**Coding Standard v 1**

**Pattern: Scope [type-(Class,void,data type)] Name**

**1.** public class ThisIsClassName

2. public static class ThisIsStaticClass

1. Casing
2. Method

-Use PascalCase.

- If a method should return a value, then prefix it with “Get”

Ex:

ThisIsSubMethod()

**Get**Age(string studentID)

1. Variables

-Use camelCase.

-Class level variables should prefix with “\_” underscore, except properties.

-Constant variable

\*All UPPERCASE

\*Separate it with “\_” underscore.

\*Use **only** in class level.

public class SampleClass

{

private int \_**classLevel**;

private const int THIS\_IS\_CONSTANT = 123;

private void ThisIsMethod(int thisIsParameter)

{

int thisIsMethodLevel;

}

}

\*Use predefined variable types: string, int

Don’t: String name;

Int32 age;

Do’s: string name;

int age;

1. Add Summary

Don’t forget to add summary in Class, methods and Properties.

/// <summary>

/// This is sample method

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

ThisIsSubMethod(object sender, EventArgs e)

1. Scope Declaration

-Public variables

\*Enclose in a Property

\*Exception is a CONSTANT

private int \_doNotAccessThis;

public int AccessThis

{

get { return \_doNotAccessThis; }

set { \_doNotAccessThis = value; }

}

Or

public int AccessThis

{get;set;}

public const int ThisIsContanct = 123;

-Session variables/Viewstate

\*Enclose in a Property

public int Age

{

get { return Convert.ToInt32(ViewState["Age"]); }

set { ViewState["Age"] = value; }

}

1. Controls
2. Button – btnOne
3. TextBoxes – txtOne
4. Dropdownlist – drpOne
5. Combobox – cmbOne
6. TextArea – areaONe
7. RadioButton – rbtn
8. RadionButtonList – rbtnlOne
9. Checkbox – chkOne
10. CheckboxList – chklOne
11. Html Table – tblOne
12. Html Table rows – trowOne
13. Html Table cell –tcellOne
14. Listview – lvwOne
15. Datagrid – grdOne
16. Gridview – grdOne
17. Datetimepicker = dteOne

**Database:**

**Pattern:** Dbo.[group]\_TableName – dbo.core\_student

* Do **not** postfix with “s” on table: dbo.core\_student**s. (use dbo.core\_student)**
* Use underscore to separate names: dbo.core\_student\_grade.
* View prefix with “v” dbo.core\_**v**student\_subject, vstudentinfo
* Function prefix with “fn” dbo.core\_fngrade.
* Stored Procedure prefix with “sp” dbo.core\_spprocess\_grades.
* Triggers prefix with “tr” dbo.core\_trprocess\_grades.