# Program 3: Visual Inheritance and Interface
## (25 points)
## Due Date: Thursday (week 5), Feb. 18, 2016 by 5:00 PM

Create a program in VB.NET 2012 for different branches of a company to manage a list of house units to be built. There are three types of houses: Platteville, Madison, and Chicago, and they are represented by three classes, each of which is a sub-class of an abstract class House. More types of houses could be available in the future.

## Program solution

The solution has three projects: Project HouseClasses, Project FormClasses, and Project Prog3GUI. Prog3GUI is the startup project.

## Project HouseClasses

1. This project is a class library project and defines the abstract class House and the three sub-classes.
2. The root namespace must be UWPCS3340, and the assembly name must be VIIHouseClasses.
3. The project is the same as that of Prog2 with the following modifications.
   - Class House has only one public event, which is Shared:

     ```
     ' Raised when a house is created, removed,
     ' or modified (price changed or not).
     Public Shared Event HouseListUpdated()
     ```

   - Class House must implement interface IDisposable.
     - The Dispose sub must be private:

       ```
       Private Sub Dispose() Implements IDisposable.Dispose
       ```

     - The Dispose(Boolean) sub should do the following:
       - Remove the house from the collection
       - Raise event HouseListUpdated
4. The House class must allow new sub-classes to be created in the future. After new sub-classes are created, the House class should work correctly without any changes.
5. Your grade will depend on how well you design the classes.
6. All the house class properties and methods MUST NOT check the house type and do things differently for different sub-classes.
7. The house class and the three sub-classes can have other private or protected members, but they **MUST NOT** have additional Public or Friend members.

## Project FormClasses

1. This is a class library project converted from project Prog2GUI of Prog2 with class Prog2 removed.
2. The assembly (DLL file) name must be FormClasses and the root namespace must be UWPCS3340.
3. Class FormClassHouse
   - The class is the same as that of Prog2 with the following additional specifications.
   - The class cannot have variables of any House sub-classes.
   - The form will use error providers instead of message boxes to warn the user when exceptions are caught while modifying the house object on the form.
   - **The class must handle the HouseListUpdated event.**

- When the rooms and/or garages of the house displayed on the form are changed (on this form or any other form), the form should display the new values.
- When the price of the house displayed on the form is changed (on this form or any other form), the form should display the new price after a message box is displayed.
- When the house displayed on the form is removed (from any FormClassList), the form should be cleared just like when the NEW button is clicked.
  - Some members of the class need to be Protected so that they could be accessed from its sub-classes.
4. Class FormClassList
    - The class is the same as that of Prog2 with the following additional specifications.
    - There is one more button with text "Remove".
    - Clicking Remove will dispose of the selected house.
    - The form will use error providers instead of message boxes to warn the user when clicking button Display or Remove without a house selected.
    - **The class must handle the HouseListUpdated event:** the displayed house objects should be updated when a house is added, removed or modified
5. Each class can have other Private members, but cannot have any other Friend or Public members.

## Project Prog3GUI

1. This is a Windows Application project and has three classes: FormClassBranch, Prog3, and FormClassMDI.
2. The startup object must be Sub Main or class Prog3.
3. Class Prog3
    - The class has one Sub Main, which runs the application with an object of FormClassMDI.
    - The class can not have any variables, even inside Sub Main.
4. Class FormClassBranch
    - This is a sub-class of class FormClassHouse by Visual Inheritance.
    - The form class must have a Public WriteOnly property BranchNo to set the branch number, which is an integer.
    - The title of the form is "Branch #N", where N is the branch number. The title of _frmList of the class should be the same.
    - Button EXIT becomes button "UNLOAD", and clicking the button will close the form and its _frmList, but will not terminate the application.
5. Class FormClassMDI
    - This is an MDI form with a ToolStrip and a StatusStrip. The title of the form is "Prog3" followed by your name, and it should be maximized at the beginning.
    - When the form is loaded at the beginning, a form object of FormClassHouse is created as a child form and displayed.
    - The ToolStrip has two ToolStripButtons followed by a ToolStripSeparator followed by a ToolStripDropDownButton.
      - Each button should have an image.
      - The first button also displays text "NEW" and will create and show a new occurrence of class FormClassBranch.
      - The branch number starts with one and increments every time a new form is created.
      - The second button displays text "EXIT" and will terminate the application when clicked.
      - The DropDownButton displays text "WINDOWS" and has four DropDown menu items: Cascade, TileHorizontal, TileVertical and ArrangeIcons. Each menu item will arrange all form occurences accordingly.
      - The shortcut keys for the the DropDown menu items are [CTL]+C, [CTL]+H, [CTL]+V, and [CTL]+A.

- All created forms, including the list forms, are MDI child form of the MDI form.
- The StatusStrip has a panel of width 100 to the left end of the StatusStrip, displaying "UWP - CS3340". There is another panel of width 150 to the right end of the StatusStrip, displaying the current date and time, which is modified every 3 seconds.

**Submission**

1. Drop your solution folder to folder Prog3 of the class DROP folder by the due time.
2. You may lose up to two points for incorrect submission, including incorrect folder name.
3. You must follow the programming rules, and you may lose up to three points on style.