

Property Sheet Component: Overview

The Property Sheet component adds a property sheet or wizard-type property sheet to an existing project.

For more information on these types, see [Property Sheet Component: Specifics](#) and [Property Sheet Component: Results](#).

Property Sheet Component: Specifics

The Property Sheet component adds a property sheet, or wizard (a modal dialog box that guides you through a difficult or complex task), to an existing project. For more information on wizards, see the **CPropertyPage** class overview in the *Class Library Reference*.

The Property Sheet component offers several variations of property sheets to choose from. However, because wizards are designed for completing or guiding the user through a difficult task, they are always modal. An optional preview window is also available. If you choose to add the preview window, the code needed to render the preview window is already implemented. In every case, you get a custom property sheet class and up to nine scratch property pages.

The first page of the Property Sheet component lets you choose whether you want a property sheet or a wizard. For this component to insert successfully, it assumes that you have something derived from **CCmdTarget** in your project. This is required because the component needs classes derived from **CCmdTarget** to attach your new property sheet to. If the project was created using AppWizard, at least one **CCmdTarget**-derived class will be available. The next step is to choose whether your property sheet supports previewing. If you are creating a property sheet instead of a wizard, you also choose whether or not your property sheet is modeless. If you create a modeless property sheet, a mini-frame for the property sheet will be provided for you. After the preview step you choose the class you want to attach the new property sheet or wizard to and the number of property pages in your property sheet. Finally, you enter the names of the classes to be created and the file names they'll be written to.

The following choices for property sheets are available:

- Custom property sheet (with all choices)
A property sheet, derived from **CPropertySheet**, with 1-9 property pages, derived from **CPropertyPage**. Initialization of the property sheet occurs in the property sheet's constructor.
- Property sheet with preview (if preview attribute is chosen)
A property sheet, derived from **CPropertySheet**, with 1-9 property pages, derived from **CPropertyPage**, and a preview window, derived from **CWnd**. Initialization of the property sheet occurs in the property sheet constructor. The preview window is initialized in the property sheet's **OnInitDialog** function.
- Modeless property sheet (if modeless attribute is chosen)
A property sheet, derived from **CPropertySheet**, with 1-9 property pages, derived from **CPropertyPage**, and a mini-frame window, derived from **CMiniFrameWnd**. This property sheet is contained within a modeless frame unlike standard property sheets. Initialization of the property sheet occurs in the property sheet's constructor. The property sheet is initialized in the modeless frame's **OnCreate** function.

Property Sheet Component: Results

Note that the property sheet can't be invoked from your user interface after a successful insertion. For information on the steps needed to do this, see the comment in the `OnProperties` function (or `OnWizards` for wizards), added by the component.

The following items are inserted by this component for all variants of the property sheet:

- A new custom property sheet class derived from **CPropertySheet**. The class and file names are determined by you.
- The desired number of property pages and corresponding dialog templates.
The dialog template ID's for the property pages are based on `IDD_PROPSHT`. When this component is inserted the first time, the ID's will be `IDD_PROPSHT1`, `IDD_PROPSHT2`, etc. If the component is added to the project a second time, interesting template IDs, like `IDD_PROPSHT22` (the second template of the second insertion), are created

Based on the options selected by you, the following items can also be inserted. If you choose:

- Wizard
The new property sheet is set up as a wizard. This means that the property sheet is modal and contains a series of property pages that can only be viewed in a sequential order.
- Supports Previewing
 - A preview window class gets added to the user's project
The preview window class is derived from **Cwnd** and the class and file names are chosen by you. This class is also linked to the property sheet created by this component.
 - Code to setup and manipulate the preview window is added to the new property sheet class.
- Modeless (non-wizard)
 - A mini-frame class, derived from **CMiniFrameWnd**, is added to the user's project.
 - A pointer to the mini-frame class is added to the attaching class's member variable list.
 - The pointer to the mini-frame class is initialized in the attached class's constructor. Comments are provided in `OnProperties` to handle a non-default constructor case.
 - The mini-frame is initialized and called from the inserted `OnProperties` function in the attaching class.
- A function, `OnProperties` (or `OnWizards` for a wizard) is added to the attaching class, the default class being **CMainFrame**, to set up the property sheet. This function mainly consists of comments describing what needs to be done and comes in two types:
 - CUSTOM** This function creates a variable of the custom property sheet type, and then invokes the property sheet.
 - MODELESS** This function, if needed, initializes a modeless property sheet frame declared in a member variable (variant on `m_pFramePointer`). After initializing the property sheet frame, this function calls the custom property sheet.
- The appropriate initialization of the property sheet done in the **CPropertySheet**-derived class.

