CSharp Coding Convention

# Naming Conventions

|  |  |  |
| --- | --- | --- |
| type | Way to name | Examples |
| Namespace | PascalCasing, tổ chức namespace với cấu trúc rõ ràng, cùng tên với folder | namespace Company.Product.Module.SubModule  namespace Product.Module.Component  namespace Product.Layer.Module.Group |
| Interface | IPascalCasing | public interface IShapeCollection  {  } |
| Class name | PascalCasing, sử dụng danh từ, cùng tên với tên file cs | public class ClientActivity  {  public void ClearStatistics()  {  //...  }  public void CalculateStatistics()  {  //...  }  } |
| Method names | PascalCasing |
| Method Arguments | camelCassing | public class UserLog  {  public void Add(LogEvent logEvent)  {  int itemCount = logEvent.Items.Count;  // ...  }  } |
| Local variable | camelCassing |
| Identifiers (định danh) | Không sử dụng Underscores(dấu gạch dưới) | // Correct  int counter;  string name;    // Avoid  int iCounter;  string strName; |
| Private Identifiers | Underscores prefix | private DateTime \_registrationDate; |
| constants | CAPITALIZED | // Correct  public static const string ShippingType = "DropShip";    // Avoid  public static const string SHIPPINGTYPE = "DropShip"; |
| Read-only variables | CAPITIALIZED |
| Abbreviations (viết tắt) | Tránh sử dụng nhưng cho phép một số ngoại lệ như **Id, Xml, Ftp, Uri,…** Sử dụng PascalCasing cho 3 kí tự trở lên. Nếu 2 kí tự thì viết hoa cả hai.  Eg:FtpTransfer ftpTransfer;  UIControl uiControl; | **// Correct**  UserGroup userGroup;  Assignment employeeAssignment;    // Avoid  UserGroup usrGrp;  Assignment empAssignment;    // Exceptions  CustomerId customerId;  XmlDocument xmlDocument;  FtpHelper ftpHelper;  UriPart uriPart; |
| Biến trong class | Khai báo tất cả các biến ở đầu class, biến tĩnh xếp cao nhất. | // Correct  public class Account  {  public static string BankName;  public static decimal Reserves;    public string Number {get; set;}  public DateTime DateOpened {get; set;}  public DateTime DateClosed {get; set;}  public decimal Balance {get; set;}    // Constructor  public Account()  {  // ...  }  } |
| Enums | Sử dụng tên duy nhất cho enum | // Correct  public enum Color  {  Red,  Green,  Blue,  Yellow,  Magenta,  Cyan  } |
| Biếu kiểu người dùng tạo ra | Khai báo kiểu dữ liệu var cho những biến này | var stream = File.Create(path);  var customers = new Dictionary();    // Exceptions  int index = 100;  string timeSheet;  bool isCompleted; |

AVOID

* System name (Int16, Single, and UInt64). Use Predefined type names.

// Correct

string firstName;

int lastIndex;

bool isSaved;

// Avoid

String firstName;

Int32 lastIndex;

Boolean isSaved;

* use Underscores in identifiers

// Correct

public DateTime clientAppointment;

public TimeSpan timeLeft;

// Avoid

public DateTime client\_Appointment;

public TimeSpan time\_Left;

* suffix enum names with Enum

public enum CoinEnum

{

Penny,

Nickel,

Dime,

Quarter,

Dollar

}

// Correct

public enum Coin

{

Penny,

Nickel,

Dime,

Quarter,

Dollar

}

# [Layout Conventions](javascript:void(0))

* Sử dụng những cài đặt Code Editor mặt định( tự thụt dòng, lùi vào 4 kí tự/1tab,…)
* Viết mỗi câu lệnh trên mỗi dòng.
* Viết mỗi câu khai báo trên mỗi dòng.
* Nếu Code Editor không tự động thụt đầu dòng, sử dụng 1 tab stop(4 kí tự trống).
* Thêm ít nhất 1 khoảng trắng giữa mỗi phương thức và các định nghĩa property.
* Sử dụng dấu ngoặc nhọn để làm một mệnh đề rõ ràng

if ((val1 > val2) && (val1 > val3))

{

// Take appropriate action.

}

## Tabs & Indenting

Nên sử dụng 4 kí tự trắng thay vì 1 tab(\0x09)

## File Organization

Tên các thành viên trong một class sắp xếp A-Z

1. **Commenting**

* Đặt comment trên một dòng, không để cuối dòng code.
* Viết hoa đầu dòng, chấm cuối câu
* Thêm một khoảng trắng sau dấu “//”. Ví dụ:

// The following declaration creates a query. It does not run

// the query.

* Đừng tạo ra khối comment /\*\*/
* Đặt trước dòng cần ghi chú.
* Trong một số trường hợp, vẫn có thể đặt ở cuối dòng: vd

public class SomethingUseful   
{  
 private int itemHash; // instance member  
 private static bool hasDoneSomething; // static member  
}

# Control Name Convention

|  |  |  |
| --- | --- | --- |
| **Control** | **Prefix** | **Example** |
| Label | lbl | lblSurname |
| Form | frm | FrmEntry |
| TextBox | txt | txtSurname |
| DataGrid | dg | dgResults |
| GridView | gv | gvResults2 |
| Button | btn | btnSave |
| ImageButton | iBtn | iBtnSave |
| Hyperlink | lnk | lnkHomePage |
| DropDownList | ddl | ddlCompany |
| ListBox | lst | lstCompany |
| DataList | dLst | dlstAddress |
| DataSet | ds | dsInvoices |
| DataTable | dt | dtClients |
| DataRow | dr | drUser |
| Repeater | rep | repSection |
| Checkbox | chk | chkMailList |
| CheckBoxList | chk | chkAddress |
| RadioButton | rBtn | rBtnSex |
| RadioButtonList | rBtn | rBtnAgeGroup |
| Image | img | imgLogo |
| Panel | pnl | pnlSevtion |
| PlaceHolder | plh | plhHeader |
| Calendar | txt | txtMyDate |
| AdRotator | adr | adrBanner |
| Table | tbl | tblResults |
| [All] Validators | val (N/A) | valCreditCardNumber |
| ValidationSummary | vals (N/A) | valsErrors |

Tham khảo thêm <http://en.wikibooks.org/wiki/C_Sharp_Programming/Naming>

Bổ sung : đặt tên form:

|  |  |  |
| --- | --- | --- |
| Form | frm | frmEntry |