

# C# Coding Conventions, Rules, Best Practices & Better Coding Standards

## C# Coding Conventions and Rules

Rules are always there to follow. Its a different thing, what is the level you are following the rules. Coding conventions are followed for

- Consistent look and feel of the code
- Easy to read and easy to follow code
- Easy to understand
- Easy maintainability of the code

Below are some examples of our C# coding standards, naming conventions, and best practices.

### Pascal Casing

The first letter in the identifier and the first letter of each subsequent concatenated word are capitalized.

### Camel Casing

The first letter of an identifier is lowercase and the first letter of each subsequent concatenated word is capitalized.

**Do use** PascalCasing for class names and method names.

// class Product

**Do use** camelCasing for method arguments and local variables.

//public void Save(Product productCatagory)

**Do not use** Abbreviations.

// **Correct**

ProductCategory productCategory;

// **Avoid**

ProductCategory prodCat;

**Do not use** Underscores in identifiers.

// **Correct**

ProductCategory productCategory;

// **Avoid**

ProductCategory product\_Category;

**Do prefix** interfaces with the letter I.

public interface IAddress

{

}

**Do declare all** member variables at the top of a class, with static variables at the very top.

public class Product

{

public static string BrandName;

public string Name{get; set;}

public DateTime DateAvailable {get; set;}

```
public Product()  
{  
}
```

**Do use singular** names for enums. Exception: bit field enums.

```
public enum Direction  
{  
    North,  
    East,  
    South,  
    West  
}
```

**Do not suffix** enum names with Enum.

**//Avoid**

```
public enum DirectionEnum  
{  
    North,  
    East,  
    South,  
    West  
}
```

Along with this few other things are there that should be taken care while coding.