# Enumerators/Enums in C#

An enum is a special class that represent a group of constants(read only/ unchangeable) variables.

To create an enum, use the enum keyword and separate the enum items with a comma:

By default, the first item of an enum has the value 0. The second has the value 1, and so on.

To get the integer value from an item, you must [explicitly convert](https://www.w3schools.com/cs/cs_type_casting.asp) the item to an int:

You can also assign your own enum values, and the next items will update the number accordingly:

Enums are lists of constants. When you need a predefined list of values which do represent some kind of numeric or textual data, you should use an enum. You should always use enums when a variable (especially a method parameter) can only take one out of a small set of possible values.

Constants can be any data type but an enum is an enum.

Example

using System;

namespace C\_Basics\_Enums

{

    class Enums

    {

        public enum DayofWeeks

        {

            Sun,

            Mon = 10,

            Tue,

            Wed,

            Thu,

            Fri,

            Sat

        }

    }

    class Program

    {

        static void Main(string[] args)

        {

            Console.WriteLine("Sunday: {0} ", (int)Enums.DayofWeeks.Sun);

            Console.WriteLine("Monday: {0} ",(int)Enums.DayofWeeks.Mon);

            Console.WriteLine("Tuesday: {0} ", (int)Enums.DayofWeeks.Tue);

        }

    }

}

Output

Sunday: 0

Monday: 10

Tuesday: 11