Coding Standards

**Variable Names (Private Variables)**

Variables should be first character of all words is upper case and the others are lower case, except for the first word that is lower case.

Variables should begin by identifying its type using the notation on the table.

|  |  |
| --- | --- |
| Type | Notation |
| Int | int |
| Bool | bln |
| String | str |
| Float | flt |
| Char | chr |
| Double | dbl |
|  |  |

Example:

bool blnCanDoSomething;

Public Properties

Properties should be first character of all words is upper case and the others are lower case. Properties must be complete names without abreviatures.

Example:

public int NumberOfTasks {get;set;}

**Method Names**

Methods should have the first character of all words is upper case and the others are lower case. Methods must be complete names without abreviatures.

void SendMessage()

{

}

**Arguments**

Arguments should have the first character of all words is upper case and the others are lower case. Arguments must be complete names without abreviatures.

void SendMessage(string Message)

{

}

**Visual Object Names (Components)**

Components should be first character of all words is upper case and the others are lower case, except for the first word that is lower case.

Components should begin by identifying its type using the notation on the table.

|  |  |
| --- | --- |
| Type | Notation |
| Textbox | txt |
| Label | lbl |
| Image | img |
| CheckBox | chk |
| Spin | spn |
| Panel | pnl |
| SplitContainer | spc |
| Button | btn |
| GroupBox | grp |
|  |  |
|  |  |

Example:

chkShowResults

**Class Names**

Classes should have the first character of all words is upper case and the others are lower case. Classes must be complete names without abreviatures.

**Comments**

Good, readable code will require very few comments. If all variables and method names are meaningful, that will make the code very readable and it will not need much commenting.

If you have to use some complex or weird logic for any reason, document it very well with sufficient comments.

If you initialize a numeric variable to a special number other than 0, -1, etc., document the reason for choosing that value.

**Enumerations**

Enumerations must be used like Classes and Properties.

**Exception Handling**

Never do a "catch exception and do nothing." If you hide an exception, you will never know if the exception happened or not. In the case of exceptions, give a friendly message to the user, but log the actual error with all possible details about the error, including the time it occurred, the method and class name, etc.