C# INTERMEDIATE

PROPERTIES

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Intermediate\_12\_Properties

{

    public class Person

    {

        //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        // This is an Auto-Implemented property

        //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        // If you create an auto-implementd property, you cannot specify

        // if you want the getter or setter excluded. The getter and setter

        // are created automatically for you.

        // To prevent users from using the setter in a property

        // you can set the property to private. This would act as a

        // safety mechanism to prevent you from accidentally setting

        // the property value at a later date.

        // If you want the ability to set this value during the

        // initialization stage of the property, you can use a

        // Constructor Method.

        public DateTime Birthdate { get; private set; }

        public Person(DateTime birthdate)

        {

            Birthdate = birthdate;

        }

        //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        // This is a Property that is not auto-implemented

        //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        // This syntax of properties allows you to perform some logic

        // on the property value and also exclude either the getter

        // or setter.

        public int Age

        {

            get

            {

                var timeSpan = DateTime.Today - Birthdate;

                var years = timeSpan.Days / 365;

                return years;

            }

        }

    }

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Intermediate\_12\_Properties

{

    public class Person

    {

        //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        // This is an Auto-Implemented property

        //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        // If you create an auto-implementd property, you cannot specify

        // if you want the getter or setter excluded. The getter and setter

        // are created automatically for you.

        // To prevent users from using the setter in a property

        // you can set the property to private. This would act as a

        // safety mechanism to prevent you from accidentally setting

        // the property value at a later date.

        // If you want the ability to set this value during the

        // initialization stage of the property, you can use a

        // Constructor Method.

        public DateTime Birthdate { get; private set; }

        public Person(DateTime birthdate)

        {

            Birthdate = birthdate;

        }

        //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        // This is a Property that is not auto-implemented

        //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

        // This syntax of properties allows you to perform some logic

        // on the property value and also exclude either the getter

        // or setter.

        public int Age

        {

            get

            {

                var timeSpan = DateTime.Today - Birthdate;

                var years = timeSpan.Days / 365;

                return years;

            }

        }

    }

}