

10. Acessando o Banco de Dados através do Visual Basic

Para acessar o Banco de Dados através do Visual Basic, pode ser utilizado dois métodos diferentes:

ADO - ActiveX Data Objects

DAO - Data Access Objects

10.1 DAO

Fazer a ligação do Banco de Dados e Visual Basic utilizando o método DAO, precisa saber os seguintes passos:

- **Adicionar a referência à DLL DAO 3.6** (Menu Project -> References -> Microsoft DAO 3.6 Object Library)

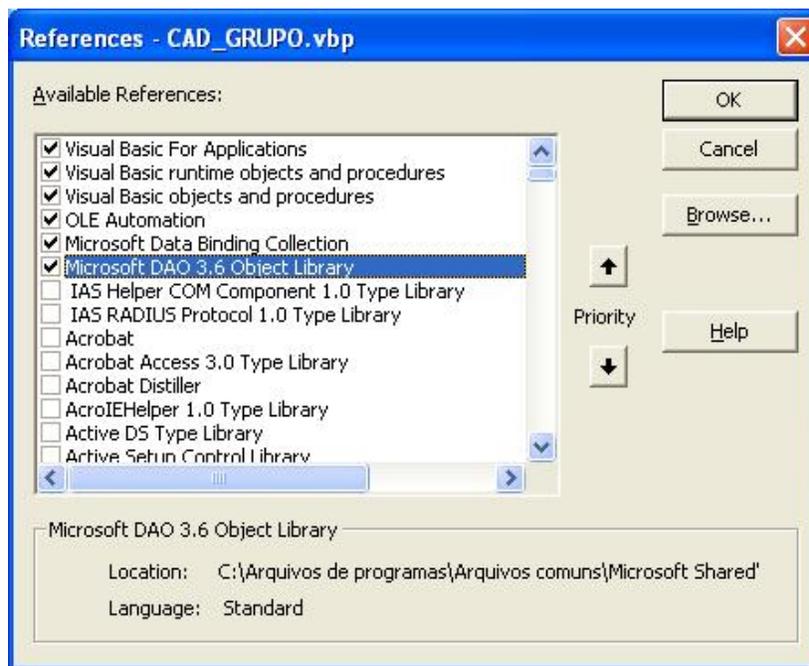


Fig. 1 – Tela de Referencias no Visual Basic

- **Declarar as variáveis**

Public BD As DAO.Database 'Banco de Dados

Public Colaboradores As DAO.Recordset 'Tabela

A variável do tipo Database é o banco de dados e a variável do tipo Recordset é uma tabela ou consulta.

- **Configurar a conexão do banco de dados**

Set bancodedados = areadetrabalho.OpenDatabase (nomedobancodedados, opções, somenteleitura, conexão)

A sintaxe do método OpenDatabase possui as partes a seguir:

Bancodedados - Uma variável de objeto que representa o objeto Database que você deseja abrir.

Áreadetrabalho - Opcional. Uma variável de objeto que representa o objeto Workspace existente que conterá o banco de dados. Se você não incluir um valor para áreadetrabalho, o **OpenDatabase** - usará a área de trabalho padrão.

Nomedobancodedados - Uma String que é o nome de um arquivo do banco de dados Microsoft Jet existente ou o nome da fonte de dados (DSN, Data Source Name) de uma fonte de dados ODBC. Consulte a propriedade Name para obter mais informações sobre a definição desse valor.

Opções - Opcional. Uma Variant que define se o banco será aberto no modo exclusivo "True" ou no modo compartilhado "False".

Somenteleitura - Opcional. Um valor Variant (subtipo Boolean) que será True se você deseja abrir o banco de dados com acesso somente leitura ou False (padrão) se você deseja abrir o banco de dados com acesso leitura/gravação.

Conexão - Opcional. Uma Variant (subtipo String) que especifica várias informações de conexão, incluindo senhas.

Exemplo

```
Set BD = OpenDatabase("C:\Banco.MDB", False, False, ";pwd=aula")
```

O Banco será aberto no modo compartilhado, ou seja, mais do que uma pessoa pode abrir o banco ao mesmo tempo, sendo possível alterar os dados, e possui uma senha para acessá-lo "aula".

- Configurar Tabelas

```
Set conjuntoderegistros = objeto.OpenRecordset (origem, tipo, opções, editarbloqueios)
```

A sintaxe do método OpenRecordset possui as partes a seguir:

Conjuntoderegistros - Uma variável de objeto que representa o objeto Recordset que você deseja abrir.

Objeto - Uma variável de objeto que representa um objeto existente a partir do qual você deseja criar o novo Recordset.

Origem - Uma String que especifica a fonte dos registros para o novo Recordset. A fonte pode ser um nome de tabela, um nome de consulta ou uma instrução SQL que retorne registros. Para objetos Recordset tipo table nos bancos de dados Microsoft Jet, a fonte só pode ser um nome de tabela.

Tipo - Opcional. Uma constante que indica o tipo de Recordset a abrir, como especificado em "Configurações".

Opções - Opcional. Uma combinação de constantes que especifica as características do novo Recordset, como indicado em "Configurações".

Editarbloqueios - Opcional. Uma constante que determina o bloqueio do Recordset, como especificado em "Configurações".

Configurações

Você pode usar uma das constantes a seguir para o argumento tipo:

DbOpenTable Abre um objeto Recordset tipo table (somente áreas de trabalho do Microsoft Jet).

DbOpenDynamic Abre um objeto Recordset tipo dynaset, que é semelhante a um cursor dynamic ODBC (somente áreas de trabalho ODBCDirect).

DbOpenDynaset Abre um objeto Recordset tipo dynaset, que é semelhante a um cursor keyset ODBC.

DbOpenSnapshot Abre um objeto Recordset tipo snapshot, que é semelhante a um cursor static ODBC.

dbOpenForwardOnly Abre um objeto Recordset tipo forward-only.

Exemplo:

```
Set Colaborador = BD.OpenRecordset("colaboradores")
```

A tabela “colaboradores” do objeto Database “BD” será armazenada no objeto Recordset “Colaboradores”, ou seja, a tabela colaboradores será reconhecido pelo VB como o objeto colaboradores.

O Recordset também aceita comandos SQL, como no exemplo abaixo:

```
Set Colaborador = BD.OpenRecordset("select * from colaborador")
```

10.1.1. Algumas propriedades do objeto Database

- OpenRecordset – Abre uma tabela ou consulta
- Close – Fecha o Banco de Dados

10.1.2. Algumas propriedades do objeto Recordset

Index – Utiliza um index para ordenar a tabela. Ex.: Colaborador.Index = "ind_nome"

Seek – Procura alguma palavra pela chave do index. Ex.: Colaborador.Seek "=", "Prog"

MoveFirst – move para o primeiro registro

MoveNext – move para o próximo registro

MoveLast – move para o último registro

MovePrevious –move para o registro anterior

EOF - testa final da tabela

BOF – indica se está no início da tabela

AddNew – adiciona novo registro na tabela

Close – fecha tabela

Exemplo da conexão utilizando DAO ligando com TextBox

```
Dim BD As DAO.Database
Dim Colaborador As DAO.Recordset

Private Sub Form_Load()
    'Configura conexão
    Set BD = OpenDatabase(sUnidade & "\Cadastro Grupo.MDB", False, False,
";pwd=aula")
    Set Colaborador = BD.OpenRecordset("colaboradores")
End Sub

Private Sub CMD_PROXIMO_Click()
    Colaborador.MoveNext
    If Colaborador.EOF = False Then
        Prepara_Consulta
    Else: MsgBox ("Último Registro"): Colaborador.MoveLast
    End If
End Sub

Private Sub CMD_ANTERIOR_Click()
    Colaborador.MovePrevious
    If Colaborador.EOF = False Then
        Prepara_Consulta
    Else: MsgBox ("Primeiro Registro"): Colaborador.MoveFirst
    End If
End Sub

Sub Prepara_Consulta()
    TXT_EMP.Text = Colaborador("empresa")
    TXT_NOME.Text = Colaborador("NOME")
    TXT_CPF.Text = Colaborador("CPF")
    TXT_END_COM.Text = Colaborador("ENDERECO_com")
    TXT_BAIRRO_COM.Text = Colaborador("BAIRRO_com")
    TXT_CIDADE_COM.Text = Colaborador("CIDADE_com")
    TXT_UF_COM.Text = Colaborador("UF_com")
    TXT_CEP_COM.Text = Colaborador("Cep_com")
    TXT_FONE_COM.Text = Colaborador("tel_com")
    TXT_EMAIL_COM.Text = Colaborador("email_com")
    TXT_END_RES.Text = Colaborador("ENDERECO_res")
    TXT_BAIRRO_RES.Text = Colaborador("BAIRRO_res")
    TXT_CIDADE_RES.Text = Colaborador("CIDADE_res")
    TXT_UF_RES.Text = Colaborador("UF_res")
    TXT_CEP_RES.Text = Colaborador("Cep_res")
    TXT_FONE_RES.Text = Colaborador("tel_res")
    TXT_EMAIL_RES.Text = Colaborador("email_res")
End Sub
```

```
TXT_OBS.Text = Colaborador("obs")
```

```
End Sub
```

```
Private Sub CMD_GRAVAR_Click()
```

```
Colaborador.AddNew
```

```
Colaborador("empresa") = TXT_EMP.Text
```

```
Colaborador("Nome") = TXT_NOME.Text
```

```
Colaborador("CPF") = TXT_CPF.Text
```

```
Colaborador("endereco_com") = TXT_END_COM.Text
```

```
Colaborador("bairro_com") = TXT_BAIRRO_COM.Text
```

```
Colaborador("cidade_com") = TXT_CIDADE_COM.Text
```

```
Colaborador("uf_com") = TXT_UF_COM.Text
```

```
Colaborador("cep_com") = TXT_CEP_COM.Text
```

```
Colaborador("tel_com") = TXT_FONE_COM.Text
```

```
Colaborador("email_com") = TXT_EMAIL_COM.Text
```

```
Colaborador("endereco_res") = TXT_END_RES.Text
```

```
Colaborador("bairro_res") = TXT_BAIRRO_RES.Text
```

```
Colaborador("cidade_res") = TXT_CIDADE_RES.Text
```

```
Colaborador("uf_res") = TXT_UF_RES.Text
```

```
Colaborador("cep_res") = TXT_CEP_RES.Text
```

```
Colaborador("tel_res") = TXT_FONE_RES.Text
```

```
Colaborador("email_res") = TXT_EMAIL_RES.Text
```

```
Colaborador("obs") = TXT_OBS.Text
```

```
Colaborador.Update
```

```
End sub
```

```
Private Sub CMD_ALTERAR_Click()
```

```
Colaborador.Edit
```

```
Colaborador("empresa") = TXT_EMP.Text
```

```
Colaborador("Nome") = TXT_NOME.Text
```

```
Colaborador("CPF") = TXT_CPF.Text
```

```
Colaborador("endereco_com") = TXT_END_COM.Text
```

```
Colaborador("bairro_com") = TXT_BAIRRO_COM.Text
```

```
Colaborador("cidade_com") = TXT_CIDADE_COM.Text
```

```
Colaborador("uf_com") = TXT_UF_COM.Text
```

```
Colaborador("cep_com") = TXT_CEP_COM.Text
```

```
Colaborador("tel_com") = TXT_FONE_COM.Text
```

```
Colaborador("email_com") = TXT_EMAIL_COM.Text
```

```
Colaborador("endereco_res") = TXT_END_RES.Text
```

```
Colaborador("bairro_res") = TXT_BAIRRO_RES.Text
```

```
Colaborador("cidade_res") = TXT_CIDADE_RES.Text
```

```
Colaborador("uf_res") = TXT_UF_RES.Text
```

```
Colaborador("cep_res") = TXT_CEP_RES.Text
```

```
Colaborador("tel_res") = TXT_FONE_RES.Text
```

```
Colaborador("email_res") = TXT_EMAIL_RES.Text
```

```
Colaborador("obs") = TXT_OBS.Text  
Colaborador.Update  
End sub
```

10.2 ADO

Fazer a ligação do Banco de Dados e Visual Basic utilizando o método ADO, precisa saber os seguintes passos:

- **Adicionar a referência para a DLL** (Menu Project -> References -> Microsoft ActiveX Data Objects 2.x Library)

- **Declarar as variáveis**

```
Dim BD As ADODB.Connection 'Banco de Dados  
Dim Tabela As ADODB.Recordset 'Tabela
```

A variável do tipo Connection é o banco de dados e a variável do tipo Recordset é uma tabela ou consulta.

- **Configurar a conexão do banco de dados**

```
Set BD = New ADODB.Connection  
BD.Open "Provider=Microsoft.Jet.OLEDB.4.0;" & _  
"Data Source=D:\myDb.mdb;" & _  
"User Id=admin;" & _  
"Password="
```

- **Configurar Tabelas**

```
sSQL = "SELECT * FROM Biblioteca"  
Set Tabela = New ADODB.Recordset  
Tabela.Open sSQL, BD, adOpenStatic, adLockOptimistic
```

As propriedades e linhas de comandos utilizando os objetos ADO são bem parecidas com os objetos DAO.

Bibliografia

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/his/htm/sna_connectionstring_property_oledb.asp – Help MSDN – 06/05/2005

<http://www.macoratti.net/vbadodao.htm> - Sobre ADO x DAO – 06/05/2005

<http://www.codelines.com/index.htm> - Manual Visual Basic – 06/05/2005

10/03/2006 <http://www.fcc.univap.br/~valdir> ou <http://www1.univap.br/~valdirgp>
e-mail: valdirgp@univap.br