coding standarads in C#

General Naming Conventions

**General Naming Conventions: Pascal Case for Types: Class and method names should be in Pascal Case. For example:**

**public class Employee { }**

**public void GetDetails() { }**

**Camel Case for Parameters and Variables: Method arguments and local variables should use camel case. For example:**

**public void PrintDetails(int employeeId, string firstName)**

**{**

**string fullName = firstName + " Doe";**

**Console.WriteLine(fullName);**

**}**

**Avoid Underscores:  Avoid the use of underscore while naming identifiers  . For example:**

**public string lastName; // Correct**

**public string last\_Name; // Avoid this**

**Prefix Interfaces with 'I': Interface names should start with the letter 'I'. For example:**

**public interface IEmployee { }**

**Constants in Upper Case: Constants should be declared in uppercase letters. For example:**

**public const int MAX\_AGE = 60;**

**Avoid the use of System data types and prefer using the Predefined data types.**

**int employeeId;**

**string employeeName;**

**bool isActive;**

**Indentation and Braces: Maintain consistent indentation and align curly braces vertically for better readability. For example:**

**class Employee**

**{**

**void PrintDetails()**

**{**

**}**

**}**

**Declare Variables Close to Their Use:** **Always declare variables as close as possible to their use.**

**int age = 25;**

**Console.WriteLine(age);**

**Always declare the properties as private so as to achieve Encapsulation and ensure data hiding.**

**class Person**

**{**

**private string name; // Private field to store the name**

**// Public property to access and modify the private field**

**public string Name**

**{**

**get { return name; } // Getter to retrieve the value of the private field**

**set { name = value; } // Setter to modify the value of the private field**

**}**

**}**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**Person myObj = new Person();**

**myObj.Name = "Liam"; // Use the public property to set the value**

**Console.WriteLine(myObj.Name); // Use the public property to get the value**

**}**

**}**

**Separate Methods and Sections by One Space:** **Always separate methods and different sections of a program by one space for better readability.**

**class Employee**

**{**

**// Private property**

**private int employeeId { get; set; }**

**// Public method**

**public void PrintDetails()**

**{**

**// Implementation of PrintDetails**

**}**

**}**

**Error Handling:**

**try**

**{**

**// Code that might throw an exception**

**}**

**catch (SpecificException ex)**

**{**

**// Handle specific exception**

**}**