

Лабораторная работа №8. Работа с элементами управления. Валидация ввода данных.

1. Создайте пустой проект ASP.Net.
2. Добавьте к проекту страницу WebForm.aspx.
3. В html-коде страницы создайте разметку для будущей формы ввода данных. Форма будет представлять из себя таблицу из двух столбцов и трех строк. Последний (третий) столбец будет иметь ширину в два столбца. Для создания разметки наберите между тегами <div> и </div> следующий код (Рис.1):

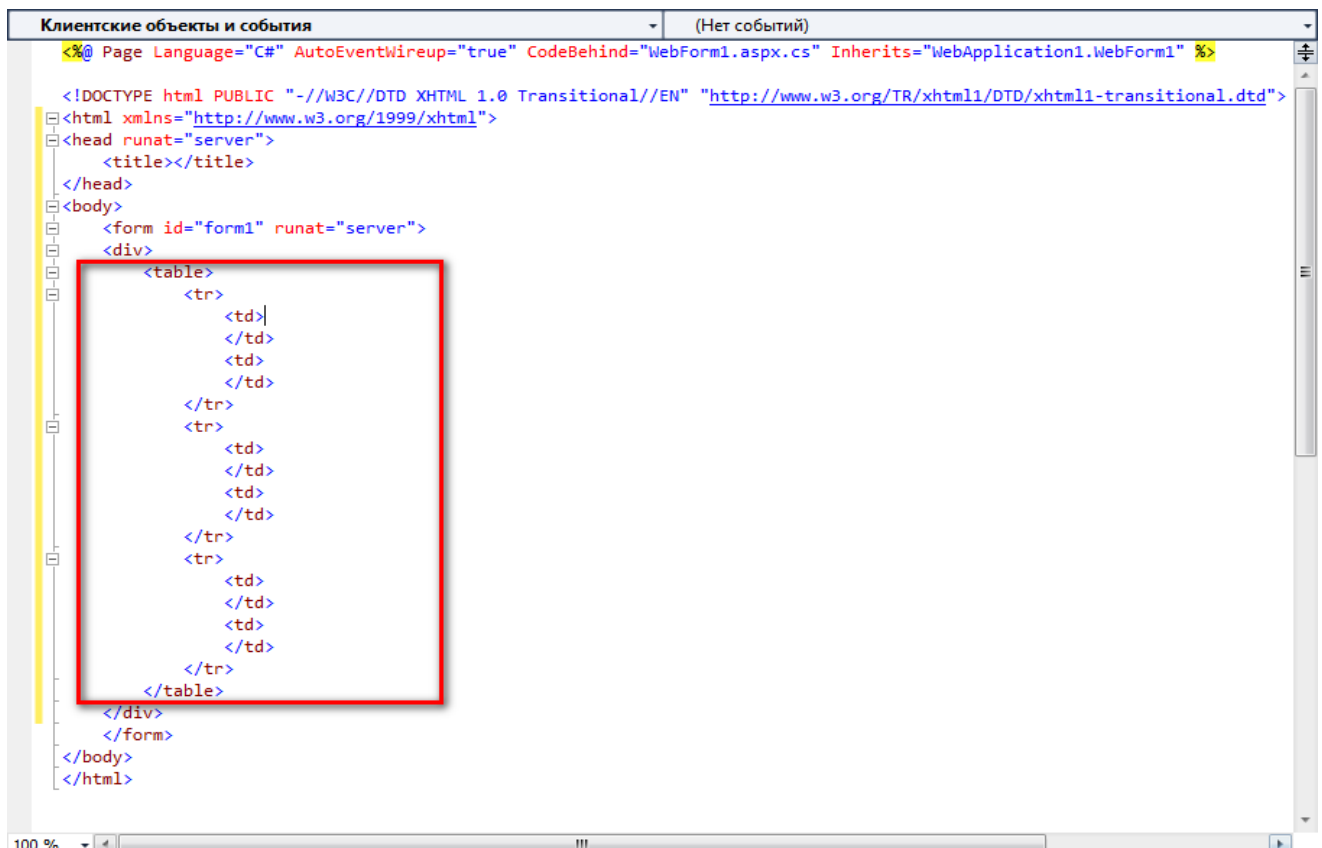


Рис. 1.

4. Добавьте необходимые элементы управления на форму: Два элемента Label, Два элемента TextBox и Один элемент Button. Из панели элементов (находится слева от страницы кода) перетащите соответствующие контролы на страницу. После добавления всех элементов, код должен выглядеть следующим образом (Рис.2):



Рис. 2.

5. Добавьте на страницу еще один элемент управления Label. Расположите его под кнопкой.
6. Создайте обработчик события нажатия на кнопку.
7. Внутри обработчика события наберите код, который будет обрабатывать данные, введенные пользователем в поля формы и выводить их в текстовую метку, расположенную под кнопкой.

Label4.Text = TextBox1.Text + " " + TextBox2.Text + " " + TextBox3.Text;

8. Запустите проект на выполнение и нажмите на кнопку для проверки отработки события.

Далее необходимо проверить данные, введенные пользователем (введены ли данные в поля, формат ввода телефона и e-mail адреса). Для этого расположим на форме для каждого из полей специальные элементы управления, валидаторы ввода, для каждого поля.

9. Перейдите в режим конструктора. Выделите мышкой TextBox1 и нажмите один раз стрелку вправо на клавиатуре. Курсор будет расположен справа от текстового поля.
10. В «Панели элементов» (расположена слева от окна разметки страницы) откройте вкладку «Проверка». Дважды нажмите мышью на элемент управления «RequiredFieldValidator» (курсор должен быть справа от текстового поля).
11. Выделите мышью добавленный валидатор и нажмите F4 для перехода в окно свойств.
12. Для свойства ControlToValidate выберите текстовое поле TextBox1 (*Свойство ControlToValidate задает элемент управления, который необходимо проверять на корректность ввода данных*).
13. Для свойства ErrorMessage (сообщение, которое будет выводиться на страницу при вводе некорректных данных) введите, например, символ *.
14. Далее проделайте операции 9-13 для TextBox2 и TextBox3.
15. Для поля e-mail, аналогично предыдущим пунктам, добавьте еще один валидатор - «RegularExpressionValidator». Расположите его рядом с валидатором «RequiredFieldValidator», добавленным ранее. Данный элемент управления позволит проверить формат введенного пользователем адреса электронной почты.
16. Откройте окно свойств данного валидатора.
17. Для свойства ControlToValidate выберите текстовое поле TextBox3.
18. Для свойства ErrorMessage введите следующий текст: e-mail введен не корректно.
19. Далее выберите свойство ValidationExpression (данное свойство позволяет установить формат данных, которые хотелось бы получить от пользователя при вводе данных) и нажмите на иконку с изображением «...».
20. В открывшемся окне выберите стандартное выражение «Адрес электронной почты в Интернет» и нажмите ОК.
21. Перейдите в режим разметки страницы и посмотрите, как изменился html-код, какие строки были добавлены в разметку страницы.

После того, как мы добавили валидаторы на страницу, форма уже будет проверять вводимые пользователем данные. Но нам необходимо изменить код логики страницы таким образом, чтобы данные проверялись до отправки запроса на сервер.

Для этого произведем следующие действия:

22. Перейдите в файл логики страницы «WebForm.aspx.cs».

23. Измените код обработчика события нажатия на кнопку следующим образом:

```
if (!Page.IsValid) return;
```

```
Label4.Text = TextBox1.Text + " " + TextBox2.Text + " " + TextBox3.Text;
```

24. Запустите проект на выполнение.

25. Нажмите на кнопку «Отправить» без ввода данных формы. Отработают валидаторы, размещенные на странице и справа от текстовых полей появятся символы «*».

26. Проверьте работоспособность валидатора формата e-mail адреса. Для этого введите в TextBox3 не верный e-mail адрес (например, без указания зоны ru или com, или без знака «@»).

*Во время отработки валидаторов на странице (если данные введены не верно) **не происходит** запроса на сервер. Инфраструктура ASP.Net формирует javascript-код для обработки вводимых данных. В случае если все валидаторы сообщили странице об успешном завершении своей работы (проверка свойства страницы Page.IsValid), происходит отправка данных формы на сервер и их обработка в соответствии с логикой (в нашем случае вывод данных в текстовую метку под кнопкой).*

Если пользователь заполняет много полей на форме, то он может допустить множество ошибок. Для того чтобы сообщить пользователю обо всех ошибках в удобном формате, существует элемент управления «ValidationSummary».

27. Перейдите в режим разметки страницы.

28. После тега <div> добавьте из вкладки «Проверка» на «Панели элементов» элемент управления «ValidationSummary».
29. Запустите проект на выполнение без заполнения полей формы. Над полями формы будет виден результат работы элемента управления «ValidationSummary» (три строки с символами «*»). Для того чтобы пользователь смог понять, какие ошибки произошли при вводе данных, необходимо изменить свойства валидаторов текстовых полей.
30. Перейдите на вкладку «Конструктор». Выделите мышкой валидатор «RequiredFieldValidator» рядом с первым текстовым полем.
31. Для свойства «ErrorMessage» установите значение «Укажите имя».
32. Для свойства «Text» установите значение «*».
33. Проведите аналогичные операции с валидаторами остальных текстовых полей («RequiredFieldValidator»).
34. Для всех валидаторов поменяйте значение свойства «Display» на «dynamic» (Свойство «Display» задает способ отображения проверки корректности ввода данных. Значение свойства «dynamic» говорит о том, что элемент управления, в который выводится сообщение об ошибке, будет генерироваться в разметке html только в случае возникновения ошибок ввода. Если оставить значение свойства равным «Static», то разметка для данного элемента будет сгенерирована при запуске проекта, а CSS свойство элемента display будет установлено в none).
35. Запустите проект на выполнение. Нажмите на кнопку без ввода данных в форму.

Ответьте на вопросы:

1. Для чего используются Валидаторы на странице?
2. Какие типы Валидаторов Вы знаете?
3. С помощью какого Валидатора можно проверить корректность ввода e-mail адреса?