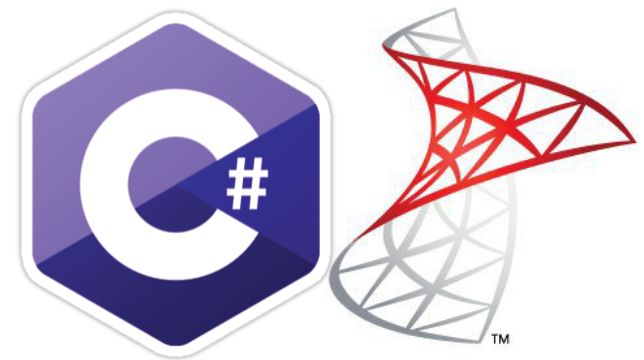


Handbook

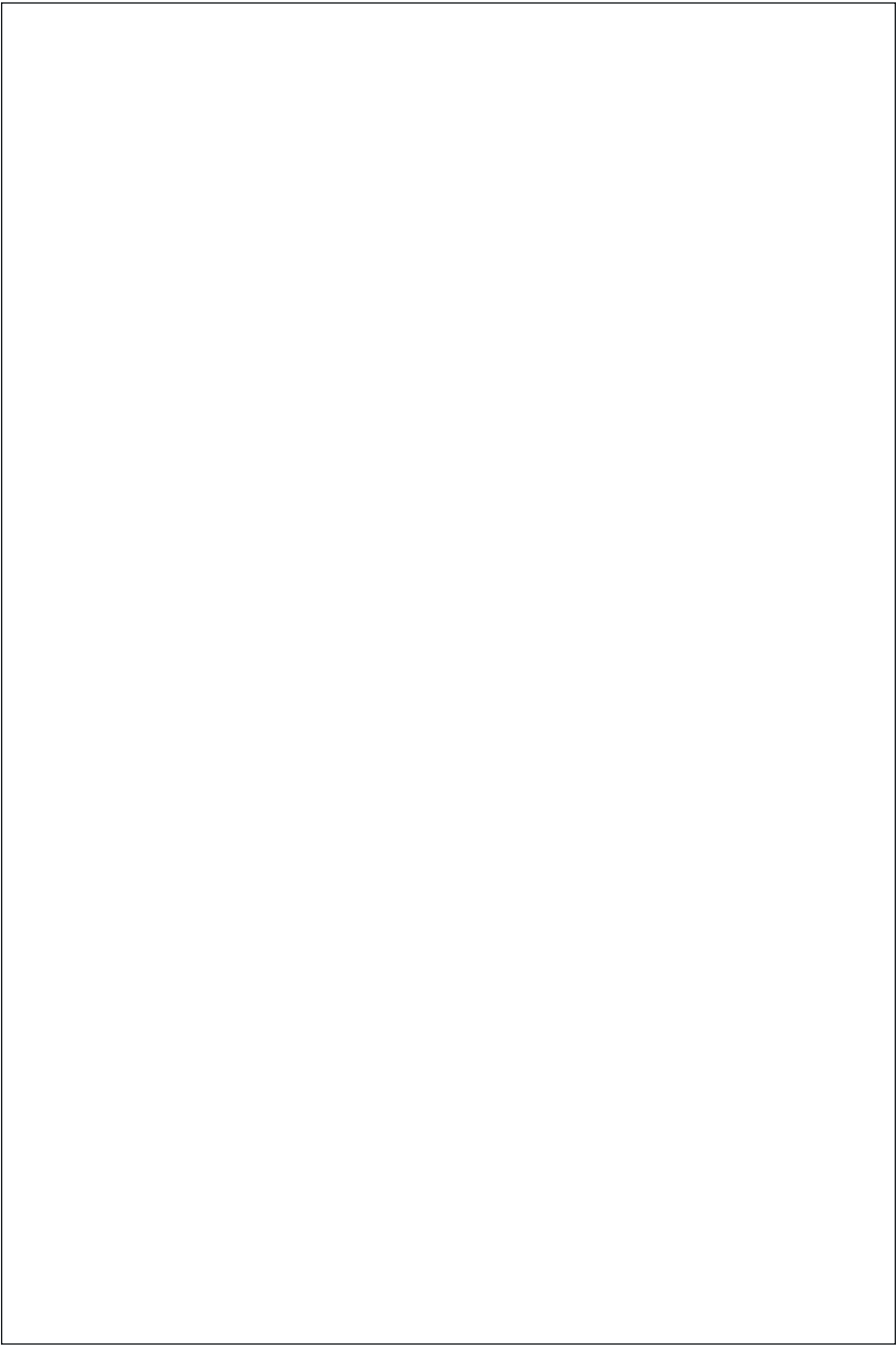
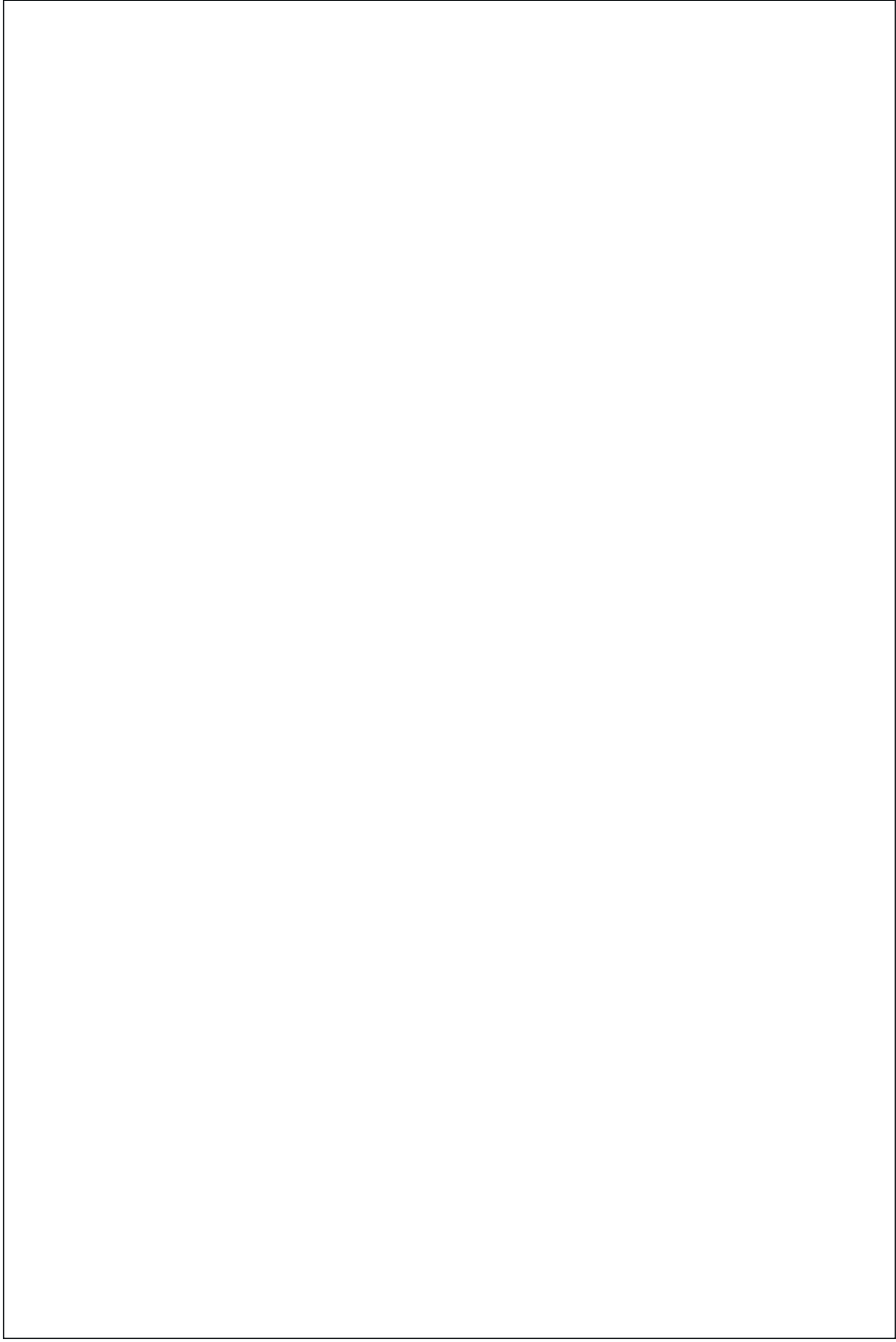
# Visual C# 2015

Marlon Falcón Hernández  
*www.marlonfalcon.cl*



# C#

Marlon Falcón Hernández  
*www.marlonfalcon.cl*



## SQL SERVER :: Procedimiento almacenados ::

```
--Procedimiento mostrar
create proc spmostrarcategoria
as
select top 200 * from categoria
order by idcategoria desc
go

--Procedimiento buscar Nombre
create proc spbuscarcategoria
@textobuscar varchar(50)
as
select * from categoria
where nombre like @textobuscar + '%'
go

--Procedimiento Insertar
create proc spinstartarcategoria
@idcategoria int output,
@nombre varchar(50),
@descripcion varchar(256)
as
insert into categoria (nombre,descripcion)
values (@nombre,@descripcion)
go

--Procedimiento Editar
create proc speditarcategoria
@idcategoria int,
@nombre varchar(50),
@descripcion varchar(256)
as
update categoria set nombre = @nombre,
descripcion = @descripcion
where idcategoria = @idcategoria
go

--Procedimiento eliminar
create proc spelimnarcategoria
@idcategoria int
as
delete from categoria
where idcategoria=@idcategoria
go
```



## Especificaciones del Lenguaje C#

### Tipos de Datos C#

sbyte  
int  
long  
byte  
float  
double  
decimal

System.Diagnostics.Process.Start("mspaint.exe");

### Ejecutar un Archivo Exe

### Salir de la aplicacion

Application.Exit();

### Definicion de clases C#

```
public class Point : Shape
{
    public int x, y;
    public Point(int x, int y) {
        this.x = x;
        this.y = y;
    }
}
```

```
Point p1 = new Point(0, 0);
Point p2 = new Point(10, 20);
```

### Mostrar un formularios C# desde un boton

```
private void button1_Click(object sender, System.EventArgs e)
{
    Form2 frm = new Form2();
    frm.Show();
}
```

### Mostrar unmensaje C#

```
MessageBox.Show("Invalid username/password, please try again");
```

## Especificaciones del Lenguaje C#

### Hola Mundo sumar dos números

```
namespace WindowsFormsApplication4
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            int numero1 = Convert.ToInt32(textBox1.Text);
            int numero2 = Convert.ToInt32(textBox2.Text);
            int resultado = numero1 + numero2;
            textBox3.Text = "El resultado es: " +
            resultado.ToString();
        }
    }
}
```

### Mostrar una página Web en Internet Explorer

```
System.Diagnostics.Process.Start(
"C:\\Program Files\\Internet Explorer\\IExplore.exe",
"www.microsoft.com");
```

### Cambiar el texto de una forma por ejecucion

```
public Form1()
{
    InitializeComponent();
    this.Text = "kaka";
}
```

### Mostrar el nombre del control Activo

```
MessageBox.Show(this.ActiveControl.Name);
```

## SQL-SERVER Basico

```
...
using System.Data.SqlClient;
using System.Data;

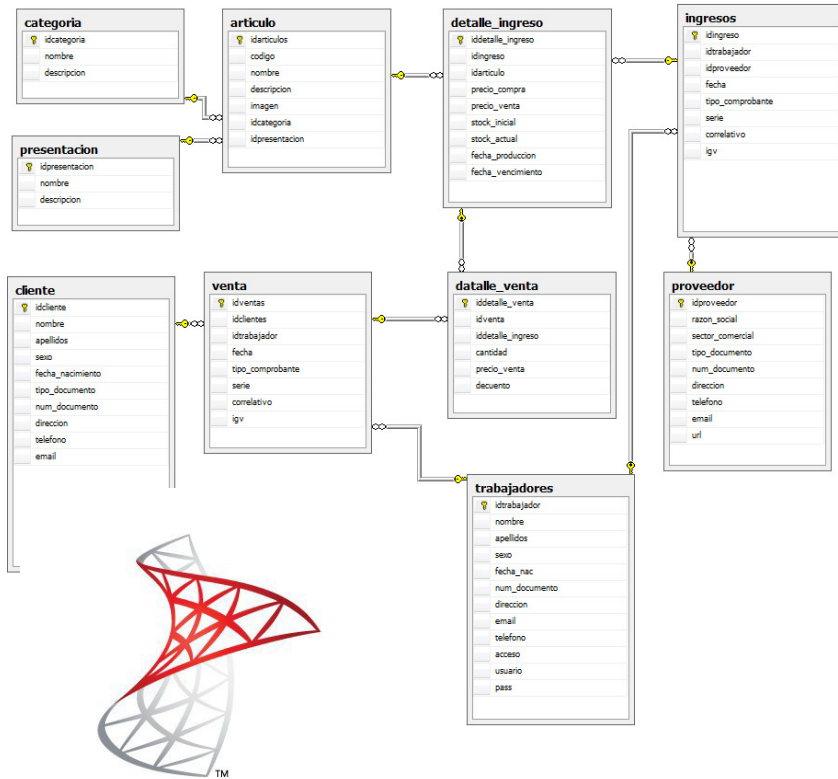
namespace ConexionSQL
{
    class Conexion
    {
        // Crear la conexion
        private string cadena = "data source = PC\\SQLEXPRESS;
initial catalog = PDHM; integrated security = true";
        public SqlConnection con;
        public DataSet ds = new DataSet();
        public SqlDataAdapter da;
        public SqlCommand comando;
        public DataTable dt;

        // Metodo para la consulta a la base de datos
        public DataTable Consultar1(string tabla) {
            con = new SqlConnection(cadena);
            string sql = "SELECT * FROM " + tabla;
            da = new SqlDataAdapter(sql, con);
            da.Fill(ds, tabla);
            dt = new DataTable();
            dt = ds.Tables[tabla];
            return dt;
        }
    }
}
```

```
this.dataGridView1.DataSource = myConexion.Consultar1("alumnos");
```



## SQL-SERVER EJEMPLO



### GridView Mostrar el valor de la celda donde se hace click

```
private void dataGridView1_CellDoubleClick(object sender,
DataGridViewCellEventArgs e)
{
    MessageBox.Show(
        dataGridView1.Rows[e.RowIndex].Cells[e.ColumnIndex].Value.ToString());
}
```

## Todo Consola

```
Console.WriteLine("This line will appear in the console
window");

namespace MiAplicacion
{
    class Personas
    {
        public string nombre = "Alex";
        public int edad = 24;
        public int[] notas = { 5, 4, 5 };
    }

    class Program
    {
        static void Main(string[] args)
        {
            Personas Miper = new Personas();
            Console.WriteLine(Miper.nombre);
            Thread.Sleep(2000);
            Console.Write("Entre la edad:");
            int age = Convert.ToInt32(Console.ReadLine());
            Console.Write("Entre el nombre:");
            string Name = Console.ReadLine();
            Console.WriteLine("Hello {0} you H {1} Years Old", Name, age);
            Console.ReadKey();
        }
    }
}
```

### Agregar elementos a un control ListBox

```
private void textBox_Leave(object sender, System.EventArgs e)
{
    listBox1.Items.Add(((TextBox)sender).Text);
}
```

## DATE

```
DateTime oldDate = new DateTime(2002,7,15);
DateTime newDate = DateTime.Now;

// Difference in days, hours, and minutes.
TimeSpan ts = newDate - oldDate;
// Difference in days.
int differenceInDays = ts.Days;

Console.WriteLine("Difference in days: {0} ", differenceInDays);
```

### Mostrar un mensaje al dar click a un boton

```
protected void btnIngresar_Click(object sender, EventArgs e)
{
    Page.ClientScript.RegisterStartupScript(GetType(), "Mi script",
    "alert('Hello')", true);
    Response.Redirect("/Inicio/");
}
```

### Formulario

```
<form runat="server" class="form-signin">
<asp:TextBox ID="txtUsuario" runat="server" type="text"
class="form-control" placeholder="Correo" required
autofocus></asp:TextBox>

<asp:TextBox ID="txtPass" runat="server" type="password"
class="form-control" placeholder="Contraseña"
required></asp:TextBox>

<asp:Button ID="btnIngresar" runat="server" class="btn btn-lg
btn-primary btn-block" type="submit" Text="Sign in"
OnClick="btnIngresar_Click"/>

<label class="checkbox pull-left">
    <input type="checkbox" value="remember-me">
    Remember me
</label>
<a href="#" class="pull-right need-help">Need help? </a><span
class="clearfix"></span>
</form>
```

## Mostrar cuadros de mensaje en C#

### Mostrar mensajes Informativo

```
MessageBox.Show("Mensaje informativo", "No
registrado", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
```

### Mostrar mensajes Error

```
MessageBox.Show("Mensaje informativo", "Message de
error", MessageBoxButtons.OK, MessageBoxIcon.Error);
```

### Mostrar mensajes Error

```
MessageBox.Show("intereaccion a continuar", "Interfaz de
Question", MessageBoxButtons.OKCancel);
```

### Mostrar mensajes con pedido de dato

```
string entrada = Interaction.InputBox("Ingrese Datos");
```

### Mostrar una página Web en Internet Explorer

```
TreeNode node = treeView1.SelectedNode;
node.Nodes.Add("New node below selected node");
TreeNode node = treeView1.Nodes.Add("Level one node");
node.Nodes.Add("Level two node");
```

## POO CONSOLA C# CLASES

```
namespace ejercicio01
{
    class Hombre
    {
        private string nombre;
        private float dinero;
        public void Inicio()
        {
            Console.WriteLine("-----BANCO
COMOPROGRAMAR.ORG-----");
            Console.WriteLine("-----");
            Console.WriteLine("INGRESE SU NOMBRE: ");
            nombre = Console.ReadLine();
            Console.WriteLine("INGRESE SU MONTO DINERO: ");
            dinero = float.Parse(Console.ReadLine());
            Console.Clear();
        }
        public void Imprimir()
        {
            Console.WriteLine("-----BANCO
COMOPROGRAMAR.ORG-----");
            Console.WriteLine("-----");
            Console.WriteLine("NOMBRE: " + nombre);
            if (dinero > 7000)
            {
                Console.WriteLine("USTED NO PUEDE EXCEDER EL
MONTO DE S/ 7000");
            }
            else
            {
                Console.WriteLine("INVERSION: " + dinero);
            }
            Console.ReadLine();
        }
    }
    static void Main(string[] args)
    {
        Hombre per = new Hombre();
        per.Inicio();
        per.Imprimir();
    }
}
```

### POO CONSOLA

## Sentencias

### Sentencia IF ELSE

```
static void Main(string[] args) {
    if (args.Length == 0) {
        Console.WriteLine("No arguments");
    }
    else {
        Console.WriteLine("One or more arguments");
    }
}
```

### Sentencia switch

```
static void Main(string[] args) {
    int n = args.Length;
    switch (n) {
        case 0:
            Console.WriteLine("No arguments");
            break;
        case 1:
            Console.WriteLine("One argument");
            break;
        default:
            Console.WriteLine("{0} arguments", n);
            break;
    }
}
```

### Sentencia While

```
static void Main(string[] args) {
    int i = 0;
    while (i < args.Length) {
        Console.WriteLine(args[i]);
        i++;
    }
}
```

## Sentencias

### Sentencia DO

```
static void Main() {  
    string s;  
    do {  
        s = Console.ReadLine();  
        if (s != null) Console.WriteLine(s);  
    } while (s != null);  
}
```

### Sentencia FOR

```
static void Main(string[] args) {  
    for (int i = 0; i < args.Length; i++) {  
        Console.WriteLine(args[i]);  
    }  
}  
  
// Sentencia interrumpida  
for (int counter = 1; counter <= 1000; counter++) {  
    if (counter == 10)  
        break;  
    Console.WriteLine(counter);  
}
```

### Sentencia foreach

```
static void Main(string[] args) {  
    foreach (string s in args) {  
        Console.WriteLine(s);  
    }  
}
```

## Ficheros de textos

### Leer Archivo de Texto

```
int counter = 0;  
string line;  
// Read the file and display it line by line.  
System.IO.StreamReader file = new  
System.IO.StreamReader("c:\\test.txt");  
while((line = file.ReadLine()) != null)  
{  
    Console.WriteLine (line);  
    counter++;  
}  
  
file.Close();  
// Suspend the screen.  
Console.ReadLine();
```

### Escribir en un archivo de texto

```
// Compose a string that consists of three lines.  
string lines = "First line.\r\nSecond line.\r\nThird line.";   
// Write the string to a file.  
System.IO.StreamWriter file = new  
System.IO.StreamWriter("c:\\test.txt");  
file.WriteLine(lines);  
file.Close();
```



## Minimizar en la barra de tarea

```
namespace WindowsFormsApplication12
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            private void minimizarToolStripMenuItem_Click(object sender, EventArgs e)
            {
                this.Hide();
                notifyIcon1.Visible = true;
                notifyIcon1.BalloonTipText = "y Texto \n\nsugerencias\n";
                notifyIcon1.BalloonTipTitle = "Aquí va el título:";
                notifyIcon1.BalloonTipIcon = ToolTipIcon.Info;
                notifyIcon1.ShowBalloonTip(5000);
            }

            private void mostrarToolStripMenuItem_Click(object sender, EventArgs e)
            {
                this.Visible = !this.Visible;
            }

            private void salirToolStripMenuItem_Click(object sender, EventArgs e)
            {
                this.Dispose();
            }

            private void Form1_Load(object sender, EventArgs e)
            {
            }
        }
    }
}
```

## C#

### Terminar proceso C#

```
{
try
{
    System.Diagnostics.Process p = new System.Diagnostics.Process();
    p.StartInfo.FileName = "tskill.exe";
    p.StartInfo.UseShellExecute = false;
    p.StartInfo.Arguments = textNombreDelProceso.Text;
    p.StartInfo.RedirectStandardInput = true;
    p.StartInfo.RedirectStandardOutput = true;
    p.StartInfo.RedirectStandardError = true;
    p.StartInfo.CreateNoWindow = true;
    p.Start();
    p.WaitForExit();

    System.Console.WriteLine(p.StandardOutput.ReadToEnd());
    System.Console.WriteLine(p.StandardError.ReadToEnd());
    p.Close();

}
catch (Exception ex)
{
    MessageBox.Show(ex.Message);
}
}
```

### Insertar Dato Data Grid

```
int rowEscribir = dataGridView1.Rows.Count - 1;

dataGridView1.Rows.Add(1);
dataGridView1.Rows[rowEscribir].Cells[0].Value = Valor1;
dataGridView1.Rows[rowEscribir].Cells[1].Value = Valor2;
dataGridView1.Rows[rowEscribir].Cells[2].Value = Valor3;
dataGridView1.Rows[rowEscribir].Cells[4].Value = Valor4;
```

## Cnectarse a Mysql dede C#

```
using MySql.Data.MySqlClient;

class BDConnect
{
    private string server;
    private string database;
    private string uid;
    private string password;

    public BDConnect( string server, string database, string
uid , string password) {
        this.server = server;
        this.database = database;
        this.uid = uid;
        this.password = password;
        string connectionString;
        connectionString = "SERVER=" + server + ";" + "DATABASE=" +
database + ";" + "UID=" + uid + ";" + "PASSWORD=" + password +
"";

        MySqlConnection connection = new
MySqlConnection(connectionString);
        try {
            connection.Open();
            MessageBox.Show("OK");
        }
        catch (MySqlException ex)
        {
            switch (ex.Number)
            {
                case 0:
                    MessageBox.Show("Cannot connect to
server. Contact administrator");
                    break;
                case 1045:
                    MessageBox.Show("Invalid
username/password, please try again");
                    break;
            }
        }
    }
}
```

## foreach

```
using System;

class ForEachTest {

    public static void Main() {
        int [] array = {11,12,13,14,15};
        string testStr = "abc123";
        for (int i = 0; i < array.Length; i++)
            Console.WriteLine(array[i]);
        foreach (int i in array)
            Console.WriteLine(i);
        foreach (char c in testStr)
            Console.WriteLine(c);
        Console.ReadLine();
    }
}
```

## Array

```
using System;
public class Tester {
    static void Main() {
        String[] myArray = {"Who", "is", "John", "Doe"};
        PrintMyArray(myArray);
        Array.Reverse(myArray);
        PrintMyArray(myArray);
        String[] myOtherArray = {"We", "Hold", "These",
"Truths", "To", "Be", "Self", "Evident"};

        PrintMyArray(myOtherArray);
        Array.Sort(myOtherArray);
        PrintMyArray(myOtherArray);
        Console.ReadLine();
    }
    public static void PrintMyArray(object[] theArray) {
        foreach (object obj in theArray)
            Console.WriteLine("Value: {0}", obj);
        Console.WriteLine();
    }
}
```

## Arreglos

```
using System;
class Test
{
    static void Main() {
        int[] a = new int[10];
        for (int i = 0; i < a.Length; i++) {
            a[i] = i * i;
        }
        for (int i = 0; i < a.Length; i++) {
            Console.WriteLine("a[{0}] = {1}", i, a[i]);
        }
    }
}
```

```
int[] notas = { 5, 4, 5 };
```

### Obtener IP de tu maquina

```
using System.Net;
using System.Net.NetworkInformation;

private void button1_Click(object sender, EventArgs e)
{
    IPHostEntry host;
    string localIP = "";
    host = Dns.GetHostEntry(Dns.GetHostName());
    foreach (IPAddress ip in host.AddressList)
    {
        if (ip.AddressFamily.ToString() == "InterNetwork")
        {
            localIP = ip.ToString();
        }
    }
    MessageBox.Show("Tú IP Local Es: " + localIP);
}
```

## Cadena

### Buscar texto dentro de una cadena

```
string searchWithinThis = "ABCDEFGHIIJKLMNOP";
string searchForThis = "DEF";
int firstCharacter = searchWithinThis.IndexOf(searchForThis);

Console.WriteLine("First occurrence: {0}", firstCharacter)
```

### Solo Números en los Textbox C# - Windows Forms

```
private void txtCaracter_KeyPress(object sender, KeyPressEventArgs e)
{
    if (!(char.IsNumber(e.KeyChar)) && (e.KeyChar != (char)Keys.Back))
    {
        MessageBox.Show("Solo se permiten numeros", "Advertencia",
            MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        e.Handled = true;
        return;
    }
}
```

### Redondear en C#

```
Math.Round(3.44, 1) 'Retorna 3.4.
```

### Salir de la Aplicacion C#

```
Application.Exit();
```

## Mostrar un mensaje al dar click a un boton

```
protected void btnIngresar_Click(object sender, EventArgs e)
{
    Page.ClientScript.RegisterStartupScript(GetType(), "Mi script",
    "alert('Hello')", true);
    Response.Redirect("/Inicio/");
}
```

## Formulario

```
<form runat="server" class="form-signin">
<asp:TextBox ID="txtUsuario" runat="server" type="text"
class="form-control" placeholder="Correo" required
autofocus></asp:TextBox>

<asp:TextBox ID="txtPass" runat="server" type="password"
class="form-control" placeholder="Contraseña"
required></asp:TextBox>

<asp:Button ID="btnIngresar" runat="server" class="btn btn-lg
btn-primary btn-block" type="submit" Text="Sign in"
OnClick="btnIngresar_Click"/>

    <label class="checkbox pull-left">
        <input type="checkbox" value="remember-me">
        Remember me
    </label>
    <a href="#" class="pull-right need-help">Need help? </a><span
class="clearfix"></span>
</form>
```

```
namespace Exepciones
{
    class Program
    {
        static void Main(string[] args)
        {
            try
            {
                Console.WriteLine("Entre el numero");
                int numero = int.Parse(Console.ReadLine());
                Console.WriteLine("Entre el valor");
                int valor = int.Parse(Console.ReadLine());
                double result = RaizEnesima(numero, valor);
                Console.WriteLine("El resultado es:"+result);
            }

            catch (FormatException) {Console.WriteLine("Ha
ocurrido un error en la entrada de datos");
            }

            catch (OverflowException)
            {
                Console.WriteLine("Valor demasiado grande para
almacenar en un int");
            }

            catch (Exception ex)
            {
                Console.WriteLine(ex.Message);
            }
            Console.ReadKey();
        }

        public static double RaizEnesima(int numero, int n)
        {
            if (n == 0)
                throw new Exception("No es posible dividir por 0");
            return Math.Pow((double)numero, 1.0 / (double)n);
        }
    }
}
```

## C# y Acces

```

/** Ejecuta un INSERT */
static void insertar(String nombre, String apellido)
{
    OleDbCommand cmd = new OleDbCommand();
    //Representa una instruccion SQL
    cmd.Connection = con;
    //asigna la conexion a la base de datos
    Console.WriteLine("Insertando registros...");
    String query = "INSERT INTO Alumno(nombre,apellido)
VALUES('"+nombre+"','"+apellido+"')";
    cmd.CommandText = @query; //asigna la instruccion SQL
    cmd.ExecuteNonQuery(); //ejecuta una instruccion SQL
    Console.WriteLine("ok");
}

/** Ejecuta un UPDATE
 * Id Llave primaria
 * Set Ej. "Nombre='nuevo valor'"
 */
static void actualizar(int Id, String Set)
{
    OleDbCommand cmd = new OleDbCommand();
    cmd.Connection = con;
    Console.WriteLine("Actualizando...");
    String query = "UPDATE Alumno SET "+Set+" WHERE Id="+Id;
    cmd.CommandText = @query;
    cmd.ExecuteNonQuery();
    Console.WriteLine("ok");
}

/** Elimina un registro dado su ID */
static void eliminar(int Id)
{
    OleDbCommand cmd = new OleDbCommand();
    cmd.Connection = con;
    Console.WriteLine("Eliminando...");
    String query = "DELETE FROM Alumno WHERE Id="+Id;
    cmd.CommandText = @query;
    cmd.ExecuteNonQuery();
    Console.WriteLine("ok");
}
}

```

3

## C# Validacion .. errorProvider1

```

private bool is_validate()
{
    bool no_error = true;

    if (textBox1.Text == string.Empty)
    {
        erroricono.SetError(textBox1, "Ingrese su
Nombre");
        no_error = false;
    }
    else {
        try
        {
            int i = Convert.ToInt32(textBox2.Text);
        }
        catch {
            erroricono.Clear();
            erroricono.SetError(textBox2, "Ingrese la Edad");
            return false;
        }

        try
        {
            double f = Convert.ToDouble(textBox3.Text);
        }
        catch
        {
            erroricono.Clear();
            erroricono.SetError(textBox3, "Ingrese un
Salario Valido");
            return false;
        }
    }
    return no_error;
}

private void button1_Click(object sender, EventArgs e)
{
    if (is_validate()) {
        erroricono.Clear();
        MessageBox.Show("Datos agregados
correctamente", "Validaciones", MessageBoxButtons.OK,
MessageBoxIcon.Information);
    }
}

```

## Balon de informacion

```
namespace WindowsFormsApplication1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
            this.toolTip1.SetToolTip(this.textBox1, "entre Nombre");
            this.toolTip1.SetToolTip(this.textBox2, "entreedad");
        }

        private void Form1_Load(object sender, EventArgs e)
        {
        }
    }
}
```

## C# Acces

```
/** Abre una conexion a la base de datos */
static void conectar()
{
    try
    {
        con = new OleDbConnection(@stringConexion);
        con.Open(); //se abre una conexion
        Console.WriteLine("Conectado a la base de datos.");
    }
    catch (System.Exception ex)
    {
        Console.WriteLine("ERROR:" + ex.Message);
    }
}

/** Cierra la conexion a la base de datos */
static void desconectar()
{
    con.Close();
    Console.WriteLine("La conexion ha terminado.");
}

/** Extrae registros y los muestra en pantalla */
static void imprimirRegistros()
{
    OleDbCommand cmd = new OleDbCommand();
    cmd.Connection = con;
    cmd.CommandText = @"SELECT Id,nombre,apellido FROM Alumno";

    //Ejecuta la instruccion SQL y retorna el resultado
    OleDbDataReader reader = cmd.ExecuteReader();
    Console.WriteLine("-----");

    //va leyendo los registros y mostrando en pantalla
    while (reader.Read())
    {
        Console.WriteLine("{0} | {1} | {2}",
            reader.GetValue(0).ToString(),
            reader.GetValue(1).ToString(),
            reader.GetValue(2).ToString());
    }
    Console.WriteLine("-----");
    reader.Close();
}
```

2

## C# Acces

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.OleDb;

namespace ConAccessOperations
{
    class Program
    {
        static OleDbConnection con; //Representa una conexión abierta a
        un origen de datos
        static String ds = "E:/dbalumno.mdb"; //direccion de la base
        de datos Access
        static String stringConexion =
        "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" + ds;

        static void Main(string[] args)
        {
            conectar();
            //Se imprimir los registros
            imprimirRegistros();
            //se añaden dos nuevos registros
            insertar("Jorge", "Nitales");
            insertar("Elsa", "Pallo");
            imprimirRegistros();
            //Se actualiza el registro con ID=4
            actualizar(4, "apellido='Gordito'");
            imprimirRegistros();
            //Se elimina
            eliminar(2);
            imprimirRegistros();

            desconectar();
            Console.WriteLine("Presione cualquier tecla para terminar");
            Console.ReadKey();
        }
    }
}
```

1

## Ini C#

```
using System.IO;
using System.Reflection;
using System.Runtime.InteropServices;

namespace IniProject
{
    class IniFile // revision 10
    {
        string Path;
        string EXE = Assembly.GetExecutingAssembly().GetName().Name;

        [DllImport("kernel32")]
        static extern long WritePrivateProfileString(string Section,
        string Key, string Value, string FilePath);

        [DllImport("kernel32")]
        static extern int GetPrivateProfileString(string Section, string
        Key, string Default, StringBuilder RetVal, int Size, string FilePath);

        public IniFile(string IniPath = null)
        {
            Path = new FileInfo(IniPath ?? EXE + ".ini").FullName.ToString();
        }
        public string Read(string Key, string Section = null)
        {
            var RetVal = new StringBuilder(255);
            GetPrivateProfileString(Section ?? EXE, Key, "", RetVal,
            255, Path);
            return RetVal.ToString();
        }
        public void Write(string Key, string Value, string Section = null)
        {
            {
                WritePrivateProfileString(Section ?? EXE, Key, Value, Path);
            }
        }
        public void DeleteKey(string Key, string Section = null)
        {
            {
                Write(Key, null, Section ?? EXE);
            }
        }
        public void DeleteSection(string Section = null)
        {
            {
                Write(null, null, Section ?? EXE);
            }
        }

        public bool KeyExists(string Key, string Section = null)
        {
            {
                return Read(Key, Section).Length > 0;
            }
        }
    }
}
```

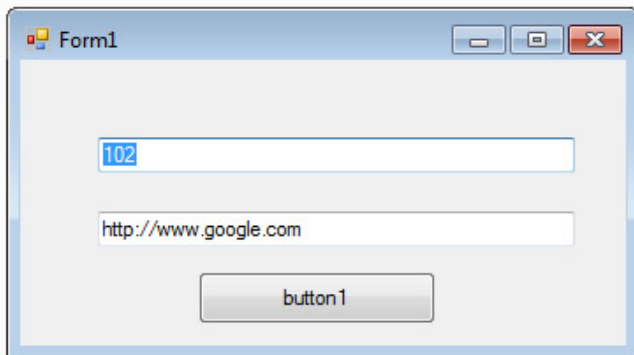
1

## Ini C# Parte 2

```
namespace IniProject
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            IniFile MyIni = new IniFile("Settings.ini");
            private void Form1_Load(object sender, EventArgs e)
            {
                this.textBox1.Text = MyIni.Read("DefaultVolume");
                this.textBox2.Text = MyIni.Read("HomePage");
            }

            private void button1_Click(object sender, EventArgs e)
            {
                MyIni.Write("DefaultVolume", this.textBox1.Text);
                MyIni.Write("HomePage", this.textBox2.Text);
            }
        }
    }
}
```



2

## C# y Acces Basico

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.OleDb; //Para base de datos Access

namespace AccessCon
{
    class Program
    {
        static void Main(string[] args)
        {
            OleDbConnection con; //Representa una conexión abierta
            String ds = "E:/dbproblem.mdb"; //direccion de la base
            de datos Access

            String stringConexion = "Provider=Microsoft.Jet.OLEDB.4.0;Data
            Source="+ds;

            try {
                con = new OleDbConnection(@stringConexion);
                con.Open(); //se abre una conexion
                Console.WriteLine("Conectado a la base de datos.");
                con.Close(); //se cierra la conexion
                Console.WriteLine("La conexion ha terminado.");
            }
            catch (System.Exception ex)
            {
                Console.WriteLine("ERROR:" + ex.Message);
            }
            Console.WriteLine("Presione cualquier tecla para terminar");
            Console.ReadKey();
        }
    }
}
```