1. Difference between Connected Architecture and Disconnected Architecture
2. DataReader and DataSet.
3. Features of DataSet
4. Working with DataSet
5. Methods of DataAdapter
6. Accessing data from DataSet
7. DataTable
8. Referring to a cell under DataTable

Let's start with the details.

ADO.NET provides the following two models for accessing data from a Data Source:

1. Connection-Oriented Architecture
2. Disconnected Oriented Architecture

**1. Connection-Oriented Architecture**

In this case, we require a continuous connection with the Data Source for accessing data in it.

Here the “DataReader” class holds the data on client machines.

**2. Disconnected Oriented Architecture**

In this case, we do not require a continuous connection with the Data Source for accessing data.

Here the “DataSet” class holds the data in the client machines.

DataSet

It is a class present under the “System.Data” namespace design for holding and managing data on client machines apart from a DataReader.

**Features Of DataSet**

1. It is also capable of holding multiple tables.
2. It is a designed disconnected architecture that doesn't require a permanent connection with a Data Source for holding data.
3. It provides scrollable navigation to data, that allows us to move in any direction.

In other words, either top to bottom or bottom to top.

1. It is updatable, in other words, changes can be performed to data present in it and also send changes back to the DB.

**Working With DataSet**

This class is responsible for the loading of data into a DataReader from a DataSource and is a command.

In the same way, the DataAdapter class is used for communication between a DataSource and DataSet.

Simply

DataReader < - Command -> DataSource.

DataSet <->DataAdapter <->DataSource.

**Methods of DataAdapter**

1. Fill (DataSet ds,string TableName).
2. Update (DataSet ds,string TableName)

Fill in the method to load data from a DataSource into a DataSet.

Update is to transfer data from a DataSet to a DataSource.

DataAdapter is internally a collection of the following 4 methods:

1. Select Command.
2. Insert Command.
3. Update Command.
4. Delete Command.

When we call the Fill() method of Adapter, the following action takes place internally.

1. Open a connection with the DataSource.
2. Execute the Select command under it on the DataSource and loads data from the table to the DataSet.
3. Close the Connection.

Siince a DataSet is updatable, changes can be made to data that is loaded into it, like adding, modifying and deleting records.

After making all the changes to the data in a dataset if we want to send those changes back to the DataSource then call the Update() method on the DataAdapter that performed the following:

1. Re-opened a connection with the DataSource.
2. Changes made in the dataset will be sent back to the table where in this process it will use the insert, update and delete commands of the DataAdapter.
3. Close the Connection.

**Accessing Data From DataSet**

DataReader provides us a pointer vases access to the data, so we can get data only in sequential order, whereas a dataset provides us index-based access to data so we can get the data from any location.

Dataset is a collection of tables where each table is represented as a DataTable class and identified by the index position.

**DataTable**

DataTable is the collection of tables.

The following syntax is used for a DataTable:

<DataSet>.Tables[Index]/[Name]

For example:

Ds.Tables[0]

Or

Ds.tables[“Company”]

Every DataTable is again a collection of Rows and Columns where each row is represented as a DataRow class and identified by its index position.

Each column is represented as a DataColumn class and identified by index position or name.

**DataRow**

It is a collection of rows.

Syntax:

<datatable>.Rows[Index].

For example:

Ds.tables[0].rows[0]

**DataColumns**

It is a collection of Columns.

Syntax:

<datatable>.Columns[Index] OR columns[Name]

For example:

Ds.Tables[0].Column[0]

Or:

Ds.Tables[0].Column[“ENO”]

Ds.tables[0].rows[0]

The following is the syntax for referring to a cell under a Data Table:

<datatable>.Rows[row] [col]

For example:

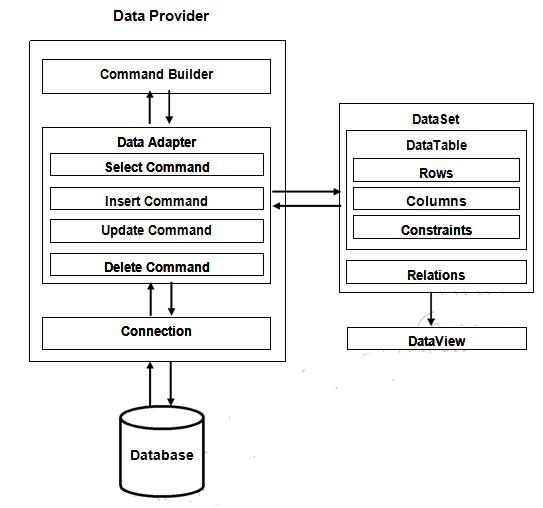
Ds.Tables[0].Rows[0][0]

Or:

Ds.Tables[0].Rows[0][“ENO”]

Ds.tables[0].rows[0]

Disconnected Architecture in ADO.NET

                            The architecture of ADO.net in which data retrieved from database can be accessed even when connection to database was closed is called as disconnected architecture. Disconnected architecture of ADO.net was built on classes connection, dataadapter, commandbuilder and dataset and dataview.  
  
  
  
  
 **Connection :**Connection object is used to establish a connection to database and connectionit self will not transfer any data.

**DataAdapter :**DataAdapter is used to transfer the data between database and dataset. It hascommands like select, insert, update and delete. Select command is used to retrieve data from database and insert, update and delete commands are used to send changes to the data in dataset to database. It needs a connection to transfer the data.

**CommandBuilder :**by default dataadapter contains only the select command and it doesn’tcontain insert, update and delete commands. To create insert, update and delete commands for the dataadapter, commandbuilder is used. It is used only to create these commands for the dataadapter and has no other purpose.

**DataSet :**Dataset is used to store the data retrieved from database by dataadapter and make itavailable for .net application.

To fill data in to dataset **fill()** method of dataadapter is used and has the following syntax.

**Da.Fill(Ds,”TableName”);**

When fill method was called, dataadapter will open a connection to database, executes select command, stores the data retrieved by select command in to dataset and immediately closes the connection.

As connection to database was closed, any changes to the data in dataset will not be directly sent to the database and will be made only in the dataset. To send changes made to data in dataset to the database, **Update()** method of the dataadapter is used that has the following syntax.

**Da.Update(Ds,”Tablename”);**

When Update method was called, dataadapter will again open the connection to database, executes insert, update and delete commands to send changes in dataset to database and immediately closes the connection. As connection is opened only when it is required and will be automatically closed when it was not required, this architecture is called disconnected architecture.

A dataset can contain data in multiple tables.

**DataView :**DataView is a view of table available in DataSet. It is used to find a record, sort therecords and filter the records. By using dataview, you can also perform insert, update and delete as in case of a DataSet.