

Objetos de Acceso a datos desde Visual Basic

Visual Basic 5.0

-DAO (Data Access Object): Utilizado para acceder a una Base de datos local, generalmente del tipo mdb.

-Funciones API de bajo nivel que permiten hacer uso de ODBC para acceder a bases de datos remotas ubicadas en un servidor.

-RDO (Remote Data Objects): Conjunto de objetos que permiten acceder a una base de datos remota a través de ODBC.

USANDO DAO:

Establecer referencia a DAO (Menú Proyecto → Referencias y tildar Microsoft DAO 3.6 Object Library)

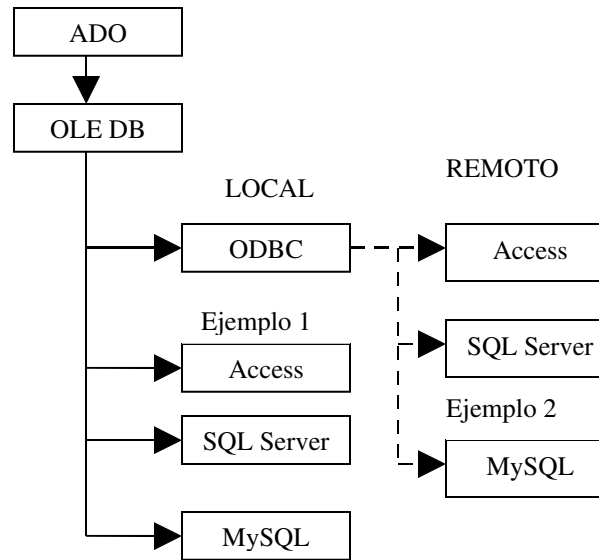
```
Dim Base As DAO.Database
Dim RstAlumnos As DAO.RecordSet
```

```
Private Sub Form_Load()
    Set Base = OpenDatabase("D:\Escuela.mdb")
    Set RstAlumnos=base.OpenRecordset("Alumnos")
    RstAlumnos.MoveFirst
    List1.Clear
    Do While Not RstAlumnos.EOF
        List1.AddItem RstAlumnos!nombre
        RstAlumnos.MoveNext
    Loop
End Sub
```

Visual Basic 6.0

-DAO para mantener compatibilidad hacia atrás.

-ADO (ActiveX Data Object): Conjunto de objetos que a través de un proveedor OLE DB pueden conectarse con bases de datos de cualquier tipo (de forma directa o mediante un origen de datos ODBC).



USANDO ADO:

Establecer referencia a ADO (Menú Proyecto → Referencias y tildar Microsoft ActiveX Data Object 2.0 Library)

En este primer ejemplo se accede a una **base de datos local de Access** (ver esquema a la izquierda) mediante ADO.

Ejemplo 1:

```
Dim Conexion As ADODB.Connection
Dim RstAlumnos As ADODB.Recordset
```

```
Private Sub Form_Load()
    Set Conexion = New ADODB.Connection
    Conexion.ConnectionString="Provider=" & _
    "Microsoft.Jet.OLEDB.4.0" & _
    ";Data Source=D:\Escuela.mdb"
    Conexion.Open
    Set RstAlumnos=Conexion.Execute("SELECT " & _
    "*" from Alumnos;")
    RstAlumnos.MoveFirst
    List1.Clear
    Do While Not RstAlumnos.EOF
        List1.AddItem RstAlumnos!Nombre
        RstAlumnos.MoveNext
    Loop
End Sub
```

En el siguiente ejemplo se accede a una **base de datos remota** que reside en el servidor de base de datos **MySQL**.

Ejemplo 2:

```
Dim Conexion As ADODB.Connection
Dim RstAlumnos As ADODB.Recordset
```

```
Private Sub Form_Load()
    Set Conexion = New ADODB.Connection
    Conexion.ConnectionString = "DRIVER=" & _
    "{MySQL ODBC 3.51 Driver};" & _
    "SERVER=192.168.1.10;" & _
    "DATABASE=medial1;" & _
    "UID=root;" & _
    "PWD=clave;" & _
    "OPTION=3"
    Conexion.Open
    Set RstAlumnos=Conexion.Execute("SELECT " & _
    "*" * from alumnos;")
    RstAlumnos.MoveFirst
    List1.Clear
    Do While Not RstAlumnos.EOF
        List1.AddItem RstAlumnos!nombre
        RstAlumnos.MoveNext
    Loop
End Sub
```

Se utilizó MySQL Server 5.0 y el driver MySQL ODBC 3.51

Visual Basic.NET

-ADO para mantener compatibilidad hacia atrás.

-ADO.NET: Se puede utilizar los drivers OLEDB o hacer uso de componentes propios del Framework para acceder de forma nativa a SQL Server.

```
Public Conexion As New
Data.OleDb.OleDbConnection()
Conexion.ConnectionString =
"Provider=Microsoft.Jet.OLEDB.4.0;" & _
"Data Source=" & Application.StartupPath & _
"\medial1.mdb;Jet OLEDB:Database " & _
"Password='clave'"
Dim Comando As New OleDb.OleDbCommand()
Dim Resul As OleDb.OleDbDataReader
Conexion.Open()
Comando.CommandText = "SELECT * FROM Alumnos;"
Comando.CommandType = CommandType.Text
Comando.Connection = Conexion
List1.Items.Clear()
Resul = Comando.ExecuteReader()
Do While Resul.Read()
    List1.Items.Add(Resul.Item(0))
Loop
Resul.Close()
Conexion.Close()
```