



Microsoft® ADO.NET

Presented by
INSB IITMS BCA & PGDCA
College, Idar
BCA SEM-5



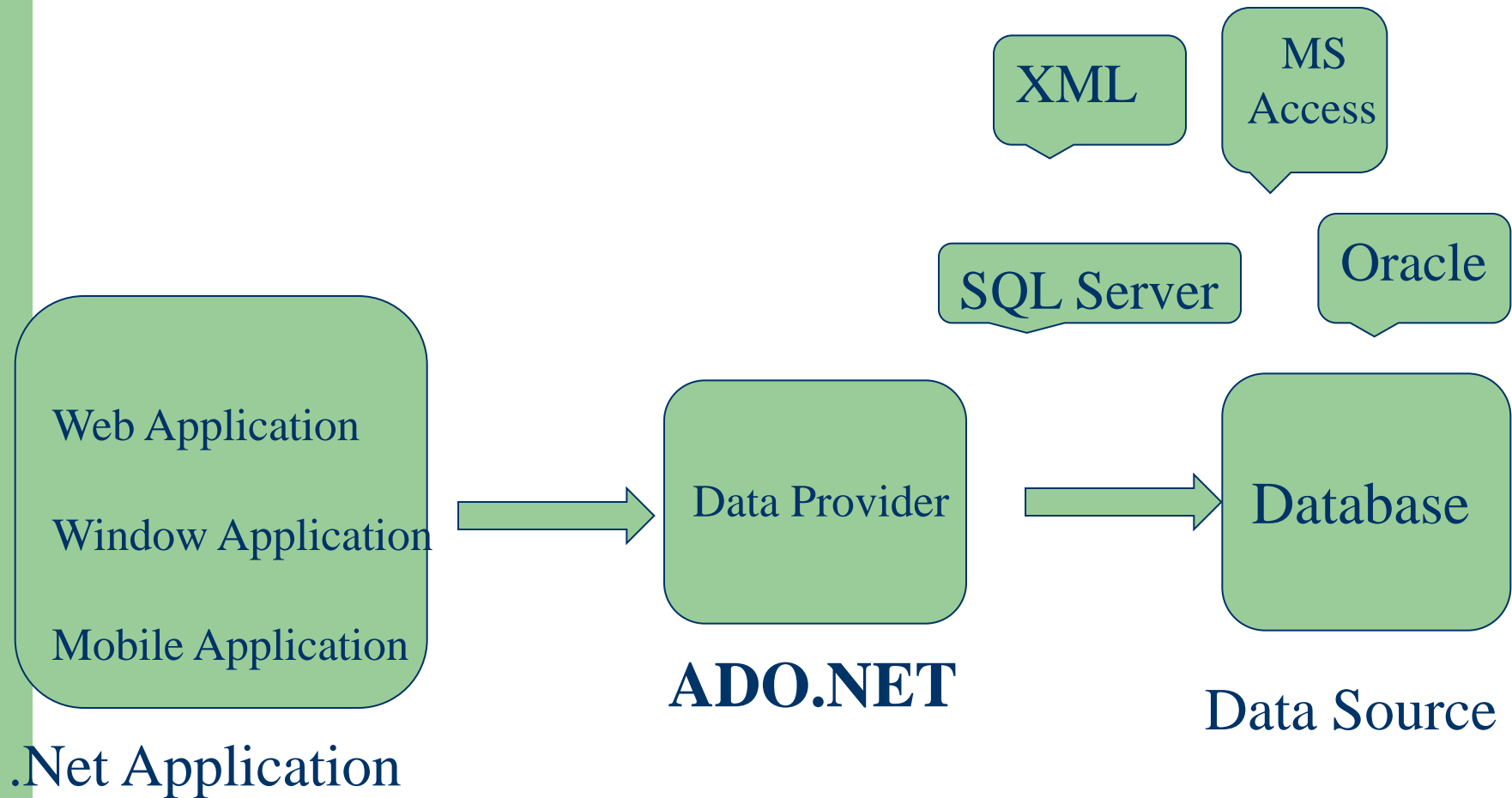
Agenda

- **ADO.NET Overview**
- ADO.NET Architecture
- DataSet
- Data Provider
- DataReader
- DataAdapter Object
- DataObject
- Data Reader VS Data Set
- ADO VS ADO.NET
- Data Binding
- Three tiers of web application

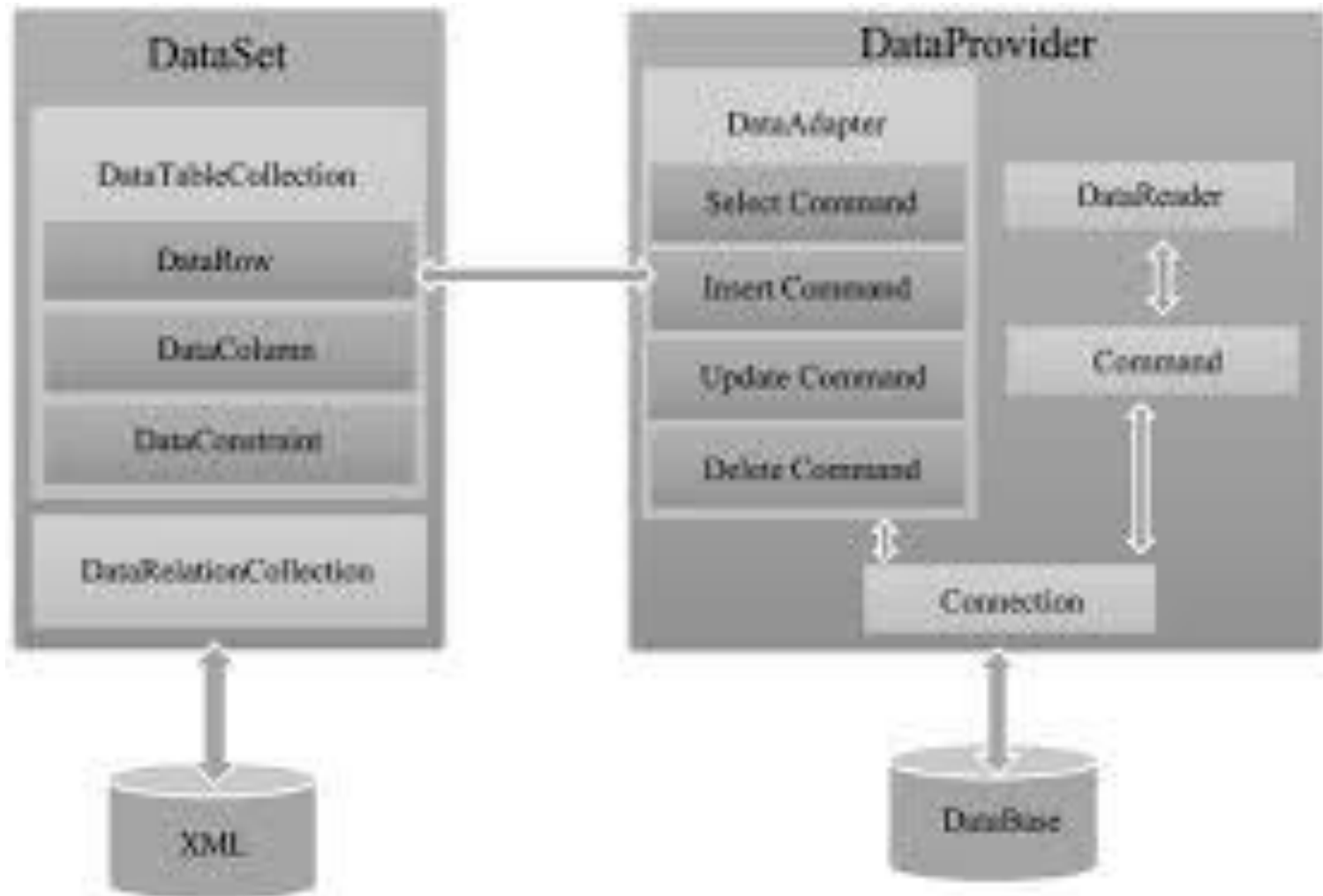
INTRODUCTION OF ADO.NET

- ADO Stand for **Active-X Data Object**.
- Microsoft ADO.Net is the latest improvement after ADO.net provider platform interoperability and scalable data access.
- .Net framework data is transmitted in the **Extensible Markup Language**(XML).
- ADO.Net technology used for working with data and database of all types. It provides access to data source as microsoft sql server and to data source exposed through OLEDB and XML.

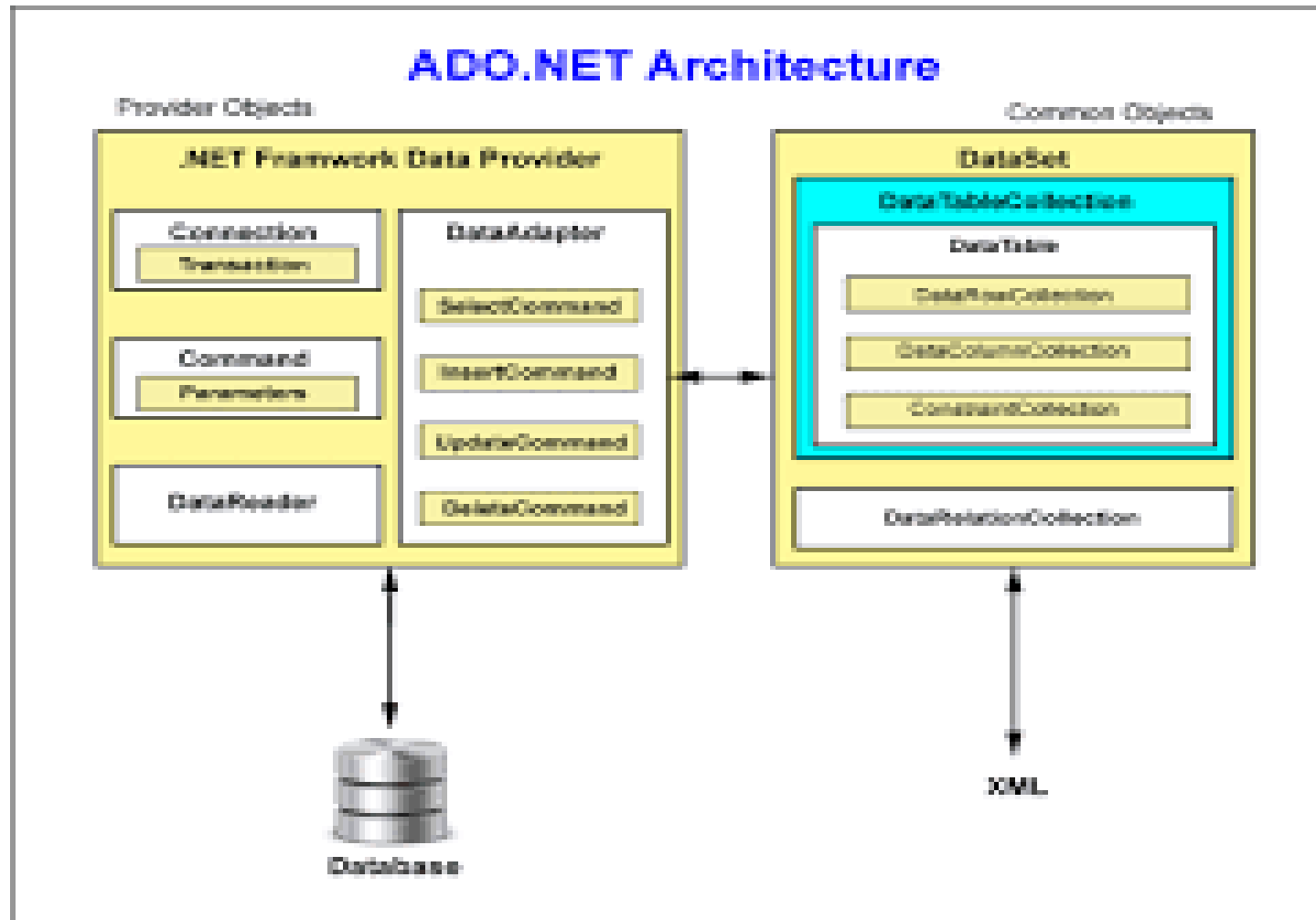
WHAT IS ADO.NET?



ADO.NET Architecture Diagram



ADO.NET Architecture Diagram





Data Source

Data Provider	Namespace	DataBases
SQLClient	System.Data.SqlClient	SQLServer
OracleClient	System.Data.OracleClient	Oracle
OLEDB	System.Data.OleDb	MS Aceess
ODBC	System.Data.Odbc	Configured Databases

Data Provider

- The connection object all of the information required to open a channel to the database in the connection string property.
- The .net application can not directly execute our SQL code. It only understands C#. If a .net application need to retrieve data or to do some insert, update, and delete operations from or to a database.
- Database Connection
- Prepare SQL Command
- Execute the Command
- Retrieve the results and display the web application

Data Set

- Data table represents a single table.
- Data set represents multiple tables as a data table objects.
- Data table and data set resides in dis-connected architecture.
- Represents a cache of data that contains tables, columns, relationships, and constraints, just like a database.
- It is collection of data tables that contain the data.
- Data set tabular representation of data. Tabular representation means it represents data into row and column format.
- We can use data set in combination with data adapter class. The data set contains the copy of the data we requested.
- A data set local copy of your database table that get populated in client pc.

Data Reader

- Provides methods and properties that deliver a forward-only stream of data rows from a data source
- When a DataReader is used, parts of the ADO.NET model are cut out, providing faster and more efficient data access

Data Adapter

- Provides a set of methods and properties to retrieve and save data between a DataSet and its source data store
- Allows the use of stored procedures
- Connects to the database to fill the DataSet and also update the database

Data Object

- Provides methods and properties that enable UI objects such as a DataGrid to bind to a DataSet
- A view of the data contained in the DataSet
- Only used in conjunction with a DataSet

ExecuteNonQuery()

- This method is commonly used for update, insert, or Delete statements, where the only returned value is the number of records affected.
- This method can be return results if you call a stored procedure that has output parameters.

Data Reader

- Data Reader is a connection oriented service.
- Data Reader is used to retrieve **Read only** (can not update) and **forward only** (can not read backward / Random) data from a database.
- Data Reader fetches the records from database and stored in the network buffer and given whenever requests.
- Data Reader is like a forward only record set. It fetches one rows at a time so very less network cost compare to DATASET.
- As a one row at a time is stored in memory in data reader it increase application performance and reduced system overheads.

Data Set

- Data Set Class in ado.net operates in a entirely disconnected nature.
- Data set is a **collection of in-memory table**. It can be used with multiple data source.
- Data Set releases the data after loading all the data in memory.
- Dataset fetches all the rows at a time. i.e. it fetches all data from the data source at a time to its memory area.
- There is more system overheads in dataset as it fetches all the data from the data source at a time in memory.

ADO	ADO.NET
➤ Classic ado regulars active connection with the data store.	➤ ADO.NET architecture works while the data store is disconnected.
➤ Data is stored in binary format	➤ Data is stored in xml.
➤ XML integration is not possible.	➤ XML integration is possible.
➤ It uses the object named Record set to reference data from the data store.	➤ It uses data set object for data access and representation.
➤ Firewall might prevent execution of classic ADO.	➤ ADO.NET has firewall proof and its execution will never be interrupted.
➤ Classic ADO architecture includes client side curser and server side curser.	➤ ADO.NET architecture doesn't include such cursors.
➤ We cannot send multiple transactions using a single connection instance.	➤ We can send multiple transactions using single connection instance.

Data Binding

- You can use dataset or the data Reader to retrieve information and add them to an HTML on a web page.
- ASP.NET adds a feature that allows you to skip this process and pop data directly into HTML elements and fully formatted controls. It called '**DATA BINDING**'.
- Data binding is where to find your data and how you want it displayed and the control handles the rest of the details
- Data binding refers to the creation of a direct connection between a data source and a control in a application window.
- Data binding is much more flexible than traditional data binding many of the most powerful data binding controls.
- There are two type of .NET data binding.
 1. Single value or “Simple” Data binding
 2. Repeated value or “List” Data binding

Single Value Data Binding

- Single value data binding is used to add information anywhere on an asp.net page.
- Single value data binding allows you to take a variable ,Property , or Expression and insert it dynamically into page.

For ex. => Public partial class sdb: system.web.ui.page

```
{  
    Protected int text;  
}  
Protected void page_load(object sender, EventArgs e)  
{  
    Test=500;  
    This.DataBind();  
}
```

Repeated Value Data Binding

- Repeated value data binding is much more useful for handling complex task such as displaying an entire table or all the values from single field in the table.
- Repeated value data binding works with the asp.net list controls such as listbox, Dropdownlist box, checkbox list and radiobutton list, gridview, detailsview, formview, and listview.
- You will know that a control supports repeated value data binding if it provides a data source property.
- You need to follow only three stapes
- Create and fill some kind of data object=>
`list<String>fruits;`
- Link the object to the control=> `listitems.data source=fruits;`
- Activate the binding => `this.databind();`
- You can use binding to bind data a collection or an array.

Three Tiers of Web Application

- Three-tier architecture is a well-established software application architecture that organizes applications into three logical and physical computing tiers.
- The presentation tier, or user interface, the application tier, where data is processed, and the data tier, where application data is stored and managed.
- There are three tier of Web Application
 - 1) Presentation Tiers
 - 2) Application Tiers
 - 2) Data(Database) Tiers

Presentation Tiers

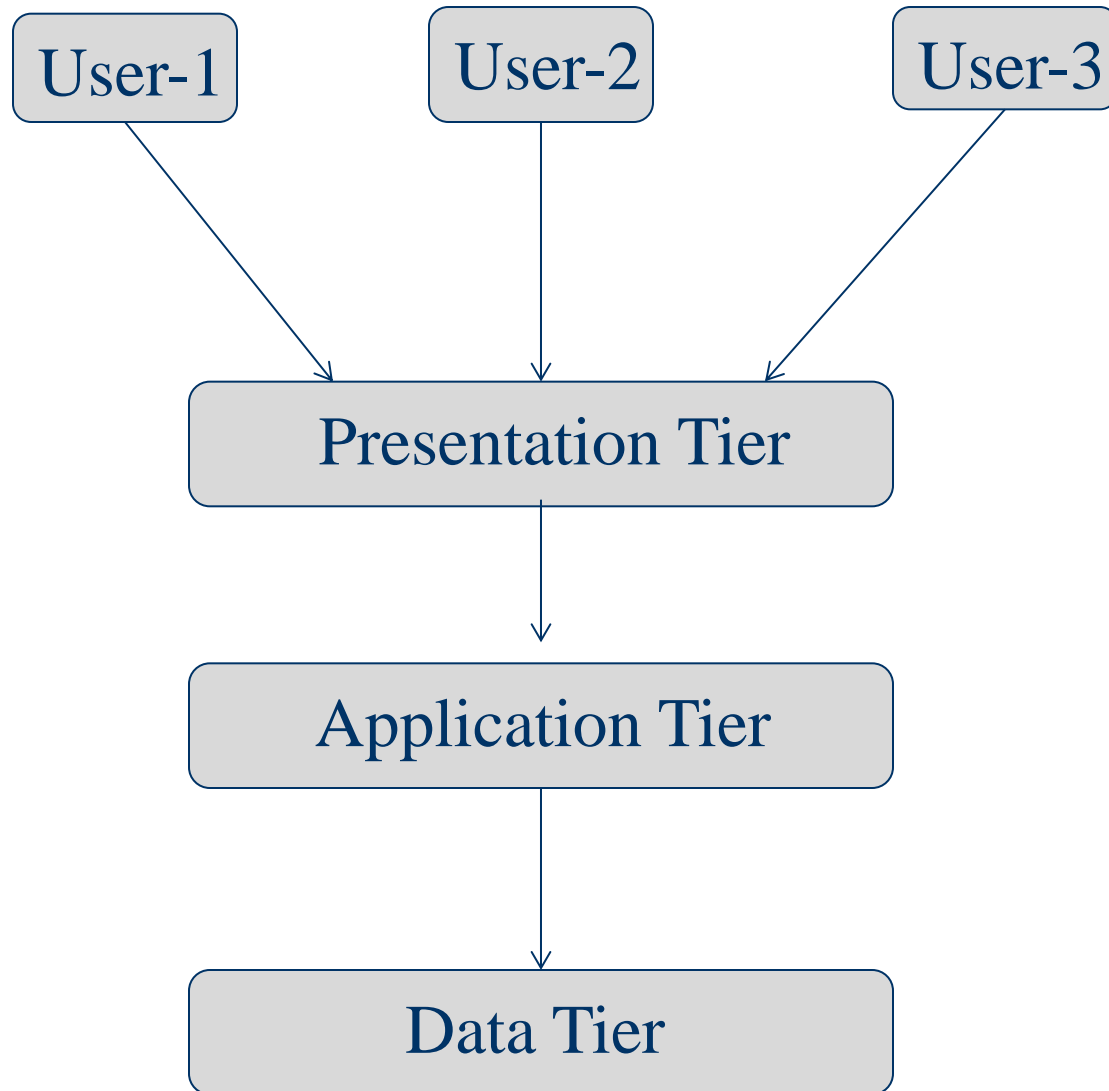
- The presentation tier is the user interface and communication layer of the application, where the end user interacts with the application.
- Its main purpose is to display information to and collect information from the user.
- This top-level tier can run on a web browser, as desktop application, or a graphical user interface (GUI), for example. Web presentation tiers are developed by using HTML, CSS, and JavaScript. Desktop applications can be written in various languages depending on the platform.

Application Tiers

- The application tier, also known as the logic tier or middle tier, is the heart of the application.
- In this tier, information that is collected in the presentation tier is processed - sometimes against other information in the data tier - using business logic, a specific set of business rules.
- The application tier can also add, delete, or modify data in the data tier.

Data Tiers

- The data tier, sometimes called database tier, data access tier or back-end, is where the information that is processed by the application is stored and managed.
- This can be a relational database management system such as MySQL, MongoDB, Oracle, Db2, Informix or Microsoft SQL Server or Database server.



Command Object

- Information submitted to a database as a query via a Connection object
- Two provider-specific classes
 - SqlCommand
 - OleDbCommand
- Input and output parameters are supported, along with return values as part of the command syntax
- Results are returned in the form of streams. Accessed by:
 - DataReader object
 - DataSet object via a DataAdapter

Connection object

- Connects to databases.
- Two provider-specific classes
 - SqlConnection
 - OleDbConnection.
- Connections can be opened in two ways:
 - Explicitly by calling the Open method on the connection
 - Implicitly when using a DataAdapter.
- Connections handle transactions

ADO.NET Namespaces

System.data	Core namespace, defines types that represent data
System.Data.Common	Types shared between managed providers
System.Data.OleDb	Types that allow connection to OLE DB compliant data sources
System.Data.SqlClient	Types that are optimized to connect to Microsoft® SQL Server
System.Data.SqlTypes	Native data types in Microsoft® SQL Server

Importing the ADO.NET Namespaces

Needed to build a data access application

- **For OLE DB:**

- Imports System.Data

- Imports System.Data.OleDb

- **For SQL Server:**

- Imports System.Data

- Imports System.Data.SqlClient