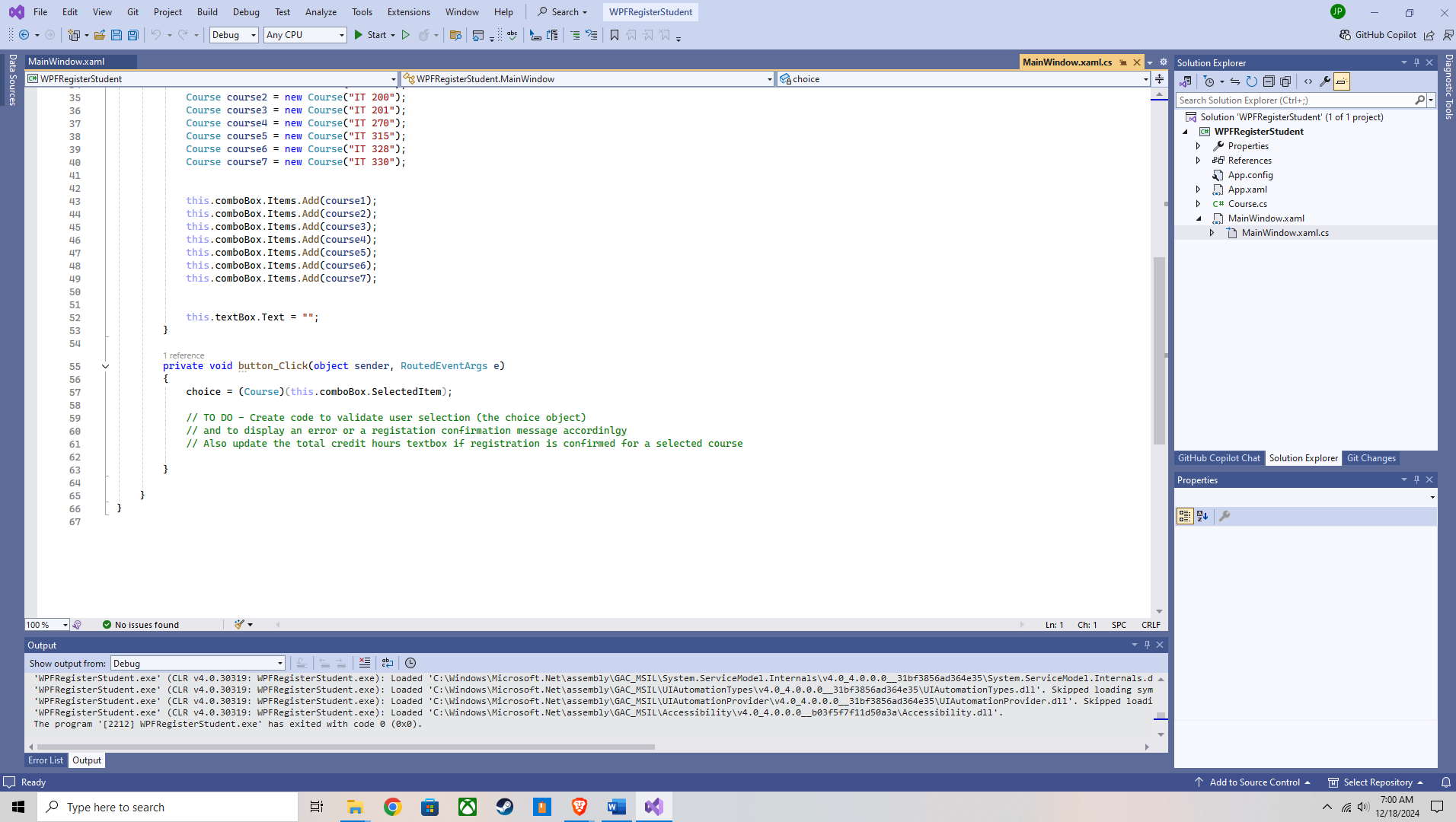
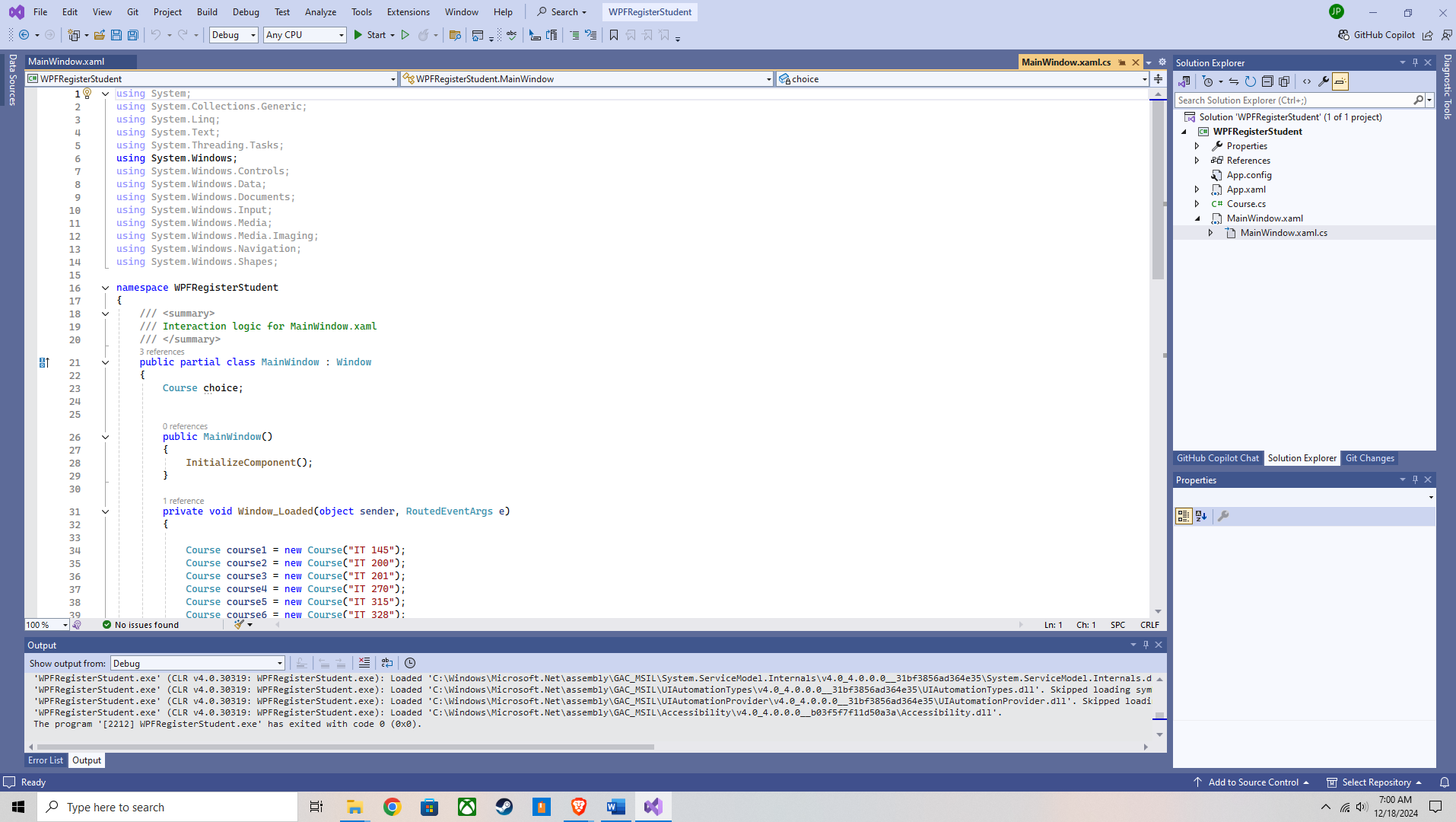
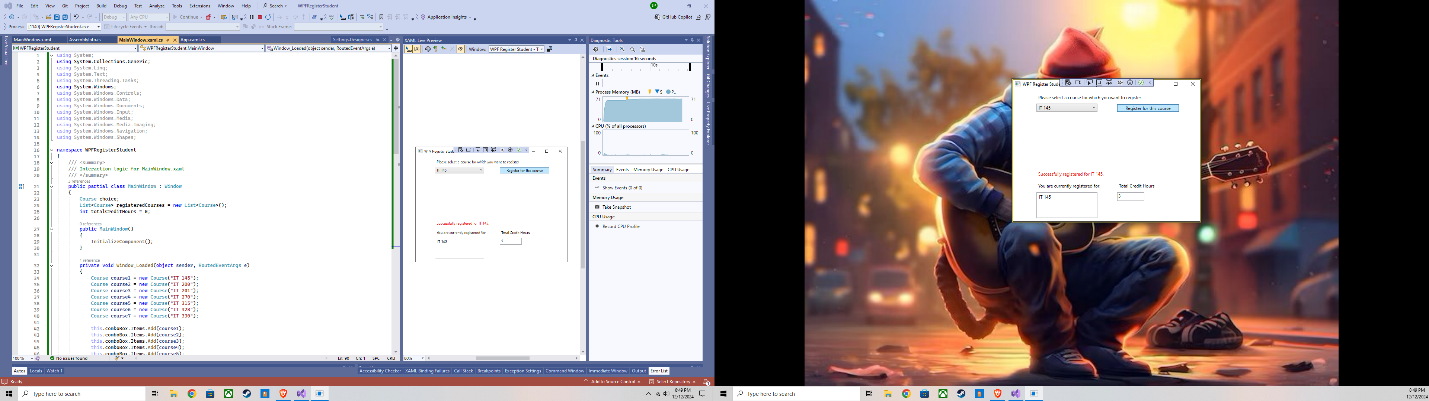
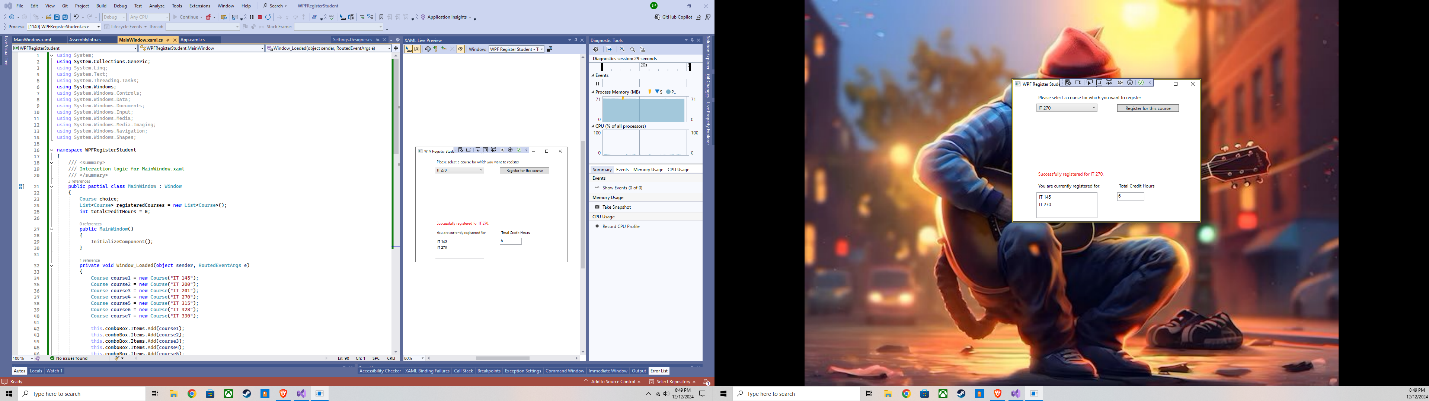
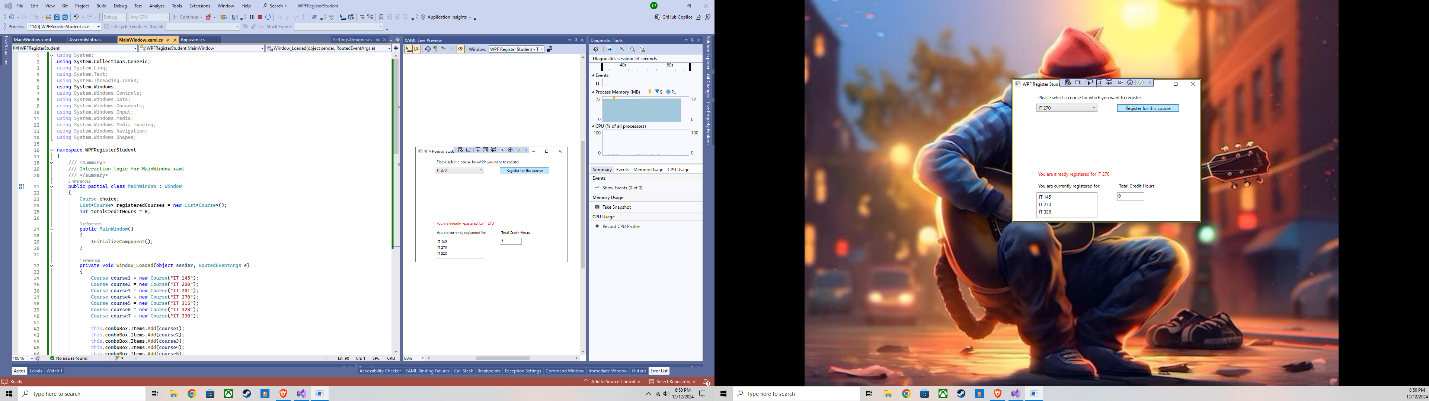
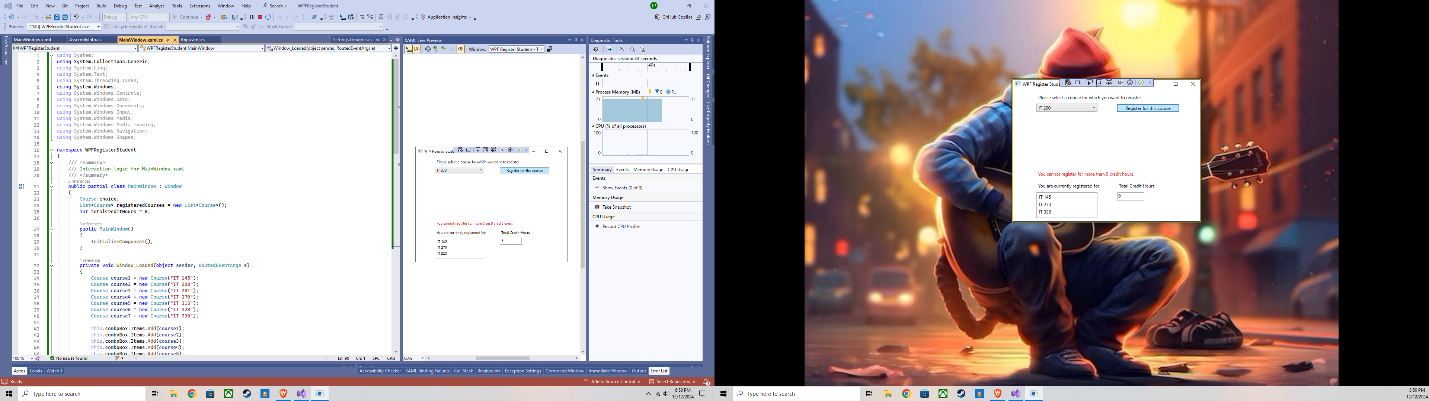
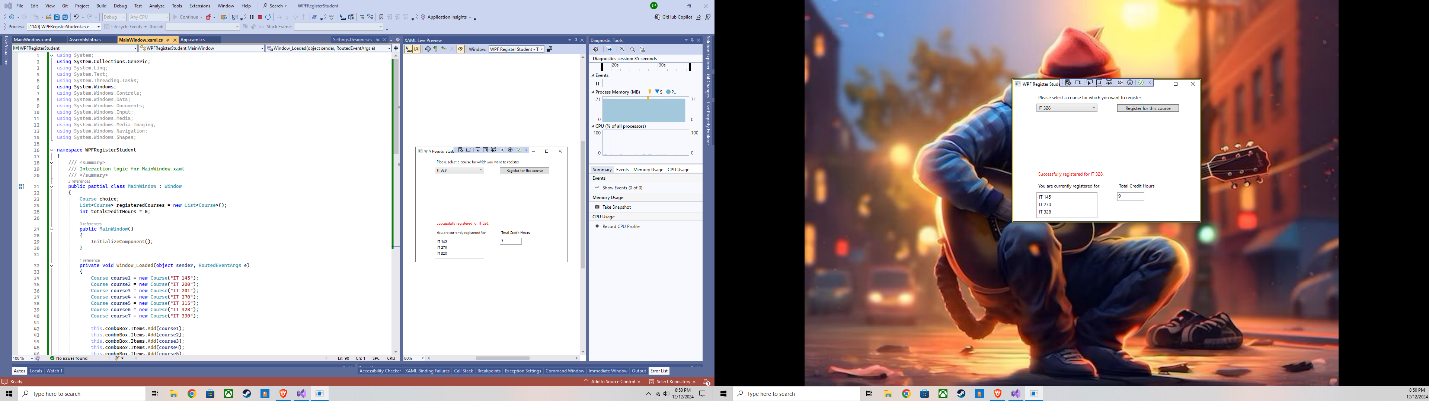
Jarrett Phillips  
  
Original code that needed a fix:



Screen shots provided below are the source code edited, fixed, with the output:  
  


Source code used:  
  
using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace WPFRegisterStudent

{

/// <summary>

/// Interaction logic for MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

Course choice;

List<Course> registeredCourses = new List<Course>();

int totalCreditHours = 0;

public MainWindow()

{

InitializeComponent();

}

private void Window\_Loaded(object sender, RoutedEventArgs e)

{

Course course1 = new Course("IT 145");

Course course2 = new Course("IT 200");

Course course3 = new Course("IT 201");

Course course4 = new Course("IT 270");

Course course5 = new Course("IT 315");

Course course6 = new Course("IT 328");

Course course7 = new Course("IT 330");

this.comboBox.Items.Add(course1);

this.comboBox.Items.Add(course2);

this.comboBox.Items.Add(course3);

this.comboBox.Items.Add(course4);

this.comboBox.Items.Add(course5);

this.comboBox.Items.Add(course6);

this.comboBox.Items.Add(course7);

}

private void button\_Click(object sender, RoutedEventArgs e)

{

choice = (Course)(this.comboBox.SelectedItem);

if (choice == null)

{

label3.Content = "Please select a course.";

return;

}

if (choice.IsRegisteredAlready())

{

label3.Content = $"You are already registered for {choice.getName()}.";

return;

}

if (totalCreditHours + 3 > 9)

{

label3.Content = "You cannot register for more than 9 credit hours.";

return;

}

// Mark the course as registered

choice.SetToRegistered();

registeredCourses.Add(choice);

// Update total credit hours

totalCreditHours += 3;

// Update the registered courses list in the ListBox

listBox.Items.Add(choice.getName());

// Update the total credit hours in the TextBox

textBox.Text = totalCreditHours.ToString();

label3.Content = $"Successfully registered for {choice.getName()}.";

}

}

}  
  
Explanation:

To solve this issue, I first reviewed the MainWindow.xaml file that was provided, I made sure that the elements like comboBox, listBox, and textBox were correctly set up. These elements were crucial for enabling course selection, displaying registered courses, and showing total credit hours. After ensuring that the interface matched the functionality needed, I updated the MainWindow.xaml.cs file, where the program's logic resides, to align with these elements. Specifically, I linked the listBox to display registered courses and made sure the textBox accurately displayed the total credit hours. Additionally, I used label3 to provide feedback messages to users, such as errors or success confirmations, which improves usability.

One of the challenges was figuring out how to properly reflect user interactions in the interface. For instance, while it seemed straightforward to update the total credit hours, ensuring that each course was tracked and visually represented in the list required a closer look at how data structures (like List<Course>) interact with the interface. Another hurdle was understanding how to provide user feedback without overwhelming or confusing the user, which required careful use of the label3 control. By breaking down the problem into steps, validating input, updating data, and refreshing the interface, I was able to finally get the program to not only work but also behave intuitively for the user.  
  
P.S. I also hope that the source code shares with you correctly, because having to redo this entire project again like I have the last two due to the file becoming corrupted is going to become quite frustrating.