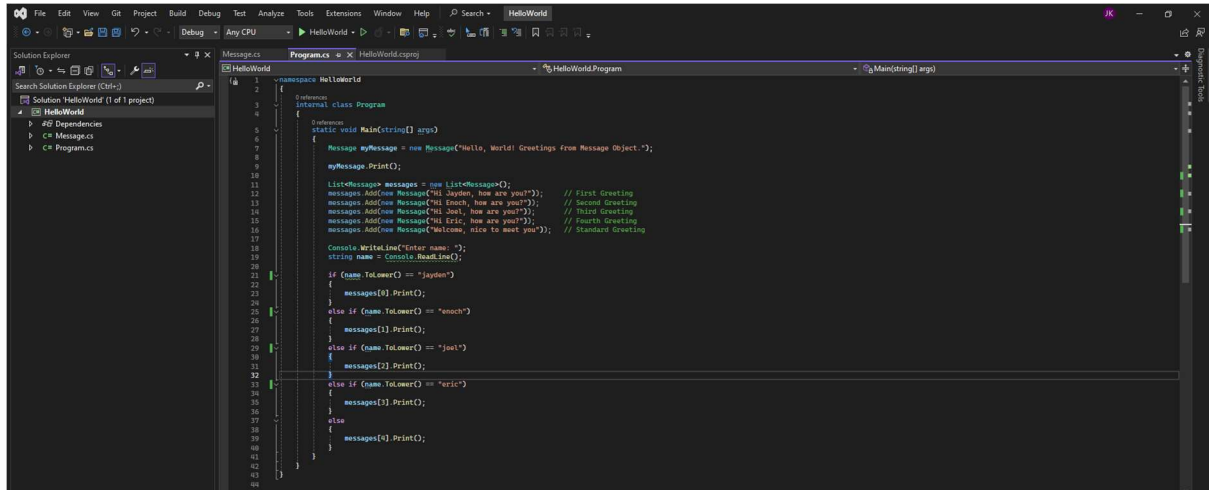


1.2P: Object-Oriented “Hello, World!”

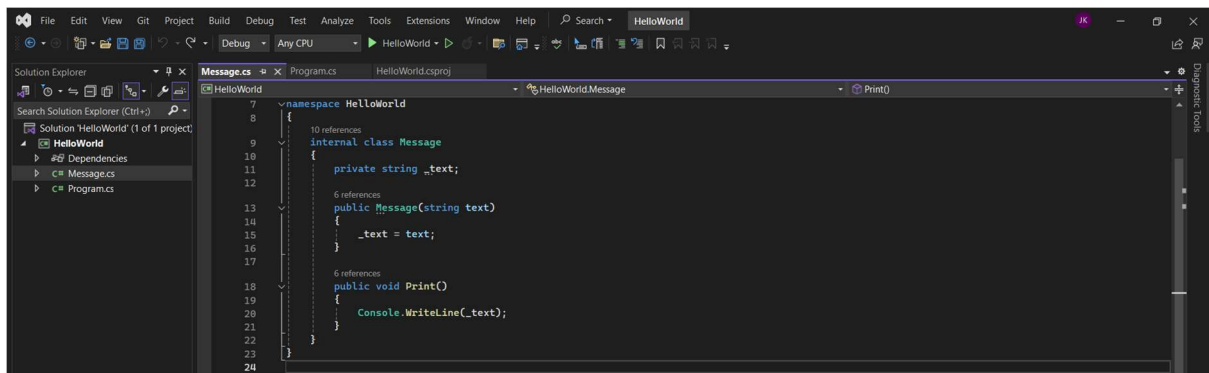
Jayden Kong, 104547242

Program.cs code (text version on page 3):

A screenshot of the Visual Studio IDE showing the Program.cs file in a project named HelloWorld. The code defines a namespace HelloWorld with an internal class Program. The Main method takes an array of strings and creates a Message object with the text "Hello, World! Greetings from Message Object.". It then prints this message. Next, it creates a list of Message objects with various greetings. A console.WriteLine prompts the user to enter a name. The code then uses a switch statement to print the corresponding greeting from the list based on the input name: "Jayden" prints the first message, "Enoch" prints the second, "Joel" prints the third, and "eric" prints the fourth. If the name is not in the list, it prints the standard greeting.

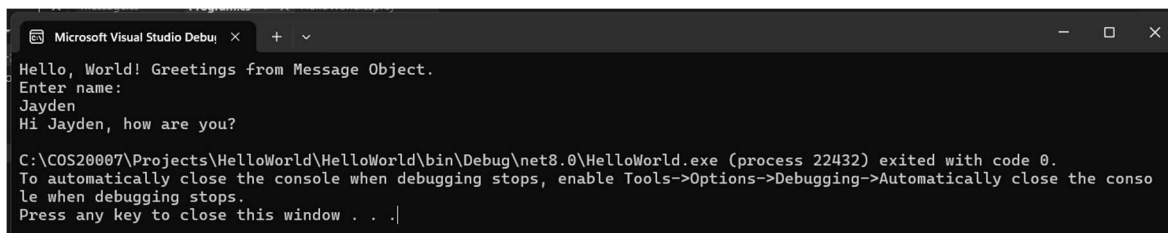
```
1 namespace HelloWorld
2 {
3     internal class Program
4     {
5         static void Main(string[] args)
6         {
7             Message myMessage = new Message("Hello, World! Greetings from Message Object.");
8             myMessage.Print();
9
10            List<Message> messages = new List<Message>();
11            messages.Add(new Message("Hi Jayden, how are you?")); // First Greeting
12            messages.Add(new Message("Hi Enoch, how are you?")); // Second Greeting
13            messages.Add(new Message("Hi Joel, how are you?")); // Third Greeting
14            messages.Add(new Message("Hi Eric, how are you?")); // Fourth Greeting
15            messages.Add(new Message("Welcome, nice to meet you")); // Standard Greeting
16
17            Console.WriteLine("Enter name: ");
18            string name = Console.ReadLine();
19
20            if (name.ToLower() == "jayden")
21            {
22                messages[0].Print();
23            }
24            else if (name.ToLower() == "enoch")
25            {
26                messages[1].Print();
27            }
28            else if (name.ToLower() == "joel")
29            {
30                messages[2].Print();
31            }
32            else if (name.ToLower() == "eric")
33            {
34                messages[3].Print();
35            }
36            else
37            {
38                messages[4].Print();
39            }
40        }
41    }
42 }
43
```

Message.cs code (text version on page 4):

A screenshot of the Visual Studio IDE showing the Message.cs file. It defines a namespace HelloWorld with an internal class Message. The class has a private string field _text. It has a public constructor Message(string text) that initializes _text. It also has a public void Print() method that calls Console.WriteLine(_text).

```
1 namespace HelloWorld
2 {
3     internal class Message
4     {
5         private string _text;
6
7         public Message(string text)
8         {
9             _text = text;
10        }
11
12        public void Print()
13        {
14            Console.WriteLine(_text);
15        }
16    }
17 }
18
```

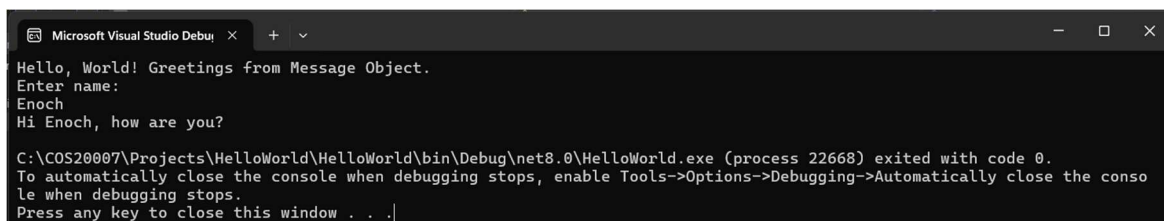
Screenshot of output with input “Jayden”:

A screenshot of the Visual Studio Debug Console window. It shows the output of the program when the input name is "Jayden". The output is: "Hello, World! Greetings from Message Object.", "Enter name:", "Jayden", and "Hi Jayden, how are you?". Below this, it shows the process exiting with code 0 and a message to automatically close the console when debugging stops.

```
Microsoft Visual Studio Debug Console
Hello, World! Greetings from Message Object.
Enter name:
Jayden
Hi Jayden, how are you?

C:\COS20007\Projects\HelloWorld\HelloWorld\bin\Debug\net8.0\HelloWorld.exe (process 22432) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

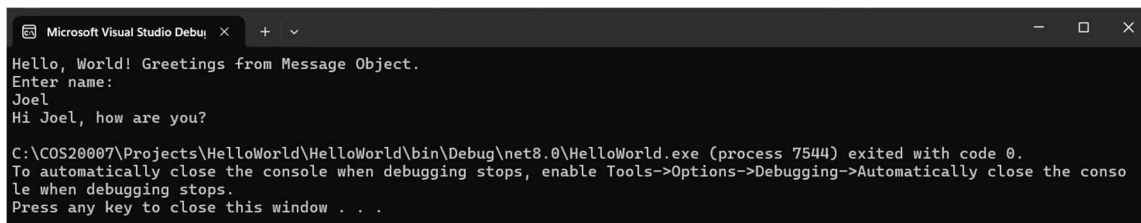
Screenshot of output with input “Enoch”:

A screenshot of the Visual Studio Debug Console window. It shows the output of the program when the input name is "Enoch". The output is: "Hello, World! Greetings from Message Object.", "Enter name:", "Enoch", and "Hi Enoch, how are you?". Below this, it shows the process exiting with code 0 and a message to automatically close the console when debugging stops.

```
Microsoft Visual Studio Debug Console
Hello, World! Greetings from Message Object.
Enter name:
Enoch
Hi Enoch, how are you?

C:\COS20007\Projects\HelloWorld\HelloWorld\bin\Debug\net8.0\HelloWorld.exe (process 22668) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

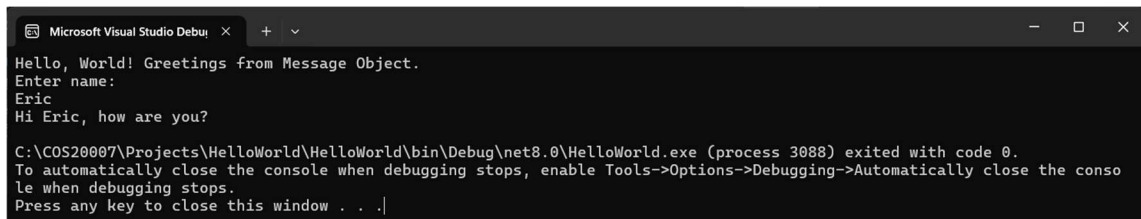
Screenshot of output with input “Joel”:



```
Microsoft Visual Studio Debug Console
Hello, World! Greetings from Message Object.
Enter name:
Joel
Hi Joel, how are you?

C:\COS20007\Projects\HelloWorld\HelloWorld\bin\Debug\net8.0\HelloWorld.exe (process 7544) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

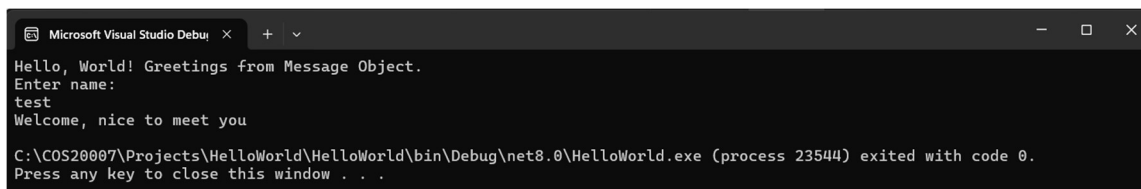
Screenshot of output with input “Eric”:



```
Microsoft Visual Studio Debug Console
Hello, World! Greetings from Message Object.
Enter name:
Eric
Hi Eric, how are you?

C:\COS20007\Projects\HelloWorld\HelloWorld\bin\Debug\net8.0\HelloWorld.exe (process 3088) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

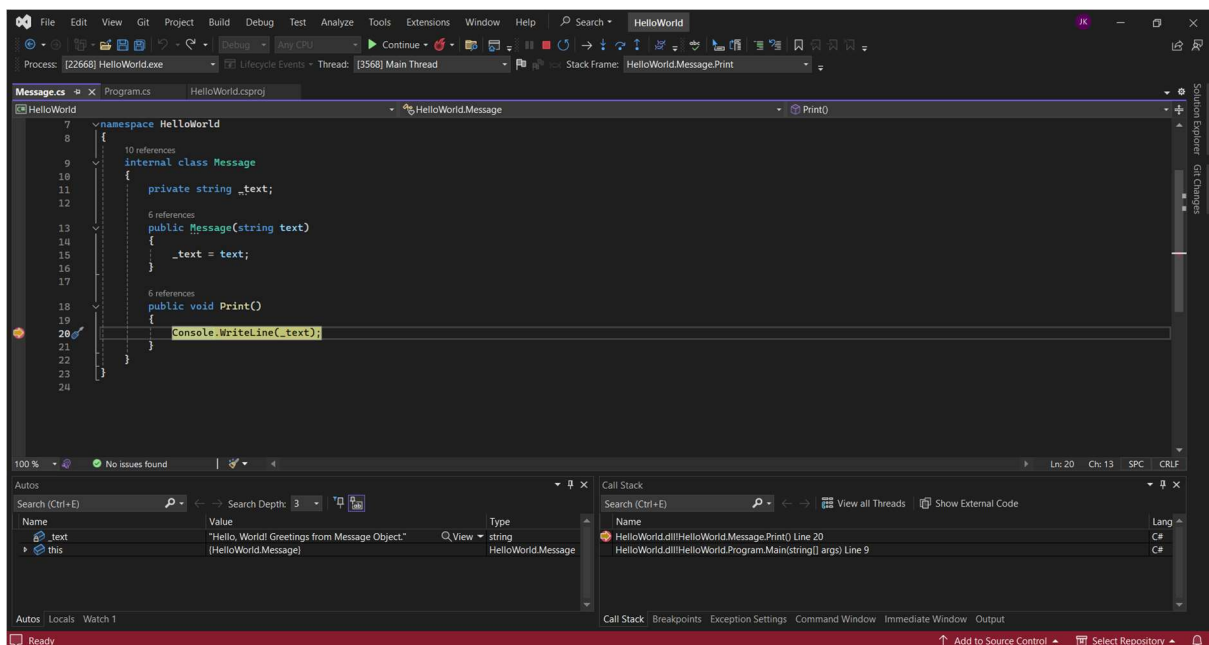
Screenshot of output with input “test”:



```
Microsoft Visual Studio Debug Console
Hello, World! Greetings from Message Object.
Enter name:
test
Welcome, nice to meet you

C:\COS20007\Projects\HelloWorld\HelloWorld\bin\Debug\net8.0\HelloWorld.exe (process 23544) exited with code 0.
Press any key to close this window . . .
```

Screenshot showing call stack and code paused within the Print method of the Message class:



```
1 namespace HelloWorld
2 {
3     internal class Program
4     {
5         static void Main(string[] args)
6         {
7             Message myMessage = new Message("Hello, World! Greetings from
8                 Message Object.");
9
10            myMessage.Print();
11
12            List<Message> messages = new List<Message>();
13            messages.Add(new Message("Hi Jayden, how are you?")); // First Greeting
14            messages.Add(new Message("Hi Enoch, how are you?")); // Second Greeting
15            messages.Add(new Message("Hi Joel, how are you?")); // Third Greeting
16            messages.Add(new Message("Hi Eric, how are you?")); // Fourth Greeting
17            messages.Add(new Message("Welcome, nice to meet you")); // Standard Greeting
18
19            Console.WriteLine("Enter name: ");
20            string name = Console.ReadLine();
21
22            if (name.ToLower() == "jayden")
23            {
24                messages[0].Print();
25            }
26            else if (name.ToLower() == "enoch")
27            {
28                messages[1].Print();
29            }
30            else if (name.ToLower() == "joel")
31            {
32                messages[2].Print();
33            }
34            else if (name.ToLower() == "eric")
35            {
36                messages[3].Print();
37            }
38            else
39            {
40                messages[4].Print();
41            }
42        }
43 }
```

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace HelloWorld
8 {
9     internal class Message
10    {
11        private string _text;
12
13        public Message(string text)
14        {
15            _text = text;
16        }
17
18        public void Print()
19        {
20            Console.WriteLine(_text);
21        }
22    }
23 }
24
```