

Design Guide: Classroom Program Requirements and Naming Standards(C#)

Note:

Programs submitted that fail, as submitted, to compile are an Automatic Grade of Zero (0). Test your programs before you submit them.

Note:

Programs submitted with only form design elements, but no appreciable code to meet the Project requirements, are an Automatic grade of Zero (0) without further review. No partial credit will be given.

Project Specifications

A. The following must be done in all C# Projects/Projects as it applies to each Project/Project.

1. Use the **Hungarian** notation with **Pascal** case (aka: upper CamelCase) to name all internal variable names, controls, etc. and make the names of them meaningful/representative of what they do or contain. **EG: intLoopCounter, decMoneyPaid, strFirstName, btnClickMe, lblLastNamePrompt, getLastName(), setFirstName(strFirstName)**
2. All project requirements must be met.
3. The first or primary form (Object/Class) and form file (frmMain.vb) in a project must always be named "**frmMain**" (this applies to the **file name, class name** and **form name**).
4. **All** controls (IE: Projectels, Textboxes, etc.) on a form must be prefixed and named to classroom standards. (See course **Prefixes** Guide)
5. When using **TRY/CATCH** validate each field individually, do not excessively group code together in the same Try/Catch. **Try/Catches** should be used in conjunction with **IF/ELSE** statements (once **IF/ELSE** has been covered as a topic).
6. When converting numeric data types, **.TryParse** will be used. Additionally, **.TryParse** should be used in conjunction with **IF/ELSE** statements, once **IF/ELSE** has been covered as a topic. [IE: if **Integer.TryParse**(tbxNumber.text, intMyNumber) then...].
7. Tabs Stops must be set from left to right across the screen and then top to bottom of the screen unless otherwise dictated by the Project/project.
8. All Forms **must** be centered on the screen unless specified by the Project not to be centered.
9. Maximize Box (button) will not be displayed unless required by the Project.
10. Forms **will NOT** have a sizable border unless required (or appropriate) for the Project.
11. **Ensure there are no UNHANDLED EXCEPTIONS** when doing error proofing, validating or calculating. (To include form load/startup). Unhandled exceptions have a substantial negative effect on your grade and could result in a grade of Zero.
12. Comments must be placed inside each **EVENT, FUNCTION, or PROCEDURE** briefly describing what it is doing.
13. **DO NOT COMMENT EACH LINE.** You may comment lines as you need to help you design the program, but do not comment each line in what you submit for grading.
14. There will be **no empty functions/methods** (functions/methods with no code).
15. All programs, unless otherwise specified by the instructor or the Project/Project, must include an **"Exit"** button with the code **"Application.Exit();"**.

16. All **MessageBox**'s must be fully formed. (All four elements: Text, Title, Button, Icon).

Example:


```
MessageBox.Show("Text to Display", "Title to Display", MessageBoxButtons.OK, MessageBoxIcon.Information);
```

17. All forms and output must be professional in appearance, function and flow.

18. No Excessive "empty space" on forms or output.

19. All forms must have an appropriate text/title for the Project.

20. All forms must have ControlBox unless otherwise specified by the Project or instructor.

21. All forms must have Minimize  Button unless otherwise specified by the Project or instructor.

22. The following remarks are required in the frmMain of every project you submit. Failure to include them will adversely affect your grade. The areas containing {} should be replaced with the appropriate required entries. (IE: 'Programmer: James Flippin):

```
*****
```

```
*****
```

```
'Programmer: {Your name goes here}'
```

```
'Course: INEW 2332.xxxx (Final Project)
```

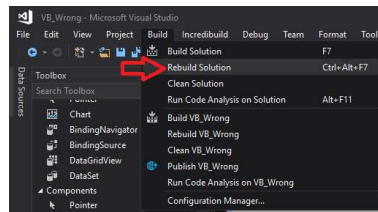
```
'Program purpose: {Program Purpose Goes here}'
```

```
*****
```

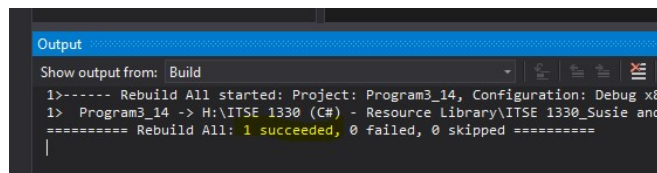
```
*****
```

B. Building, Debugging and running your programs:

To test your code, First select "**Build**" from the Menu, then select "**Rebuild Solution**".

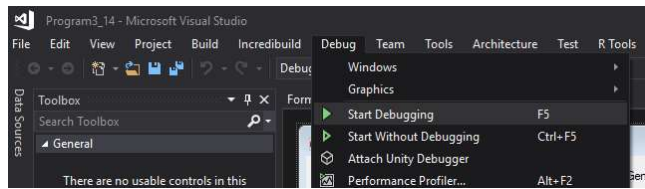


Check the **Output window** for errors, if errors were found. Stop and correct them. Then rebuild again. Repeat this process until there are no more errors.



If a line of code is in error, you can double click on the error message in the output window to be taken directly to that line of code.

If there are no errors, you can run the program by either selecting "**Start Debugging**" or "**Start Without Debugging**" depending upon what you are trying to do. Until you have fully "debugged" your program, you should be using "**Start Debugging**". Please note the Keyboard Shortcut (F5 – Start Debugging, Ctrl+F5 Without debugging) keys that are available for each of these.



C. Control/Object Name Prefixes (Use the Hungarian notation with Pascal aka upper CamelCase):

<u>Object</u>	<u>Prefix</u>	<u>Description</u>
Button	btn	Presents a standard button that the user can click to perform actions. IE: btnClickMe.
CheckBox	cbx	Indicates whether a condition is on or off. IE: cbxPurchaseInsurance
CheckedListBox	clb	Displays a list of items with a check box next to each item. IE: clbSelectedCampsites
ColorDialog	clr	Allows the user to select a color from a palette in a pre-configured dialog box and to add custom colors to that palette. IE: clrPickColor
ComboBox	cbo	Displays data in a drop-down combo box. IE: cboCityName
ContextMenu	ctx	Provides users with an easily accessible menu of frequently used commands that are associated with the selected object. IE: ctxClipboardOptions
DataGrid	dgrd	Displays tabular data from a dataset and allows for updates to the data source. IE: dgrdEmployees
DataGridView	dgv	Provides a flexible extensible system for displaying and editing tabular data. IE: dgvStaffMembers
DateTimePicker	dtp	Allows the user to select a single item from a list of dates or times. IE: dtpStartDate
FileDialog	fil	Class Provides base-class functionality for file dialog boxes. IE: filOpenInventoryFile
FolderBrowserDialog	fldr	Displays an interface with which users can browse and select a directory or create a new one. IE: fldrLocateInventory
FontDialog	fnt	Exposes the fonts that are currently installed on the system. IE: fntSelectFont
Form	frm	A window form (contains thing such as buttons check boxes text). Initial forms in these classes should be named frmMain
GroupBox	gbx	Provides an identifiable grouping for others. IE: gbxEmployeeDetails
HelpProvider	hlp	Associates an HTML Help file with a Windows-based application. IE: hlpUserHelp
ImageList	img	Used with image display IE: imgEmployeePicture
Label	lbl	Displays text that cannot be edited/changed by the user. IE: lblLastName
ListBox	lbx	Allows the user to select one or more items from a predefined list. IE: lbxCountyNames

ListView	lv	Displays a list of items with icons in the manner of Windows Explorer. IE: lvStaffSelections
MaskedTextBox	msk	Constrains the format of user input in a form. IE: mskPaymentAmount
MenuStrip	mnu	Provides a menu system for a form. IE: mnuMainMenu
Modules	mod	Modules (code). IE: modCalculations
MonthCalendar	cal	Presents an intuitive graphical interface for users to view and set date information. IE: calSelectDate
OpenFileDialog	ofd	Allows users to open files by using a pre-configured dialog box. IE: ofdOpenTextFile
PageSetupDialog	psd	Sets page details for printing through a pre-configured dialog box. IE: psdPageSetup
Panel	pnl	Provide an identifiable grouping and allows for scrolling. IE: pnlPolicyChoices
PictureBox	pbx	Displays graphics in bitmap GIF JPEG metafile or icon format. IE: pbxEmployeePhoto
PrintDialog	pd	Selects a printer chooses the pages to print and determines other print-related settings. IE: pdPrinter
PrintDocument	pdoc	Sets the properties that describe what to print and prints the document in Windows-based applications. IE: pdocPayroll
PrintPreview	ppv	Allows you to create your own PrintPreview or dialog box instead of using the pre-configured version. IE: ppvPrintPayroll
PrintPreviewDialog	ppd	Displays a document as it will appear when it is printed. IE: ppdPreviewPayroll
ProgressBar	pbar	Graphically indicates the progress of an action towards completion. IE: pbarLoadingStatus
RadioButton	rad	Presents a set of two or more mutually exclusive options to the user. IE: radPremiumCustomer
RichTextBox	rtb	Allows users to enter display and manipulate text with formatting. IE: rtbReportWriteup
SaveFileDialog	sfd	Selects files to save and where to save them. IE: sfdFileSave
StatusStrip	ss	Represents a Windows status bar . Although StatusStrip replaces and extends the StatusBar of previous versions StatusBar is retained for both backward compatibility and future use if you choose. IE: ssMain
TextBox	tbx	Allows editable multiline input from the user. IE: tbxLastName
Timer	tmr	Raises an event at regular (clock like) intervals. IE: tmrFadeOut
ToolStrip	ts	Creates custom toolbars and menus in your Windows Forms applications. IE: tsQuickAccess

ToolStripProgressBar	spb	Graphically indicates the progress of an action towards completion. The ToolStripProgressBar is typically contained in a StatusStrip. IE: spbCompletionStatus
ToolStripStatusLabel	ssl	Represents a panel in a StatusStrip. IE: sslFeedbackMessage
ToolTip	tt	Displays text when the user points at others. IE: ttExpectedEntries

***Common (C#) variable/object prefixes**

Arrays	arr	Arrays (subscripted). IE: arrEmployees()
Collections	col	Collections. IE: colAddresses()
List	lst	List (Item lists). IE: lstAddresses()
StreamReader	sr	Stream File Reader. IE: srInputFile
StreamWriter	sw	Stream File Writer. IE: swOutputFile

<u>Data Type</u>	<u>Prefix</u>	<u>Example</u>
Boolean	bol	blnFoundIt
Byte	byt	bytRasterData
Character	chr	chrMiddleInitial
Collection	col	colBooks
Date(Time)	dt	dtGradDate or dtmGradDate
Decimal	dec	decNetPay
Double	dbl	dlbTolerance
Integer	int	intNoDecimalNumber
Long	lng	lngBigNumber
Float	flt	fltBudget
Object	obj	objChair
Short	shrt	shrtCount
Single	sng	sngAverage
String	str	strFirstName

<u>Data Type</u>	<u>Prefix</u>	<u>Example</u>
Boolean	bln	blnFoundIt
Byte	byt	bytRaster
Short Byte	sbyt	sbytRaster
Character	char	charMiddleInitial
Decimal	dec	decNetPay
Double	dbl	dblTolarance
Float	flt	fltBudget
Integer	int	intLoopCount
Unsigned Integer	uint	uintNoNegativeCount
Long	lng	lngBigNumber

Unsigned Long	ulng	ulngBigNumNoNeg
Short	shrt	shrtCount
Unsigned Short	ushrt	ushrtNoNegCount