try/catch/finally

This lesson briefs over the usage of try/catch/finally block in C#.

WE'LL COVER THE FOLLOWING

Introduction

Catching the problem is a good idea, but it can sometimes leave your program in an invalid state.

For example, if you open a connection to a database, an error occurs and you throw an exception. Where would you close the connection? In both the try AND catch blocks? Well, problems may occur before the close is carried out.

Therefore, the **finally statement allows you to cater for the "in all cases do this" circumstance.**

Example

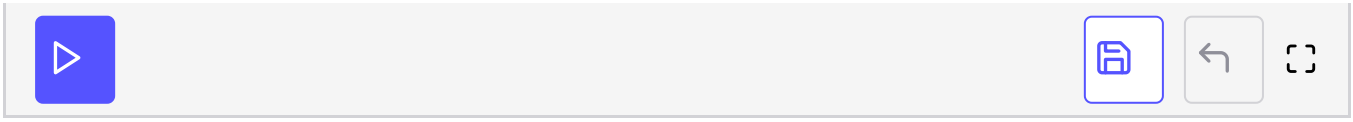
See the example below:

```
using System;
public class excepation
{
        public double num1, num2,result;

        public void add()
        {
                try
                {
                        int num1 = 55;
                        int num2 = 0;
                        result = num1/num2;
                }
                catch(DivideByZeroException  e) //FormatException
                {
```

```
                Console.WriteLine("{0}",e.Message);
                }
                catch(FormatException ex)

                {
                        Console.WriteLine("{0}",ex.Message);
                }
                finally
                {
                        Console.WriteLine("turn over");
                }
        }
        public void display()
        {
                Console.WriteLine("The Result is: {0}",result);
        }
        public static void Main()
        {
                excepation ex = new excepation();
                ex.add();
                ex.display();
        }
}
```



Now let's get into `try/finally` block next!