C# Coding Standards

Capitalization Guidelines

John F. Gnazzo

A software coding standard is a set of rules and guidelines for the formatting and organizing source code. Software coding standards are used to define a specific programming style.

In professional environments, the benefits of coding standards include readability, maintainability and compatibility. Any member of a development team needs to be able to quickly read and understand the code of another member. The developer who maintains a piece of code tomorrow may not be the developer who programmed it today. Many of today's enterprise solutions are so complex that multiple development teams commonly unite to build a singular enterprise software application. With coding standards in place, disparate teams can rely on the way that they can interface with the code built by a separate team.

This blog, the second in a series of articles discussing coding standards, will discuss **Capitalization Guidelines** in C#, a fourth generation language commonly used by developers to develop desktop, mobile and web applications. The blog will also address the benefits of using appropriate capitalization to support software code readability, understandability for developers and those maintain software code.

Types of Capitalization Conventions

Three basic naming conventions are used in C# programming and include:

- Pascal Case
- Camel Case
- Upper Case

Pascal Case

Pascal Casing involves capitalization of the first letter in the identifier and the first letter of each subsequent concatenated word. Generally, Pascal case can be used for identifiers of three (3) or more characters. For example:

GrainElevator

Camel Case

Camel casing involves using lowercase for the first letter of an identifier and capitalizing the first letter of each subsequent concatenated. For example:

fooBar

Upper Case

Upper casing involves capitalization of all letters in the identifier. Use this convention only for identifiers that consist of two or fewer letters. For example:

Data.IO

The following table illustrates the proper naming convention based upon Microsoft Standards (1).

Identifier	Casing	Example
Namespace	Pascal	namespace System.Security { ... }
Type	Pascal	public class StreamReader { ... }
Interface	Pascal	public interface IEnumerable { ... }

Method	Pascal	public class Object {
		public virtual string ToString();
		}

Property	Pascal	public class String {
		public int Length { get; }
		}

Event	Pascal	public class Process {
		public event EventHandler Exited;
		}

Field	Pascal	public class MessageQueue {
		public static readonly TimeSpan InfiniteTimeout;
		}
		public struct UInt32 {
		public const Min = 0;
		}

Enum value	Pascal	public enum FileMode {
		Append,
		...
		}

Parameter	Camel	public class Convert {
		public static int ToInt32(string value);
		}

Benefits of Capitalizations Conventions

The benefits of using a capitalization convention in software code include .code readability, maintainability and compatibility. This Blog. illustrated a practical coding standard for capitalization.

References

1. *Capitalization Conventions.* **Microsoft.** s.l. : Microsoft Corporation, 2015.