

Try - catch

1, Try catch is to handle the unexpected issues at runtime - Eg below

- Cannot read or write the file
- unable to connect to database due to network issues
- due to wrong inputs from user

2, To know about the exceptions hover over constructor to get all exceptions

```
StreamReader myStreamReader = new StreamReader(@"sample1.txt");
string line = "";
while (line != null)
{
    line = myStreamReader.ReadLine();
    if (line != null)
        Console.WriteLine(line);
}

myStreamReader.Close();
```

StreamReader.StreamReader(string path) (+ 10 overload(s))
Initializes a new instance of the System.IO.StreamReader class for the specified file name.

Exceptions:

- System.ArgumentException
- System.ArgumentNullException
- System.IO.FileNotFoundException
- System.IO.DirectoryNotFoundException
- System.IO.IOException

3, **Best Practice** : - Catch specific exceptions first and generic to the last

4, only **one catch block** is executed among multiple.

```
static void Main(string[] args)
{
    try
    {
        // currently there is file sample.txt in \bin\debug
        //renamed here to create exception
        StreamReader myStreamReader = new StreamReader(@"sample1.txt");
        string line = "";
        while (line != null)
        {
            line = myStreamReader.ReadLine();
            if (line != null)
                Console.WriteLine(line);
        }
        myStreamReader.Close();
    }
    catch (DirectoryNotFoundException de)
    {
        Console.WriteLine("Couldn't find the file, are u sure directory exists");
    }
    catch (FileNotFoundException fe)
    {
        // here in this case it will come here and go to finally , at a time only one catch is executed
        Console.WriteLine("Couldn't find the file, are u sure you're looking for the correct file ");
    }
    catch (Exception unexpectedException)
    {
        Console.WriteLine("unexpectedException happened");
    }

    finally {
        // perform any cleanup or close the connections
    }

    Console.ReadLine();
}
```