INSTITUTO KHIPU

Semestre III

Sesión 11

PROGRAMA DE ESTUDIOS

DESARROLLO DE SISTEMAS DE INFORMACIÓN

HERRAMIENTAS DE PROGRAMACIÓN – C#

Tema:

CLASE GENERICA LISTA

CLASE GENERICA LISTA

En C#, la clase genérica List<T> es una estructura de datos poderosa que forma parte del espacio de nombres System.Collections.Generic. Se utiliza para almacenar colecciones de elementos de un tipo específico, y su tamaño puede crecer o reducirse dinámicamente, a diferencia de los arrays, cuyo tamaño es fijo.

A lo largo de este artículo, te explicaré de manera clara y precisa qué es una lista genérica, cómo se usa, y algunos ejemplos prácticos para que puedas comprender su funcionamiento.

¿Qué es List<T>?

List<T> es una colección genérica que permite almacenar elementos del tipo especificado por T. El uso de la clase List facilita el manejo de conjuntos de datos que pueden cambiar en tamaño, ya que es dinámica. A diferencia de un array que necesita tener un tamaño fijo, una lista genérica puede crecer o reducirse según se agreguen o eliminen elementos.

Características Principales de List<T>

1. **Genérica**: Es fuertemente tipada, lo que significa que todos los elementos en la lista deben ser del mismo tipo, definido en el momento de su creación (por ejemplo, List<int> o List<string>).

2. **Mutable**: Puedes agregar, modificar y eliminar elementos de la lista después de que ha sido creada.

3. **Tamaño dinámico**: Se ajusta automáticamente conforme agregas o eliminas elementos, sin necesidad de definir un tamaño fijo.

4. **Indexada**: Los elementos de la lista se pueden acceder mediante un índice, comenzando desde 0.

5. **Permite duplicados**: A diferencia de otras colecciones como HashSet, las listas permiten almacenar elementos duplicados.

¿Cómo Crear una Lista Genérica?

La creación de una lista en C# es muy sencilla. Debes especificar el tipo de dato que deseas almacenar en la lista, usando la sintaxis genérica List<T>.

csharp

Copiar código

```
List<int> numeros = new List<int>();  // Lista de enteros

List<string> nombres = new List<string>();  // Lista de cadenas de texto
```

Métodos Principales de List<T>

A continuación, te mostraré los métodos más importantes y comunes que puedes utilizar para manipular una lista genérica en C#:

1. `Add(item)`: Agrega un elemento al final de la lista.

```csharp
numeros.Add(10);  // Agrega el número 10 a la lista
```

2. `Insert(index, item)`: Inserta un elemento en una posición específica de la lista.

```csharp
numeros.Insert(1, 5);  // Inserta 5 en la posición 1
```

3. `Remove(item)`: Elimina la primera ocurrencia de un elemento específico.

```csharp
numeros.Remove(10);  // Elimina el primer 10 encontrado en la lista
```

4. `RemoveAt(index)`: Elimina el elemento en una posición específica.

```csharp
numeros.RemoveAt(0);  // Elimina el elemento en la posición 0
```

5. `Contains(item)`: Verifica si un elemento existe en la lista.

```csharp
bool existe = numeros.Contains(5);  // Devuelve true si 5 está en la li
```

6. `Count`: Devuelve el número total de elementos en la lista.

```csharp
int cantidad = numeros.Count;  // Devuelve la cantidad de elementos en l
```

7. `Clear()`: Elimina todos los elementos de la lista.

```csharp
numeros.Clear();  // Vacía la lista
```

8. `Sort()`: Ordena los elementos en orden ascendente.

```csharp
numeros.Sort();  // Ordena la lista de menor a mayor
```

9. `Reverse()`: Invierte el orden de los elementos en la lista.

```csharp
numeros.Reverse();  // Invierte el orden de la lista
```

10. `ToArray()`: Convierte la lista a un array.

```csharp
int[] arrayNumeros = numeros.ToArray();  // Convierte la lista en un ar
```

Ejemplo Práctico

A continuación, te dejo un ejemplo sencillo que muestra cómo utilizar varios de estos métodos en una lista genérica:

```
using System;
using System.Collections.Generic;

namespace ConsoleApp1

{
   class Program
   {
      static void Main(string[] args)
      {
         // Crear una lista de enteros
         List<int> numeros = new List<int>();

         // Agregar elementos a la lista
         numeros.Add(5);
         numeros.Add(10);
         numeros.Add(15);

         // Insertar un número en la posición 1
         numeros.Insert(1, 7);  // La lista ahora es [5, 7, 10, 15]

         // Eliminar un número
         numeros.Remove(10);  // La lista ahora es [5, 7, 15]

         // Verificar si un número está en la lista
         if (numeros.Contains(7))
         {
            Console.WriteLine("El número 7 está en la lista.");
         }

         // Ordenar la lista
         numeros.Sort();  // La lista ahora es [5, 7, 15]

         // Imprimir los números de la lista
         foreach (int numero in numeros)
         {
            Console.WriteLine(numero);
         }

      }
   }
}
```

INSTITUTO

KHIPU