

CHAPTER TEN

WINDOW PROGRAMMING-OBJECT ORIENTED

10.1 Microsoft Foundation classes

Window programming so far done with the help of API is known as procedure oriented programming. Microsoft visual C++ provides an environment to develop two types of windows application. First procedure oriented C/C++ language Windows programs using only Win 32 API. Second, Object oriented Approach using various visual C++ tools such as –Application Wizard, Class Wizard and Resource editor, to make low-level Win-32 programming easier. Window programming using Microsoft Foundation Classes(MFC) is called object Oriented Programming. The Microsoft Foundation Classes are the collection of predefined classes in visual C++ which provide a new approach towards Window programming. These classes facilitate the programmer to do programming in object oriented manner. By using and creating MFC objects or derived class objects one can write a window program.

10.2 MFC Notation

All the classes present in MFC begin with C, such as CObject, CWinApp, or CView. Data member of class are prefixed with m_. Hungarian notation is also used by MFC for most of the variables, particularly those that were originated in Window API.

10.3 Classes and Files in Application Wizard

In this section we will use classes and files that are built by the Application Wizard.

1. Open the Visual C++ IDE Environment for working.
2. Click the File menu and select NEW for creating a new application.
3. Click the Project tab for creating project files, select MFC AppWizard (exe).
4. Click the Project –name Box and write the project name: Mypro. New project Dialogue Box looks like in Figure 10.1.
5. Click OK. Now step 1 dialogue Box of MFC AppWizard is displayed, select single document. (Figure 10.2)
6. Step-2 dialogue box of MFC is displayed, accept the default setting , click Next.(See Figure 10.3).

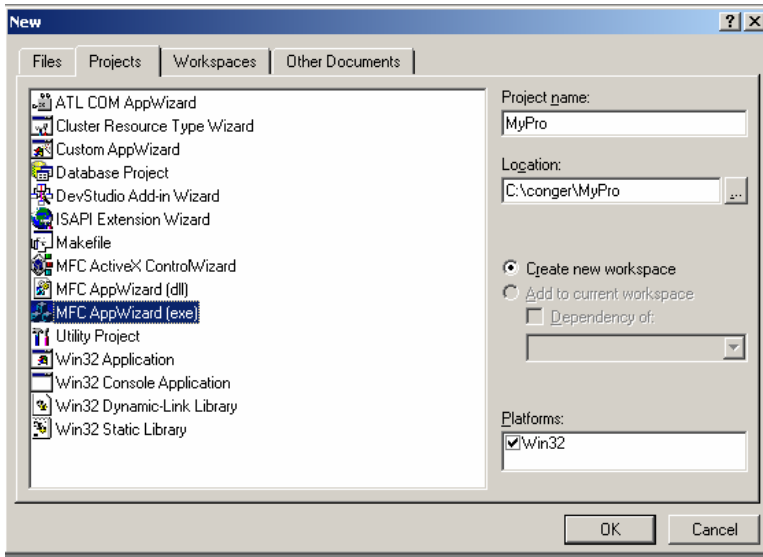


Figure 10.1 Project Dialogue Box

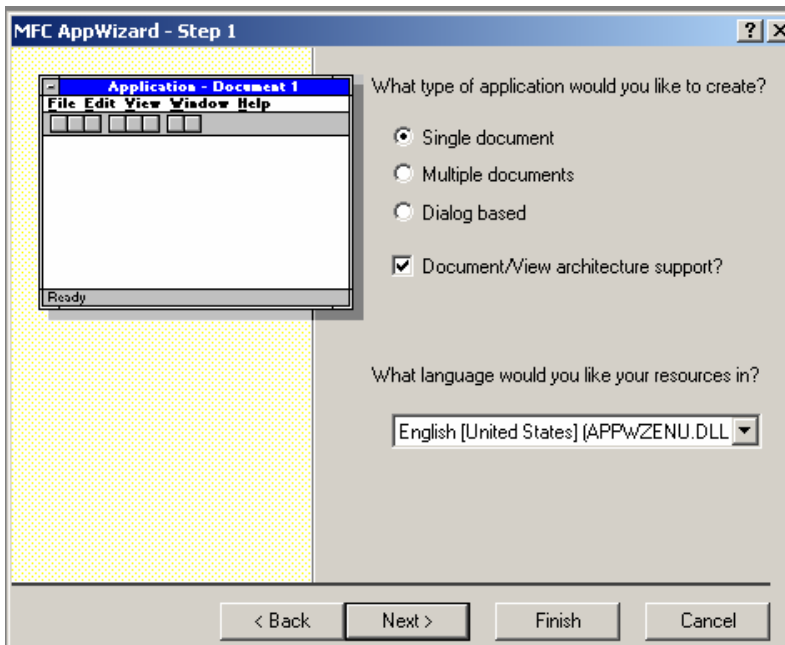


Figure 10.2 MFC Application Wizard-Step 1 dialogue Box.

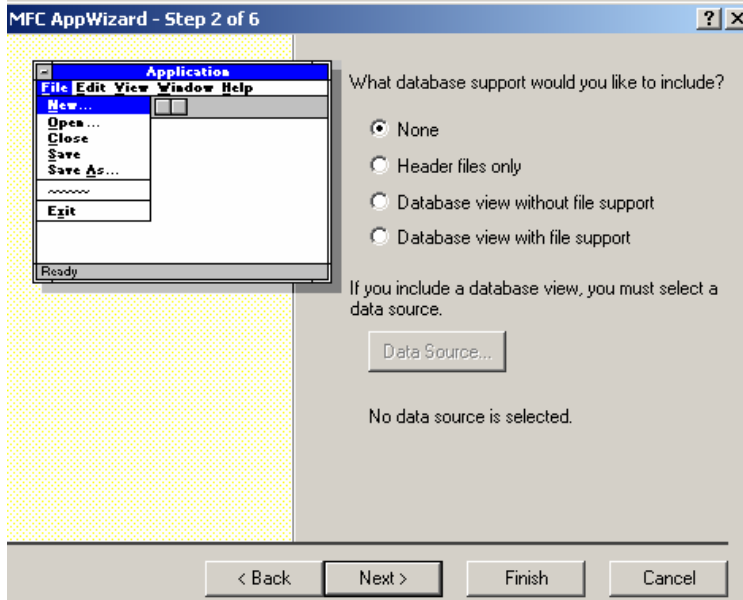


Figure 10.3 : MFC Application Wizard –step 2

7. Step-3 dialogue Box of MFC Application Wizard is displayed. Click next to continue. (figure 10.4)

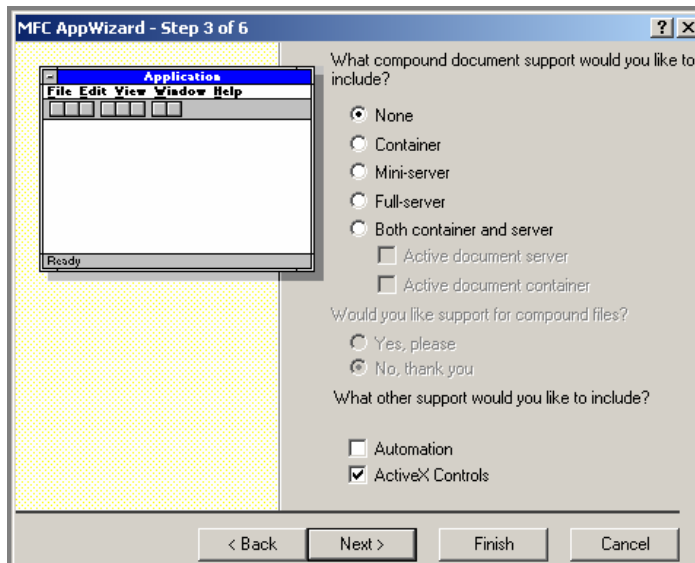


Figure 10.4 . MFC Application Wizard –step 3

8. Step-4 dialogue Box of MFC Application Wizard is displayed. Click next to continue. (figure 10.5)

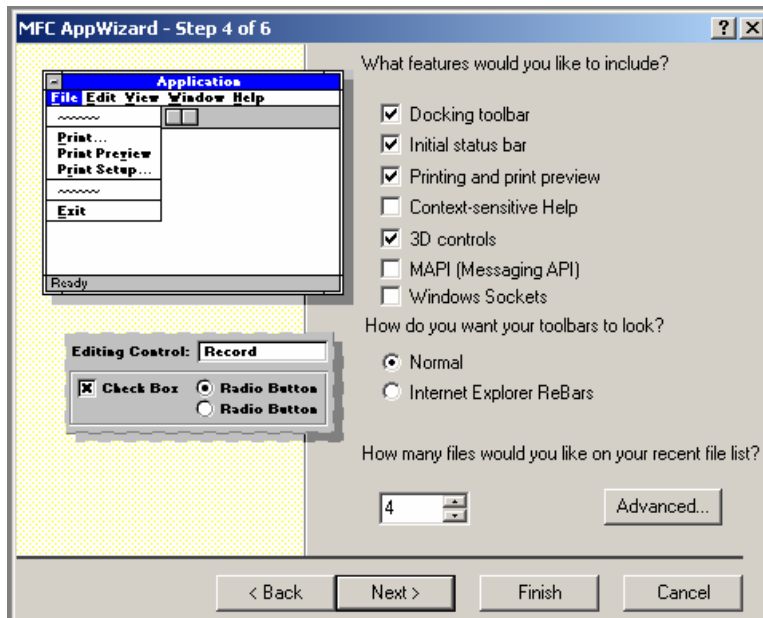


Figure 10.5 . MFC Application Wizard –step 4

9. Step-5dialogue Box of MFC Application Wizard is displayed. Click next to continue. (figure 10.6)

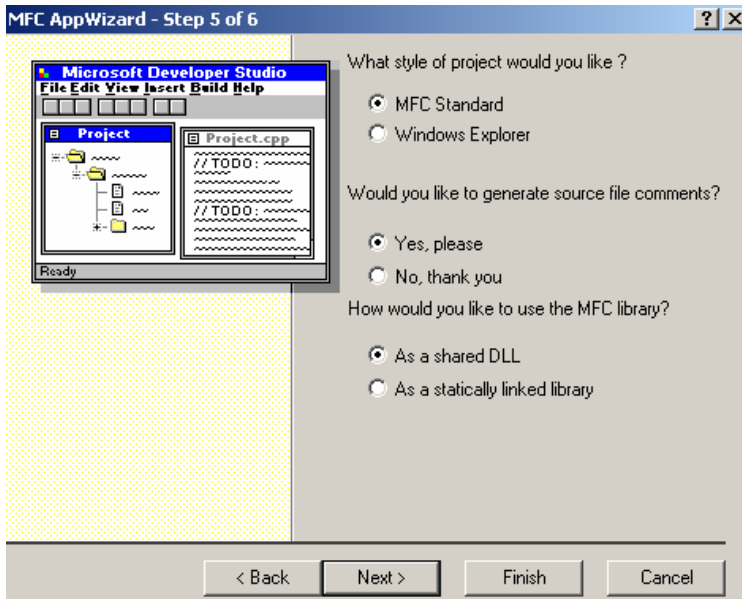


Figure 10.6 . MFC Application wizard –step-5

10.. Step-6 5dialogue Box of MFC Application Wizard is displayed. Click Finish to continue. (figure 10.7)

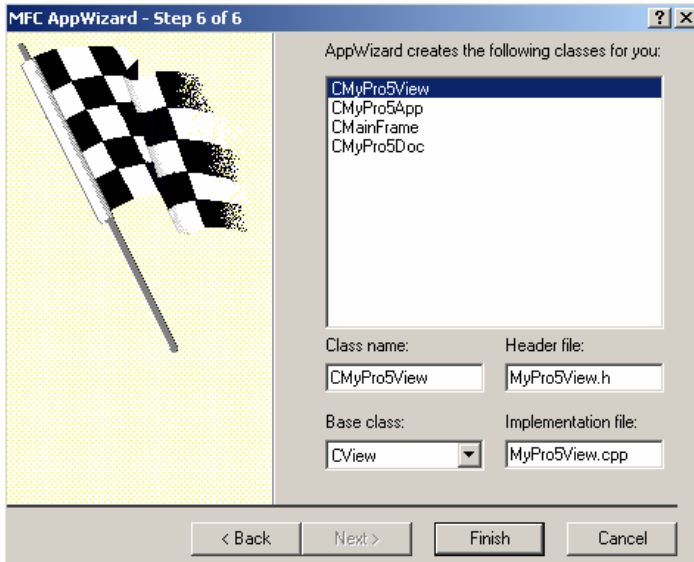


Figure 10.7 . MFC Application wizard –step-6

11. New Project information window is open, it shows the project skeleton specifications created by the AppWizard. Click OK to continue. See figure 10.8.

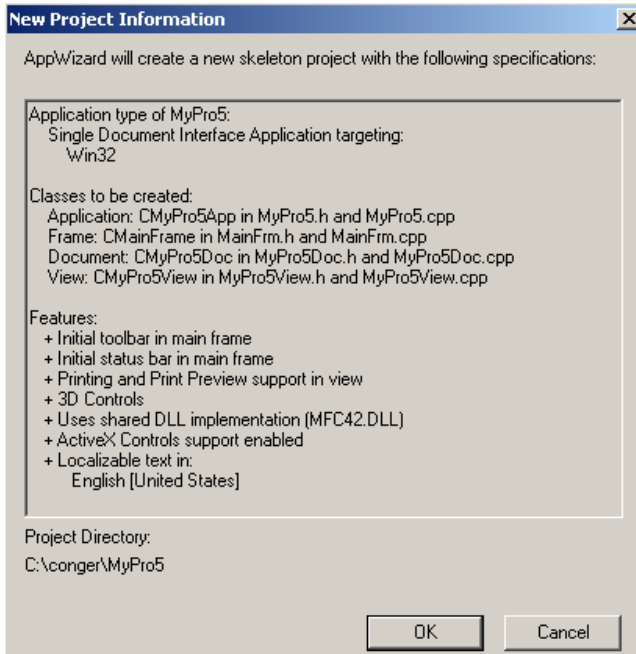


Figure 10.8. new project Information.

12. Click at +/- sign next to the Mypro classes to view the classes. See figure 10.9. Double click at CmainFrame class to view the source code of CmainFrame Class. Execute this program we will get output as a window (see figure 10.10).

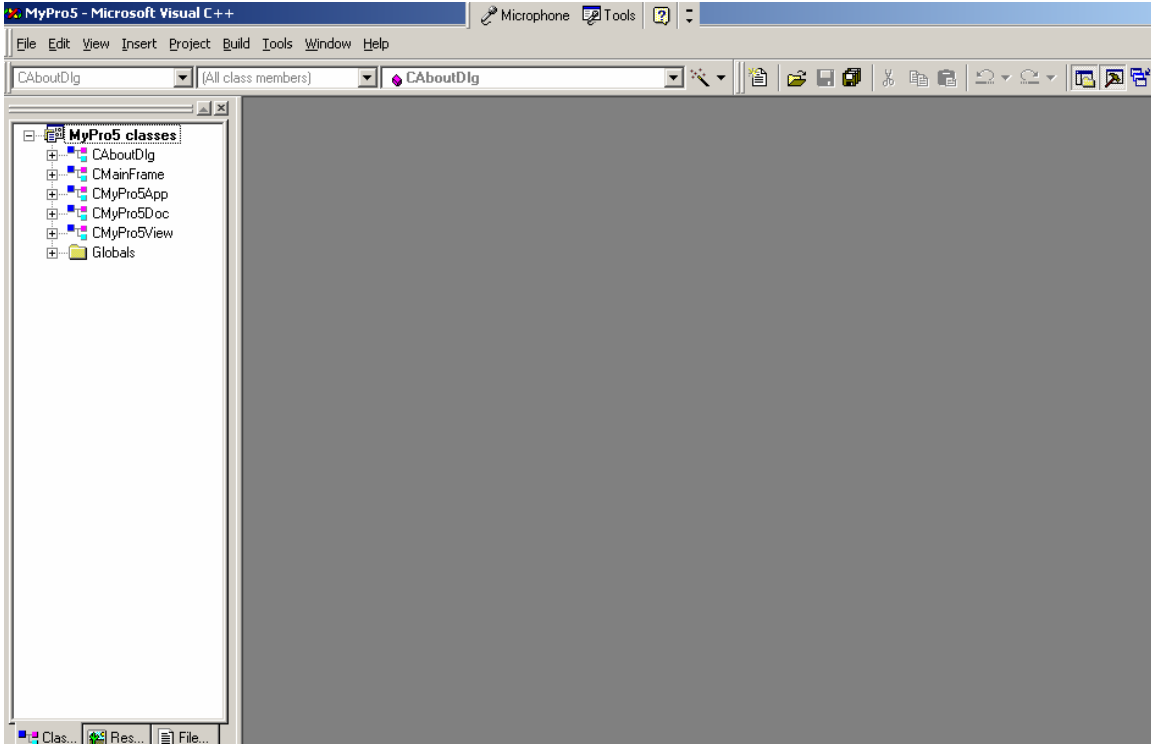


Figure 10.9. Class View

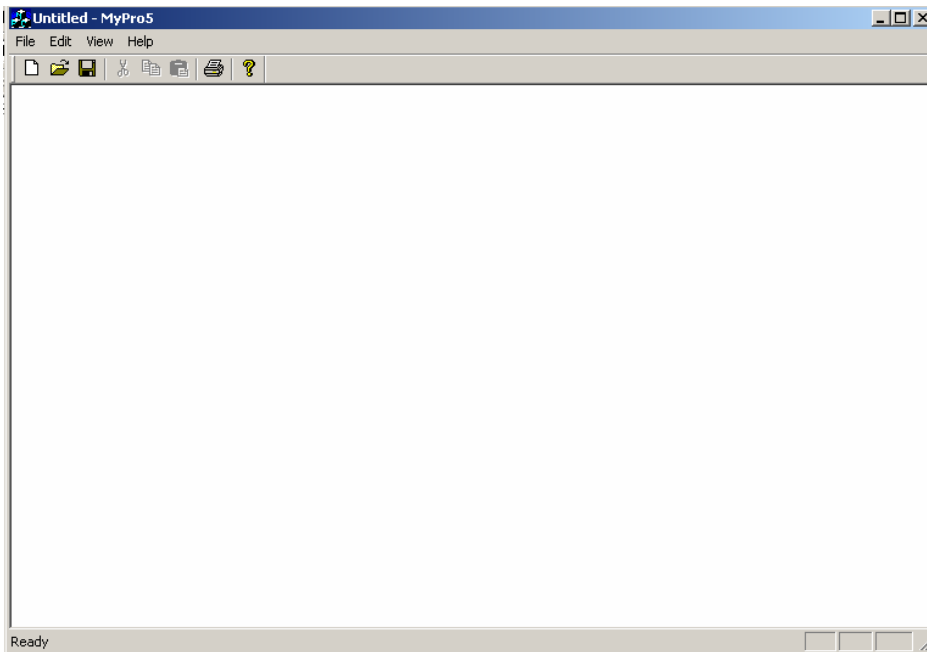


Figure 10.10 . Output of CMainFrame class using MFC

10.4 Creating Dialog based application.

Let us create a dialog based application in the way, we have created a window. Following are the steps.

1. Create a new project Mydialog. Press Ok.
2. In the new window opened, press Dialog based and press next.
3. Press next by clicking at About Box, 3D Controls and Active X control.
4. Press next in new window with default setting and again press next and then OK for the window showing project information. We will see the following interactive window.

