Files and exceptions

Introduction to Computer Programming

Maciej Piernik

Review of Lecture 5

- Methods: procedures and functions
- Returning values
- Passing parameters
  - Reference vs value types
  - Output parameters
  - Varying number of parameters
  - Default values
- Namespaces
- Structures

Outline

- Writing to files
- Reading from files
- Handling errors
- Causing errors **!?**

Writing to files

```
StreamWriter sw = new StreamWriter("asdf.txt");

sw.WriteLine("First line of text");
sw.WriteLine("Second line of text");

sw.Close();
```

```
using (StreamWriter sw = new StreamWriter("asdf.txt"))
{
    sw.WriteLine("First line of text");
    sw.WriteLine("Second line of text");
}
```

Reading from files

```
StreamReader sr = new StreamReader("asdf.txt");
while (!sr.EndOfStream)
{
    Console.WriteLine(sr.ReadLine());
}
sr.Close();
```

```
using (StreamReader sr = new StreamReader("asdf.txt"))
{
    while (!sr.EndOfStream)
    {
        Console.WriteLine(sr.ReadLine());
    }
}
```

Handling errors

```
StreamReader sr = new StreamReader("nonexistent file");

while (!sr.EndOfStream)
{
    Console.WriteLine(sr.ReadLine());
}

sr.Close();
```

Handling errors

```
try
{
    StreamReader sr = new StreamReader("asdf.txt");

    while (!sr.EndOfStream)
    {
        Console.WriteLine(sr.ReadLine());
    }

    sr.Close();
}
catch (FileNotFoundException)
{
    Console.WriteLine("File does not exist!");
}
```

Handling errors

```
try
{
    StreamReader sr = new StreamReader("asdf.txt");

    try
    {
        while (!sr.EndOfStream)
        {
            Console.WriteLine(sr.ReadLine());
        }
    }
    finally
    {
        sr.Close();
    }
}
catch (FileNotFoundException e)
{
    Console.WriteLine(e.Message);
}
```

Handling errors

```
try
{
    int a = 0;
    Console.WriteLine(5 / a);
}
catch (DivideByZeroException)
{
    Console.WriteLine("Division by zero!");
}
```

Handling errors

```
try
{
    int a = 0;
    Console.WriteLine(5 / a);
}
catch (DivideByZeroException)
{
    Console.WriteLine("Division by zero!");
}
catch (Exception e)
{
    Console.WriteLine(e.Message);
}
```

Throwing exceptions

```
try
{
    throw new Exception("An unexpected error has occured!");
}
catch (Exception e)
{
    Console.WriteLine(e.Message);
}
```

Outline

- Writing to files
- Reading from files
- Handling errors
- Causing errors