Bài: Event với Delegate trong C#

Xem bài học trên website để ủng hộ Kteam: Event với Delegate 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 DELEGATE TRONG C#. Hôm nay chúng ta sẽ cùng tìm hiểu về **Event trong C#**.

Nội dung

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

- BIẾN và KIỂU DỮ LIỆU, TOÁN TỬ trong C#
- CÂU ĐIỀU KIỆN trong C#
- Cấu trúc cơ bản của VÒNG LẶP , HÀM , MẢNG trong C#
- LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG trong C#

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

- Event là gì?
- Khai báo Event trong C#
- Khởi tạo Event trong C#
- Cách dùng Event trong C#

Event là gì?

Event là Delegate với mục đích để cho lớp khác hoặc **đối tượng cha của đối tượng hiện tại ủy thác(định nghĩa) hàm vào trong đó**.

Mục đích chính của chuyện này là để **thông báo** lên cho đối tượng cha biết mà xử lý.

Khai báo Event trong C#

Khai báo Event trong C# sẽ tương tự như khai báo một biến. Nhưng biến này sẽ nhận kiểu dữ liệu là Delegate đã được tạo trước đó. Cần có từ khóa event để chương trình biết đây là một biến event.

Công thức:

```
event <Kiểu delegate> <tên event>;
```

Ví dụ:

C#:

```
event UpdateNameHandler NameChanged;
```

Ví dụ code đầy đủ:

C#:

```
namespace Event_Voi_Delegate
{
		public delegate void UpdateNameHandler(string name);
		class Program
		{
			static void Main(string[] args)
			{
			}
		}

		public class HocSinh
		{
			public event UpdateNameHandler NameChanged;

			private string _Name;
			public string Name
			{
			get => _Name;
			set
			{
					_Name = value;
			}
			}
		}
}
```

> **Lưu ý:** Chữ event viết thường

Mục đích là mình mong muốn mỗi khi **Name** của class HocSinh thay đổi mình sẽ biết và có thể code xử lý tương ứng.

Ta tạo một class HocSinh có filed là _Name kiểu dữ liệu là string. Được đóng gói thành property Name.(Bạn để ý rằng mình cố ý tạo property và filed cùng tên nhưng khác nhau là filed có dấu _ phía trước tên. Như vậy vừa dễ quản lý vừa tiện cho việc code)

Lúc này chúng ta đã tạo một Delegate có tên là UpdateNameHandler cùng cấp với class Program và class HocSinh . UpdateNameHandler có kiểu trả về là void, một tham số đầu vào là string. Ta cần tạo Delegate ở phạm vi này là vì muốn có thể được dùng cả trong class HocSinh và class Program.

Ta tạo event NameChanged thuộc class HocSinh lúc này có kiểu dữ liệu là Delegate UpdateNameHandler . Lưu ý cách đặt tên event thể hiện được tính chất là Name Changed(đã thay đổi)

> **Lưu ý:** event phải public.

Khởi tạo và sử dụng Delegate trong C#

Event phải được ủy thác từ đối tượng cha của đối tượng chứa event . Bằng cách **+=** hàm Delegate tương ứng vào event của đối tượng(Tương tự có thể loại bỏ bằng cách **-=**).

Ví dụ:

C#:

```
HocSinh hs = new HocSinh();

hs.NameChanged += Hs_NameChanged;
```

Hàm Hs_NameChanged lúc này được tự tạo ra bằng cách gõ cú pháp sau và nhấn phím tab một lần:

> hs.NameChanged **+=**

Hàm Hs_NameChanged lúc này được tự tạo ra bên dưới hàm **Main** với kiểu trả về và tham số đầu vào tương ứng với Delegate UpdateNameHandler.

Hoặc bạn cũng có thể dùng **anonymous method** trong tình huống này:

C#:

```
hs.NameChanged += (name) =>
      {
            Console.WriteLine("Tên mới: " + name);
      };
```

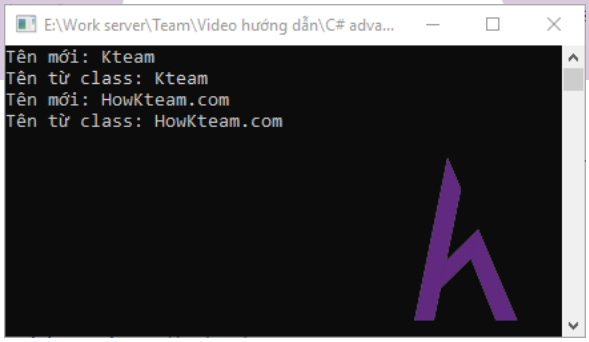
Trong phương thức set của property Name. Mình sẽ gọi event NameChanged. Nhưng nếu như NameChanged chưa từng được ủy thác thì khi gọi sẽ bị null exception. Nên mình sẽ kiểm tra NameChanged có null hay không? Nếu không null thì sẽ gọi event như hàm và truyền param tương ứng vào là Name.

Code hoàn chỉnh:

C#:

```
namespace Event_Voi_Delegate
{
    public delegate void UpdateNameHandler(string name);
    class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = Encoding.Unicode;

            HocSinh hs = new HocSinh();

            hs.NameChanged += Hs_NameChanged;

            hs.Name = "Kteam";
            Console.WriteLine("Tên từ class: " + hs.Name);
            hs.Name = "HowKteam.com";
            Console.WriteLine("Tên từ class: " + hs.Name);

            Console.ReadLine();
        }

        private static void Hs_NameChanged(string name)
        {
            Console.WriteLine("Tên mới: " + name);
        }
    }

    public class HocSinh
    {
        public event UpdateNameHandler NameChanged;

        private string _Name;
        public string Name
        {
            get => _Name;
            set
            {
                _Name = value;
                if(NameChanged != null)
                {
                    NameChanged(Name);
                }
            }
        }
    }
}
```

Kết quả: Khi chạy chương trình:



Bạn có thể thấy là mỗi khi mình set lại giá trị cho hs.Name thì hàm Hs_NameChanged đã được gọi ngay sau đó.

Kết luận

Nội dung bài này giúp các bạn nắm được:

- Event với Delegate là gì?
- Khai báo Event trong C#
- Khởi tạo Event trong C#
- Cách dùng Event trong C#

Bài học sau chúng ta sẽ cùng tìm hiểu về EVENT CHUẨN .NET 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**".