研究员的支持与关心。

最后衷心感谢北京邮电大学白中英教授的关心。

孟祥武摇张玉洁  
北京邮电大学计算机科学与技术学院  
二〇〇四年 八月

# 目摇摇录

第一章 摇摇语言的背景 .....	员
摇摇语言的发展与现状 .....	员
摇摇语言的特点 .....	圆
摇摇语言开发环境 .....	源
摇摇语言程序实例 .....	缘
第二章 摇摇数据类型 .....	怨
摇摇标识符 .....	怨
摇摇语言关键字 .....	圆
摇摇注释语句 .....	员
摇摇数据类型 .....	员
摇摇各类数值型数据间的混合运算 .....	缘
第三章 摇摇运算符 .....	苑
摇摇 运算符和优先级 .....	苑
摇摇 算术运算符 .....	愿
摇摇 关系运算符 .....	圆
摇摇 布尔逻辑运算符 .....	员
摇摇 位运算符 .....	圆
摇摇 赋值运算符 .....	圆
摇摇 条件运算符 .....	圆
摇摇 其它 .....	圆
第四章 摇摇控制流 .....	圆
摇摇 分支语句 .....	圆
摇摇 循环语句 .....	猿
第五章 摇摇数组 .....	源
摇摇 一维数组 .....	源
摇摇 多维数组 .....	源



---

第十二章 网络通信.....	12
网络基础.....	12
网络系统通信.....	12
网络数据报通信.....	12

# 第一章 摇佬语言背景

本章主要讲述 摇佬语言的发展与现状、摇佬语言的特点及开发环境 ,并结合一些简单的实例使读者对 摇佬语言有个初步的认识。

## 员 摇佬语言的发展与现状

计算机的主要用途将从单机应用转向网络应用 ,这是一大趋势 ,不可逆转。网络就是计算机 ,摇佬是网络上的世界语。它已成为 网络编程事实上的标准。它是一种跨平台的、适合于网络计算环境的、面向对象的程序设计语言。

最初 ,杂士希望进入消费类电子产品市场。对于消费类电子产品 ,从商业上考虑 ,要求 悦裁要轻便小巧 ,软件可靠 ,价格合理等 ,最重要的是能够标准化。这就要求编程语言简明、可靠。杂士专门成立了一个小组开始寻找满足条件的编程语言 ,小组负责人是 晃碧。他们认为 :目前已存在的程序设计语言 ,没有能满足要求的。所以要设计一种新的程序设计语言。

考虑到 悦旦语言是当前广泛使用的语言 ,因此 ,他们以 悦旦语言为基础 ,设计出了一种新的程序设计语言 晃碧。

晃碧语言的基础是 悦旦语言 ,但去除了其中易出错的部分。

晃碧是 杂士公司为家电编程而设计的一种语言 ,简单紧凑 ,而且适应不同的 悦裁。在晃碧语言开发完成之时 ,正赶上 网络大潮 ,晃碧语言建立的“虚拟运行环境”解决了跨平台问题。晃碧语言于 晃碧年 缘月推出 ,获得了极大的成功 ,甚至改变了计算模式 ,在世界范围内 ,引起了一股 晃碧热。

最近几年 ,在 网络上出现的特别吸引人的事件 ,就是 晃碧语言和用 晃碧语言编写的浏览器 晃碧。

对于 晃碧语言程序 ,首先由 晃碧语言编译器将源程序编译为字节码 ,字节码是一种中间码 ,这种中间码由 晃碧虚拟机解释和执行。

晃碧语言的诞生必将对整个计算机产业产生深远的影响 ,对传统的计算模型提出了新的挑战。

有人预言 :晃碧语言将是网络上的“世界语” ,今后所有用其它语言编写的软件统统都要用 晃碧语言来改写。

晃碧语言的出现 ,将会引起一场软件革命 ,这是因为传统的软件往往都是与具体的实现环境有关 ,换了一个环境就需要做一番改动 ,耗时费力 ;而用 晃碧语言编写的程序能运行在不同的机器上 ,只要所用的机器能提供 晃碧语言解释器即可。

晃碧语言将对未来软件的开发产生很大的影响 ,这可从以下几个方面考虑 :

(员) 软件的需求分析。可以将用户的需求进行动态地可视化描述 ,以提供设计者更

加直观的要求。用户的需求是各色各样的,受地区、行业、部门、爱好的影响。用 允槽语言都可以描述清楚。

(圆) 软件的开发方法。由于 允槽语言的面向对象的特性,所以完全可以用面向对象的技术与方法来开发,这是符合最新的软件开发规范要求的。

(猿) 允槽语言的动画效果远比 赋斌技术更加逼真,尤其是利用 宰宰宰提供的巨大动画资源空间,可以共享全世界动态画面的资源。

(源) 软件最终产品。用 允槽语言开发的软件可以具有可视化、可听化、可操作化和可交互性。对于动画与动作,要它停就停,要它继续就继续,而这是在电影与电视播放过程中难以做到的。

(缘) 其它。使用 允槽语言对开发效益和开发价值都有比较明显的影响。

允槽语言程序和支持它的浏览器(如 匀兑槽宰宰宰糟素的 宰宰宰浏览器以上版本, 配部糟兑槽宰宰宰糟素的 宰宰宰浏览器以上版本)的出现正是为了给 宰宰宰浏览提供动画(粤兑槽宰宰宰)、速度(杂兑槽宰宰)、交互性(附兑槽宰宰宰)。允槽语言能从你的浏览器里直接播放声音,播放页面里的动画,还能告诉你的浏览器怎样处理新的类型文件。

允槽语言有着广泛的应用前景,大体上可以从以下几个方面来考虑其应用:

- (员) 所有面向对象的应用开发,包括面向对象的事件描述、处理、综合等;
- (圆) 计算过程的可视化、可操作化的软件的开发;
- (猿) 动态画面的设计,包括图形、图像的调用;
- (源) 交互操作的设计(选择交互、定向交互、控制流程等);
- (缘) 附兑槽宰宰宰的系统管理功能模块的设计,包括 宰宰宰页面的动态设计、管理和交互操作设计等;
- (远) 附兑槽宰宰宰的企业内部网)上的软件开发(直接面向企业内部用户的软件);
- (苑) 与各类数据库连接查询的 杂兑槽宰宰宰语句实现;
- (愿) 其它应用类型的程序。

## 员兑槽允槽语言的特点

允槽语言具有简单性、面向对象、分布性、编译和解释性、健壮性、安全性、中性的体系结构、可移植性、高性能、多线程和动态性等特点。它可在消费类电子产品、个人计算机、网络计算机、工作站和主机上使用。

允槽语言不仅能创建安全的能在网络上使用的应用程序,而且还改变了人们对未来计算机的看法。

允槽语言是一种很有潜力的程序设计语言。随着计算机网络的普及,允槽语言会变得越来越重要。

允槽语言的特点主要是:

(员) 简单性

语言本身的组成成份较少,结构较小。与已有语言类似,用户容易熟悉掌握。允槽语言把 悦兑槽宰宰宰中一般程序员很少使用的、容易出错的特征加以剔除。允槽语言略去了运算

符重载、多重继承等模糊的概念,并且通过实现自动无用单元收集,大大简化了程序设计师的内存管理工作。

系统简单,它的基本解释器及类的支持只有 100K 左右,加上标准类库和线程的支持也只有 100K 左右。

猿语言的创造者希望通过创建一门简单的语言,以鼓励大家编写简单易懂的程序。各类猿语言程序便于在小型机器上执行,通过网络也容易下载。

#### (圆) 面向对象

猿语言从基础上就被设计成面向对象的,不能在类外面定义单独的数据和函数。所有对象都派生自同一个基类,并共享它所有的功能。

也就是说,猿语言最外部的数据类型是对象。猿语言中不允许有独立的变量、常量或者函数。所有的元素都要通过类和对象来访问。

#### (猿) 分布性

猿语言从设计上就支持网络应用,通过它提供的类库可以处理 网络协议,用户可以通过 地址在网络上很方便地访问其它对象,它是分布式语言。

#### (源) 编译和解释性

猿语言编译器把猿语言源程序编译成猿字节码。在不同的平台上,用该平台上的猿语言解释器来执行猿字节码。通过编译和解释两步实现了跨平台。字节码本身携带了许多编译信息,使得连接过程更加简单。

#### (缘) 鲁棒性

猿语言是一个强类型语言,它要求以显示的方法声明,它不支持指针。猿语言还具有无用单元回收,异常处理等功能。猿解释程序也执行许多运行时的检查。用猿语言可编写具有鲁棒性的软件。

#### (远) 安全性

猿语言实现了几种安全机制,保护用户免遭病毒和恶意代码的袭击。程序员永远不需要显式地释放一块和对对象相关联的内存。收集无用单元(由猿实现)是猿内建的机制。猿语言不允许对指针进行算术运算,具有很高的安全性。

#### (苑) 中性的体系结构

猿语言编译程序生成的字节码与体系结构无关。猿字节码适合各种体系结构。猿语言程序能在任何系统上运行,只要该系统实现了猿虚拟机(猿)。这些字节码指令对应于猿虚拟机中的表示,猿解释器得到字节码后,对它进行转换,使之能够在不同的平台上运行。

#### (愿) 可移植性

中性体系结构是确保可移植性的重要组成部分。猿语言规范中没有与机器相关的特性。猿语言的基本数据类型(如浮点数和整数)的范围是确定的,不依赖于具体平台的实现。猿语言环境本身也是可移植的。猿语言编译器是用猿语言编写的,解释器是用悦语言编写的。猿的类库中也实现了与不同平台的接口,使这些类库可以移植。

#### (怨) 高性能

和其它解释执行的语言(如 月粤院)不同, 编程语言字节码的设计使其很容易直接转换成对应于特定 悦哉的机器码或汇编码,从而得到较高的性能。

#### (员) 多线程性

编程语言对多线程提供支持。多线程机制使应用程序能够并行,同步机制可以保证对共享数据的正确操作。

通过使用多线程,程序设计者可以分别用不同的线程完成特定的行为,带来了更好的交互能力和实时运行能力。

#### (员) 动态性

编程语言设计成适应于变化的环境, 编程中的类是根据需要而装入的。在类库中可以自由地加入新的方法和实例变量而不会响应用户程序的执行。并且 编程语言通过接口来支持多重继承,使之比严格的类继承具有更灵活的方式和扩展性。

总之, 编程是一种程序设计语言,但它不仅仅是一种程序设计语言。

#### (员) 编程首先是一种面向对象的程序设计语言。

(圆) 编程是 悦哉上的通用语言,也是 悦哉上第一个实际应用的语言,它获得了广泛的支持。

(猿) 编程是一种编程环境,它包括一个相当规范、较大的类库。这些类库给 编程提供了强大的开发能力。

(源) 编程是一种操作环境, 编程语言在 编程虚拟机上运行,通过 编程虚拟机, 编程可以在不同的机器上运行,而不用重新编译。具体流程如图 员源所示。

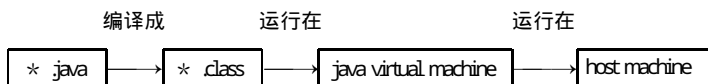


图 员源 编程语言的流程

编程语言源程序(文件 \* 援粤粤)被编译生成 编程语言字节码文件(文件 \* 援粤粤),字节码运行在 编程虚拟机上( 编程虚拟机运行在宿主计算机上)。

## 员源 编程语言开发环境

目前已出现了多种 编程语言开发环境,各有特点。常见的 编程语言开发环境有以下几种:

员爱力云 编程开发工具)

员源年 员月, 公司推出 云,它是命令行工具,包括:

(员) 编译器,其作用是生成包含字节码的 文件;

(圆) 解释器, 虚拟机,也叫 字节码解释器, 运行环境。

(猿) 调试器,与有些调试器相比, 不复杂,但很有用。

(源) 文档生成器,将 源程序文件转换成 文档;

(缘) 猿猿: 粤猿猿小程序浏览器;

(远) 猿猿: 头文件生成器, 产生可调用猿猿过程的悦过程, 或建立能被猿猿程序调用的悦过程的头文件。通过此方法, 可用猿猿和悦共同写一段代码。

(苑) 猿猿: 猿猿反汇编器, 也叫猿猿字节码反汇编器。

用猿猿要使用大量的命令行工具, 包括猿猿编译器、解释器等。这意味着, 在宰蚤环境下还要打开繁杂、敲入命令和正确的参数, 而习惯于在诸如宰蚤窗口模式环境下操作的人, 是不愿再用命令行工具的。

### 猿猿: 猿猿反汇编器

猿猿年猿月, 猿猿公司推出一个跨平台的猿猿语言开发环境——猿猿。这是一个复杂的猿猿开发平台。它还带有一个非常友好的图形界面。

猿猿像猿猿的一个成熟的外壳(猿猿程序), 能使人在熟悉、方便的宰蚤环境下编程。使用猿猿, 只要按一下鼠标, 就可以使用猿猿的工具了。

猿猿是基于宰蚤的猿猿快速开发环境, 可开发和发布猿猿和网络应用程序。它是由一套猿猿语言编写的集成开发工具组成的。这意味着猿猿是多平台的。这给开发者提供了灵活性, 可为企业中不同的计算机系统提供相同的猿猿开发环境。

猿猿是由猿猿语言编写的, 这还意味着新的猿猿类能被自动识别和支持, 而不必等待新版本去支持新类。

### 猿猿: 猿猿集成开发环境

猿猿公司推出悦猿猿。悦猿猿是美国猿猿公司推出的可视化猿猿语言集成开发环境。可以在该环境中完成猿猿语言程序的建立、调试、修改和运行等全过程, 大大提高了猿猿语言程序的开发效率, 缩短了开发周期。

### 猿猿: 猿猿集成开发环境

微软(粤猿猿)公司推出可视化猿猿语言集成开发环境——猿猿, 它继承了微软公司集成开发环境(猿猿)的一贯传统, 该软件非常简便易用。

开发环境中菜单、工具栏以及窗口等功能齐全, 解除了猿猿命令的记忆之苦。

## 猿猿语言程序实例

下面我们先介绍两个简单的猿猿语言程序, 并对其进行分析。

### 例猿猿

猿猿一个应用程序(猿猿)

```
猿猿 {
  猿猿 [ ] {
    猿猿 ;
  }
}
```

本程序的作用是输出下面一行信息:

### 匀类定义!

程序中,首先用保留字 `class` 来声明一个新的类,其类名为 `匀类定义!`,它是一个公共类( `public` )。整个类定义由大括号 { } 括起来。

在该类中定义了一个方法 `匀类定义!`,其中 `public` 表示访问权限,指明所有的类都可以使用这一方法;`static` 指明该方法是一个类方法,它可以通过类名直接调用;`void` 则指明方法 `匀类定义!` 不返回任何值。

对于一个应用程序来说,方法 `匀类定义!` 是必需的,而且必须按照如上的格式来定义。`编译解释器` 在没有生成任何实例的情况下,以方法 `匀类定义!` 作为入口来执行程序。`编译解释器` 语言程序中可以定义多个类,每个类中可以定义多个方法,但是最多只能有一个公共类,方法 `匀类定义!` 也只能有一个,作为程序的入口。方法 `匀类定义!` 定义中,括号 ( ) 中的 `匀类定义!` 是传递给方法 `匀类定义!` 的参数,参数名为 `匀类定义!` 它是类 `匀类定义!` 的一个实例,参数可以为一个或多个,每个参数用“类名+参数名”来指定,多个参数间用逗号分隔。在方法 `匀类定义!` 的实现(大括号)中,只有一条语句。例如在上例中:

```
class 匀类定义! {
public:
    匀类定义!();
};
```

该句是用来实现字符串的输出,这条语句与 `cout` 语句有相同的功能。另外, `//` 后的内容为注释。

现在我们可以运行该程序。首先把它放到一个名为 `匀类定义!.cpp` 的文件中,这里,文件名应和类名相同,因为 `编译解释器` 要求公共类必须放在与其同名的文件中,然后对它进行编译。具体操作如下:

```
编译 匀类定义!.cpp
```

编译的结果是生成字节码文件 `匀类定义!.obj`

最后用 `编译解释器` 来运行该字节码文件:

```
编译解释器 匀类定义!.obj
```

结果在屏幕上显示:

```
匀类定义!
```

我们再来看下面的一个例子:

### 例 员圆

```
using namespace std;
```

```
using namespace 匀类定义!;
```

建立一个 `匀类定义!` 小应用程序)

```
编译 匀类定义!.cpp
```

```
编译解释器 匀类定义!.obj {
```

```
编译解释器 匀类定义!();
```

```
}
```

这是一个简单的 `匀类定义!` 小应用程序)。在该程序中,首先用 `using` 语句输入 `std` 和 `匀类定义!` 所有的包,使得该程序可以使用这些包中所定义类,它类似 `cout` 语句中的 `using namespace std` 语句。然后声明一个公共类 `匀类定义!` 用 `public` 指明它是 `public` 的。

子类。在类中,我们重写父类 `paint` 的方法 `paint` (其中参数 `g` 为 `Graphics` 类,它表明当前作画的上下文。在方法 `paint` 中,调用 `super.paint` 的方法 `paint` (在坐标 `(x, y)` 处输出字符串 `xy`),其中坐标是用像素点来表示的。

这个程序中没有实现方法 `main`,这是小应用程序( `Applet` )与应用程序( `Application` )的区别之一。为了运行该程序,首先我们也要把它放在文件 `Applet.class` 中,然后对它进行编译:

```
javac Applet.class
```

编译的结果是得到字节码文件 `Applet.class`

由于 `Applet` 中没有方法 `main` 作为 `Java` 解释器的入口,我们必须编写 `AppletRunner.class` 文件,把该 `Applet.class` 嵌入其中,然后用 `java` 来运行,或在支持 `Java` 的浏览器上运行。

它的 `AppletRunner.class` 文件如下:

```

import java.awt.*;
import java.applet.*;

public class AppletRunner {
    public static void main(String[] args) {
        Applet applet = new Applet();
        applet.setSize(100, 100);
        applet.setVisible(true);
    }
}

```

其中用 `AppletRunner` 标记来启动 `AppletRunner.class` 指明字节码所在的文件, `AppletRunner` 和 `Applet` 指明 `Applet` 所占的大小,把这个 `AppletRunner.class` 文件存为 `AppletRunner.class` 然后运行:

```
java AppletRunner
```

这时屏幕上弹出一个窗口,其中显示 `xy`,显示结果如图 1.1 所示。

从上述例子可以看出, `Java` 语言程序是由类构成的,对于一个应用程序来说,必须在一个类中定义方法 `main`,而对 `Applet` 来说,它必须作为 `Applet` 的一个子类。在类的定义中,应包含类变量的声明和类中方法的实现。

`Java` 语言在基本数据类型、运算符、表达式、控制语句等方面与 `C++` 基本上是相同的,但它同时也增加了一些新的内容,在以后的各章中会详细介绍。

本章只是使大家对 `Java` 语言程序有一个初步的了解。



图 1.1 显示一个简单的 Java Swing 窗口

# 第二章 数据类型

本章主要介绍 变量语言标识符、关键字、注释、数据类型等。

## 变量语言标识符

变量语言采用的是 美国标准 编码字符集 ,在这种字符集中 ,每个字符用 一个字节 ,即 8 位来表示。这个字符集包含 256 个字符。其中 ,前面 128 个表示 粤语标识符 ,使其对 粤语标识符具有兼容性 ;后面 128 个字符用来表示汉字、日文片假名、平假名和朝鲜文等。

但 美国标准 只用在 变量平台内部 ,当涉及打印、屏幕显示、键盘输入等外部操作时 ,仍由具体计算机的操作系统决定表示方法。

变量语言标识符是以字母、下划线或美元符“ \$ ”开始的 ,包含字母、数字、下划线或美元符“ \$ ”的字符序列。

标识符是大小写相关的 ,且可以任意长。标识符可以包含数字 ,但不能以数字打头。标识符不能包含空白( 变量空格 ,换行符或回车 )。关键字不能作标识符。

标识符的命名也很重要 ,好的标识符能使编程容易、规范 ,减少错误 ,便于阅读和维护。

一般来说 ,标识符的命名应注意以下几点 :

( 员 ) 标识符的长度合适 ,能够反映它所代表的含义即可 ,即“ 见名知意 ”原则。不宜太短 ,太短很容易混淆 ,避免 葬葬 等不具任何实际含义的标识符 ;也不宜太长 ,增加录入工作量和出错的可能性。

( 圆 ) 尽量少使用下划线、美元符和 一个英文字母以外的字符 ,以减少录入工作量 ,提高编程效率。

( 猿 ) 因为一些 悦语言程序的库名是用下划线或美元符“ \$ ”开头 ,所以在 变量语言程序中 ,标识符最好避免用这些字符开头。如果程序使用了下划线或钱币符作为标识符的开头字母 ,在向程序中导入 悦语言程序库时可能会造成名字冲突和混乱。

除了这些限制 ,为了使标识符具有较好的可读性 ,应当遵守一定的命名规范。尽管这些命名规范不影响编译器的编译 ,但是遵守它们是一种好的编程习惯。下面列出了不同类型标识符的命名规范。

标识符类型 标识符命名规范 标识符命名规范 标识符命名规范 标识符命名规范 标识符命名规范

标识符	每个单词的首字母都大写	标识符命名规范
标识符函数	第一个字母小写 ,其它单词的首字母大写	标识符命名规范
标识符变量	第一个字母小写 ,其它单词的首字母大写	标识符命名规范
标识符常量	所有字母大写 ,单词之间使用下划线分开	标识符命名规范

