|  |  |
| --- | --- |
|  | D:\Dokumen Mocher\desktop\logo UMB.jpg |
|  | **MODUL PERKULIAHAN** |
|  |  |
|  | **PEMROGRAMAN VISUAL DOT NET** |
|  |  |
|  | Penggunaan Object ADO.Net |
|  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  | **Fakultas** | **Program Studi** | **Tatap Muka** | **Kode MK** | **Disusun Oleh** |  |
|  | Ilmu Komputer | Sistem Informasi | **12** | **W151700005** | Hanna Yunita, ST.,MKom |  |

|  |  |
| --- | --- |
| **Abstract** | **Kompetensi** |
|  |  |
| Pada modul ini akan mempelajari tentang penggunaan object pada ADO.Net | Mahasiswa mampu dan dapat memahami program menggunakan database ADO.Net dan menggunakan object ADO.Net. |
|  |  |

# 12.1 Introduction ADO.NET

Pada teknologi .NET Microsoft telah menyediakan Universal Data Access baru untuk mengakses database yang dikenal dengan nama ADO.NET. ADO.NET sebagai data provider (penyedia data) terdiri dari 4 komponen utama yaitu :

1. Komponen / Object Connection
2. Command
3. DataReader
4. DataAdapter

Object DataReader digunakan untuk membaca baris data dalam jumlah banyak dan bersifat *read and forward only*, umumnya baris data yang diperoleh akan langsung digunakan untuk ditampilkan pada layar computer atau menjadi bagian dari baris record suatu laporan.

Object DataAdapter berfungsi sebagai perantara untuk mengantar perintah objek command ke database atau sumber data melalui hubungan yang dispesifikasikan pada object connection, selanjutnya baris data yang diperoleh akan ditampung pada object DataSet.

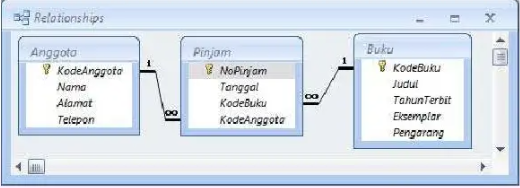
Sedangkan Object DataSet berfungsi untuk menampung baris data (table data) sebelum baris data ditampilkan pada layar computer (baik dalam bentuk individual record maupun dalam bentuk datagrid)

ADO.NET dikelompokkan dalam Namespace sesuai dengan bagian/kelompoknya. Sperti System.Data.OleDB digunakan untuk mengakses berbagai tipe sumber data/Database, sedangkan System.Data.SqlClient hanya digunakan untuk mengakses SQL Server Database. Namespace dapat disertakan pada individu program dengan menambahkan kata **import** pada awal program dilanjutkan dengan nama *Namespace* yang ingin digunakan. Biasanya penggunaan namespace dideklarasikan melalui project properti yaitu melalui opsi **reference.**

# 12.2 ADO.NET via Toolbox

Koneksi ke Microsoft Access via Toolbox. Untuk membuat aplikasi database dengan bahasa pemrograman apapun, yang harus disiapkan terlebih dahulu adalah koneksi aplikasi tersebut ke dalam database. Yang banyak digunakan jenis koneksi Visual Basic saat ini adalah **Microsoft Data Control** untuk semua koneksi ke database, selain itu koneksi khusus ke database Oracle visual basic memiliki **Oracle Data Control(ORADC).**

Contoh kasus Skema database Sistem Perpustakaan :

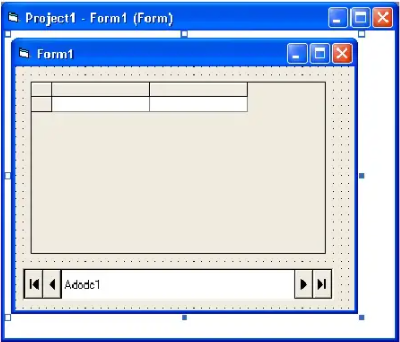


**Microsoft Data Control (ADODC)**

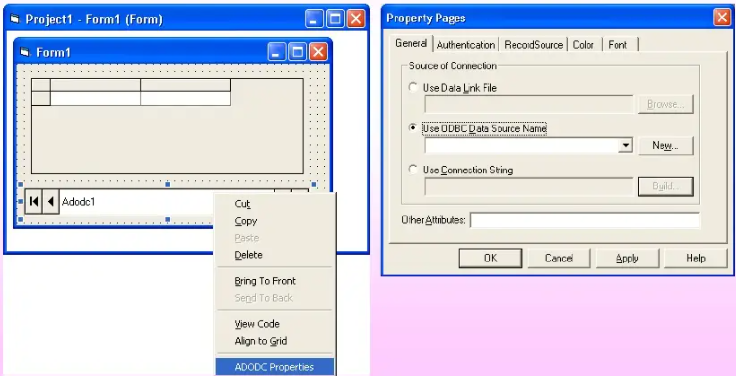
Keunggulan menggunakan koneksi ADODC adalah semua database bisa dikoneksi dengan ADODC dengan memilih provider database yang sesuai dengan database yang dipakai baik untuk database local maupun database jaringan.

Cara menambahkan komponen MS ADO Data Control :

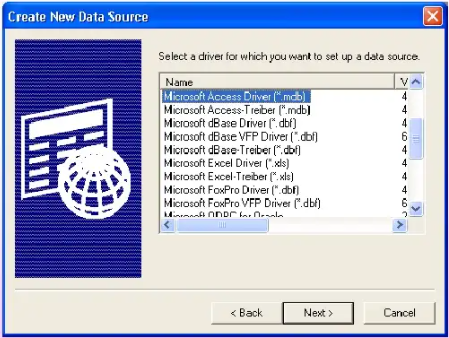
* Pilih menu : Project - Components atau
* Arahkan mouse pada area toolbox dan klik kanan pilih Components
* Pilih komponen : MS ADO Data Control 6.0 dan MS DataGrid Control 6.0 dan tekan OK.
* Pilih komponen ADODC an DataGrid pada toolbox dengan double click mouse maka akan muncul komponen tersebut dalam form.
* Atur ukuran dan posisi komponen tersebut.



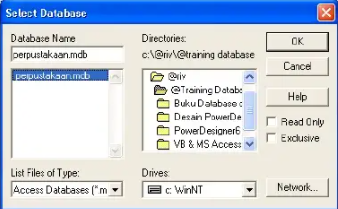
* Hubungkan ADODC ke database MS Access Perpustakaan.
* Klik kanan pada ADODC yang berada di form tersebut. Lalu pilih ADODC Properties.



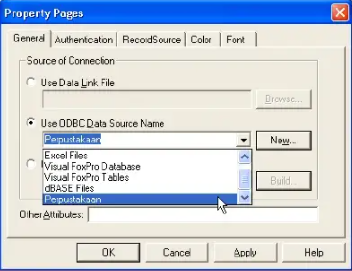
* ODBC (Open Database Connectivity) merupakan teknik membuka dan mengkoneksi database secara umum. Semua database yang dikoneksi dapat dilakukan dengan teknik ODBC karena ODBC pada dasarnya diletakkan atau berada di setiap administrasi Operating System.
* Pilih Use ODBC Data Source Name untuk membuat nama koneksi ODBC yang dihubungkan dengan database dan klik New untuk memulai.
* Kotak dialog ini menanyakan bagaimana anda ingin membagi database yang ingin diakses pada program visual basic.
* Pilihan “File Data Source” menentukan bahwa database anda akan digunakan oleh pemakai pada computer lain (by network/internet) pilihan fleksible tapi tidak perlu untuk database system tunggal
* Pilihan “User Data Source” : Database hanya untuk anda sendiri.
* Pilihan “System Data Source” : database bisa digunakn oleh oranglain dengan computer anda namun Login yang berbeda.
* Pilih yang ketiga, lalu tekan Next.
* Pilih driver Microsoft Access Driver (\*.MDB) yang menghubungkan ke database Ms.Access.

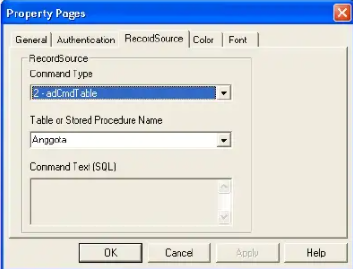


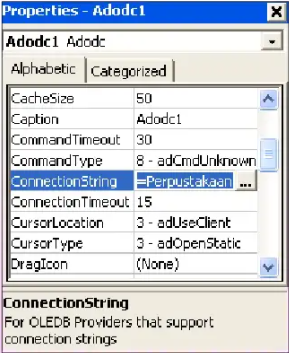
* Tekan Finish
* Beri nama Data Source Name, misalkan Perpustakaan
* Lalu tekan select, untuk memilih file database Perpustakaan.mdb yang tersimpan
* Tekan OK dan OK lagi.



* Kembali ke form dialog semula dan pada user ODBC Data Source Name, pilih nama DSN (Data Source Name) : Perpustakaan.



* Pilih Tab RecordSource, masukkan :
* Command Type pilih 2 – adCmdTable
* Table or Stored Procedure Name : “Anggota”
* Tekan OK
* Dan cek **Properties** nilai ConnectionString berisi DSN=Perpustakaan



* Jalankan program



# 12.3 Object Connection

Connection object memegang peran penting dalam koneksi antara aplikasi dengan data source. ADO.NET connection object sendiri terhubung dengan database dalam aplikasi yang kita bangun. Ketika koneksi database sudah dibuat, kita dapat memberikan perintah SQL untuk menerima data atau memanipulasi database. Setelah aktivitas berakhir kita perlu menutup koneksi ke database. Pada ADO.NET jenis koneksi ditentukan oleh sistem database yang kita gunakan. Adapun jenis-jenis koneksi database yaitu sebagai berikut:

1.      OLEDB Connection

2.      ODBC Connection

3.      SQL Connection

Kita bahas satu persatu tentang jenis-jenis koneksi database.

1. OLEDB Connection

Koneksi database OLEDB Connection menggunakan driver yang sudah tersedia. Contoh syntax yang digunakan adalah sebagai berikut:

**ConnectionString = “Provider = Microsoft.Jet.OLEDB.4.0; Data Source = yourdatabasename.mdb;”**

Contoh penggunaannya dalam program adalah sebagai berikut:

**Dim connetionString As String**

**Dim cnn As OleDbConnection**

**connetionString="Provider=Microsoft.Jet.OLEDB.4.0;Data Source=yourdatabasename.mdb;"**

**cnn = New OleDbConnection(connetionString)**

**Try**

**cnn.Open()**

**MsgBox("Connection Open ! ")**

**cnn.Close()**

**Catch ex As Exception**

**MsgBox("Can not open connection ! ")**

**End Try**

1. ODBC Connection

Cara terakhir untuk membangun koneksi database adalah dengan menggunakan ODBC Connection. Untuk cara ini kita harus mendeskripsikan koneksi database dengan syntax berikut:

**connetionString = "Driver = {Microsoft Access Driver (\*.mdb)}; DBQ = yourdatabasename.mdb;"**

**Contoh source code adalah berikut ini:**

**Dim connetionString As String**

**Dim cnn As OdbcConnection**

**connetionString="Driver={Microsoft Access Driver (\*.mdb)};DBQ=yourdatabasename.mdb;"**

**cnn = New OdbcConnection(connetionString)**

**Try**

**cnn.Open()**

**MsgBox("Koneksi berhasil ! ")**

**cnn.Close()**

**Catch ex As Exception**

**MsgBox("koneksi database gagal ! ")**

**End Try**

1. SQL Connection

Membangun koneksi database dengan menggunakan SQL Server Connection string. Syntax yang digunakan adalah sebagai berikut:

**connetionString = "Data Source=ServerName;Initial Catalog=DatabaseName; User ID=UserName;Password=Password"**

Jika sudah memberikan nama pada instance di SQL Server kita perlu mengubahnya menjadi “Server = localhost\Sqlexpress” untuk menutup koneksi kita gunakan method close(). Dengan method close() transaction pending akan di rools back dan koneksi akan diputus. Jika kita ingin membangun koneksi database dengan menggunakan authentikasi windows kita dapat menggunakan syntax berikut ini:

**"Server= localhost; Database= employeedetails; Integrated Security=SSPI;"**

Sedangkan jika hendak melakukan koneksi dengan menggunakan IP syntaxnya adalah sebagai berikut:

**connetionString="Data Source=IP\_ADDRESS,PORT; Network Library=DBMSSOCN; Initial Catalog=DatabaseName; User ID=UserName;Password=Password"**

Contoh program untuk membangun koneksi Database dengan SQL Connection adalah sebagai berikut:

**Private Sub Button1\_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click Dim connetionString As String Dim cnn As SqlConnection connetionString = "Data Source=ServerName;Initial Catalog=DatabaseName;User ID=UserName;Password=Password" cnn = New SqlConnection(connetionString) Try cnn.Open() MsgBox("Koneksi berhasil ! ") cnn.Close() Catch ex As Exception MsgBox("koneksi database gagal ! ") End Try**

Untuk melakukan pengujian kali ini saya membuat project baru, kemudian menambahkan form baru dengan sebuah button. Saya menambahkan event click pada button. Kemudian saya menambahkan source code berikut untuk melakukan koneksi dan melakukan perintah SQL. Koneksi database dapat disesuikan dengan kondisi di PC anda.

**Dim connetionString As String Dim connection As SqlConnection Dim command As SqlCommand Dim sql As String connetionString = "Data Source=ServerName;Initial Catalog=DatabaseName;User ID=UserName;Password=Password" sql = Select \* from employe" connection = New SqlConnection(connetionString) Try connection.Open() command = New SqlCommand(sql, connection) Dim sqlReader As SqlDataReader = command.ExecuteReader() While sqlReader.Read() MsgBox(sqlReader.Item(0) & " - " & sqlReader.Item(1) & " - " & sqlReader.Item(2)) End While sqlReader.Close() command.Dispose() connection.Close() Catch ex As Exception MsgBox("Koneksi gagal ! ") End Try**

Nilai-nilai yang ada pada Object Connection:

* Provider

Berisi nama piranti / driver yang akan digunakan oleh objek model ADO.NET untuk mengakses sumber data, untuk SQL Server, pirantinya bernama SQLOLEDB.

* Server atau Data Source

Nama server dimana SQL server dijalankan. Anda dapat menggunakan nama server PC anda (misalnya privacy), atau menggunakan nama generic localhost jika anda menggunakan Windows Server atau nama local

* Initial catalogue

berisi nama database yang akan digunakan.

* UserID: Adalah nama user database SQL Server
* Password : Adalah kode sandi dari User ID

Biasanya untuk memudahkan maintenance program spesifikasi connection string disimpan sebagai public variable pada induk program. Selanjutnya masing-masing program akan mengambil spesifikasi tersebut sebagai informasi untuk berhubungan dengan database ketika membuat atau mendeklarasikan object connection.

# 12.4 Object Data Reader

Walaupun object DataReader memiliki kemampuan untuk membaca baris record dalam jumlah yang banyak, tetapi kenyataannya objek ini jarang digunakan dibanding dengan object DataAdapter. Oleh karena itu object DataReader biasanya digunakan untuk proses pembuatan laporan.

Object DataReader memiliki banyak properti dan metode. Umumnya program hanya memanfaatkan metode *Read* untuk membaca baris dan metode *close* untuk menutup DataReader. Proses pengambilan nilai field/kolom dapat langsung dilakukan dengan menyebutkan nama field yang diinginkan. Object ini diaktifkan melalui metode ExecuteReader yang dimiliki oleh Object Command.

# 12.5 Object Command

Object command umumnya digunakan untuk proses terkait dengan aktivitas manipulasi data baik itu melalui perintah SQL maupun melalui Stored Procedure dan metode Execute Non Query digunakan untuk mengeksekusi / mengirim perintah tersebut.

Properti yang sering digunakan pada objek ini adalah properti Connection, Command Type dan CommandText.

* Properti connection digunakan untuk hubungan ke sumber data biasanya objek ini memanfaatkan object connection.
* CommandText berisi pernyataan SQL, nama tabel atau nama Stored Procedure.
* CommandType berisi nilai perintah yang harus disesuaikan dengan jenis CommandText yang diberikan.

Jika proses (Object Command) digunakan untuk memperoleh baris data, maka object command akan dikapsulasi melalui DataAdapter. Sebelum dikirim ke sumber data. Jika tidak ada baris yang perlu dikembalikan maka digunakan metode *Execute NonQuery* untuk mengeksekusi perintah. Penggunaan metode *Execute Scalar* hanya akan mengembalikan nilai pada kolom pertama dari baris pertama yang dihasilkan, sedangkan metode *Execute Reader* digunakan bersamaan dengan objek DataReader.

Inti dari penggunaan Object Command adalah untuk memanipulasi database melalui ActionQuery (Insert/Update/Delete).

# 12.6 Object DataSet

Istilah DataSet menggantikan istilah RecordSet yang dikenal pada model ADO pada VB6. DataSet adalah representasi dari suatu kelompok data yang terdiri dari kolom, tabel-tabel, hubungan baris dan berbagai constraint dimana proses interaksi antara object DataSet dan sumber data (Database) dilakukan melalui provider.

Beberapa contoh bentuk pernyataan object dataset :

* Menghitung jumlah baris dalam tabel dapat digunakan sebagai control keberadaan record, misal jika jumlah baris = 0 berarti tidak diketemukan record yang sesuai untuk proses view, update maupun delete. Sebaliknya jika jumlah record = 1 maka proses penambahan record tidak diijinkan. Berikut adalah contoh bentuk pernyataan untuk menghitung jumlah baris dalam tabel.

mTotal\_Record.Text=objdataset(“Album”).rows.count

* Mengambil nilai pada kolom dan baris tertentu dibuthkan terutama pada saat record akan diubah. Ini dilakukan melalui properti item yang dimiliki object Row. Jika anda ingin mengambil dari baris pertama, maka index row untuk baris tersebut bernilai=0. Berikut ini contoh pengambilan nilai masing-masing kolom dan record album untuk keperluan edit.

if mKode\_Proses.text=”EDIT” then

with objdataset.tables(“Album”).rows(0)

txtAlbum\_ID.text = .item(“Album\_ID”)

txtAlbum\_desc.txt = .item(“Album\_desc”)

end with

end if

# Daftar Pustaka

1. Programming Visual Basic .NET, Dave Grundgeiger, Publisher: O'Reilly First Edition January 2002 (Chapter 1).
2. Modul Mata Kuliah Pemrograman Visual Basic .Net, Abd. Rahmat Dayat, Manajemen Informatika, AMIK UMEL Mandiri, Jayapura, 2016.
3. Dasar-dasar Pemrograman dengan .Net, Ade rahmat Iskandar, Elex Media Komputindo, 2019