**Part 90 - Passing data to the Thread function in a type safe manner**

In this video we will discuss **passing data to the Thread function without loosing the type safety feature of C# programming language**. This is continuation to [Part 89](http://csharp-video-tutorials.blogspot.com/2014/03/part-89-parameterizedthreadstart.html). Please watch [Part 89](http://csharp-video-tutorials.blogspot.com/2014/03/part-89-parameterizedthreadstart.html) before proceeding. We will be working with the same example we worked with in [Part 89](http://csharp-video-tutorials.blogspot.com/2014/03/part-89-parameterizedthreadstart.html).

**To pass data to the Thread function in a type safe manner**, encapsulate the thread function and the data it needs in a helper class and use the ThreadStart delegate to execute the thread function. An example is shown below.

usingSystem**;**

usingSystem.Threading**;**

namespaceThreadingExample

**{**

classProgram

**{**

publicstaticvoidMain**()**

**{**

// Prompt the user for the target number

Console.WriteLine**(**"Please enter the target number"**);**

// Read from the console and store it in target variable

inttarget=Convert.ToInt32**(**Console.ReadLine**());**

// Create an instance of the Number class, passing it

// the target number that was read from the console

Numbernumber=newNumber**(**target**);**

// Specify the Thread function

ThreadT1=newThread**(**newThreadStart**(**number.PrintNumbers**));**

// Alternatively we can just use Thread class constructor as shown below

// Thread T1 = new Thread(number.PrintNumbers);

T1.Start**();**

**}**

**}**

// Number class also contains the data it needs to print the numbers

classNumber

**{**

int\_target**;**

// When an instance is created, the target number needs to be specified

publicNumber**(**inttarget**)**

**{**

// The targer number is then stored in the class private variable \_target

this.\_target=target**;**

**}**

// Function prints the numbers from 1 to the traget number that the user provided

publicvoidPrintNumbers**()**

**{**

for**(**inti=1**;**i<=\_target**;**i++**)**

**{**

Console.WriteLine**(**i**);**

**}**

**}**

**}**

**}**

**Next Video:** Retrieving data from Thread function using callback method