# **TITULO PORTADA TRABAJO**

# Ejemplo de subtítulo

NOMBRE: Luis Jiménez

CARRERA: Analista Programador

ASIGNATURA: Programación Avanzada de Aplicaciones

PROFESOR: Fernando Reveco

FECHA: 20 de abril 2022

Listas: Lista es muy parecida a la clase ArrayList, que era la opción de ir a la lista antes de que C # admitiera listas genéricas. Por lo tanto, también verá que List puede hacer muchas de las mismas cosas que una Matriz. Las listas son mas simple y fácil de trabajar a diferencia del ArrayList las listas no se tiene que crear una Lista con un tamaño específico; en su lugar, puede crearla y .NET la expandirá automáticamente para ajustarse a la cantidad de elementos a medida que los agregue.

Ejemplo como se crea una lista:

List<string> lista1 = new List<string>();

El comando anterior crea una lista vacía para agregar algo se utiliza el método Add.

listat1.Add("a string");

Ejemplo

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Pila (Stack)

Una pila permite introducir un nuevo elemento en la cima y quitar el elemento que hay en la cima utilizando el push y pop.

Para utilizar la clase "Stack" y la mayoría de las que veremos en este tema, necesitamos incluir en nuestro programa una referencia a "System.Collections".

Ejemplo

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Colas (Queue)

Las colas nos permiten almacenar información de tipo FIFO es decir las que entran primero salen primero

Alternativamente en .NET tenemos el tipo Stack<T> que almacena la información en un estilo LIFO (el último que entra sale el primero)

Ejemplo de como añadir elementos a la cola

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Peek, el cual es un método que nos devuelve el valor del primer elemento, pero no lo remueve de la cola.

Dequeue el cual devuelve el elemento y lo elimina de la cola

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Diccionario

Es una colección que te la opción de almacenar datos de una forma única como clave-valor que usan tipo específicos genéricos tanto para la clave como para el valor. Antes de esto se usaba el hashtable pero era más complicado realizar el guardado de objetos. El cual con Dictionary es más simple y es una ventaja que implemento con la llegada de genéricos para llamar a dictionary es “dictionary<tkey, tvalue>”

Ejemplo: //inicialización y crear a valores

Dictionary<string, string> comunidadesCapitales = new Dictionary<string, string>()

{

{"Aragon", "Zaragoza"},

{"Navarra", "Pamplona"}

};

//añadir valores

comunidadesCapitales.Add("Castilla la mancha", "Toledo");