## ACESSO AOS DADOS NUMA DATATABLE (EX. COM TABELA TEMPORÁRIA – NÃO LIGADA A UMA B.D.)

### DATA BINDING E BINDING NAVIGATOR

```
Private Sub IniDados()
    'Criar tabela temporária
    DataTable = New DataTable

    DataTable.Columns.Add("ID")
    DataTable.Columns("ID").DataType = GetType(Integer)
    DataTable.Columns("ID").AllowDBNull = False

    DataTable.Columns.Add("Name")
    DataTable.Columns("Name").DataType = GetType(String)
    DataTable.Columns("Name").AllowDBNull = False

    'Criar novas linhas
    'Primeira linha
    Dim NewRow As DataRow = DataTable.NewRow
    NewRow.Item("ID") = 1
    NewRow.Item("Name") = "John"
    DataTable.Rows.Add(NewRow)

    'Segunda linha
    NewRow = DataTable.NewRow
    NewRow.Item("ID") = 2
    NewRow.Item("Name") = "Steve"
    DataTable.Rows.Add(NewRow)

    'Bind controls
    'Textboxes
    TextBoxID.DataBindings.Add(New Binding("text", DataTable, "ID"))
    TextBoxName.DataBindings.Add(New Binding("text", DataTable, "Name"))

    'Navigator
    Dim BS As New BindingSource
    BS.DataSource = DataTable
    BindingNavigator1.BindingSource = BS
```

## *ACESSO AOS REGISTOS DEVOLVIDOS POR QUERY*

```
Connection = New MySqlConnection

Connection.ConnectionString = "Server=localhost;port=3306;userid=root;password=root;database=databasename"

Dim reader As MySqlDataReader

...

Private Sub XXXX...


if IsNumeric(id.text) Then

        cmd.CommandText = "Select * from student where Student_id=@p1"

        cmd.Prepare

        cmd.Parameters.AddWithValue("@p1", id.text)


        Try

         Connection.Open()

         Dim query As String

         query= "Select * from Databasename.tablename where fieldname='" & textbox1.text & "'"

         Command = New MySqlCommand(query, Connection)

         reader = Command.ExecuteReader


         While reader.Read

                Dim sname As String

                sname = reader.GetString("Fieldname")                    ‘GetDateTime, GetDouble, GetGuid, GetInt32,...
```

‘You can access each column of the returned row by passing the name or ordinal reference of the column to the **DataReader**. However, for best performance, the **DataReader** provides a series of methods that allow you to access column values in their native data types.

```
                textbox1.Items.Add(sname)

          End While

          Connection.Close()


         Catch e MySqlException

          MsgBox (ex.Message)


         Finally

          Connection.Dispose

         End Try

Else

         Exit Sub

End If

End Sub
```

-----------------

## *CARREGAR REGISTOS DA BASE DE DADOS NUMA COMBOBOX (Via DataTable):*

```
Private Sub Form5_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

        Dim con As New SqlConnection
        con.ConnectionString = "Data Source=.\SQLEXPRESS;AttachDbFilename=C:\Users\lito\Documents\QMP_DB.mdf;Integrated Security=True;Connect Timeout=30;User Instance=True"
        con.Open()
        Dim da As New SqlDataAdapter("select * from Subject_Info", con)
        Dim ds As New DataSet
        da.Fill(ds, "subjectInfo")
        Dim dt As DataTable
        dt = ds.Tables("Subject_Info")
        Dim dr As DataRow
        Dim i As Integer
        For i = 0 To dt.Rows.Count
            dr = dt.Rows(i)
            cbSubj.Items.Add(dr(1).ToString)
        Next
end sub
```

```
Private Sub Form5_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

         Try
                 Dim con As New SqlConnection
                 Dim objDataAdapter As New SqlDataAdapter()
                 Dim objDataSet As New DataSet()

                 'set database connection
                 con.ConnectionString = "Data Source=.\SQLEXPRESS;AttachDbFilename=C:\Users\lito\Documents\QMP_DB.mdf;Integrated Security=True;Connect Timeout=30;User Instance=True"

```

```
                '//state dataset of combo box
                ' Set the SelectCommand properties...
                objDataAdapter.SelectCommand = New SqlCommand()
                objDataAdapter.SelectCommand.Connection = con
                objDataAdapter.SelectCommand.CommandText = "select * from Subject_Info"
                objDataAdapter.SelectCommand.CommandType = CommandType.Text

                ' Open the database connection...
                con.Open()
                ' Fill the DataSet object with data...
                objDataAdapter.Fill(objDataSet, "subject_info")
                ' Close the database connection...
               con.Close()

                With originComboBox
                    .DataSource = objDataSet
                    .DisplayMember = "subject_info.subjectinfo"
                End With

            Catch ex As Exception

            End Try
```

Idem:

```
Dim dtColours As New DataTable
        Dim daColours As New SqlDataAdapter
        Dim i As Integer

        ConnectToSQL()
        daColours = New SqlDataAdapter("SELECT DISTINCT Rtrim(UPPER(Colour)) As Colour FROM invStockColour WHERE InUse = 1
ORDER BY Colour", dbSQL)
```

```
        daColours.Fill(dtColours)

        For i = 0 To dtColours.Rows.Count - 1
            cboColours.Items.Add(dtColours.Rows(i).Item(0).ToString)
        Next

        dbSQL.Close()
```

Idem:

```
sqlstr = "SELECT * FROM Class WHERE State= Not Started"
DBCmd = New MySql.Data.MySqlClient.MySqlCommand(sqlstr, DBConn)
DBDR = DBCmd.ExecuteReader
While (DBDR.Read())
CB_Class.Items.Add(DBDR("Code"))
End While
DBCmd.Dispose()
DBDR.Close()
```

----------------------------

## *INSERIR IMAGEM A PARTIR DE UMA TABELA:*

**SÓ PATH:**

```
cod = New MySqlCommand("select imagem from produtos where modelo='" + produto + "'", ligacao)
        ligacao.Open()
        valor = cod.ExecuteScalar
        str = CType(valor, String)
        ligacao.Close()
        'str fica com a localização do ficheiro da imagem
        pct1.Image = System.Drawing.Image.FromFile(str)
```

**INSERIR UMA IMAGEM IDENTIFICADA NA DGV:**

```
        Dim produto As String
        Dim cod As MySqlCommand
        Dim valor As Object
        Dim str As String
        produto = dgv1.CurrentCell.Value

        cod = New MySqlCommand("select imagem from produtos where modelo='" + produto + "'", ligacao)
        ligacao.Open()
        valor = cod.ExecuteScalar
        str = CType(valor, String)
        ligacao.Close()
        'str fica com a localização do ficheiro da imagem
        pct1.Image = System.Drawing.Image.FromFile(str)
```

**IMAGEM GUARDADA NA BD:**

Regardless of what data access technology or database you use, you need to convert am `Image` to a `Byte` first and then save that. On retrieval, you convert the `Byte` array back to an `Image`.

To save:

```
Dim connection As New SqlConnection("connection string here")
Dim command As New SqlCommand("UPDATE MyTable SET Picture = @Picture WHERE ID = 1", connection)

'Create an Image object.'
Using picture As Image = Image.FromFile("file path here")
    'Create an empty stream in memory.'
    Using stream As New IO.MemoryStream
        'Fill the stream with the binary data from the Image.'
        picture.Save(stream, Imaging.ImageFormat.Jpeg)

        'Get an array of Bytes from the stream and assign to the parameter.'
        command.Parameters.Add("@Picture", SqlDbType.VarBinary).Value = stream.GetBuffer()
    End Using
End Using

connection.Open()
command.ExecuteNonQuery()
connection.Close()
```

**PARA RECUPERAR (BD para FORM):**

```
Dim connection As New SqlConnection("connection string here")
Dim command As New SqlCommand("SELECT Picture FROM MyTable WHERE ID = 1", connection)

connection.Open()

Dim pictureData As Byte() = DirectCast(command.ExecuteScalar(), Byte())

connection.Close()

Dim picture As Image = Nothing
```

```
'Create a stream in memory containing the bytes that comprise the image.'
Using stream As New IO.MemoryStream(pictureData)
    'Read the stream and create an Image object from the data.'
    picture = Image.FromStream(stream)
End Using
```

That example is for ADO.NET and SQL Server but the principle of using a `MemoryStream` for the conversion is the same regardless.

Idem:

*vb.net Code:*

```
1. Dim cmd As New MySqlCommand("Insert Into " + tables.SelectedItem.ToString() & " (clientid, priority, img)
   values (?uid, ?pri, ?image);", conn)
2.
3. cmd.Parameters.Add("uid", MySql.Data.MySqlClient.MySqlDbType.Int32)
4. cmd.Parameters.Add("pri", MySql.Data.MySqlClient.MySqlDbType.VarChar)
5. cmd.Parameters.Add("image", MySql.Data.MySqlClient.MySqlDbType.Blob)
6.
7. cmd.Parameters("uid").Value = 123
8. cmd.Parameters("pri").Value = "hi"
9.
10.
11.           Dim path As String = "c:\sample.jpg"
12.           Dim sz As Integer
13.           Dim picbytes() As Byte
14.           Try
15.               Using fs As New IO.FileStream(path, FileMode.Open)
16.                   sz = CInt(fs.Length)
17.                   ReDim picbytes(sz - 1)
18.                   fs.Read(picbytes, 0, sz)
19.                   fs.Close()
20.               End Using
21.           Catch ex As Exception
22.               MsgBox(ex.Message)
23.           End Try
24.
25.   cmd.Parameters("image").Value = picbytes
```

## Insert image in mysql db

Lines 12 to 25 could essentially be replaced with:
vb.net Code:

```
cmd.Parameters("image").Value = IO.File.ReadAllBytes(path)
```

Idem:

BLOB into MySQL using VB.NET:

```
Dim conn As New MySqlConnection
Dim cmd As New MySqlCommand

Dim SQL As String

Dim FileSize As UInt32
Dim rawData() As Byte
Dim fs As FileStream

conn.ConnectionString = "server=127.0.0.1;" _
& "uid=root;" _
& "pwd=12345;" _
& "database=test"

Try
fs = New FileStream("c:\image.png", FileMode.Open, FileAccess.Read)
FileSize = fs.Length

rawData = New Byte(FileSize) {}
fs.Read(rawData, 0, FileSize)
fs.Close()

conn.Open()

SQL = "INSERT INTO file VALUES(NULL, ?FileName, ?FileSize, ?File)"

cmd.Connection = conn
cmd.CommandText = SQL
cmd.Parameters.Add("?FileName", strFileName)
cmd.Parameters.Add("?FileSize", FileSize)
cmd.Parameters.Add("?File", rawData)

cmd.ExecuteNonQuery()

MessageBox.Show("File Inserted into database successfully!", _
"Success!", MessageBoxButtons.OK, MessageBoxIcon.Asterisk)

conn.Close()
Catch ex As Exception
MessageBox.Show("There was an error: " & ex.Message, "Error", _
MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try
```

-----------------

```
PictureBox1.Image = Image.FromFile("C:\myFile.???")
```

-------------------

# DATA GRID VIEW

**'COLOCA A INFORMAÇÃO DA TABELA DEFINIDA NUM DATASET NUMA DGV:**

```
da.Fill(ds, "myTable")


' Define que a fonte de dados da DataGridView é a nossa DataSet

' criando automáticamente as colunas e linhas de dados

Me.DataGridView1.DataSource = ds.Tables("myTable")
```

-----------

**LARGURA DAS COLUNAS A DGV:**

```
DataGridView1.Columns.Item("Adress").Width = 60

DataGridView1.Columns.Item("Phone").Width = 30

DataGridView1.Columns.Item("Name").Width = 40

DataGridView1.Columns.Item("Etc.").Width = 30
```

**PROCURAR DADOS:**

```
cod = New MySqlCommand("select count(codp) from produto", ligacao)
        ligacao.Open()
        valor = cod.ExecuteScalar
        cont = CType(valor, Integer)
        ligacao.Close()

        If dgv1.CurrentCellAddress.Y <= cont - 1 Then
            pk = dgv1.Rows(dgv1.CurrentCellAddress.Y).Cells(0).Value
            txtdescricao.Text = dgv1.Rows(dgv1.CurrentCellAddress.Y).Cells(3).Value
            txtreferencia.Text = dgv1.Rows(dgv1.CurrentCellAddress.Y).Cells(4).Value
            txtquantidade.Text = dgv1.Rows(dgv1.CurrentCellAddress.Y).Cells(5).Value
            txtpreco.Text = dgv1.Rows(dgv1.CurrentCellAddress.Y).Cells(6).Value
            Dim index As Integer = lst1.FindString(dgv1.Rows(dgv1.CurrentCellAddress.Y).Cells(2).Value)
            Dim index1 As Integer = lst2.FindString(dgv1.Rows(dgv1.CurrentCellAddress.Y).Cells(1).Value)
            If index = -1 Then
                MessageBox.Show("Item não disponivel na Lst1")
            Else
                lst1.SetSelected(index, True)
            End If
```

**INSERIR LINHAS**

```
DataGridView1.Rows(1).Visible = False
```

-------

```
DataGridView1.ColumnCount = 3

DataGridView1.Columns(0).Name = "Product ID"
DataGridView1.Columns(1).Name = "Product Name"
DataGridView1.Columns(2).Name = "Product_Price"

Dim row As String() = New String() {"1", "Product 1", "1000"}
DataGridView1.Rows.Add(row)
row = New String() {"2", "Product 2", "2000"}
DataGridView1.Rows.Add(row)
row = New String() {"3", "Product 3", "3000"}
DataGridView1.Rows.Add(row)
row = New String() {"4", "Product 4", "4000"}
DataGridView1.Rows.Add(row)

DataGridView1.Rows(1).Visible = False
```

-------

## SELECIONAR

```
DataGridView1.SelectionMode = DataGridViewSelectionMode.FullRowSelect


DataGridView1.Rows.RemoveAt(DataGridView1.SelectedRows(0).Index)
```

## ORDENAR

```
DataGridView1.Sort(DataGridView1.Columns(1), ListSortDirection.Ascending)
```

## FILTRAR

```
Dim dv As DataView
dv = New DataView(ds.Tables(0), "type = 'business' ", "type Desc", DataViewRowState.CurrentRows)
DataGridView1.DataSource = dv
```

## APAGAR LINHAS

‘Deixar o user selecionar uma linha e

'no evento delete key click.

‘É recomendado deixar o user selecionar uma única linha e não um conjunto (myDataGridView.MultiSelect = false)

```
Private Sub pbtnDelete_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnDelete.Click


    If myDataGridView.SelectedRows.Count > 0 Then


              'you may want to add a confirmation message, and if the user confirms delete

           myDataGridView.Rows.Remove(myDataGridView.SelectedRows(0))

    Else
          MessageBox.Show("Select 1 row before you hit Delete")

    End If


   End Sub


‘Atenção: isto não apaga da BD até que se execute o query adequado.


Try

          If Not DataGridView1.CurrentRow.IsNewRow Then

              'Query string

              q = "delete * from Table Name where Column Name='" & DataGridView1.CurrentRow.Cells(0).Value & "'"


              'oledb connection

              cn = New OleDbConnection("microsoft.jet.oledb.4.0;data source=File Path";)

              cn.Open()


              'command button

              cmd = New OleDbCommand(q, cn)

              cmd.ExecuteNonQuery()

              DataGridView1.Rows.Remove(DataGridView1.CurrentRow)

              MessageBox.Show("Record Deleted")

          End If
```

```
Catch ex As Exception


        MessageBox.Show(ex.Message)

End Try
```

-----------

**COMBOBOX NUMA CÉLULA DA DGV:**

```
DataGridView1.Columns.Add("ID", "Codigo")

DataGridView1.Columns.Add("Nome", "Nome Aluno")


'---cria uma coluna do tipo combobox---

Dim comboBoxCol As New DataGridViewComboBoxColumn


'---define o cabeçalho (header) ---

comboBoxCol.HeaderText = "Disciplinas"


'---inclui itens no controlo---

comboBoxCol.Items.Add("Matemática")

comboBoxCol.Items.Add("Português")

comboBoxCol.Items.Add("História")
```

```
'---Incluir a coluna combobox na DataGridView---

DataGridView1.Columns.Add(comboBoxCol)
```