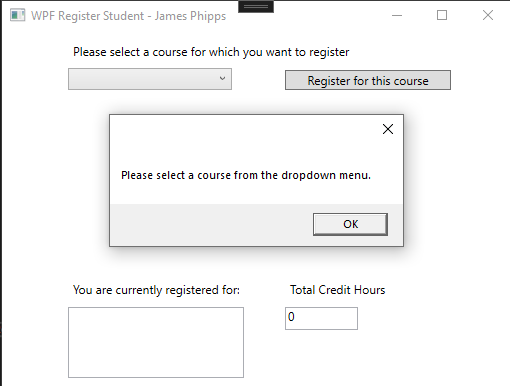
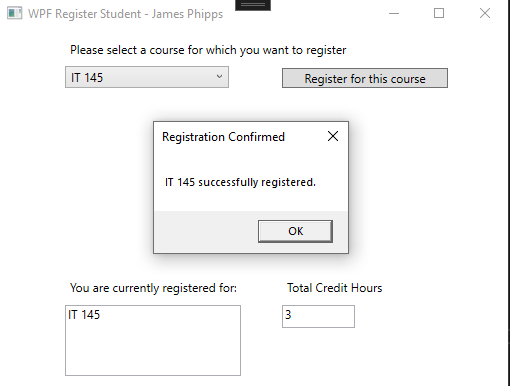
**James Phipps**

**IT-230**

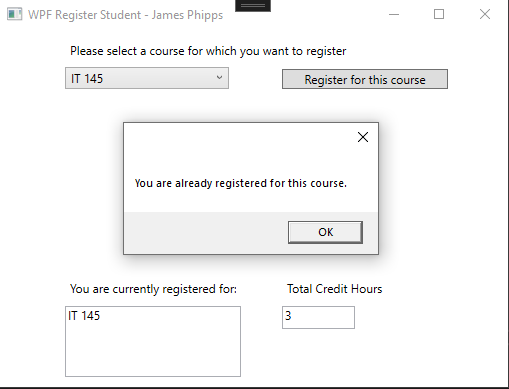
**Final Project Part II – WPFRegisterStudent**



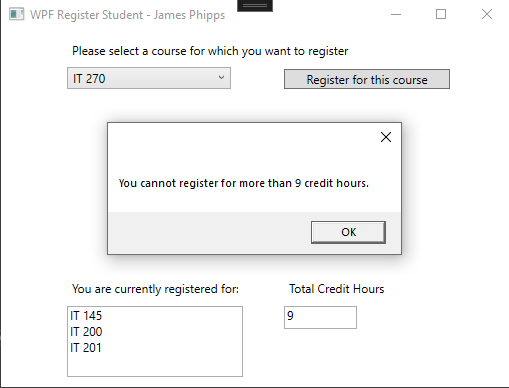
This screenshot shows the behavior when a user clicks the “Register for this course” button without selecting a course. A message box appears prompting the user to make a valid selection from the dropdown. This ensures the program only proceeds with registration when a valid input is provided.



This screenshot confirms the successful registration of “IT 145.” After selecting a course and clicking the register button, a confirmation message box appears and the course name is added to the “currently registered for” display. The total credit hours box also updates from 0 to 3.



This screenshot shows the program correctly preventing the user from registering for a course they have already registered for. A message box alerts the user that they are already enrolled in “IT 145,” and the program does not duplicate the course or update the credit hours.



This screenshot demonstrates the program blocking registration once the student has reached the 9-credit-hour maximum. Attempting to register for a fourth course displays a message box stating the limit has been reached. The registration is not confirmed, and the credit hour total remains unchanged.

**Source Code**

MainWindow.xaml.cs

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;

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);

this.textBox.Text = "";

}

private int totalCredits = 0;

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

{

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

if (selectedCourse == null)

{

MessageBox.Show("Please select a course from the dropdown menu.");

return;

}

if (selectedCourse.IsRegisteredAlready())

{

MessageBox.Show("You are already registered for this course.");

return;

}

if (totalCredits >= 9)

{

MessageBox.Show("You cannot register for more than 9 credit hours.");

return;

}

// Register the course

selectedCourse.SetToRegistered();

totalCredits += 3;

// Update registered course display

this.textBox.AppendText(selectedCourse.getName() + "\n");

// Update total credit hours

this.creditHourBox.Text = totalCredits.ToString();

MessageBox.Show($"{selectedCourse.getName()} successfully registered.", "Registration Confirmed");

}

private void textBox\_TextChanged(object sender, TextChangedEventArgs e)

{

}

}

}

Course.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace WPFRegisterStudent

{

class Course

{

private string name = "";

private bool isRegisteredAlready = false;

public Course(string name)

{

this.name = name;

}

public void setName(string name)

{

this.name = name;

}

public string getName()

{

return name;

}

public bool IsRegisteredAlready()

{

return isRegisteredAlready;

}

public void SetToRegistered()

{

isRegisteredAlready = true;

}

public override string ToString()

{

return getName();

}

}

}