# Bài: Event chuẩn .Net trong C#

*Xem bài học trên website để ủng hộ Kteam: Event chuẩn .Net trong C#*

*Mọi vấn đề về lỗi website làm ảnh hưởng đến bạn hoặc thắc mắc, mong muốn khóa học mới, nhằm hỗ trợ cải thiện Website. Các bạn vui lòng phản hồi đến Fanpage How Kteam nhé!*

## Dẫn nhập

Ở các bài học trước, chúng ta đã cùng nhau tìm hiểu về EVENT VỚI DELEGATE TRONG C#. Hôm nay chúng ta sẽ cùng tìm hiểu về **Event chuẩn .Net trong C#**.

---

## Nội dung

Để đọc hiểu bài này tốt nhất các bạn nên có kiến thức cơ bản về các phần:

- LẬP TRÌNH C# CƠ BẢN
- LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG trong C#
- EVENT VỚI DELEGATE TRONG C#

Trong bài học này, chúng ta sẽ cùng tìm hiểu các vấn đề:

- Event chuẩn .Net là gì?
- Cách dùng Event chuẩn .Net trong C#
- Dùng Event chuẩn .Net với tham số truyền vào trong C#

## Event chuẩn .Net là gì?

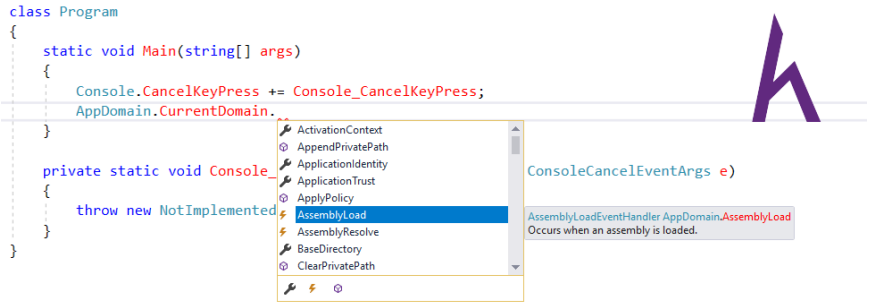
**Event chuẩn .Net** là event với Delegate nhưng thỏa mãn các điều kiện:

- Delegate có kiểu trả về là void
- Delegate có hai tham số, tham số thứ nhất có kiểu dữ liệu là object, tham số thứ hai có kiểu EventArgs. object chính là đối tượng phát sinh sự kiện, EventArgs chính là class giữ thông tin mà đối tượng gửi kèm trong quá trình phát sinh sự kiện.
- Lúc này thay vì chúng ta dùng Delegate do chúng ta tự tạo thì **.Net** có sẵn Delegate tên là EventHandler theo chuẩn ở trên.

Vậy các event có sẵn trong **.Net** như Console.CancelKeyPress hay nhiều event khác nữa

```
class Program
{
    static void Main(string[] args)
    {
        Console.CancelKeyPress += Console_CancelKeyPress;
    }

    private static void Console_CancelKeyPress(object sender, ConsoleCancelEventArgs e)
    {
        throw new NotImplementedException();
    }
}
```

Các event này thường được đánh ký hiệu là **tia sét**:




# Cách dùng Event chuẩn .Net trong C#

Với mong muốn tạo ra một class HocSinh để quản lý Tên của học sinh. Vì muốn biết khi nào tên của học sinh thay đổi sẽ ghi lại thành **log** sau này đối chiếu lịch sử thay đổi này. Chúng ta sẽ tạo một Event NameChanged với Delegate là EventHandler có sẵn của .Net và ủy thác việc ghi log lại.

Mình cũng đóng gói event mình sẽ tạo. Nhưng với event thay vì get set thì sẽ là add và remove. Đồng thời tạo hàm OnNameChanged để thông báo event đã có sự thay đổi.

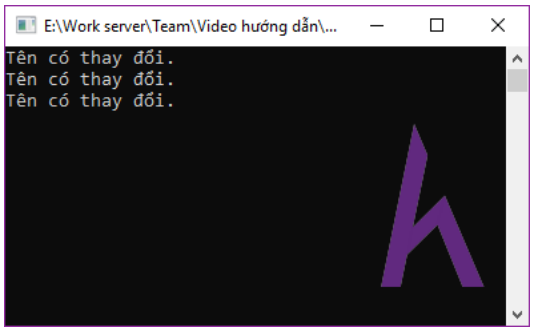
> **Lưu ý:** sử dụng _NameChanged chứ không phải NameChanged. Vì filed _NameChanged chính là event thật sự. còn event NameChanged chính là event nhận hàm ủy thác add và remove mà thôi.

Chúng ta truyền mặc định this vào tham số đầu tiên để thể hiện class HocSinh gọi event này, tạo một đối tượng EventArgs vào tham số thứ hai để thỏa mãn đầy đủ tham số cho Delegate EventHandler. Sau này, bạn có thể dùng các tham số này ở phần sau.

**C#:**

```
class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = Encoding.Unicode;

            HocSinh hs = new HocSinh();
            hs.NameChanged += Hs_NameChanged;
            hs.Name = "Tên lần 1";
            hs.Name = "Tên lần 2";
            hs.Name = "Tên cuối";

            Console.ReadLine();
        }

        private static void Hs_NameChanged(object sender, EventArgs e)
        {
            Console.WriteLine("Tên có thay đổi.");
        }
    }

    public class HocSinh
    {
        private string _Name;
        public string Name
        {
            get => _Name;
            set
            {
                _Name = value;
                OnNameChanged();
            }
        }


        private event EventHandler _NameChanged;
        public event EventHandler NameChanged
        {
            add
            {
                _NameChanged += value;
            }
            remove
            {
                _NameChanged -= value;
            }
        }

        void OnNameChanged()
        {
            if(_NameChanged != null)
            {
                _NameChanged(this, new EventArgs());
            }
        }
    }
```

Mỗi khi có sự thay đổi tên của học sinh thì event NameChanged được thông báo lên. Vậy là chúng ta có thể biết để ghi ra màn hình cho biết tên học sinh đã thay đổi.




## Dùng Event chuẩn .Net với tham số truyền vào

Chúng ta đã biết khi nào tên của học sinh thay đổi và thông báo. Nhưng không thể biết học sinh đó đã đổi tên thành gì. Vậy để có thể biết được giá trị sau khi cập nhật là gì? Chúng ta sẽ dùng tới tham số thứ hai đó là EventArgs.

Vì tham số thứ hai chỉ cần có kiểu dữ liệu là EventArgs nên chúng ta sẽ tạo một class mới là NameChangedEventArgs kết thừa lại EventArgs .

Class này sẽ có **Constructor** với tham số đầu vào là tên mới của học sinh. Và có thuộc tính là Name public ra.

Ngay tại vị trí khai báo event NameChanged và _NameChanged. Chúng ta sẽ thêm generic NameChangedEventArgs vào sau EventHandler để thông báo rằng mình sẽ dùng kiểu dữ liệu NameChangedEventArgs thay cho EventArgs.

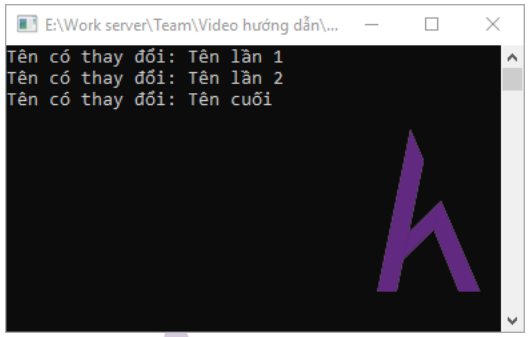
Khi thông báo event được thực hiện trong hàm OnNameChanged. Chúng ta sẽ thay new EventArgs thành new NameChangedEventArgs đồng thời truyền tên mới của học sinh vào. Hàm OnNameChanged cũng phải thêm tham số đầu vào kiểu dữ liệu là string để có thể truyền tên mới của học sinh vào. Đồng thời cũng phải truyền value vào khi gọi hàm OnNameChanged tại hàm set của Name.

Lúc này, phía ủy thác event cũng phải thay đổi EventArgs thành NameChangedEventArgs có thể lấy tên này ra dùng thông qua tham số thứ hai đang có tên là **e**. Chúng ta sẽ thông báo ra tên mới của học sinh là gì bằng cách cộng thêm chuỗi e.Name.

**C#:**

```
class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = Encoding.Unicode;

            HocSinh hs = new HocSinh();
            hs.NameChanged += Hs_NameChanged;
            hs.Name = "Tên lần 1";
            hs.Name = "Tên lần 2";
            hs.Name = "Tên cuối";

            Console.ReadLine();
        }

        private static void Hs_NameChanged(object sender, NameChangedEventArgs e)
        {
            Console.WriteLine("Tên có thay đổi: " + e.Name);
        }
    }

    public class HocSinh
    {
        private string _Name;
        public string Name
        {
            get => _Name;
            set
            {
                _Name = value;
                OnNameChanged(value);
            }
        }


        private event EventHandler<NameChangedEventArgs> _NameChanged;
        public event EventHandler<NameChangedEventArgs> NameChanged
        {
            add
            {
                _NameChanged += value;
            }
            remove
            {
                _NameChanged -= value;
            }
        }

        void OnNameChanged(string name)
        {
            if(_NameChanged != null)
            {
                _NameChanged(this, new NameChangedEventArgs(name));
            }
        }
    }

    public class NameChangedEventArgs : EventArgs
    {
        public string Name { get; set; }
        public NameChangedEventArgs(string name)
        {
            Name = name;
        }
    }
```

**Kết quả:**




## Kết luận

Nội dung bài này giúp các bạn nắm được:

- Event chuẩn .Net là gì?
- Cách dùng Event chuẩn .Net trong C#
- Dùng Event chuẩn .Net với tham số truyền vào trong C#

Bài học sau chúng ta sẽ cùng tìm hiểu về MULTI THREADING TRONG C#.

Cảm ơn các bạn đã theo dõi bài viết. Hãy để lại bình luận hoặc góp ý của mình để phát triển bài viết tốt hơn. Đừng quên "**Luyện tập – Thử thách – Không ngại khó**".