**In this video we will discuss**  
**1.** The use of Concat operator  
**2.** Difference between Concat and Union operators   
  
   
  
**Concat operator concatenates two sequences into one sequence.**  
  
The following code will **concatenate both the integer sequences** (numbers1 & numbers2) into one integer sequence. Notice that the duplicate elements ARE NOT REMOVED.

int[] numbers1 = { 1, 2, 3 };

int[] numbers2 = { 1, 4, 5 };

var result = numbers1.Concat(numbers2);

foreach (var v in result)

{

    Console.WriteLine(v);

}

**Output :**   
linq concat example   
  
Now let us perform a **union**between the 2 integer sequences (numbers1 & numbers2). Just like concat operator, union operator also combines the 2 integer sequences (numbers1 & numbers2) into one integer sequence, but notice that the **duplicate elements ARE REMOVED.**

int[] numbers1 = { 1, 2, 3 };

int[] numbers2 = { 1, 4, 5 };

var result = numbers1.Union(numbers2);

foreach (var v in result)

{

    Console.WriteLine(v);

}

**Output :**   
linq concat example c#   
  
**What is the difference between Concat and Union operators?**  
Concat operator combines 2 sequences into 1 sequence. Duplicate elements are not removed. It simply returns the items from the first sequence followed by the items from the second sequence.   
  
Union operator also combines 2 sequences into 1 sequence, but will remove the duplicate elements.