**Snippet 10 — Properties (get; set;)**

* **Code Recap**

public class MyClass

{

    public string MyProperty { get; set; }

    public int OtherProperty { get; set; }

}

1. **Working Theory (keywords & concepts)**

**What is a Property?**

* A **property** is a special member that provides **controlled access to a field**.
* Instead of exposing fields directly, we use **get; and set; accessors**.
* Compiler generates a **backing field** behind the scenes for auto-properties.

**get**

* Retrieves the value of the property.
* Acts like a “read”.

**set**

* Assigns a new value to the property.
* Acts like a “write”.
* Special variable value represents the incoming assignment.

Example with custom logic:

private int age; // Variable

public int Age // function

{

    get { return age; }

    set

    {

        if (value >= 0) age = value;

        else throw new ArgumentException("Age must be positive and greater than zero");

    }

}

**Auto-Properties (snippet example)**

public string MyProperty { get; set; }

* Shorthand syntax introduced in C# 3.
* Compiler auto-creates the hidden field.
* Equivalent to writing a full property with a backing field.

**Access Modifiers on Accessors**

You can restrict access individually:

public string Name { get; private set; }

* Anyone can read Name,
* Only inside the class can Name be changed.

**Practical Example:**

1. **Person Class: Create a class name as “Person.cs” and write this code.**

public class Person

{

    public string Name { get; set; }

    public int Age { get; private set; }

    public Person(string name, int age)

    {

        Name = name;

        Age = age;

    }

}

1. **Program.cs**

class Program

{

    static void Main()

    {

        var person = new Person("Anas", 22);

        Console.WriteLine(person.Name); // Anas

        //person.Age = 30; // ❌ not allowed (private setter)

        Console.WriteLine(person.Age); // Anas

        person.Name = "Ali"; // ✅ allowed

        Console.WriteLine(person.Name); // Ali

    }

}

**Output**

Anas

22

Ali

**🔧 Extras**

* **Auto-properties with default values**

public string Country { get; set; } = "Pakistan";

* **Read-only property**

public string Id { get; } = Guid.NewGuid().ToString();

* **Expression-bodied properties**

public int DoubleAge => Age \* 2;

✅ snippet 10 explained: you now understand **properties, auto-properties, get/set, and why they’re safer than public fields**.