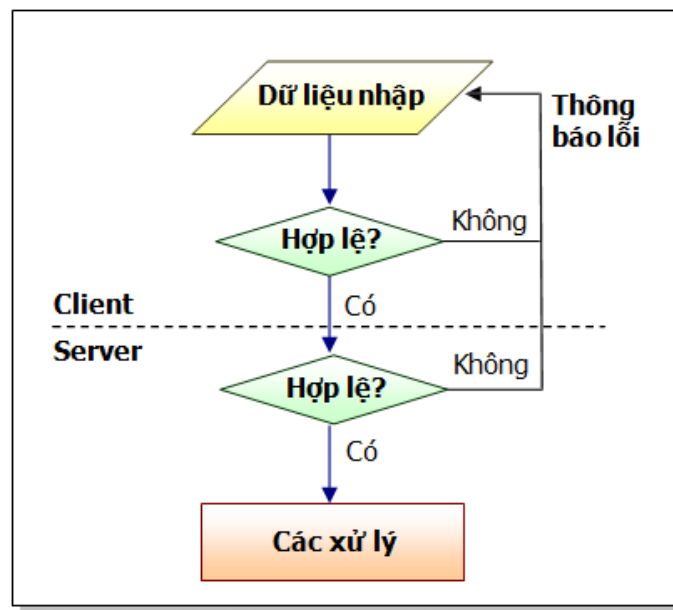


Khi xây dựng ứng dụng, chúng ta nên kiểm tra dữ liệu nhập từ người dùng để hạn chế các sai sót dữ liệu nhập nhằm đảm bảo việc thực hiện xử lý dữ liệu được chính xác theo các yêu cầu nghiệp vụ. Nếu chúng ta viết mã để kiểm tra phải mất nhiều thời gian (sử dụng JavaScript hoặc VBScript). Webform hỗ trợ các validation controls để kiểm tra dữ liệu nhập từ người dùng trong các sever controls, mục đích là tránh để người dùng nhập sai hoặc không được bỏ trống các thông tin quan trọng bắt buộc,...



Sơ đồ xử lý kiểm tra dữ liệu nhập tại Client và Server

Như các bạn đã biết, mỗi khi PostBack về Server, trang Web luôn kiểm tra tính hợp lệ dữ liệu (nếu có yêu cầu khi thiết kế). Nếu dữ liệu không hợp lệ (bỏ trống, vi phạm miền giá trị, mật khẩu nhập lại không đúng, ...), trang web sẽ không thể PostBack về Server.

➤ Các thuộc tính chung của các validation control

Thuộc tính	Ý nghĩa
ControlToValidate	Tên điều khiển cần kiểm tra. Đây là thuộc tính mà các bạn phải xác định khi sử dụng Validation Control.
Text	Chuỗi thông báo xuất hiện khi có lỗi.
ErrorMessage	Chuỗi thông báo xuất hiện trong điều khiển Validation Summary. Giá trị này sẽ được hiển thị tại vị trí của điều khiển nếu chúng ta không gán giá trị cho thuộc tính Text.
Display	Qui định hình thức hiển thị: <ul style="list-style-type: none"> ▪ None: Không hiển thị thông báo lỗi (vẫn có kiểm tra dữ liệu) ▪ Static: Trong trường hợp không có vi phạm dữ liệu, điều khiển không có hiển thị nhưng vẫn chiếm vị trí như trong lúc thiết kế. ▪ Dynamic: Trong trường hợp không có vi

	phạm dữ liệu, điều khiển không chiếm dụng vị trí trên màn hình.
EnableClientScript	Có cho phép thực hiện kiểm tra ở phía Client hay không ?. Giá trị mặc định là true - có kiểm tra.

7.2.1 Điều khiển Required Field Validator

Điều khiển này được dùng để kiểm tra giá trị trong điều khiển phải được nhập. Sử dụng điều khiển này để kiểm tra rằng buộc dữ liệu khác rỗng (bắt buộc nhập).

➤ Thuộc tính

- **InitialValue:** Giá trị khởi động. Giá trị bạn nhập vào phải khác với giá trị của thuộc tính này. Giá trị mặc định của thuộc tính này là chuỗi rỗng.

7.2.2 Điều khiển Compare Validator

Điều khiển này được dùng để so sánh giá trị của một điều khiển với giá trị của một điều khiển khác hoặc một giá trị được xác định trước.

Thông qua thuộc tính Operator, chúng ta có thể thực hiện các phép so sánh như: =, <, >, >=, <=, <= hoặc dùng để kiểm tra kiểu dữ liệu (DataTypeCheck).

Sử dụng điều khiển này để kiểm tra rằng buộc miền giá trị, kiểu dữ liệu, liên thuộc tính.

Lưu ý: Trong trường hợp không nhập dữ liệu, điều khiển sẽ không thực hiện kiểm tra vi phạm.

➤ Các thuộc tính

- **ControlToCompare:** Tên điều khiển cần so sánh giá trị. Nếu bạn chọn giá trị của thuộc tính Operator = DataTypeCheck thì không cần phải xác định giá trị cho thuộc tính này.
- **Operator:** Qui định phép so sánh, kiểm tra kiểu dữ liệu
 - Equal: = (Đây là giá trị mặc định)
 - GreaterThan: >
 - GreaterThanEqual: >=
 - LessThan: <
 - LessThanEqual: <=

- NotEqual: <>
- DataTypeCheck: Kiểm tra kiểu dữ liệu
- Type: Quy định kiểu dữ liệu để kiểm tra hoặc so sánh.
 - String
 - Integer
 - Double
 - Date
 - Currency
- **ValueToCompare: Giá trị cần so sánh.** Trong trường hợp bạn xác định giá trị của cả 2 thuộc tính ControlToCompare và ValueToCompare thì giá trị của điều khiển được quy định bởi thuộc tính ControlToCompare được ưu tiên dùng để kiểm tra.

7.2.3 Điều khiển Range Validator

Điều khiển này được dùng để kiểm tra giá trị trong điều khiển phải nằm trong đoạn [min-max]. Sử dụng điều khiển này để kiểm tra ràng buộc miền giá trị của dữ liệu.

Lưu ý: Trong trường hợp không nhập dữ liệu, điều khiển sẽ không thực hiện kiểm tra vi phạm.

➤ Các thuộc tính

- MinimumValue: Giá trị nhỏ nhất.
- MaximumValue: Giá trị lớn nhất.
- Type: Xác định kiểu để kiểm tra dữ liệu. Ta có thể thực hiện kiểm tra trên các kiểu dữ liệu sau:
 - String
 - Integer
 - Double
 - Date
 - Currency

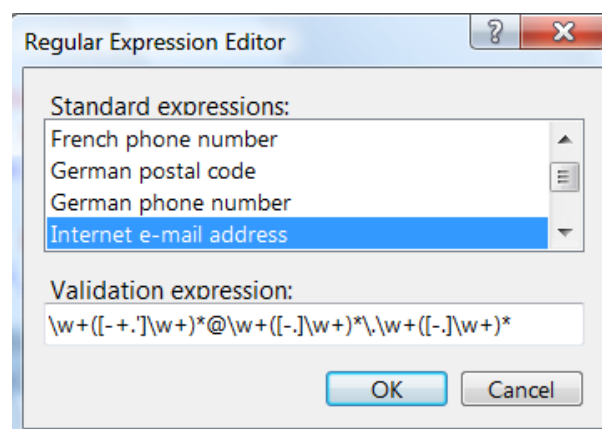
7.2.4 Điều khiển Regular Expression Validator

Điều khiển này được dùng để kiểm tra giá trị của điều khiển phải theo mẫu được qui định trước: địa chỉ email, số điện thoại, mã vùng, số chứng minh thư, ...

Lưu ý: Trong trường hợp không nhập dữ liệu, điều khiển sẽ không thực hiện kiểm tra vi phạm.

➤ Thuộc tính:

- ValidationExpression: Qui định mẫu kiểm tra dữ liệu.



Bảng mô tả các ký hiệu thường sử dụng trong Validation Expression

Ký hiệu	Mô tả
A	Ký tự chữ cái (đã được xác định). Ở đây là chữ a
1	Ký tự số (đã được xác định). Ở đây là số 1
[0-n]	Một ký tự số từ 0 đến 9
[abc]	Một ký tự: hoặc a hoặc b hoặc c
 	Lựa chọn mẫu này hoặc mẫu khác
\w	Ký tự thay thế phải là một ký tự chữ cái
\d	Ký tự thay thế phải là một ký tự số
\	Thể hiện các ký tự đặc biệt theo sau.
\.	Ký tự thay thế phải là dấu chấm câu (.)
?	Qui định số lần xuất hiện: 0 hoặc 1 lần
*	Qui định số lần xuất hiện: 0 hoặc nhiều lần
+	Qui định số lần xuất hiện: 1 hoặc nhiều lần (ít nhất là 1)
{n}	Qui định số lần xuất hiện: đúng n lần

7.2.5 Điều khiển Custom Validator

Điều khiển này cho phép bạn tự viết hàm xử lý kiểm tra lỗi.

➤ Sự kiện

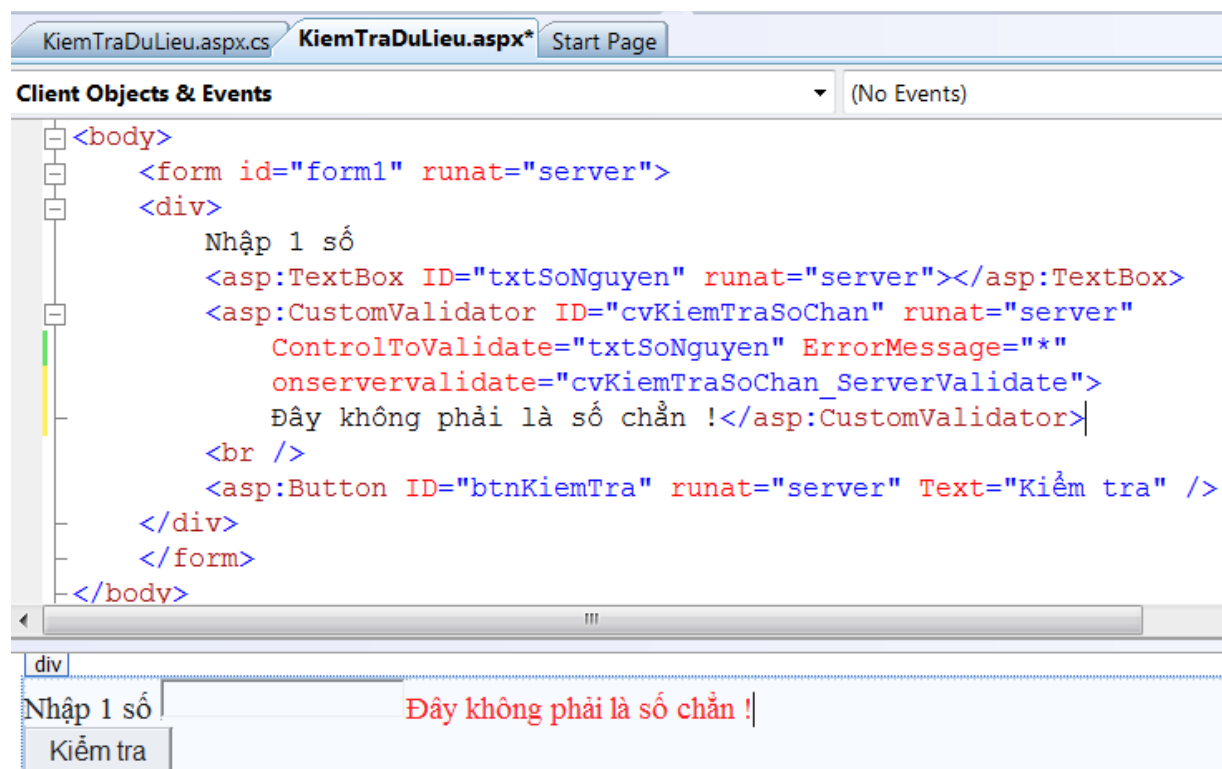
- **ServerValidate:** Đặt các xử lý kiểm tra dữ liệu trong sự kiện này. Việc kiểm tra này được thực hiện ở Server.

Ví dụ: Minh họa xử lý kiểm tra dữ liệu nhập tại điều khiển txtSoNguyen có phải là số chẵn hay không ?

Bước 1 : Tạo 1 trang KiemTraDuLieu.aspx có giao diện như hình 7.6

Bước 2 : Viết lệnh xử lý sự kiện , hình 7.7

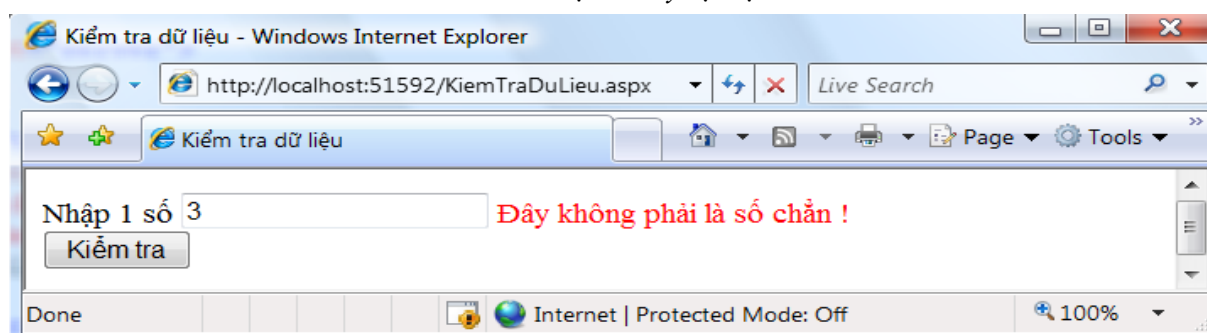
Bước 3 : Nhấn Ctrl+F5 thi hành ứng dụng , nhập vào 1 số nguyên và nhấn nút **Kiểm tra** , xem kết quả hình 7.8



Hình 7.6 : Giao diện trang KiemTraDuLieu.aspx

```
public partial class KiemTraDuLieu : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void cvKiemTraSoChan_ServerValidate(object source,
        ServerValidateEventArgs args)
    {
        int n = int.Parse(args.Value);
        if (n % 2 == 0)
            args.IsValid = true;
        else
            args.IsValid = false;
    }
}
```

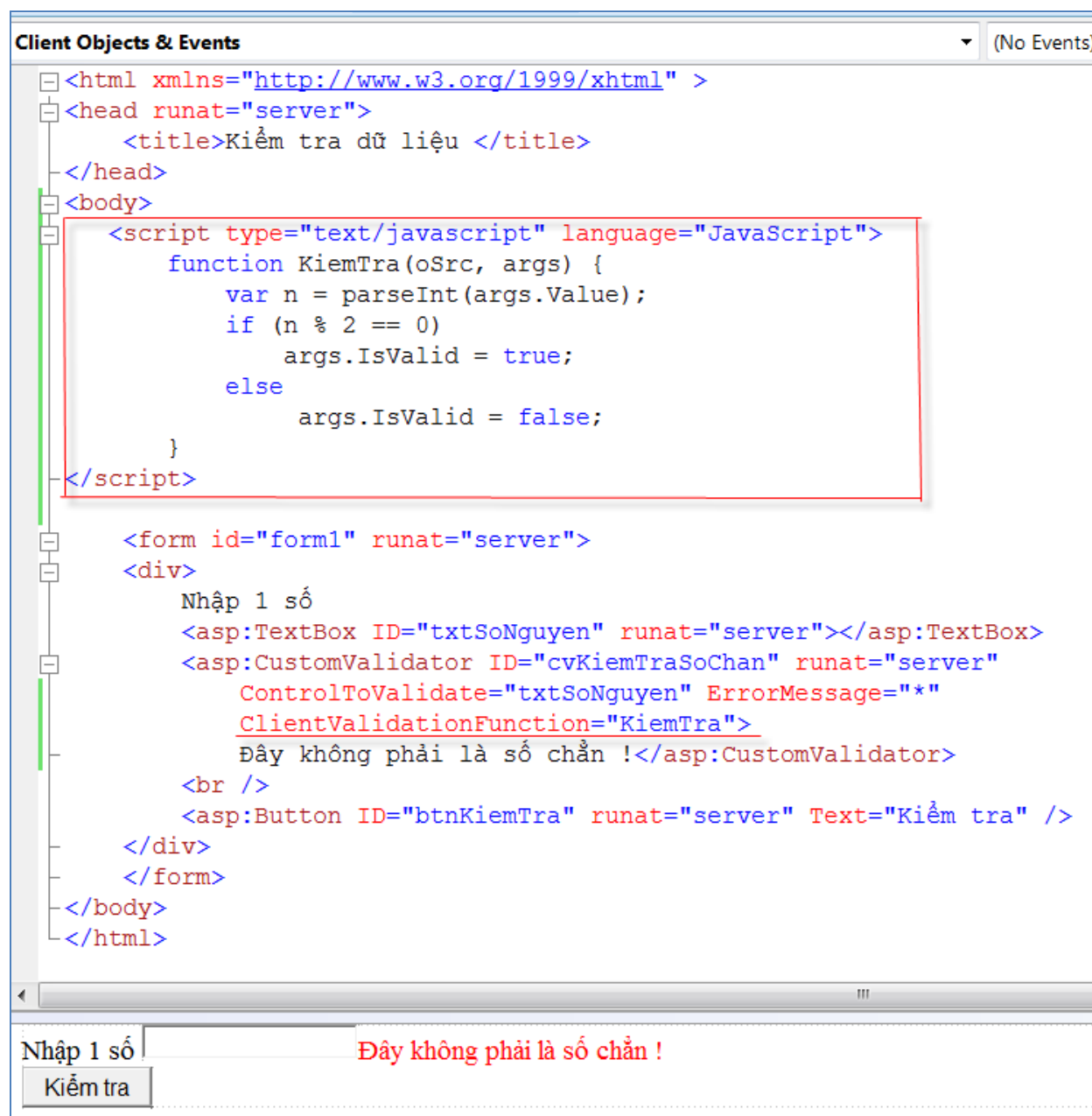
Hình 7.7 : Viết lệnh xử lý sự kiện



Hình 7.8 : Kết quả kiểm tra

Ví dụ : Minh họa kiểm tra tại phía Client

Để kiểm tra tại phía Client chúng ta viết lệnh bằng JavaScript trong phần HTML như sau :



Lưu ý : Khi kiểm tra tại phía Server , sau khi chúng ta nhập vào số nguyên và nhấn **Kiểm tra** để xem kết quả nhưng khi kiểm tra tại phía Client chúng ta chỉ cần nhập vào số nguyên và nhấp chuột ra khỏi TextBox để xem kết quả .

7.2.6 Điều khiển Validation Summary

Điều khiển này được dùng để hiển thị ra bảng lỗi - tất cả các lỗi hiện có trên trang Web. Nếu điều khiển nào có dữ liệu không hợp lệ, chuỗi thông báo lỗi - giá trị thuộc tính ErrorMessage của Validation Control sẽ được hiển thị. Nếu giá trị của thuộc tính ErrorMessage không được xác định, thông báo lỗi đó sẽ không được xuất hiện trong bảng lỗi.

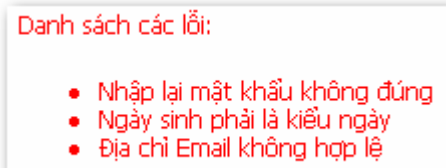
➤ Các thuộc tính

- HeaderText: Dòng tiêu đề của thông báo lỗi
- ShowMessageBox: Qui định bảng thông báo lỗi có được phép hiển thị như cửa sổ MessageBox hay không. Giá trị mặc định của thuộc tính này là False - không hiển thị.



Thông báo lỗi xuất hiện qua cửa sổ MessageBox

- ShowSummary: Qui định bảng thông báo lỗi có được phép hiển thị hay không. Giá trị mặc định của thuộc tính này là True - được phép hiển thị.



Thông báo lỗi hiển thị trực tiếp trên trang Web

Ví dụ : Minh họa sử dụng các validation control

Tạo 1 trang ASP.Net tên DangKy.aspx với giao diện và điều kiện ràng buộc như sau :

Phiếu Đăng Ký Thành Viên Diễn Đàn Tin Học

Tên đăng nhập Bạn phải nhập tên !

Tuổi 18<= Tuổi <=40

Mật khẩu Mật khẩu không hợp lệ

Nhập lại mật khẩu:

Email: Email không hợp lệ

◆ Các ràng buộc dữ liệu được mô tả như sau:

- Tên đăng nhập không được bỏ trống
- Tuổi phải nằm trong khoảng từ 18 đến 40
- Mật khẩu nhập phải chính xác(nhập 2 lần).
- Email phải hợp lệ(ví dụ: abc@yahoo.com)

◆ Bảng mô tả các thuộc tính của các controls

STT	Control	Tên thuộc tính	Giá trị thuộc tính
1	Lable	Text	Tên đăng nhập
2	Lable	Text	Tuổi
3	Lable	Text	Mật khẩu
4	Lable	Text	Nhập lại mật khẩu
5	Lable	Text	Email
6	Lable	Text	Thông báo
7	TextBox	ID	txtTenDangNhap
8	TextBox	ID	txtTui
9	TextBox	ID	txtMatKhau
10	TextBox	ID	txtNhapLaiMatKhau
11	TextBox	ID	txtEmail
12	RequiredFieldValidator	ControlToValidate	txtTenDangNhap
		ErrorMessage	Bạn phải nhập tên !

13	RequiredFieldValidator	ControlToValidate	txtNhapLaiMatKhau
		ErrorMessage	Bạn phải nhập lại mật khẩu !
14	RangeValidator	ControlToValidate	txtTuoi
		ErrorMessage	18<= tuổi <=40
		MaximumValue	40
		MinimumValue	18
		Type	Integer
15	CompareValidator	ControlToCompare	txtNhapLaiMatKhau
		ControlToValidate	txtMatKhau
		ErrorMessage	Mật khẩu không hợp lệ
16	RegularExpressionValidator	ErrorMessage	Email không hợp lệ
		ValidationExpression	Nhấp vào nút ... và chọn Internet e-mail address.
		ControlToValidate	txtEmail
13	RequiredFieldValidator	ControlToValidate	txtNhapLaiMatKhau
		ErrorMessage	Bạn phải nhập lại mật khẩu !
14	RangeValidator	ControlToValidate	txtTuoi
		ErrorMessage	18<= tuổi <=40
		MaximumValue	40
		MinimumValue	18
		Type	Integer
15	CompareValidator	ControlToCompare	txtNhapLaiMatKhau
		ControlToValidate	txtMatKhau
		ErrorMessage	Mật khẩu không hợp lệ
16	RegularExpressionValidator	ErrorMessage	Email không hợp lệ
		ValidationExpression	Nhấp vào nút ... và chọn Internet e-mail address.
		ControlToValidate	txtEmail

❖ Sau khi chạy trang DangKy.aspx, nếu ta nhập vào giá trị không hợp lệ và nhấn nút **Kiểm Tra** thì sẽ xuất hiện thông báo lỗi tương ứng như hình trên.