

用C Builder写COM版的Helloworld PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/253/2021\\_2022\\_\\_E7\\_94\\_A8C\\_\\_Buil\\_c97\\_253807.htm](https://www.100test.com/kao_ti2020/253/2021_2022__E7_94_A8C__Buil_c97_253807.htm) 开始做 Automation : 1.File|New

Application (做 Automation 要有 Form , 即使是个空的也不要拿掉) . 2.File|New... 选 ActiveX 页上的 Automation Object. 在对话框中输入 COM 名 : AutoTest , 其它就用默认值。 3.在打开的 Type Library Editor (就是那个标题为 Project1.tlb 东东) 中, 左边的树中的 IAutoTest 上点右键, New 一个 Property , 会产生两个同名的东东, 分别用于 Get 和 Set , 取名为 Hello (两个会自动变为一样的) , 然后在右边找到 Parameters 页, 将 Type 改为 BSTR(Set) 和 BSTR \*(Get , 注意, 列表中的 BSTR 没有 \* , 要自己输入一个) 。 4.然后点顶上那个 Refresh Implementation 的按钮 (按钮上的小图片是一张纸, 里面有两个绿色的箭头成环绕状) , 这一步很重要, 每次修改完 TLB 文件都要点这个按钮再存盘, 它将产生相应的代码。 5.在 AutoTestImpl.h 文件中找到如下内容 : // IAutoTest public: STDMETHOD(get\_Hello(BSTR\* Value)).

STDMETHOD(set\_Hello(BSTR Value)). 这是 Refresh Implementation 时自动产生的代码, 在后面加上 : private : WideString FData. Type Library Editor 产生的 CPP/H 文件除了这个以 Impl (即 Implementation ) 结尾的以外, 还有两个, 分别是以 \_ATL 和 \_TLB 结尾的 : \_ATL 是自动产生的, 通常不用变它, 当然如果你有特别的要求并且你对 ATL 又很熟, 也可以改改它 ; \_TLB 是由 Type Library Editor 自动维护, 无须修改, 改了也没用, 你的任何修改都会在 Refresh

Implementation 时被改回来的。再看 AutoTestImpl.cpp 文件，找到如下内容：STDMETHODIMP

```
TAutoTestImpl::get_Hello(BSTR* Value) { try { } catch(Exception  
amp.e) { ShowMessage( e.Message ). } 5.Save All , 编译。 6.运行  
后，点 Button ，你会看到 Project1 也运行起来，同时 Label1  
显示 "Hello world!" 。当 Project2 退出时，Project1 也自动退出  
。会了吧。另一种使用 Automation 的方法（后期联编用法）  
： 1.File|New Application 2.在 Unit3.cpp 中加入： #include 3.在  
Form 上放一个 Label 和一个 Button 。双击 BUTTON ，在事件  
响应中输入： Variant v = CreateOleObject( "Project1.AutoTest" ).  
v.Exec( PropertySet( "Hello" ) Label1->Caption = v.Exec(  
PropertyGet( "Hello" ) ). 4.Save All , 编译。 5.运行后，点  
Button ，你会看到 Project1 也运行起来，同时 Label1 显示  
"Hello world!" ，然后 Project1 就退出了。 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
```