Nếu chúng ta bỏ đi get và set ở khai báo thuộc tính Name trong lớp Person, từ public virtual string Name { get; set; } chuyển thành public string Name;, một số vấn đề sẽ xảy ra. Dưới đây là chi tiết về các vấn đề này và cách chúng ảnh hưởng đến chương trình của chúng ta

**Vấn đề khi bỏ get và set**

1. **Không thể truy cập thuộc tính từ bên ngoài lớp**: Nếu bạn không định nghĩa các accessor get và set cho thuộc tính Name, nó sẽ trở thành một trường (field) thay vì một thuộc tính (property). Điều này có nghĩa là bạn sẽ không thể truy cập hoặc thiết lập giá trị của nó từ bên ngoài lớp.
2. **Không thể ghi đè thuộc tính trong lớp dẫn xuất**: Thuộc tính virtual với get và set cho phép lớp dẫn xuất ghi đè nó. Nếu bạn bỏ get và set, bạn không thể sử dụng từ khóa override trong lớp Student để cung cấp logic riêng cho việc lấy và thiết lập giá trị.

**Điều gì xảy ra nếu bỏ get và set**

Dưới đây là mã nguồn có vấn đề nếu chúng ta bỏ get và set của Name trong Person.

**Person.cs**

public abstract class Person

{

public string Name; // Không có get và set

}

**Interface.cs**

public interface KPI

{

void kpi();

}

**Student.cs**

public class Student : Person, KPI

{

public string Major { get; set; }

// Sẽ không thể ghi đè thuộc tính Name vì nó không còn là property

public void kpi()

{

Console.WriteLine($"Student {Name} is in the {Major} department.");

}

}

**Program.cs**

using System;

class Program

{

static void Main(string[] args)

{

// Khai báo một đối tượng kiểu Person, cấp phát cho nó một đối tượng Student

Person person = new Student { Name = "Nguyễn Văn Nam", Major = "ICT" };

// Kiểm tra đối tượng có phải là kiểu Student hay không

if (person is Student student)

{

// Gọi phương thức kpi

student.kpi();

}

// Thử cài đặt giá trị không hợp lệ cho Name

try

{

person.Name = ""; // Điều này sẽ không ném ra ArgumentException như mong đợi

}

catch (ArgumentException ex)

{

Console.WriteLine(ex.Message);

}

}

}

**Vấn đề phát sinh**

1. **Không thể sử dụng initializer**:

Person person = new Student { Name = "Nguyễn Văn Nam", Major = "ICT" }; // Lỗi biên dịch

Do Name không có get và set, bạn không thể sử dụng initializer để thiết lập giá trị cho nó.

1. **Không thể ghi đè thuộc tính Name trong lớp Student**:

public override string Name // Lỗi biên dịch

{

get { return base.Name; }

set

{

if (string.IsNullOrWhiteSpace(value))

{

throw new ArgumentException("Name cannot be empty or null.");

}

base.Name = value;

}

}

Bạn sẽ gặp lỗi biên dịch vì Name không còn là một property virtual có thể ghi đè.

**Kết luận**

Việc bỏ đi get và set sẽ làm cho thuộc tính Name trở thành một trường (field) thông thường và không thể truy cập hay ghi đè từ bên ngoài hoặc trong lớp dẫn xuất. Điều này sẽ phá vỡ logic của chương trình và gây ra nhiều lỗi biên dịch. Để tránh những vấn đề này, bạn nên giữ lại get và set cho thuộc tính Name.