Processing text, testing, and documentation

Introduction to Computer Programming

Maciej Piernik

Review of Lecture 6

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

Outline

- Processing text
- Unit tests
- Documentation

Processing text

- Concatenation

```
string a = "Culture ";
string b = "Clash";
Console.WriteLine(a + b);
```

- Special characters (np. \t, \n, \\, \”)

```
string text = "Some loooong text we want to divide \ninto two separate
lines and add backslash at the end... \\";
//or
text = @"Some loooong text we want to divide
into two separate lines and add backslash at the end... \";
```

Processing text

- Formatting text

```
string variable = "asdf";
string result = string.Format("Inserting the value of variable here {0}.",
variable);
```

- Formatting numbers

```
double variable = 23.200234;
Console.WriteLine(string.Format("Rounding: {0:0.00}", variable));
Console.WriteLine(string.Format("Rounding and trimming zeros: {0:0.##}",
variable));
```

Processing text

- Text as an array of characters

```
string text = "A year from now you will wish you had started today.";

foreach (char c in text)
{
    Console.WriteLine(c);
}

for (int i = 0; i < text.Length; i++)
{
    Console.WriteLine(text[i]);
}
```

Processing text

- Checking if text is a number

```
string text = "23";
int number = 0;

if (int.TryParse(text, out number))
{
    Console.WriteLine("Yes it is!");
}
```

Processing text

- Useful methods

```
string text = "If you change nothing, nothing will change.";

text.IndexOf("change");          //Index of the first occurrence of a given word
text.LastIndexOf("change");      //Index of the last occurrence of a given word

text.Contains("change");         //Does the text contain...
text.StartsWith("If");           //Does the text begin with...
text.EndsWith(".");              //Does the text end with...


string[] words = text.Split(' ');           //Splits the text whenever a given
                                            //character is found

string fragment = text.Substring(4);        //Returns a substring beginning at...

string replaced = text.Replace("nothing", "something");
//Replaces all occurrences of the first word with the second word
```

Unit tests

- Separate project in solution – Unit Test Project

```
[TestClass]
public class TestOfSomeClass
{
    [TestMethod]
    public void TestOfSomeMethod()
    {
        //Initialization
        //Execution
        //Verification
    }
}
```

Unit tests

- Verification using assertions

```
[TestMethod]
public void Booking_ValidBooking()
{
    int numberOfRoomsBeforeBookiing = 4;
    int numberOfRoomsBeingBooked = 2;
    int numberOfRoomsAfterBooking = numberOfRoomsBeforeBooking -
numberOfRoomsBeingBooked;

    Hotel.AvailableRooms = numberOfRoomsBeforeBooking;

    Hotel.Book(numberOfRoomsBeingBooked);

    Assert.AreEqual(numberOfRoomsAfterBooking, Hotel.AvailableRooms, "The
number of available rooms is incorrect!");
}
```

Unit tests

- Example assertions

```
Assert.AreEqual(expectedAvailability, Hotel.AvailableRooms);

Assert.Fail("We shouldn’t be here!");

Assert.IsTrue(Hotel.NoAvailableRooms);
Assert.IsFalse(Hotel.NoAvailableRooms);
```

Documentation

```
/// <summary>
/// Class for managing hotel room reservations
/// </summary>
public class Hotel
{
    /// <summary>
    /// A function responsible for making a room reservation
    /// </summary>
    /// <param name="numberOfRooms">Number of rooms being booked</param>
    public static void Book(int numberOfRooms)
    {
        ...
    }
}
```

Summary

- Processing text
- Unit tests
- Documentation