

Uso de Genericos

Los genéricos introducen en .NET el concepto de parámetros de tipo, lo que le permite diseñar clases y métodos que aplazan la especificación de uno o varios tipos hasta que el código de cliente declare y cree una instancia de la clase o el método

```
// Declare the generic class.
public class Genericos<T>{
    public void Add(T input) { }
}
class TestGenericos {
    private class ExampleClass { }

    static void Main() {
        // Declare a list of type int.
        Genericos<int> list1 = new Genericos<int>();
        list1.Add(1);

        // Declare a list of type string.
        Genericos<string> list2 = new Genericos<string>();
        list2.Add("");

        // Declare a list of type ExampleClass.
        Genericos<ExampleClass> list3 = new Genericos<ExampleClass>();
        list3.Add(new ExampleClass());
    }
}
```

El tipo genérico se define como **T**. Cuando inicializamos el objeto sobre la clase indicando que **T** es un tipo de dato(string, int, float, otros tipos de clase) automáticamente en ese objeto, **T** es igual al tipo que hemos añadido.

Para que usar los genéricos?

- Use tipos genéricos para maximizar la reutilización del código, la seguridad de tipos y el rendimiento.
- El uso más común de los genéricos es crear clases de colección.
- La biblioteca de clases .NET Framework contiene varias clases de colección genéricas nuevas en el espacio de nombres System.Collections.Generic. Estas se deberían usar siempre que sea posible en lugar de clases como ArrayList en el espacio de nombres System.Collections.
- Puede crear sus propias interfaces, clases, métodos, eventos y delegados genéricos.
- Puede limitar las clases genéricas para habilitar el acceso a métodos en tipos de datos determinados.
- Puede obtener información sobre los tipos que se usan en un tipo de datos genérico en tiempo de ejecución mediante la reflexión.