**Módulo: Características del Lenguaje C#**

**Tutorial: Ambito**

**Ejemplo 1**

1. Ambito de Bloques.

//.1 Ambito de Bloque

for (int i = 0; i <= 10; i++)

{

i++; //Dentro del ámbito del bloque

}

i++;// Fuera del ámbito del bloq

1. Otro ejemplo pero con otra variable diferente de

for (int i = 0; i <= 10; i++)

{

int j = 0; //Dentro del ámbito del bloque

j++;

}

j++; // Fuera del ámbito del bloque

1. Otro ejemplo mismo identificador de variables, pero en declarados en distintos bloques

for (int l = 0; l <= 10; l++)

{

l++;

}

if (1 == 1)

{

int l = 9;

l++;

}

1. Ambito de procedimiento (método). Declaración y uso de la variable dentro del método

public void AmbitoMetodo()

{

int j = 0; //declaración en el método

for (int l = 0; l <= 10; l++)

{

j++; // Dentro del Ambito del método

}

j = +j;// Dentro del Ambito del método

}

public void AmbitoMetodoB()

{

j++; // Fuera del ámbito del método

}

1. Declaración de un mismo identificador de variable dentro del método y dentro del bloque:

public void AmbitoMetodoC()

{

int j = 0; //declaración en el método

for (int l = 0; l <= 10; l++)

{

int j = 1; //declaración en el bloque

j++;

}

}

1. Ambito de clase.

class Program

{

int m = 0;

static void Main(string[] args)

{

1. Uso de la variable de clase.

public void AmbitoMetodoD()

{

int j = 0;

for (int l = 0; l <= 10; l++)

{

j++;

j = m; // uso en el bloque

}

j = +j;

m ++; // uso en el metodo

}

1. Ambito de Espacio de nombres. Declarar un clase dentro del mismo espacio de nombres.

class ClaseA

{

public int a = 0;

public string b = "Hola";

}

1. Dentro de otra clase del mismo espacio de nombre se usa la clase definida.

class Program

{

static void Main(string[] args)

{

//Uso de una clase dentro del espacio de nombres

ClaseA objClaseA = new ClaseA();

Console.WriteLine(objClaseA.a);

Console.ReadLine();

}

**Ejemplo 2**

1. Con using es palabra clave que corresponde a una directiva que se utiliza para crear un alias para un espacio de nombres o para importar tipos definidos en otros espacios de nombres.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

1. Uso del tipo de acceso protected. Declaración de una variable dentro de una clase

public class Persona

{

protected string nombre = string.Empty;

}

1. Uso de la variable protegida en otra clase:

class Program

{

static void Main(string[] args)

{

Persona p = new Persona();

p.nombre //no es factible

}

}

1. Creación de una subclase y uso de la variable en esa subclase.

public class cliente : Persona

{

public cliente()

{

this.nombre = "Sin nombre";

}

}