
	<b>GUÍA N° 1</b>		
	<b>Ingeniería Electrónica 2011 - III</b>	<b>Programación Aplicada Página 1 de 4</b>	

## RECONOCIMIENTO DEL ENTORNO DE DESARROLLO .NET.

**OBJETIVO:** Explorar el entorno de desarrollo .Net a través de programas con rutinas de código básicas.

### 1. Características y lenguajes del Entorno de Desarrollo .Net

La plataforma .Net a través de su entorno de desarrollo (Visual Studio .Net) proporciona una serie de oportunidades para desarrollo usando múltiples y variados lenguajes de programación entre los que se destacan: C#. Visual Basic, C++ y J#

El entorno de desarrollo .Net (Visual Studio .Net) ofrece variedad de alternativas para el desarrollo software en cualquiera de los lenguajes antes mencionados, estas alternativas son básicamente:

- ✓ Aplicación de Consola
- ✓ Aplicación para Windows
- ✓ Sitio Web

Los desarrollos trabajados como aplicación de consola conservan la misma estructura tanto en lenguaje como en tipo de datos que las aplicaciones trabajadas en cualquier compilador usado por ejemplo para C++ como Turbo C++, Dev C++, entre otros.

### 2. Ejercicio: Aplicación de Consola CLR Usando C++

a. Cree un proyecto nuevo en Visual Studio .Net; seleccione el lenguaje de programación C++; en plantillas instaladas de visual Studio seleccione Aplicación de Consola CLR.

b. Adicione las librerías de C stdio.h y conio.h;



c. En el bloque del código del main() escriba las líneas de código del programa.

El código a implementar es:

```
#include "stdafx.h"
#include "conio.h"
#include "stdio.h"

using namespace System;

int main(array<System::String ^> ^args)
{
    Console::WriteLine(L"Aplicacion de Consola CLR");
}
```

	<b>GUÍA Nº 1</b>		
	<b>Ingeniería Electrónica 2011 - III</b>	<b>Programación Aplicada Página 2 de 4</b>	

```

int f=1,i,n;
printf("Programa para calcular el factorial de un numero\n\n");
printf("Digite el numero para calcularle el factorial: ");
scanf("%d",&n);
for(i=1;i<=n;i++)
    f*=i;
printf("El factorial es: %d",f);
getchar();

    getch();
return 0;
}

```

d. Ejecute la aplicación haciendo clic en el botón de Debug y observe los resultados.

Las aplicaciones trabajadas bajo este entorno conservan la extensión cpp.

### 3. Aplicación Windows

Las aplicaciones trabajadas bajo este entorno conservan la mayoría de las características y estructura de la programación realizada en una aplicación de consola pero provee un entorno gráfico para el desarrollo de las mismas.



#### 3.1 Ejercicio: Aplicación para Windows Usando Visual Basic

Las aplicaciones trabajadas bajo este entorno conservan la mayoría de las propiedades de los trabajados en las versiones anteriores de visual Basic (3.0, 4.0, 5.0). Sin embargo los elementos (formularios y controles) son tratados como objetos y sus propiedades, métodos y eventos como miembros del objeto. Para acceder a un miembro de un objeto se hace a través del operador punto (.).

a. Cree un proyecto nuevo en Visual Studio .Net; seleccione el lenguaje de programación Visual Basic; en plantillas instaladas de visual Studio seleccione Aplicación para Windows.

b. Sobre el formulario Form1 coloque una etiqueta (label) y en el cuadro de propiedades, en la propiedad text escriba "Digite el número:"; coloque un cuadro de texto (textBox); coloque un boton (button) y en el cuadro de propiedades en la propiedad text Escriba "Calcular"; Finalmente coloque otra etiqueta (label)

c. haga doble clic sobre el botón para acceder al diseñador de código y escriba el siguiente código:

	<b>GUÍA N° 1</b>		
	<b>Ingeniería Electrónica 2011 - III</b>	<b>Programación Aplicada Página 3 de 4</b>	

```

Dim i As Integer
Dim n As Integer
Dim f As Integer
f = 1
n = Val(TextBox1.Text)
For i = 1 To n
    f = f * i
Next

Label2.Text = "El factorial es: " & Format(f)

```

d. Ejecute la aplicación haciendo clic en el botón de Debug y observe los resultados.

### 3.2 Aplicación para Windows Usando C#

Las aplicaciones trabajadas bajo este entono son aplicaciones orientadas a objetos. Todos los elementos (formularios y controles) son clases. Las propiedades o atributos y los métodos son miembros de cada una de sus clases. Además cada elemento posee eventos. Cada clase conserva las características de los objetos como la herencia y el polimorfismo. Para acceder a un miembro de un objeto se hace a través del operador punto (.).

a. Cree un proyecto nuevo en Visual Studio .Net; seleccione el lenguaje de programación C#; en plantillas instaladas de visual Studio seleccione Aplicación para Windows.

b. Sobre el formulario Form1 coloque una etiqueta (label) y en el cuadro de propiedades, en la propiedad text escriba “*Digite el número:*”; coloque un cuadro de texto (textBoxt); coloque un boton (button) y en el cuadro de propiedades en la propiedad text Escriba “*Calcular*”; coloque una etiqueta (label) y en cuadro de propiedades en la propiedad Text escriba “El factorial es: “.



c. haga doble clic sobre el botón para acceder al diseñador de código y escriba el siguiente código:

```

int n, f = 1;
n = int.Parse(textBox1.Text);
for (int i = 1; i <= n; i++)
{
    f *= i;
}
label2.Text = "El Factorial es: " + f.ToString();

```

d. Ejecute la aplicación haciendo clic en el botón de Debug y observe los resultados.

 UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS	<b>GUÍA N° 1</b>		 UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS
	<b>Ingeniería Electrónica 2011 - III</b>	<b>Programación Aplicada Página 4 de 4</b>	

**TALLER:**

Diseñe un programa que calcule la serie Fibonacci en cada uno de los lenguajes vistos en la guía. Se debe pedir al usuario cuantos términos desea generar. Recuerde que cada término de la serie resulta de la suma de los dos anteriores:

1 1 2 3 5 8 13 21 .....