try/finally

This lesson briefs over using try/finally block in C#.

WE'LL COVER THE FOLLOWING

- Try/Finally Example
- Re-throwing Exceptions

Try/Finally Example

The `try/finally` block allows you to do the same as you learnt in the previous lessons, but instead, errors that are thrown are dealt with by the catch (if possible) and then thrown up the call stack.

Therefore, this example will still throw an exception since it doesn't have a `catch` block.

```
using System;

class ExceptionTest
{
  public static void Main(string[] args)
  {
    try
    {
      int[] arr = null; // OK, declares a null reference to an array.
      //int first = arr[0]; // Throws System.NullReferenceException because there is no actua
      Console.WriteLine(arr[0]);
    }
    finally
    {
      Console.WriteLine("In Finally!");
    }
  }
}
```

Re-throwing Exceptions

Re-throwing Exceptions

Sometimes it is better to throw the error up the call stack for two reasons.

- It is not something you would expect to happen.
- You are placing extra information into the exception, to help diagnosis.

Now that you’ve now pretty much learnt about exceptions, let’s have a quiz!