

编写高质量的VisualBasic代码常注意的几项问题 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/449/2021_2022__E7_BC_96_E5_86_99_E9_AB_98_E8_c97_449454.htm 销毁对象 无论编写的是

是什么软件，程序员都需要考虑在用户决定终止软件运行后释放软件占用的内存空间。但遗憾的是很多程序员对这一点好像并不是很在意。正确的做法是在退出程序前需要销毁程序中使用的对象。例如：`Dim FSO as New FileSystemObject` 执行操作 销毁对象 `Set FSO = Nothing` 对于窗体，可以进行卸载：`Unload frmMain` 或 `Set frmMain = Nothing` 变长和定长字符串从技术上来说，与变长字符串相比，定长字符串需要较少的处理时间和空间。但是定长字符串的缺点在于在很多情况下，你都需要调用Trim函数以去除字符串末的空字符，这样反而会降低代码效率。所以除非是字符串的长度不会变化，否则还是使用变长字符串。使用类模块，而不是ActiveX控件除非ActiveX控件涉及到用户界面，否则尽量使用轻量的对象，例如类。这两者之间的效率有很大差异。使用内部对象在涉及到使用ActiveX控件和DLL的时候，很多程序员喜欢将它们编译好，然后再加入工程中。我建议你不要这样做，因为从VB连接到一个外部对象需要耗费大量的CPU处理能力。每当你调用方法或存取属性的时候，都会浪费大量的系统资源。如果你有ActiveX控件或DLL的源代码，将它们作为工程的私有对象。减少模块的数量 有些人喜欢将通用的函数保存在模块中，对于这一点我表示赞同。但是在一个模块中只写上二三十行代码就有些可笑了。如果你不是非常需要模块，尽量不要使用它。这样做的原因是因为只有在模块中的函数

或变量被调用时，VB才将模块加载到内存中.当VB应用程序退出时，才会从内存中卸载这些模块。如果代码中只有一个模块，VB就只会进行一次加载操作，这样代码的效率就得到了提高.反之如果代码中有多个模块，VB会进行多次加载操作，代码的效率会降低。使用对象数组 当设计用户界面时，对于同样类型的控件，程序员应该尽量使用对象数组。你可以做一个实验：在窗口上添加100个PictureBox，每个PictureBox都有不同的名称，运行程序。然后创建一个新的工程，同样在窗口上添加100个PictureBox，不过这一次使用对象数组，运行程序，你可以注意到两个程序加载时间上的差别。使用Move方法 在改变对象的位置时，有些程序员喜欢使用Width，Height，Top和Left属性。例如：Image1.Width = 100 Image1.Height = 100 Image1.Top = 0 Image1.Left = 0 实际上这样做效率很低，因为程序修改了四个属性，而且每次修改之后，窗口都会被重绘。正确的做法是使用Move方法：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