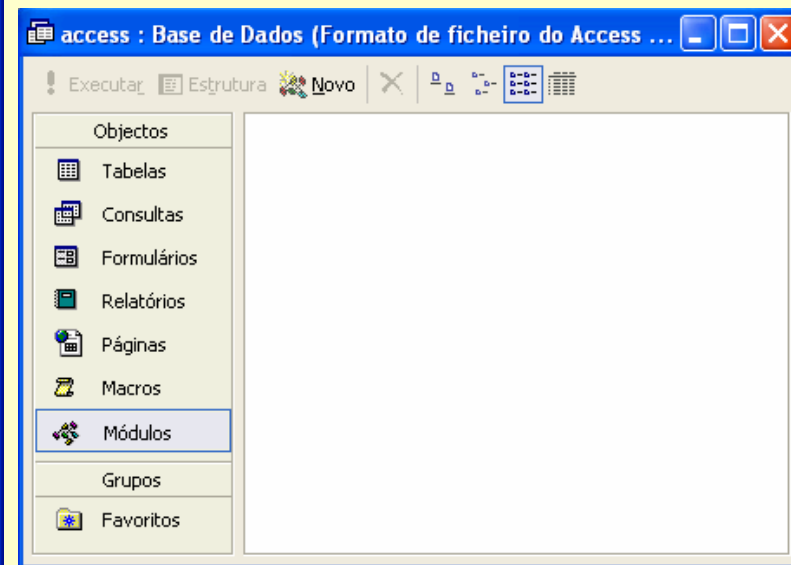
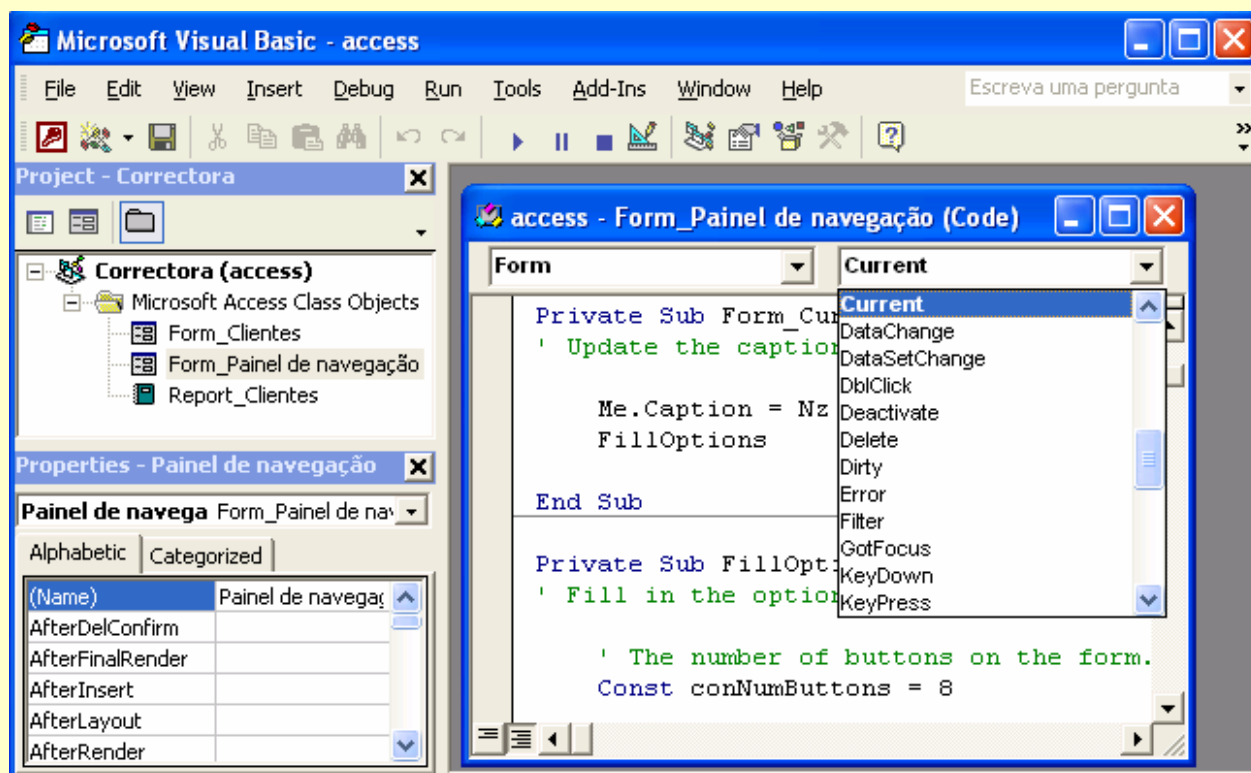


VBA e Access

● **Modo de interação**

- As funcionalidades do VBA estão completamente integradas com o Access
- Um bom exemplo dessa integração é o facto dos formulários e dos módulos fazerem por defeito parte integrante de uma base de dados em Access



Hierarquia de objectos

● **Objectos de nível um**

- [Application.]**Forms**: *collection* dos formulários que se encontram abertos na base de dados
- [Application.]**Reports**: *collection* dos relatórios que se encontram abertos na base de dados
- [Application.]**DataAccessPages**: *collection* das páginas que se encontram abertas na base de dados
- [Application.]**Modules**: *collection* dos módulos que se encontram abertos na base de dados
- [Application.]**DoCmd**: permite utilizar directamente no código o conjunto de acções disponíveis como macros (abrir formulários, imprimir relatórios, ...)
- [Application.]**Screen**: permite referenciar o controle, página, folha de dados, formulário ou relatório que tem o foco num dado momento
 - **Screen.ActiveControl**
 - **Screen.ActiveDataAccessPage**
 - **Screen.ActiveDatasheet**
 - **Screen.ActiveForm**
 - **Screen.ActiveReport**

Objecto DoCmd I

● **Abrir formulários/relatórios**

- `DoCmd.OpenForm FormName,,, WhereCond, DataMode,, Args`: abre o formulário `FormName` em modo `DataMode`, filtra a informação mediante a condição `WhereCond` e instancia a propriedade `OpenArgs` do formulário com o valor indicado em `Args`. `DataMode` pode ser uma das seguintes constantes:
 - `acFormEdit`: permite inserir novos registos e alterar os existentes
 - `acFormAdd`: apenas permite inserir novos registos (os registos existentes são ocultos)
 - `acFormReadOnly`: não permite inserir novos registos nem editar os existentes
 - `acFormPropertySettings`: usa as permissões definidas nas propriedades do formulário
- `DoCmd.OpenReport ReportName,,, WhereCond,, Args`: abre o relatório `ReportName`, filtra a informação mediante a condição `WhereCond` e instancia a propriedade `OpenArgs` do relatório com o valor indicado em `Args`

● **Exemplos**

```
DoCmd.OpenForm "Clientes",,, "Apelido = 'Silva'", acFormEdit  
DoCmd.OpenReport "Clientes",,, "Nome Like 'A*'",, "10"
```

Objecto DoCmd II

● **Fechar formulários/relatórios**

- **DoCmd.Close ObjectType, ObjectName, Save**: fecha o objecto do tipo **ObjectType** com nome **ObjectName** e guarda-o segundo a opção **Save**
 - **ObjectType** pode ser uma das seguintes constantes (entre outras):
 - **acDefault**: para fechar a janela activa
 - **acForm**: para fechar formulários
 - **acReport**: para fechar relatórios
 - **Save** pode ser uma das seguintes constantes:
 - **acSavePrompt**: questiona o utilizador para guardar possíveis alterações
 - **acSaveYes**: guarda possíveis alterações
 - **acSaveNo**: não guarda possíveis alterações

● **Exemplos**

```
DoCmd.Close acForm, "Clientes", acSaveYes  
DoCmd.Close acReport, "Clientes", acSaveNo
```

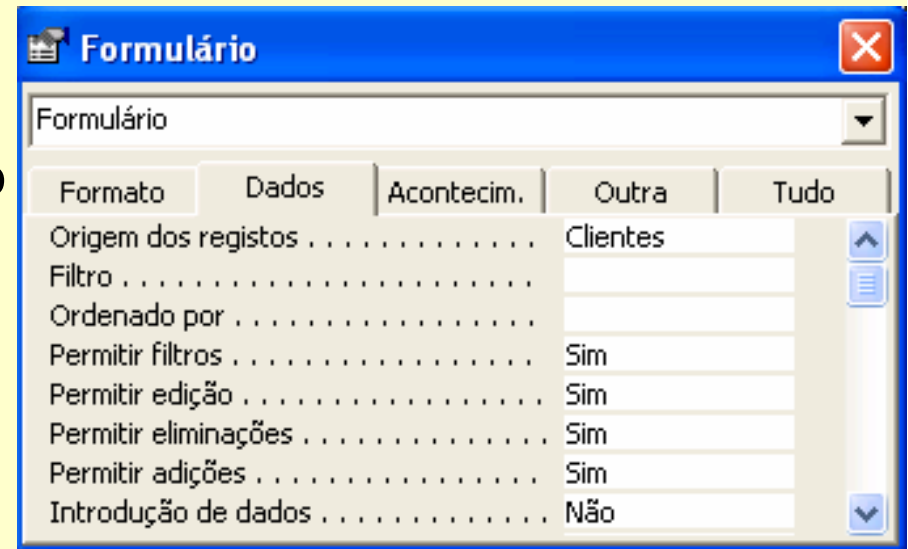
Objectos Form I

● Referenciação

- `Forms("Clientes")`: referência directa ao nome do formulário
- `Forms!Clientes`: referência directa ao nome do formulário
- `Forms![Clientes e Facturas]`: para nomes com espaços
- `Forms(0)`: referência ao primeiro formulário a ser aberto
- `Screen.ActiveForm`: referência ao formulário que tem o foco

● Propriedades

- `form.Name`: nome do formulário
- `form.Caption`: título do formulário
- `form.Visible`: visibilidade do formulário
- `form.Modal`: se `True` requer que o formulário seja fechado para que o foco possa ser movido para outro objecto
- `form.RecordSource`: origem dos registos do formulário (tabela/consulta da base de dados ou expressão SQL)



Objectos Form II

● *Propriedades*

- `form.NavigationButtons`: esconde/apresenta os botões de navegação
- `form.AllowEdits`: se `True` permite editar os registos previamente guardados
- `form.AllowDeletions`: se `True` permite remover registos
- `form.AllowAdditions`: se `True` permite adicionar novos registos
- `form.DataEntry`: se `True` apenas permite adicionar novos registos
- `form.Filter`: filtra os registos a apresentar no formulário
- `form.FilterOn`: activa/desactiva o filtro
- `form.OrderBy`: modo de ordenação dos registos do formulário
- `form.OrderByOn`: activa/desactiva o modo de ordenação

● *Exemplo*

```
Private Sub Form_Load()  
    Me.Filter = "País = 'Portugal'" : Me.FilterOn = True  
    Me.OrderBy = "País, Cidade DESC"  
End Sub  
Private Sub BotãoOrdenação_Click()  
    Me.OrderByOn = Not Me.OrderByOn  
End Sub
```

Objectos Form III

● **Métodos**

- **form.Requery**: executa novamente a consulta na qual o formulário é baseado
- **form.Refresh**: actualiza o conjunto actual de registos do formulário. Os registos que entretanto tenham sido adicionados/removidos (por outro utilizador num ambiente multi-utilizador) não serão incluídos/excluídos do conjunto actual. Tão pouco serão excluídos os registos que já não satisfaçam a consulta associada ao formulário
- **form.Undo**: desfaz todas as alterações feitas ao registo actual desde que foi guardado pela última vez

● **Exemplo**

```
Private Sub Form_BeforeUpdate(Cancel As Integer)
    If ... then
        ...
        Me.Undo
    End If
End Sub
```

Objectos Report I

Referenciação

- `Reports("Clientes")`: referência directa ao nome do relatório
- `Reports!Clientes`: referência directa ao nome do relatório
- `Reports![Clientes e Facturas]`: para nomes com espaços
- `Reports(0)`: referência ao primeiro relatório aberto
- `Screen.ActiveReport`: referência ao relatório que tem o foco

Propriedades

- `report.Name`: nome do relatório
- `report.Caption`: título do relatório
- `report.Visible`: visibilidade do relatório
- `report.RecordSource`: origem dos registos do relatório (tabela/consulta da base de dados ou expressão SQL)
- `report.Filter`: filtra os registos a apresentar no relatório
- `report.FilterOn`: activa/desactiva o filtro
- `report.OrderBy`: modo de ordenação dos registos do relatório
- `report.OrderByOn`: activa/desactiva o modo de ordenação

Objectos Report II

■ **Secções**

- `report.Section(Index)`: secções do relatório. `Index` pode ser uma das seguintes constantes:
 - `acDetail`: secção detalhe
 - `acHeader`: secção cabeçalho
 - `acFooter`: secção rodapé
 - `acPageHeader`: secção cabeçalho de página
 - `acPageFooter`: secção rodapé de página
 - `acGroupLevel1Header`: cabeçalho do nível de agrupamento 1 (só relatórios)
 - `acGroupLevel1Footer`: rodapé do nível de agrupamento 1 (só relatórios)
 - ...
 - `acGroupLevel10Header`: cabeçalho do nível de agrupamento 10 (só relatórios)
 - `acGroupLevel10Footer`: rodapé do nível de agrupamento 10 (só relatórios)

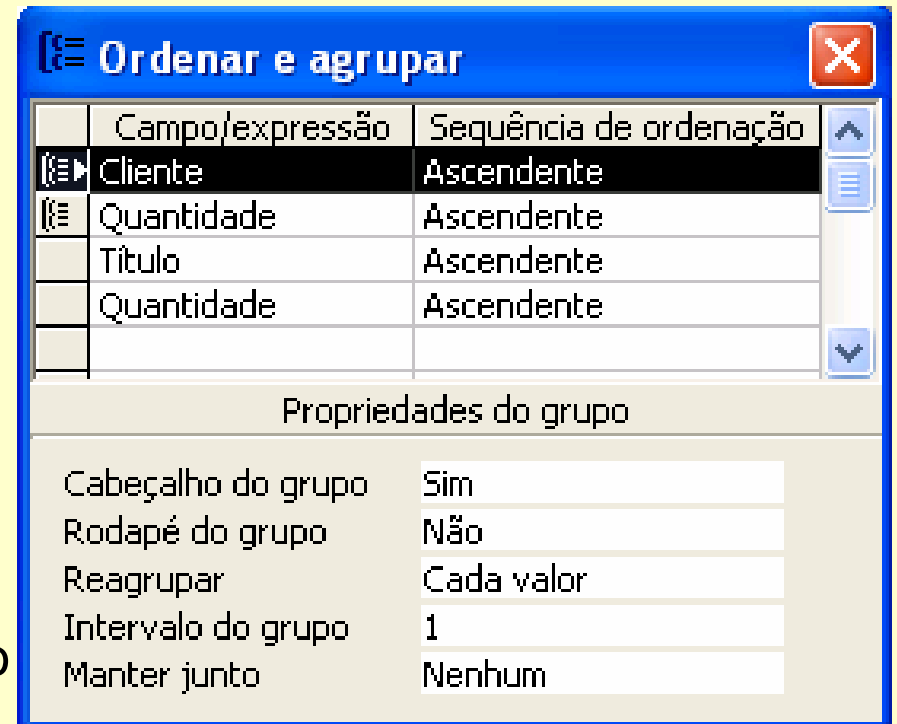
■ **Exemplo**

```
Reports("Alunos").Section(acGroupLevel1Header).Visible = True
```

Objectos Report III

● Grupos

- `report.GroupLevel(Index)`: níveis de agrupamento/ordenação do relatório
- `grouplevel.SortOrder`: modo de ordenação do grupo (**False** corresponde a ordenação ascendente)
- `grouplevel.GroupHeader`: se **True** activa o cabeçalho do grupo
- `grouplevel.GroupFooter`: se **True** activa o rodapé do grupo
- `grouplevel.GroupOn`: especifica como os dados são agrupados
- `grouplevel.KeepTogether`: impõe restrições ao modo de apresentar o cabeçalho, rodapé e secção de detalhe do grupo nas páginas do relatório
- `grouplevel.GroupInterval`: especifica o intervalo para agrupar os dados



Objectos Report IV

● Exemplo

`as propriedades `SortOrder`, `GroupHeader`, `GroupFooter`,
``GroupOn`, `GroupInterval` e `KeepTogether` só podem ser alteradas
`quando ocorre o evento `Open` do relatório em causa

```
Private Sub Report_Open(Cancel As Integer)
```

```
    With Me.GroupLevel(0)
```

```
        .SortOrder = False
```

`ordenação ascendente

```
        .GroupOn = 4
```

`agrupar os registos por ...

```
        .GroupInterval = 6
```

`... semestre (6 meses)

```
        .KeepTogether = 0
```

`sem restrições de agrupamento

```
    End With
```

```
End Sub
```

Objectos Control I

● Referenciação

- [form|report].Controls("Nome"): referência directa ao nome do controlo
- [form|report].Controls!Nome: referência directa ao nome do controlo
- [form|report]!Nome: referência directa ao nome do controlo
- [form|report].Controls(0): referência ao primeiro controlo no contexto da ordem que se encontram no formulário/relatório respectivo
- Screen.ActiveControl: referência ao controlo que tem o foco

● Exemplos

'formas de referenciar o controlo [xpto] do formulário [Novo]

```
Forms("Novo").Controls("xpto")
```

```
Forms("Novo").Controls!xpto
```

```
Forms("Novo")!xpto
```

```
Forms!Novo!xpto
```

```
Forms.Novo.xpto
```

'ao utilizar o símbolo ! não pode seguir-se uma collection

```
Application.Forms!Novo!xpto
```

'ok

```
Application!Forms!Novo!xpto
```

'erro

```
Forms!Novo!Controls!xpto
```

'erro

Objectos Control II

● **Propriedades comuns**

- **ControlSource**: origem dos dados do controlo (campo ou expressão utilizando campos da tabela/consulta em que o formulário é baseado)

● **Combo / list boxes**

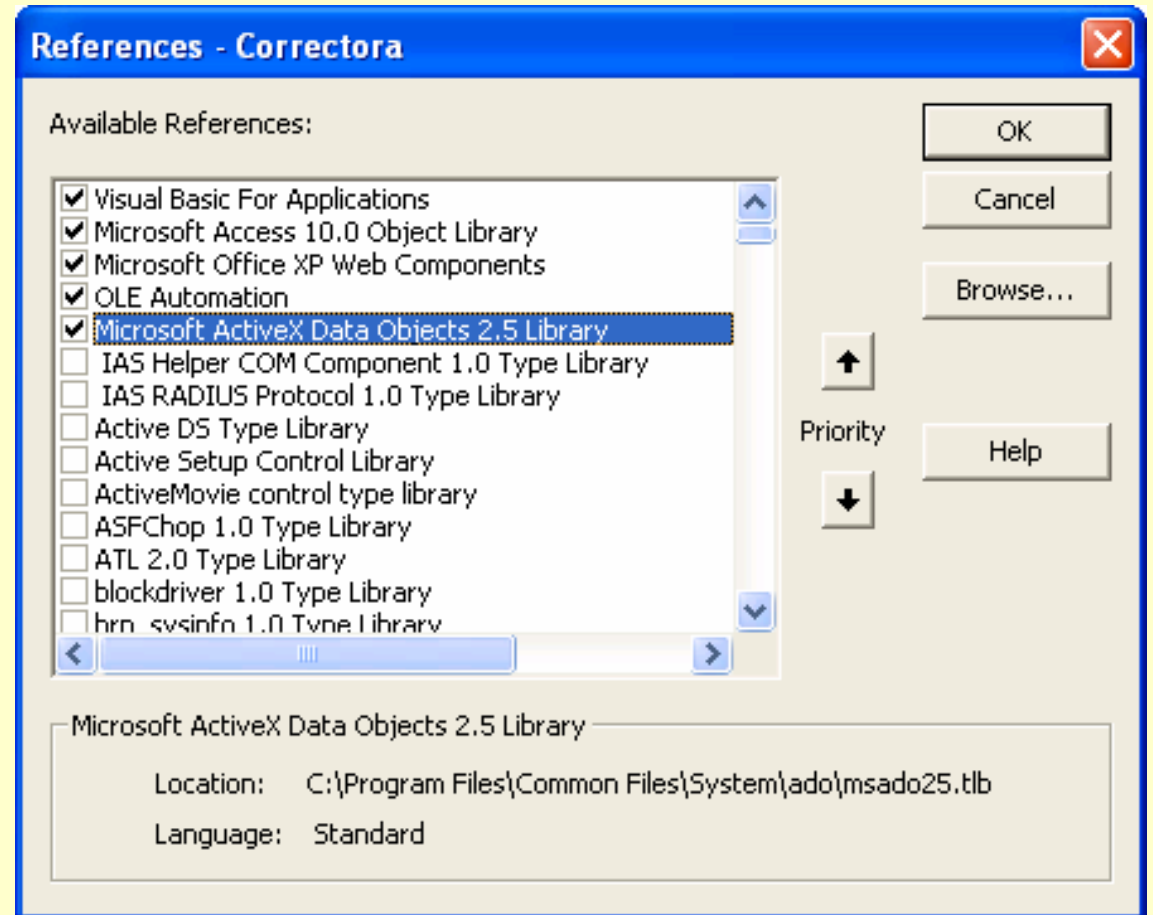
- **RowSource**: origem dos dados do controlo
- **RowSourceType**: tipo da origem dos dados do controlo
 - **Table/Query**: os dados são os registos de uma tabela/consulta ou expressão SQL
 - **Value List**: os dados são uma lista de itens

● **Exemplos**

```
Forms.Clientes.Combo.RowSourceType = "Table/Query"  
Forms.Clientes.Combo.RowSource = "Clientes"  
Forms.Clientes.Combo.ColumnCount = 2      `campos 'Id' e 'Nome'  
Forms.Clientes.Combo.ColumnWidths = "0;1"  
Forms.Clientes.Combo.BoundColumn = "1"  
Forms.Clientes.Lista.RowSourceType = "Value List"  
Forms.Clientes.Lista.RowSource = "1;2;3"  
Forms.Clientes.Lista.AddItem "4"
```

ActiveX Data Objects I

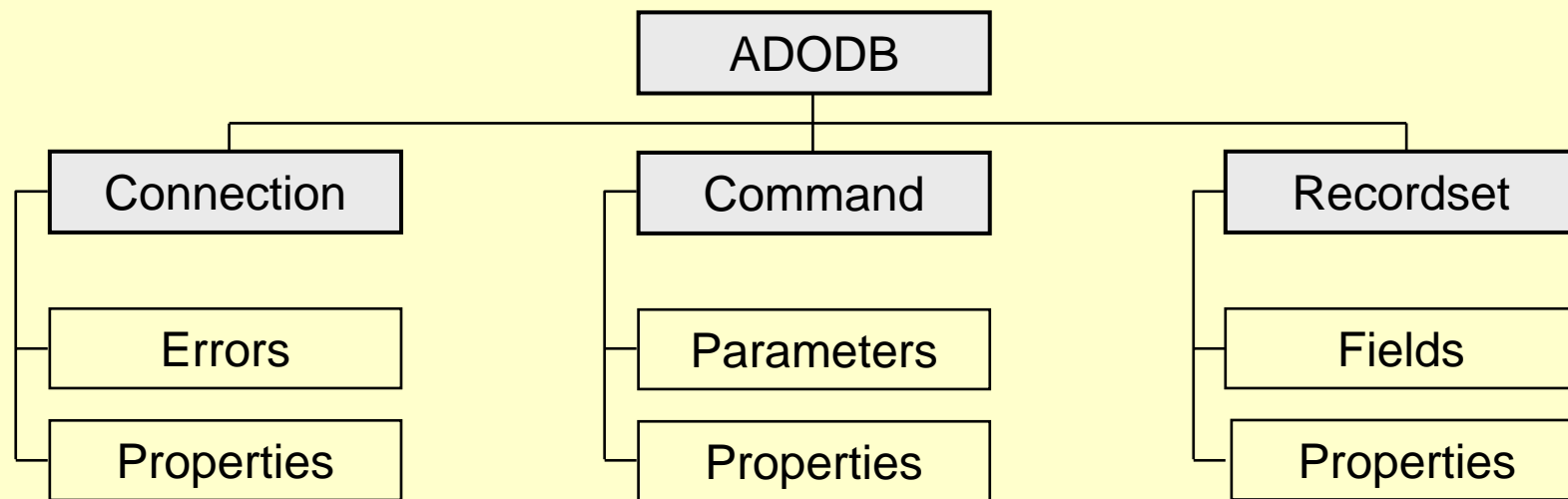
- **Aspectos gerais**
 - A biblioteca ADO oferece uma interface genérica para acesso a dados
 - Permite comunicar com um amplo conjunto de diferentes formatos de dados
 - Define uma única sintaxe para manipular o mesmo tipo de objectos (é independente do formato dos dados)
 - Constitui por si só uma hierarquia de objectos independente da hierarquia de objectos da aplicação principal



ActiveX Data Objects II

■ Hierarquia de objectos

- **ADODB**: objecto de topo da hierarquia ADO
- **Connection**: para definir e estabelecer ligações com diferentes bases de dados
- **Command**: para executar instruções sobre uma dada ligação
- **Recordset**: para manipular conjuntos de registos
- **Errors**: conjunto de erros ocorridos durante a última operação dum ligação
- **Parameters**: conjunto de parâmetros de um dado comando com parâmetros
- **Fields**: conjunto de atributos (campos) de um dado registo
- **Properties**: conjunto de propriedades específicas de um dado objecto



Objectos Connection I

● Métodos

- `connection.Open ConnectionString`: estabelece uma nova ligação
- `connection.Execute(CommandText,,Options)`: para executar comandos
 - `adCmdTable`: indica que `CommandText` refere o nome de uma tabela ou consulta
 - `adCmdText`: indica que `CommandText` refere uma expressão SQL
- `connection.Close`: fecha a ligação

● Exemplos

```
`estabelecer ligação explícita a uma base de dados do Access
Dim con As New Connection      `a declaração New cria desde logo
                                `uma nova instância do objecto
con.Open "Provider=Microsoft.Jet.OLEDB.4.0;
         Data Source=teste.mdb"

con.Close
`ligação corrente ao Access
Dim con As Connection          `se vou usar Set posso omitir New
Set con = CurrentProject.Connection
con.Close
```


Objectos Connection II

■ Exemplos

`criar recordset a partir duma ligação explícita (3 exemplos)

```
Dim con As Connection : Dim rst As Recordset
Set con = ...
Set rst = con.Execute("Clientes", , adCmdTable)
Set rst = con.Execute("[Query sem Parâmetros]", , adCmdTable)
Set rst = con.Execute("SELECT * FROM Clientes", , adCmdText)
rst.Close
con.Close
```

`criar recordset a partir da ligação corrente ao Access

```
Dim rst As New Recordset
rst.Open "Clientes", CurrentProject.Connection
rst.Close
```

`criar recordset a partir duma ligação implícita

```
Dim rst As New Recordset
rst.Open "Clientes", "Provider=Microsoft.Jet.OLEDB.4.0;
                    Data Source=teste.mdb"
rst.Close
```

Objectos Command

● **Propriedades e métodos**

- `command.ActiveConnection`: ligação associada ao comando
- `command.CommandText`: texto do comando
- `command.CommandType`: tipo de comando (`adCmdTable` / `adCmdText`)
- `command.Parameters`: parâmetros do comando
- `command.Execute`: para executar o comando

● **Exemplos**

`Dim com As New Command`

``comando sem parâmetros`

`com.ActiveConnection = CurrentProject.Connection`

`com.CommandText = "SELECT * FROM Clientes"`

`com.CommandType = adCmdText : Set rst = com.Execute`

``comando com parâmetros`

`com.ActiveConnection = CurrentProject.Connection`

`com.CommandText = "[Query com Parâmetros]"`

`com.CommandType = adCmdTable`

`com.Parameters.Refresh` ``lê os parâmetros`

`com.Parameters("[Limite?]") = 100 : Set rst = com.Execute`