

مفهوم التعميم باستخدام الجنيريكس Generics في البرمجة يعني إنك تقدر تكتب كود مرّن وقابل لإعادة الاستخدام مع أنواع بيانات مختلفة بدون ما تضطر تكتب نفس الكود لكل نوع بيانات على حدة.

مثلاً، لو عندك دالة بتعمل عملية معينة على مصفوفة من الأرقام، وعاليز تستخدم نفس الدالة على مصفوفة من النصوص، بدل ما تكتب دالة جديدة للنصوص، ممكن تستخدم الجنيريكس وتكتب دالة واحدة تشتغل مع أي نوع بيانات.

ببساطة، الجنيريكس ببساطة تكتب كود أكثر مرونة وكفاءة، وببقلل من تكرار الكود. يعني بتكتب الكود مرة واحدة وتستخدمه مع أنواع بيانات مختلفة.

```

public class GenericList<T>
{
    private T[] elements;
    private int count;

    public GenericList(int size)
    {
        elements = new T[size];
        count = 0;
    }

    public void Add(T element)
    {
        if (count < elements.Length)
        {
            elements[count] = element;
            count++;
        }
    }

    public T GetElement(int index)
    {
        if (index >= 0 && index < count)
        {
            return elements[index];
        }
        else
        {
            throw new IndexOutOfRangeException("Index out of range");
        }
    }
}

```

مثلاً، لو عندك دالة بتعمل عملية معينة على مصفوفة من الأرقام، وعايز تستخدم نفس الدالة على مصفوفة من النصوص، بدل ما تكتب دالة جديدة للنصوص، ممكن تستخدم الجنيريكس وتكتب دالة واحدة تشتغل مع أي نوع بيانات.

في المثال ده، عندي كلاس اسمه **GenericList** بيستخدم الجنيريكس. الكلاس ده بيقدر يخزن عناصر من أي نوع بيانات (زي الأرقام أو النصوص) ويضيفها للمصفوفة ويسترجعها باستخدام نفس الكود.

الجنيريكس بيساعدك تكتب كود أكثر مرونة وكفاءة، وبيققل من تكرار الكود. يعني بتكتب الكود مرة واحدة وتستخدمه مع أنواع بيانات مختلفة.