

CST8253 Web Programming II

Lab 7

Objective

1. Use ASP.NET Session object
2. Use inheritance in object-oriented programming

Due Date

See Brightspace posting for the due date of this lab. To earn 5 points, you are required:

1. Complete the lab as required.
2. Zip the application's project folder and submit to the Brightspace before due date.
3. Demo your lab work during the lab session in the week after the due date.

Preparations

1. In Visual Studio create a new Web Form ASP.NET Web Application (.NET Framework) project **Lab7**
2. Add the following class definitions to the project's Models folder

- **Student** class

This is the base class for all types of students. This class should at least have the following properties:

Id – an integer, readonly

Name – a string, readonly

This class should have a constructor taking one parameter of string type for initialize the **Name** property. The constructor will initialize Id property with a randomly generated 6 digits number.

- **FulltimeStudent** class

This class should inherit from Student class. This class should have the following property:

MaxWeeklyHours – a static integer, read and write.

This class should have a constructor taking one parameter of string type to pass to the base class's constructor to initialize the base classes **Name** property.

- **ParttimeStudent** class

This class inherits from Student class. This class should have the following property:

MaxNumOfCourses – a static integer, read and write

This class should have a constructor taking one parameter of string type to pass to the base class' constructor to initialize the base classes **Name** property.

- **CoopStudent** class

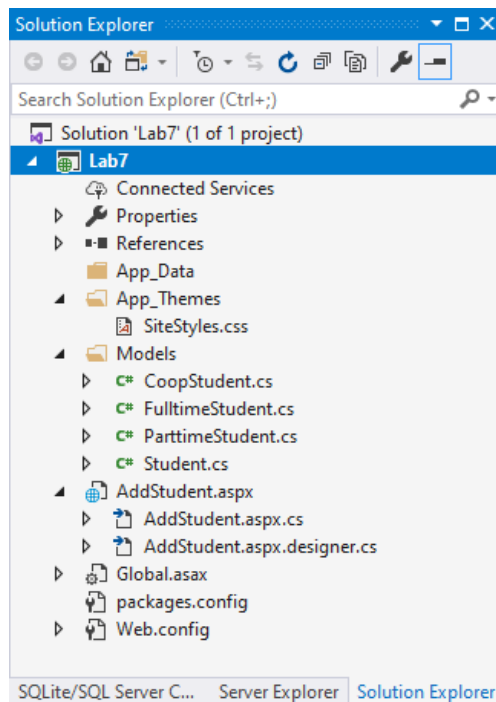
This class inherits from Student class. This class should have the following two properties:

MaxWeeklyHours – a static integer, read and write

MaxNumOfCourses – a static integer, read and write

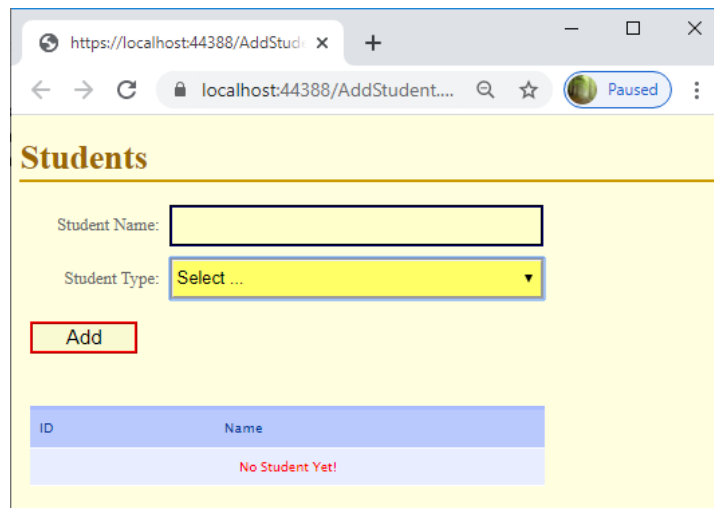
This class should have a constructor taking one parameter of string type to pass to the base class' constructor to initialize the base classes **Name** property.

3. You are encouraged to use Bootstrap to style your pages. But if you prefer, you use **SiteStyles.css** to style your pages as in Lab 6.
4. Add a new web page **AddStudent.aspx** to the project. After completion this step, your Solution Explorer should look like:



Requirements

1. Web page AddStudent.aspx is for a user to enter new students into the system. This page should have the follow elements:
 - A text box to enter student name.
 - A dropdown list for selecting student type from **Full Time**, **Part Time** or **Coop**.
 - A table showing the ID and Name of the students currently in the system.



The screenshot shows a web browser window with the URL `https://localhost:44388/AddStudent.aspx`. The page has a yellow background. At the top, there is a title "Students" in bold. Below the title, there is a form with two fields: "Student Name" with a text box and "Student Type" with a dropdown menu showing "Select ...". Below these fields is a red "Add" button. At the bottom, there is a table with two columns: "ID" and "Name". The table is currently empty, and a red message "No Student Yet!" is displayed below the table header.

You are required to style the page. At minimum, all elements should be sized and spaced properly, and horizontally and vertically aligned.

No style or poorly styled page will result in up to 2 point deduction.

The web application should have the following functionality:

- Once the user entered the **Student Name** in the text box, selected **Student Type** from the dropdown list and clicked the **Add** button, the web application a student object of the selected type and add the student object to student list stored in the **Session** of the web application.
- The application shows the page again with the newly added student showing in the table. The Student Name text box should be blanked, and the student type dropdown list should be set back to the initial **Select ...** so that the user will not accidentally add the same student into the system again.

The screenshot shows a web browser window with the address bar displaying `https://localhost:44388/AddStudent.aspx`. The page has a yellow background and a title **Students**. Below the title, there is a form with two input fields: "Student Name:" followed by a text box, and "Student Type:" followed by a dropdown menu showing "Select ...". Below these fields is a red-bordered button labeled "Add". At the bottom of the page, there is a table with two columns: "ID" and "Name". The table contains three rows of data.

ID	Name
9450730	Wei Gong
9891412	John Smith
9046402	Peter Davison

Notes:

- For simplicity, if the user clicks the **Add** button without entering Student Name or select a Student Type, the web application will do nothing. We will learn how to validate the user entries and show proper validation error messages in the following weeks.