# 这个类型的作用是帮助你理解windows API函数，以后我们会学习MFC也就是c++的方式

## 项目结构

|  |
| --- |
|  |

## Winapp.cpp的内容如下

|  |
| --- |
| #include"stdafx.h"  #include"winapp.h"  /\*\*  ctrl+j :列出成员  ctrl+shift+space ： 参数信息  ctrl+space： 单词完成  \*/  LRESULT CALLBACK WinProc(HWND hwnd,UINT umsg,WPARAM wparam,LPARAM lparam);  int APIENTRY \_tWinMain(HINSTANCE hInstance,  HINSTANCE hPrevInstance,  LPTSTR lpCmdLine,  int nCmdShow)  {    UNREFERENCED\_PARAMETER(hPrevInstance);  UNREFERENCED\_PARAMETER(lpCmdLine);  WNDCLASSEX wcex;  wcex.cbClsExtra = 0;  wcex.cbWndExtra = 0;  wcex.cbSize = sizeof(WNDCLASSEX);  wcex.hbrBackground = (HBRUSH)GetStockObject(WHITE\_BRUSH);  wcex.hIcon = LoadIcon(NULL,IDI\_ASTERISK);  wcex.hCursor = LoadCursor(NULL,IDC\_HAND);  wcex.hInstance = hInstance;  wcex.lpfnWndProc = WinProc;  wcex.lpszClassName = \_T("MyWin");  wcex.style =CS\_HREDRAW| CS\_VREDRAW;  wcex.hIconSm =LoadIcon(wcex.hInstance, MAKEINTRESOURCE(IDI\_SMALL));  wcex.lpszMenuName = NULL;  //注册  RegisterClassEx(&wcex);    //创建窗口  HWND hwnd;  hwnd = CreateWindow(  \_T("MyWin"),  \_T("MyApp"),  WS\_OVERLAPPEDWINDOW,  CW\_USEDEFAULT,  CW\_USEDEFAULT,  CW\_USEDEFAULT,  CW\_USEDEFAULT,  NULL,  NULL,  hInstance,  NULL);  if(!hwnd)  {  MessageBox(NULL,\_T("创建窗口失败"),\_T("result"),0);  return -1;  }  ShowWindow(hwnd,SW\_SHOWNORMAL);  UpdateWindow(hwnd);  //信息循环  MSG msg;  while(GetMessage(&msg,NULL,0,0))  {  TranslateMessage(&msg);  DispatchMessage(&msg);  }  return (int)msg.wParam;  }  LRESULT CALLBACK WinProc(HWND hwnd,UINT umsg,WPARAM wparam,LPARAM lparam)  {  PAINTSTRUCT ps;  HDC hdc;  switch(umsg)  {  case WM\_PAINT:  hdc = BeginPaint(hwnd,&ps);  TextOut(hdc,200,200,\_T("First Win32 Application!!"),wcslen(\_T("First Win32 Application!!")));  EndPaint(hwnd,&ps);  break;  case WM\_CHAR:  TCHAR info[50];  swprintf(info,50,L"按键的ascII是：%d",(int)wparam);  MessageBox(hwnd,info,\_T("char"),0);  break;  case WM\_LBUTTONDOWN:  MessageBox(hwnd,\_T("Mouse clicked"),\_T("mouse"),0);  break;  case WM\_CLOSE:  if(IDOK == MessageBox(NULL,\_T("退出？"),\_T("提示？"),MB\_OKCANCEL))  {  DestroyWindow(hwnd);  }  break;  case WM\_DESTROY:  PostQuitMessage(0);  break;  default:  return DefWindowProc(hwnd,umsg,wparam,lparam);  }  return 0;  } |

## 这里需要注意的是vs2010使用宽字符，需要使用\_T()把字符串包裹起来，否则会报错，strlen的宽字符串的版本是wcslen里面的参数也需要使用\_T()把字符串包裹起来。sprintf的宽字符版本是swprintf