C# Properties

C# Properites doesn't have storage location. C# Properites are extension of fields and accessed like fields.

The Properties have accessors that are used to set, get or compute their values.

Usage of C# Properties

1. C# Properties can be read-only or write-only.
2. We can have logic while setting values in the C# Properties.
3. We make fields of the class private, so that fields can't be accessed from outside the class directly. Now we are forced to use C# properties for setting or getting values.

C# Properties Example

```
using System;

public class Customer
{
    private int m_id = -1; // Default Value remain same.

    public int GetID()
    {
        return m_id;
    }

    public void SetID(int id)
    {
        m_id = id;
    }

    private string m_name = " Haris "; // Default Value remain same.


    public string GetName()
    {
        return m_name;
    }

    public void SetName(string name)
    {
        m_name = name;
    }
}



public class CustomerManagerWithAccessorMethods
{
```

```
    public static void Main()
    {
        Customer cust = new Customer();

        cust.SetID(10);
        cust.SetName("Ahmed");

      Console.WriteLine("ID: {0}, Name: {1}",cust.GetID(), cust.GetName());

        Console.ReadKey();
    }
}
```