

windows用户界面构件及思想Microsoft认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/623/2021_2022_windows_E7_94_A8_c100_623926.htm

Windows最初由Microsoft公司在1983年11月宣布，1985年11月推出了第一公开发行人版本，即1.01版。此后两年，1.01版进行了几次修改以满足国际市场的需求，并增减了一些显示器和打印机的驱动程序。1987年11月发行的Windows 2.0版在用户界面上做了些改进。例如：重叠式窗口的引入，还增强了键盘和鼠标接口，特别是增强了选单和对话框的功能．．．（略）

1、用户界面的构件

1.1.1 窗口

窗口是屏幕上与一个应用程序相关的矩形区域，它是用户与产生该窗口的应用程序之间的可视界面。对应用程序来说，窗口是应用程序控制下的屏幕上的一个矩形区域，应用程序创建并控制窗口的所有方面。当用户启动一个应用程序时，一个窗口就被创建。每当用户操作窗口中的对象时，程序就有所响应。

1.1.2 边框

绝大多数窗口都有一个环绕着它的边框，边框不仅作为窗口的边界，它也用来指明窗口的状态，即窗口是否是一个活动窗口。当我们将鼠标指针放在边框上按下鼠标键并移动鼠标时，就可以改变窗口的大小。

1.1.3 标题栏

标题栏位于窗口的顶部，其中显示的文本信息用于标注程序，一般是应用程序的名字，这便于用户知道哪个应用程序正在运行。标题栏的颜色反映一个窗口是否是一个活动窗口。

1.1.4 控制框

控制框是每个窗口左上方的一个小图片，每个应用程序都使用它。在控制图标上单击鼠标键会使Windows显示系统菜单。系统菜单它提供了诸如还原、移动、大小、最小化、最大化以及关闭这样的标准操作。

1.1.5 最小化图标

每

个Windows 98及Windows NT应用程序都在窗口的右上角显示三个图标。最左边的图标是一段短下划线，这就是最小化图标。它可以使用程序被最小化。

1.1.6 最大化图标 最大化图标是三个图标中中间的那一个，看起来象两个小窗口。使用最大化图标可以使用应用程序占满整个屏幕。如果选择了这个图标，其它应用程序窗口都会被盖住。

1.1.7 垂直滚动条 如果有必要，应用程序可以显示一个垂直滚动条。垂直滚动条显示在应用程序窗口的右边，在两端有两个方向相反的箭头。它还有一个着色的棒和一个透明的窗口块。后者被用于显示当前显示内容与整个文档的关系。你可以用滚动条来选择显示哪一页。一般在任何一个箭头上单击一下会使显示内容移动一行。单击向上箭头下方的窗口块并拖动它会使屏幕输出快速更新到应用程序屏幕输出的任意位置。

1.1.8 水平滚动条 也可以显示一个水平滚动条。水平滚动条显示在窗口的底部，具有与垂直滚动条类似的功能。你用它来选择要显示哪些列。一般在任何一个箭头上单击一个会使显示内容移动一列。单击向左箭头右边的窗口块并拖动它会使屏幕输出快速更新到应用程序屏幕输出的任意位置。

1.1.9 选单栏 一个可选择的菜单条可以显示在标题条的下方。通过菜单条来选择菜单和子菜单。这种选择可以通过用鼠标单击，也可以用热键组合来实现。热键组合经常是ALT与命令中带下划线的字母的组合，比如File命令中的“F”。

1.1.10 用户区 通常用户区占据了窗口最大的部分。这是应用程序的基本输出区域。应当由应用程序来复杂管理用户区。另外，应用程序可以输出到用户区。

2、面向对象的思维方法

对于程序员来说，面向对象意味着非常熟悉的事物：将世界看成是一组彼此相关并相

互通信的实体即对象组成，每个对象有一个名字来标识，这是人们通常看待世界的方式。例如，当看见一辆汽车时，所见到的是一辆汽车，而不是一大堆原子。人们可以将汽车分解为车轮、发动机、车门、油箱等，它们都是具体的实体即对象。对象之间的通信被称为发送消息，即一个对象请求另一个对象执行某种方式的操作。例如，交叉路口的红灯“请求”驾驶员停车，驾驶员在接受到消息之后，他所执行的动作是踏下制动踏板，这有向汽车发送了一条消息，汽车在接受到此消息之后，又将该消息分解之后发送到相关的对象上：制动器作用于车轮上，将动能转变成为势能，使车速降下来；尾灯又向它后面的其它车辆的驾驶员发送消息；各种仪表盘向驾驶员反馈出所发送的消息的动作结果。从程序员角度而言，对象是内存中一块有名的存储单元。我们通常所谓的变量就是一种数据对象，但对象的概念比变量的含义更广义，通常将对象定义成为包含有数据和代码的内存区域，数据表征对象的特征，而代码用于响应消息，使对象进行某些动作。从屏幕上显示的一个可视的窗口对象为例，我们可以对比分析一下用户心目中的对象和程序员心目中的对象的关系。窗口对象的特征，例如颜色、长度、其中显示的信息等，在程序中被表示为数据，用户对窗口所做的操作，例如移动窗口、改变窗口的大小等，使得用户向窗口发送了消息，这些消息引起了计算机（内存中的）对象执行相应的代码，代码执行的结果改变了对象中的数据，使对应的可以视对象的位置和大小发生了变化。对象为响应消息所执行的代码被称为方法，对象中保存的数据构成对象的属性，对象的抽象定义就是执行某些动作，否则，没有其它途径可以使一个

对象动作起来。向一个对象发送消息在程序中表示为：
functionName(id, arg1, arg2, ...). 其中，消息是functionName，id是标识对象的一个对象名，或称其为对象的标识符，Windows使用某种类型的数据来作为对象的标识符，这个标识符号常被称为对象的句柄。arg1等为消息所带的参数。虽然发消息类似于标准的函数调用，但消息也有函数调用中没有的特性，例如，消息始终在执行一选择机制，其参数与其它消息区别开来，从而告诉该对象完成什么样的操作。一个函数名始终指向内存中该函数的代码所位于的确定地址，消息并不指向内存中的某地址，但却告诉接受消息的对象要引用的内存地址是什么。函数不仅说明操作，而且还要执行如何完成该操作的方法。消息只说明该操作，在对象中定义的方法说明如何完成该操作。当向不同的对象发送相同的消息时，所执行的方法是不同的。在面向对象的程序设计中，每个对象由一个类来定义，类是对一组性质相同的对象的程序描述，它是由概括了一组对象共同性质的方法和数据组成。从一组对象中抽象出公共的方法与数据，将它们保存在一个类中是面向对象程序设计的核心。在日常生活中，我们也以类这种方式来定义客观对象。通过对客观对象进行抽象，我们将性质相同的对象归为一类，形成概念，例如，人类、苹果类、食品类等。通过对客观对象分类，我们也可以更好地认识客观对象，例如，当知道张三是一个人时，不用对张三进行更多的描述，我们已知道张三作为一个人所具有的特征和行为，因为它们已经在“人”类中进行了描述。在面向对象的程序中，类被用作样板来生产具有相同行为方式的对象。类就像是生产对象的一个工厂，在生产对象时，对象具

有类中所描述的同样的数据结构和方法，同时，对象的每个数据在创立之初取得一个初始值，形成对象的初始状态。对象通过发送消息相互作用，对象的状态从一种状态过渡到另一种状态，当所有的有关对象到达某种特定的状态时就得程序的运行结果。使用类产生对象的过程也称为生成该类的一个实例。因此，对象也可以定义为对象是类的一个实例。定义类也意味着将该类的对象公用代码放在内存的公共区域中，而不必对每个对象都将它们的代码和数据重新进行一次描述，这减轻了程序员的劳动强度。我们可以将一些常用对象定义放在一个公用库中，而在程序中需要该类的一个对象时，就创建该类的一个实例。Windows已为程序员预定义了许多像按钮、滚动杠和对话框等对象的类，当程序员需要这些类的对象时，仅需创立该类的实例即可。对于同一个类的不同对象，在建立对象时其初始状态不同，因而这些对象在屏幕上显示的位置、大小等属性也不相同，但同类的对象的操作是相同的（因为它们共用相同的方法）。这也就是为什么不同的Windows应用程序对用户表现出一致的操作特性的原因之一。编辑特别推荐: 微软WindowsServer2008认证体系介绍 MCSE认证考试全程心得 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com