



  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  




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;

// Declare List to hold courses chosen

// Using a list makes it easier to check for duplicates

// As well as count the number of courses registered

//

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

// This integer variable sets the maximum number of courses a student may take

//

// The default is 3, but it's possible some students may be able to take more

// But you would need a method to set that.

//

int maxNumberOfCourses = 3;

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 = "";

}

//

// This Method Simply returns the number of courses \* 3 as a string

//

//

private string courseHours()

{

string courseToHours = "";

courseToHours = Convert.ToString(chosenCourseList.Count \* 3);

return courseToHours; }

//

// This method comes up with a string that is sent to the output

// label that tells the user they are registered for the specified class

//

private string registerForCourseMessage(Course course)

{

string courseName = course.getName();

string registerMessage = "You have been registered for this course " + courseName;

return registerMessage;

}

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

{

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

bool alreadyChosenCourse = false;

// If there are no courses chosen, automatically add it to the list

if (chosenCourseList.Count == 0)

{

chosenCourseList.Add(choice);

this.listBox.Items.Add(choice);

this.textBox.Text = courseHours();

this.label3.Content = registerForCourseMessage(choice);

}

//

// Check to see if there is more than one class but less than the maximum allowed

//

else if (chosenCourseList.Count >= 1 && chosenCourseList.Count < maxNumberOfCourses)

{

//

// Loop through list containing chosen courses to see if the currently chosen course

// Has already been picked

//

for (int i = 0; i < chosenCourseList.Count; i++)

{

//

// If it is, change flag and print error message

//

if (chosenCourseList[i] == choice)

{

//MessageBox.Show("You are already registered for this class!","Error",MessageBoxButton.OK,MessageBoxImage.Error);

alreadyChosenCourse = true;

this.label3.Content = "You are already registered for this class!";

}

}

//

// If not, register the class

//

if (alreadyChosenCourse == false)

{

chosenCourseList.Add(choice);

this.listBox.Items.Add(choice);

this.textBox.Text = courseHours();

this.label3.Content = registerForCourseMessage(choice);

}

}

//

// Check if student is at the maximum number of classes allowed

// And if so, give an error message.

//

else if (chosenCourseList.Count >= maxNumberOfCourses)

{

//MessageBox.Show("You already are registered for the maximum number of classes!", "Error", MessageBoxButton.OK, MessageBoxImage.Error);

this.label3.Content = "You are already registered for the maximum number of classes!";

}

}

}

}