**Estándar de codificación en C#.**

**Ajustes y estándares de nombres.**

* Notación pascal: El primer carácter de todas las palabras en mayúsculas y los otros caracteres en minúsculas. (Exclusivo solo para declaración de clases)
  + Ejemplo. HolaMundo.
* Notación CamellCase: El primer carácter de todas las palabras, excepto de la primera palabra en mayúsculas y los otros caracteres en minúsculas.
  + Ejemplo. diasDelMes.
* Utilizar palabras entendibles y descriptivas. No usar abreviaciones.
* No declarar variables con solo una letra.
  + Ejemplo. i,n,x,j.
* Una excepción de la utilización de declaración en una sola letra pude ser en ciclos como for, while, do while etc…
  + Ejemplo. for(int i =0; i<numeroMaximo; i++){

}

* No utilizar guiones bajo para la declaración de variables.
* Usa el prefijo apropiado para cada elemento de la Interfaz Gráfica de manera que puedas identificarlos para el resto de las variables.
  + Una recomendación es utilizar el prefijo de IU para los elementos de la interfaz.
* Nombre de los archivos deben de coincidir con el nombre de la clase.

**Sangría y espacios.**

* Utilizar TAB para la sangría, no usar espacio.
* Los comentarios deben estar al mismo nivel que el código (usar el mismo nivel de sangría).
* Las llaves deben de estar al nivel en el que se declara el método, ciclo, etc…
* Utiliza un espacio después de la declaración de cada método.
* Evita escribir métodos muy largos. Un método debe típicamente tener entre 1 a 25 líneas de código.
* Los nombres de los métodos deben decir lo que hacen.
  + Ejemplo: void GuardarNombreDelDocente.
* Un método debe de tener una sola tarea.
* Evitar utilizar muchas variables, en la medida posible utiliza locales a menos que sean necesarias para otros métodos.
* Los mensajes de error deben de ayudar al usuario a resolver el problema.
  + Ejemplo. “Fallo en la actualización de la base de datos”.
* Evitar archivos muy grandes.
* Evita métodos y propiedades públicas, a menos que ellas realmente necesiten ser accedidas desde afuera de la clase.
* No utilizar muchos parámetros en un método.
* Organizar lógicamente los archivos dentro de carpetas apropiadas.
* Si estas abriendo conexiones a una base de datos, sockets, archivos etc, siempre cierra dichas conexiones en el bloque finally.
* Declarar variables lo más cerca posible de donde son usadas. Usa una declaración de variable por línea.