COS20007
Object Oriented Programming

Thomas Horsley

Contents

List of Figures

1 UML Class Diagrams

Class diagrams are designed to clearly depict what an object knows (the objects variables, constants and custom data types) and what it can do (the methods within the object and relationships between the objects data). A class diagram consists of the class title, the attributes held within the class and the methods the class can call.

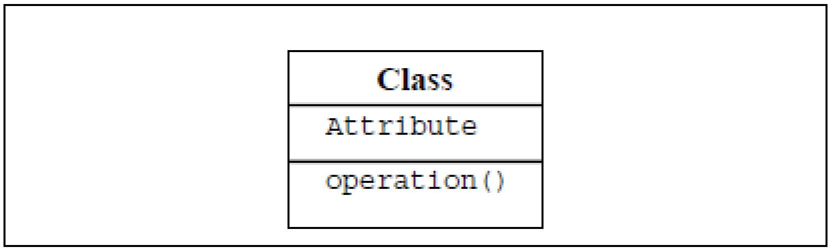


Figure 1: General class diagram structure

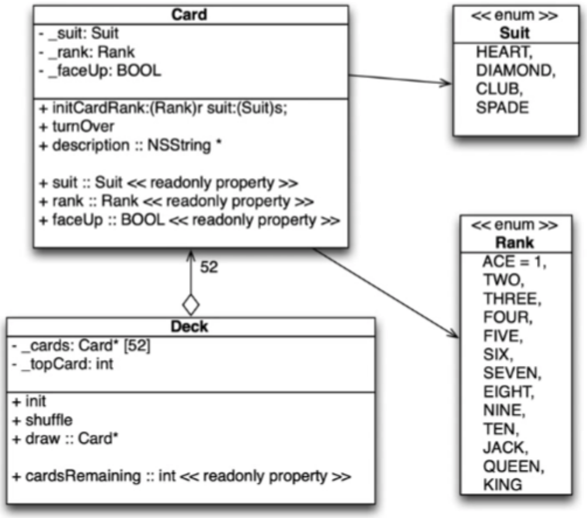
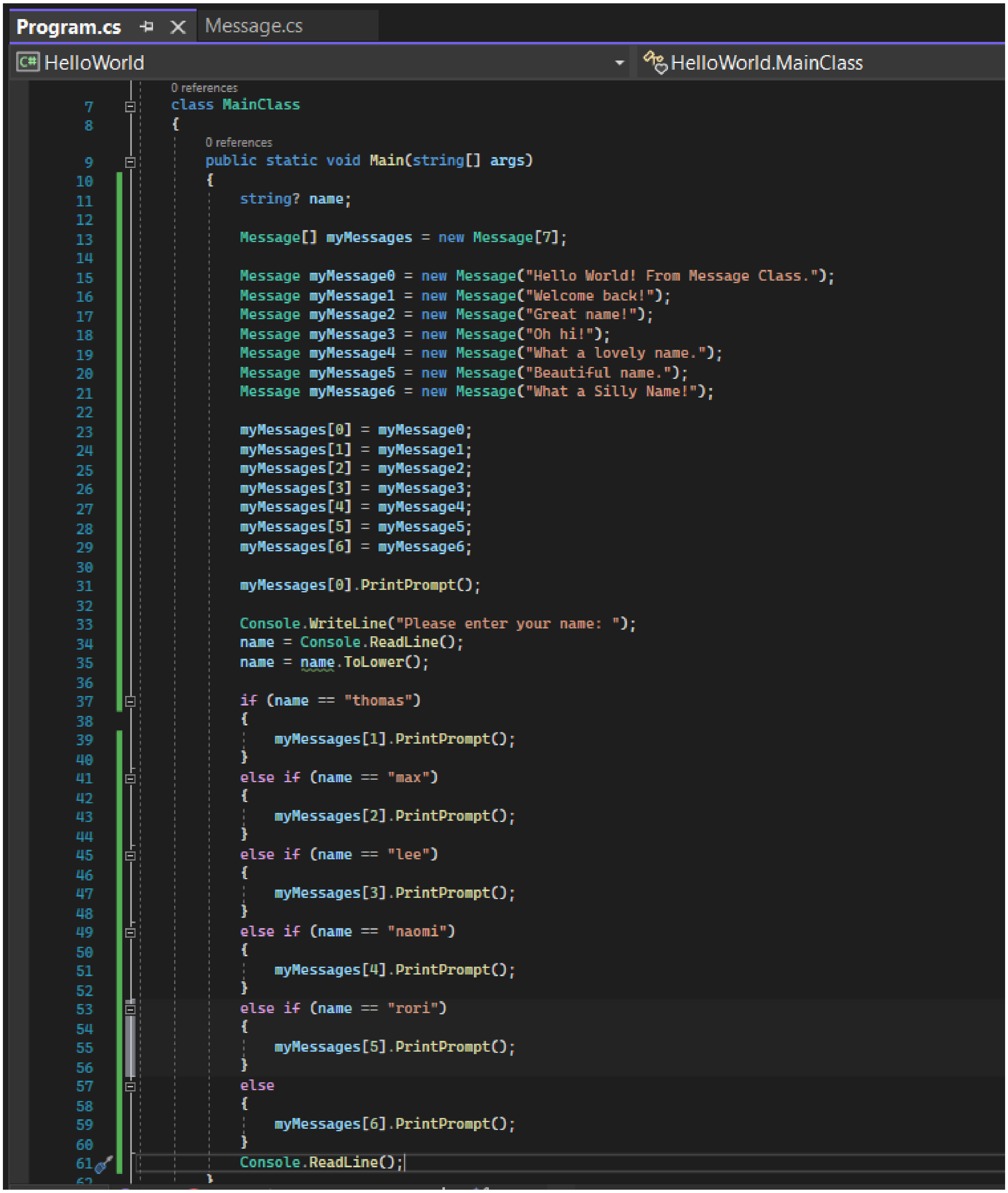


Figure 2: Example of a set of UML Class Diagrams for a deck of playing cards

The access modifier (the '+' and '−' found before each field) indicates whether the field is private (denoted '−') or public (denoted '+').

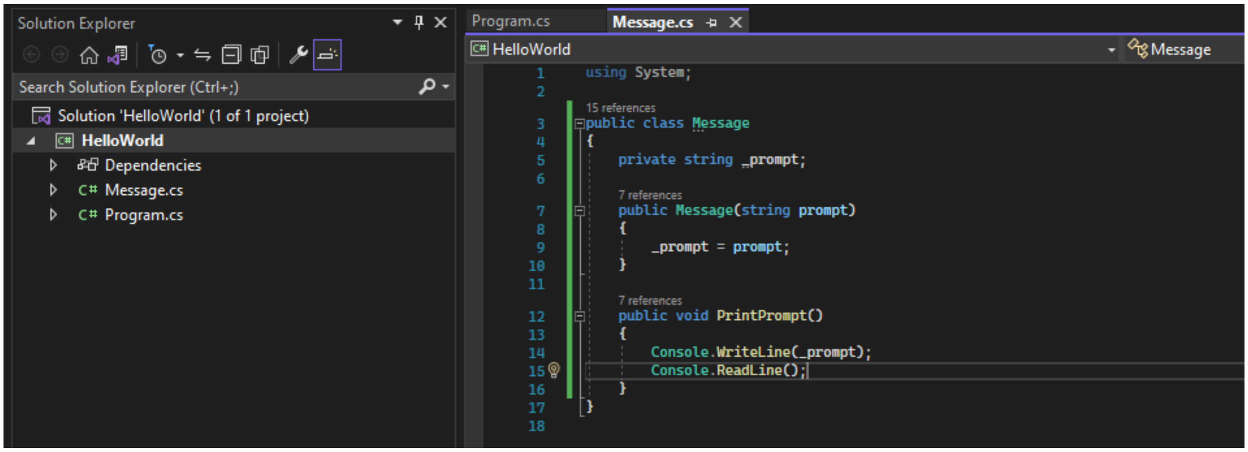
Operations are written first by explicitly denoting their access modifier and name, followed by the parameters passed into the operation, followed by the return value.

2 Portfolio Task 1.2P



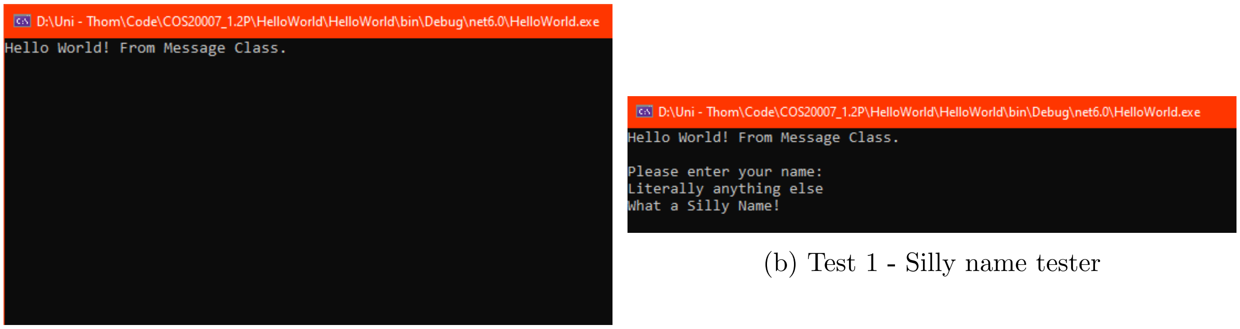
```
0 references
class MainClass
{
    0 references
    public static void Main(string[] args)
    {
        string? name;

        Message[] myMessages = new Message[7];

        Message myMessage0 = new Message("Hello World! From Message Class.");
        Message myMessage1 = new Message("Welcome back!");
        Message myMessage2 = new Message("Great name!");
        Message myMessage3 = new Message("Oh hi!");
        Message myMessage4 = new Message("What a lovely name.");
        Message myMessage5 = new Message("Beautiful name.");
        Message myMessage6 = new Message("What a Silly Name!");

        myMessages[0] = myMessage0;
        myMessages[1] = myMessage1;
        myMessages[2] = myMessage2;
        myMessages[3] = myMessage3;
        myMessages[4] = myMessage4;
        myMessages[5] = myMessage5;
        myMessages[6] = myMessage6;

        myMessages[0].PrintPrompt();

        Console.WriteLine("Please enter your name: ");
        name = Console.ReadLine();
        name = name.ToLower();

        if (name == "thomas")
        {
            myMessages[1].PrintPrompt();
        }
        else if (name == "max")
        {
            myMessages[2].PrintPrompt();
        }
        else if (name == "lee")
        {
            myMessages[3].PrintPrompt();
        }
        else if (name == "naomi")
        {
            myMessages[4].PrintPrompt();
        }
        else if (name == "rori")
        {
            myMessages[5].PrintPrompt();
        }
        else
        {
            myMessages[6].PrintPrompt();
        }
        Console.ReadLine();
```

Figure 3: *Program.cs* is the entry point for the Hello World programand houses most of the functionality



```
using System;

public class Message
{
    private string _prompt;

    public Message(string prompt)
    {
        _prompt = prompt;
    }

    public void PrintPrompt()
    {
        Console.WriteLine(_prompt);
        Console.ReadLine();
    }
}
```

Figure 4: Code encapsulating the Message class



D:\Uni - Thom\Code\COS20007_1.2P\HelloWorld\HelloWorld\bin\Debug\net6.0\HelloWorld.exe
D:\Uni - Thom\Code\COS20007_1.2P\HelloWorld\HelloWorld\bin\Debug\net6.0\HelloWorld.exe

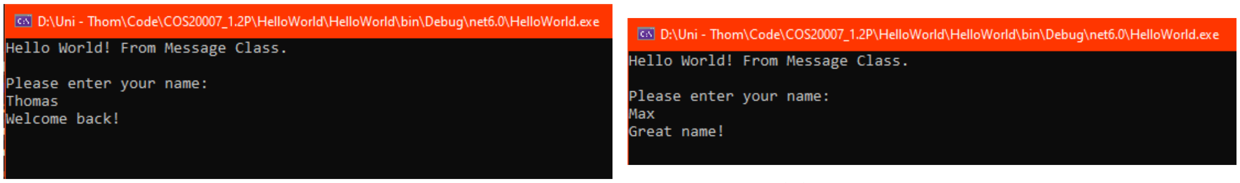
```
Hello World! From Message Class.
```

```
Hello World! From Message Class.

Please enter your name:
Literally anything else
What a Silly Name!
```

(b) Test 1 - Silly name tester

(a) Hello World!

Figure 5: **1.2P** Hello World Solution and Silly Name Tester



D:\Uni - Thom\Code\COS20007_1.2P\HelloWorld\HelloWorld\bin\Debug\net6.0\HelloWorld.exe
D:\Uni - Thom\Code\COS20007_1.2P\HelloWorld\HelloWorld\bin\Debug\net6.0\HelloWorld.exe

```
Hello World! From Message Class.

Please enter your name:
Thomas
Welcome back!
```

```
Hello World! From Message Class.

Please enter your name:
Max
Great name!
```

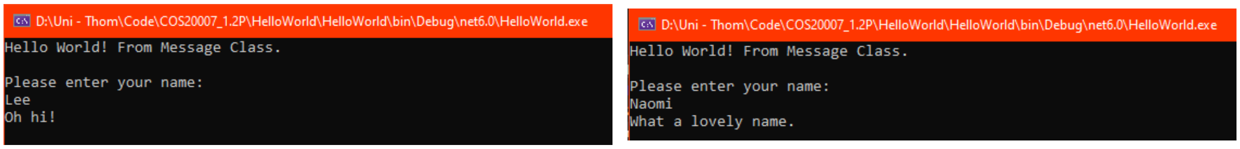
(a) Test 2 - Thomas's Prompt Test

(b) Test 3 - Max's Prompt

Figure 6: Thomas and Max prompt tests

NOTE:

My solution and images can also be found at https://github.com/KingSchlock/COS20007


D:\Uni - Thom\Code\COS20007_1.2P\HelloWorld\HelloWorld\bin\Debug\net6.0\HelloWorld.exe
D:\Uni - Thom\Code\COS20007_1.2P\HelloWorld\HelloWorld\bin\Debug\net6.0\HelloWorld.exe

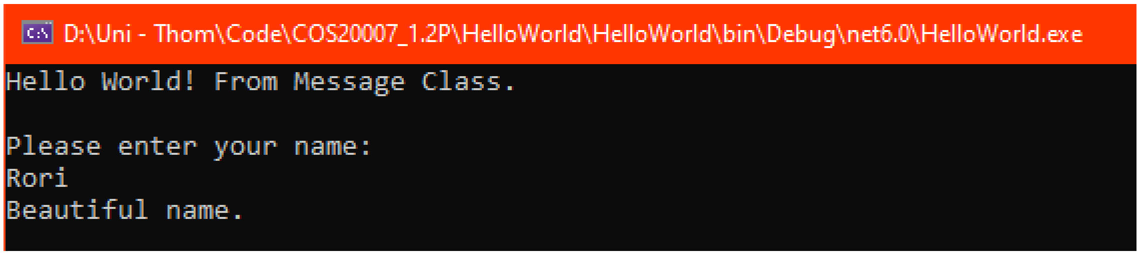
```
Hello World! From Message Class.

Please enter your name:
Lee
Oh hi!
```

(a) Test 4 - Lee’s Prompt Test

```
Hello World! From Message Class.

Please enter your name:
Naomi
What a lovely name.
```

(b) Test 5 - Naomi’s Prompt

Figure 7: Lee and Naomi prompt tests


D:\Uni - Thom\Code\COS20007_1.2P\HelloWorld\HelloWorld\bin\Debug\net6.0\HelloWorld.exe

```
Hello World! From Message Class.

Please enter your name:
Rori
Beautiful name.
```

Figure 8: Test 6 - Rori’s Prompt