25+ Most Popular ADO.NET Interview Questions And Answers

Last Updated: November 10, 2019

Most popular ADO.NET Interview Questions and Answers:

If you are preparing for an interview, this set of most frequently asked Ado.net Interview Questions and Answers will help you a lot.

We have designed the same to get you introduced to the questions which you may likely encounter during your technical interview. Hope this tutorial will be of immense help to you during your interview preparation journey.

Just have a look at them!

About ADO.NET

ADO.net is a database technology and is a part of the .NET framework architecture. ADO.net can be used in all .NET framework complaint programming language, such as Visual Basic.NET, VisualC++ etc. It supports disconnected architecture with which Data access and manipulation are performed completely independent of the data source.

List of the Frequently Asked ADO.NET Interview Questions

Let's begin !!!

Q #1) What is ADO.NET?

Answer: ADO.net Stands for Microsoft ActiveX Data Object. Ado.net is a database technology which we can think like a set of classes that can be used to interact with the data sources like databases and XML files.

Asp.net application, windows application, console application are the few types of .net applications that use ADO.NET to connect with the databases to execute commands and retrieve data.

Q #2) What are the key features of ADO.NET?

Answer:

- Disconnected Data Architecture.
- Data cached in DataSet.
- Scalability
- Data transfer in XML Format.
- Strongly typed language.

Q #3) Why is it important to close an ADO.NET application?

Answer: Connections need to be closed properly because it affects the scalability and reliability of the applications.

For open connections it is always vulnerable to attack, so to be short, 'Open connections as late as possible and close it as early as possible'. We can 'Close' the connections by 'final' block or 'using' the USING statement.

Q #4) What are the two uses of a 'using' statement in c#?

Answer:

#1) To import a namespace.

For Example: using System;

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.OleDb;
```

#2) To Close a connection properly, refer the below figure. The connection is automatically closed at the end of 'using' block.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.OleDb;

namespace WindowsFormsApp1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        private void Form1_Load(object sender, EventArgs e)
        {
            String constr = "Provider=Microsoft.ACE.OLEDB.12.0;" +
                            "Data Source=***" +
                            "Jet OLEDB:Database Password=***";
            using (OleDbConnection con = new OleDbConnection(constr))
            {
                con.Open();
                MessageBox.Show("Connection is Opened!");
            }

        }
    }
}
```

Q #5) Give an example of a .NET application which connects to Microsoft Access Database using Ado.net classes.

Answer: **Example** given below describes only how to establish a connection to Microsoft Access Database, it won't explain about the executing commands against the database.

Please refer the below figure.

```csharp
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.OleDb;

namespace WindowsFormsApp1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        private void Form1_Load(object sender, EventArgs e)
        {
            String constr = "Provider=Microsoft.ACE.OLEDB.12.0;" +
                            "Data Source= ***" +
                            "Jet OLEDB:Database Password=***";

            OleDbConnection con = new OleDbConnection(constr);

            try
            {
                con.Open();
                MessageBox.Show("Connection is Opened!");

            } catch (Exception ex) {
                MessageBox.Show("Connection is unable to Open!");
            }
        }
    }
}
```

For establishing a connection to the database and retrieve the values from a database we need to follow the below steps:

- Import reference library class. **Example:** using System.Data.OleDB.
- Create connection object.
- Specify the connection string, you may copy that from the Database Property.
- Use Connection object constructor and specify the 'connection String 'as Argument.
- Open an instance of Connection (It's better to open it in a try-catch statement).

Q #6) What is Connection Pooling?

Answer: Data providers implement 'Connection pooling' services. 'Connection pooling' improves performance. Connection pooling creates a set of connections.

When you are establishing a connection with the database, you will be given a connection from a pool. When you disconnect from the database, the connection will return to the pool and that connection will be used by the next applicant who wants a connection.

With 'Connection pooling' the 'opening' and 'closing' of connections no longer become an expensive task.

Please refer the below figure.
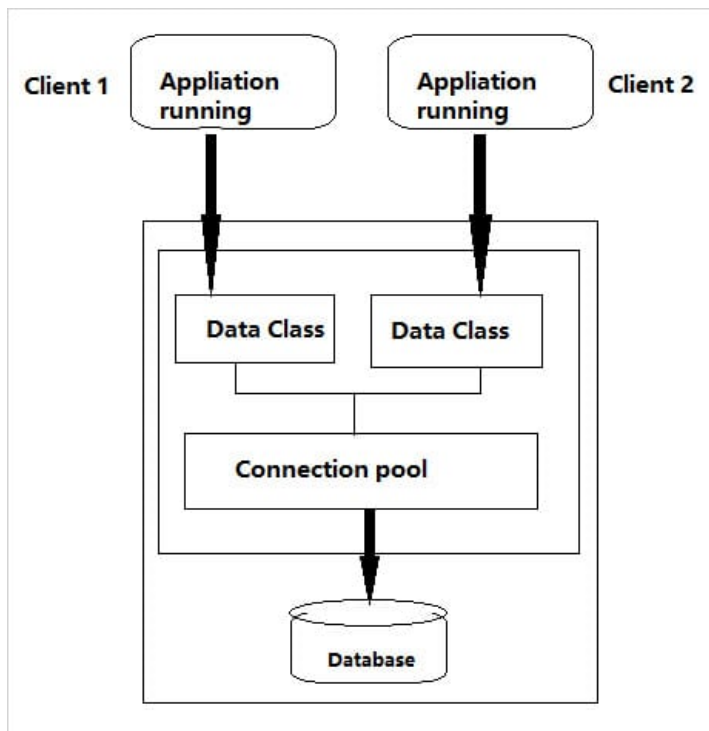
Q #7) What is a Dataset?

Answer: Dataset is a disconnected copy of data that gets populated in the client PC. It contains Datatables and Datarelations. Generally, DataAdapter is required for populating a Dataset with the data.

Since it is disconnected, the user no longer need to be connected to the database every time and data manipulations are performed without interacting with the data source. It is very fast and reliable as it resides in the local system.

Two Objects in Dataset

- DataTableCollection
- DataRelationCollection

Two Types of Dataset

- Typed Dataset
- Untyped Dataset

Q #8) What are the different methods by which we can populate a Dataset?

Answer: We can populate the Dataset using different the approaches mentioned below:

- Using DataAdapter objects and 'fill' method.
- Programmatically creating Datatable, Datarow, and Data column objects.
- Data from Xml Documents.
- Merging with another Dataset.

Q #9) What is DataAdapter?

Answer: DataAdapter helps in linking the database and connection object. DataAdapter provides the communication between Dataset and data sources.

Data moves 'in and out' of the Dataset through DataAdapter.

Properties of DataAdapter that allow the user to control the database are:

- Select Command
- Update Command
- Insert Command

- Delete Command

Q #10) What are DataReaders?

Answer: DataReader object is 'stream-based', 'read-only' and 'forward-only', which provides a connection based data access from a database. This contains a 'Read ()' method that retrieves the data stored in a data source.

A Connection Object has only one DataReader at a time. 'Read ()' method retrieves only one row at a time. That is data need not be completely read into the application before it is processed.

Please refer the below figure.

```csharp
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.OleDb;

namespace WindowsFormsApp1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        private void Form1_Load(object sender, EventArgs e)
        {
            String constr = "Provider=Microsoft.ACE.OLEDB.12.0;" +
                            "Data Source=*******" +
                            "Jet OLEDB:Database Password=***** ";
            String command = "Select * from [UrTableName] ";
            OleDbConnection con = new OleDbConnection(constr);
            OleDbCommand cmd = new OleDbCommand(command,con);
            try
            {
                con.Open();
                MessageBox.Show("Connection is Opened!");
                OleDbDataReader drRead = cmd.ExecuteReader();
                while (drRead.Read())
                {
                    Console.WriteLine(drRead[0]);
                }
            } catch (Exception ex) {
                MessageBox.Show("Connection is unable to Open!");
            }
        }
    }
}
```

Q #11) Give few examples of DataReader that is used in different DataProviders.

Answer: Each DataProvider has a DataReader class.

Few are given in the table below:

DataProvider	DataReader
OleDb	OleDbDataReade
Sql	SqlDataReader
ODBC	ODBCDataReader

Q #12) What is Databinding?

Answer: Databinding is the process of binding the data with graphical elements (controls in a window form). After binding the data in a window form, you can navigate through the records with the help of the Binding Navigator Control.

One of the advantages of data binding is, the user does not need to write the codes explicitly, for establishing the connections and creating a data set, this feature will write the necessary ADO.NET code for the user.

Q #12) What are the types of Databinding?

Answer: This classification is based on the number of bound values that can be displayed through a control of the window form.

Two types of Databinding

- Simple Data Binding
- Complex Data Binding

Simple Data Binding: It is the process of binding a control to a single data element.

For Example: Binding a textbox or label with a single value.

Complex Data Binding: It is the process of binding a component to display one data element or multiple data elements.

For Example: GridView or Listbox control binding one data element or more than one records in a table.
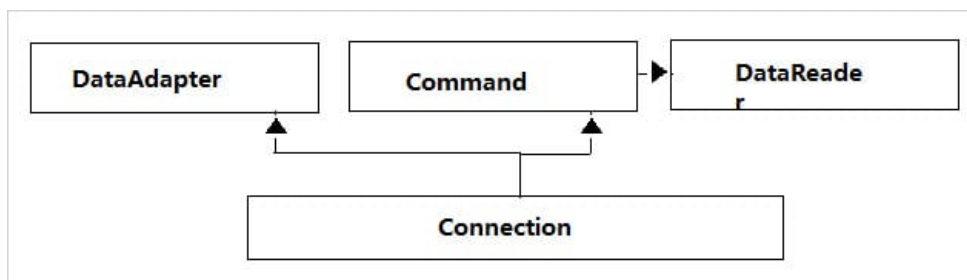
Q #13) What are DataProviders?

Answer: DataProviders are used for connecting to a database, retrieving data storing data, updating the data etc.

Example of DataProviders: SQL, OleDb, ODBC

Four Components of DataProviders are:

- Connection
- Commands
- DataReader
- DataAdapter

Please refer the below figure.

Q #14) Describe ADO.NET object model in detail.

Answer: ADO.net is based on an Object Model. The data from Datasource is retrieved with Data Provider. The Data providers include components such as connection, DataAdapter, DataReader, and Command. It provides data to the application, updates the database with the changes made in the application.

An Application can access the data either through a data set or through a DataReader object.

ADO.net also enables to create an XML representation of a dataset. In an XML representation of a dataset, data is written in XML format.

Please refer the below figure.

DataProvider for .NET framework
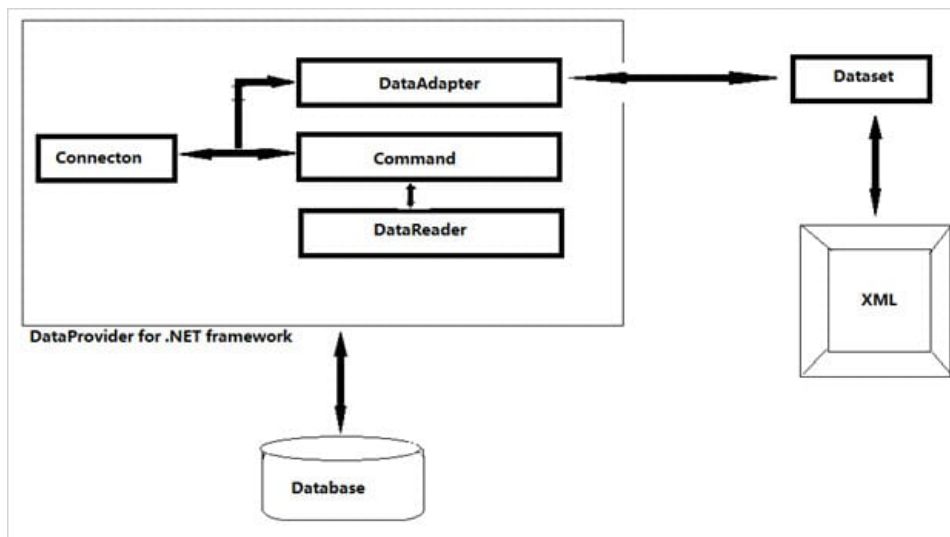
Q #15) What are the key events of SqlConnection Class?

Answer: **The two key events of SqlConnection are:**

- StateChange event
- InfoMessage event

StateChange event: This event occurred when the state of the Connection changes.

The event handler receives an argument (Datatype: StateChangeEventArgs) which contains the data related to that particular event.

InfoMessage event: This event occurred when an info message or Warning is returned from a data source. The event handler receives an argument (Datatype: SqlInfoMessageEventArgs) which contains the data related to that particular event.

Q #16) What you mean by Filtering of data?

Answer: Filtering of data is done when you need to display only selective records.

Given below are the two methods for filtering data:

- Creating parameterized queries.
- Filtering data using control of a window form.

Q #17) What is the difference between Connected and Disconnected environment?

Answer: **Difference between Connected and Disconnected environment is stated in the below table.**

Connected Environment	Disconnected Environment
It requires a constant connection to transfer data between the application and database	It doesn't require a constant connection to transfer data between the application and database
Data concurrency is easy to control	Data concurrency is not easy to control
Data is up-to-date since user is always connected to the database	Data is not up-to-date since user is always connected to the database
It has scalability and performance issues for the client application	It improves scalability and performance of the client application
Lead to network traffic logging	Less network traffic logging issues

Q #18) What do you mean by performing Asynchronous Operation using Command Object?

Answer: Sometimes execution of the commands in the database may take a large amount of time to complete as they are linked to each other.

A solution for such a situation has asynchronously executed the commands against the database without waiting for the command

execution to finish, which can be handy in the situation in which, when you try to execute the long-running base commands.

Advantages of Asynchronous Execution:

- Improves performance.
- Improve responsiveness of the client application.

Q #19) What do you mean by 'Batch Updates'?

Answer: A batch update can be defined as a batch of updates grouped together. To improve the performance of the data updates in a database is to update and send the changes in batches to the database, rather than one by one.

Advantages of Batch Updates:

- Less network traffic is involved as data is sent in batches.
- A database might be able to execute some of the updates in parallel.

Q #20) What is the difference between Typed and Untyped Dataset?

Answer: **The differences are explained below:**

Typed Dataset: A typed dataset is derived from the Dataset class and has an associated XML schema, which is created at the time of creation of the dataset.

The XML schema contains information about the dataset structure such as tables, columns, and rows. Data is transferred from a database into a dataset and from the dataset to another component in the XML format.

Untyped Dataset: Untyped dataset doesn't have an XML schema associated with it. Untyped Dataset, the tables, and columns are represented as a collection.

Q #21) What are the rules to implement connection pooling?

Answer: **To establish a connection pooling, following rules must be followed:**

- The connection string must be the same for every User.
- The UserID must be the same for every user.

Q #22) What is a Datatable?

Answer: Datatable is a collection of rows, columns, and constraints. A Datatable is able to perform disconnected data access.

Q #23) What is DataView?

Answer: A DataView control creates a customized view of data stored in Datatable. The changes in DataView will affect the Datatable. By default, every table has a data view attached to it.

You can then add the DataView control and bind it to a Windows form. It allows the user to filter, navigate and sort the data in it.

Q #24) What are the major challenges in accessing data from a database?

Answer: **The challenges include:**

- More than one user might need to access the data simultaneously from one database.
- More than one user might need to access the data anytime, anywhere.

The solution to this problem is attained by implementing a **'Database locking'** during the time of transaction execution.

Q #25) What is meant by 'Transaction' in a database and what are the 'Properties of Transaction'?

Answer: Transaction can be defined as a series of operation. If the transaction is successful, all the data modifications performed in the database will be committed and saved.

Properties of Transaction

- Atomicity
- Consistency
- Isolation
- Durability

Atomicity: Consider a bank transaction in which an amount, say Rs 1000/- is withdrawn from the Account 'AB'.

If a system failure occurrs, the amount will rollback to the Account 'AB' itself. This property is termed as 'Atomicity'. This property states that either all the changes to the data are performed or none of them are performed.

This means that the transactions should be completed successfully or transactions should not begin at all.

Consistency: <u>For Example</u>, consider the bank transaction in which an amount of Rs 500/- is transferred from Account 'A' to Account 'B'. A system failure occurs and that Rs 500/- is reduced from Account 'A' and at the same time Rs 500/- is not credited to Account 'B'. Such a system is considered an **'Inconsistent'.**

'Consistency' states that the data in the system should be in a consistent state after a transaction is completed successfully, thus maintain the integrity of the same.

As per the above Example, the transaction will be rolled back to add Rs 500/- to Account 'A' and thus maintain system consistency.

Isolation: Consider the bank transaction process in which an Account 'A' is transferring Rs 500/- to Account 'B' and Account 'C' at the same time. Account 'A' having Rs 1500/- as total balance. For a 'Non-Isolated System' both transactions will read the balance for Account 'A' as Rs 1000/ instead of Rs 500/- which is incorrect.

'Isolation' states that the changes in data of a system made by one transaction must be isolated from the changes made by the other transaction.

For an 'Isolated System,' the first transaction will read a balance of Rs 1000/- and the second transaction will read a balance of Rs 500/- for Account 'A'.

Durability: Here it states that any change made in the data by a successful transaction will be permanent. Recovery management system is responsible to maintain the durability of a system.

Q #26) What are two types of transaction supported by ADO.net?

Answer:

Two types of Transaction supported by ADO.net

- Local Transaction
- Distributed Transaction

Local Transaction: A local transaction is based on a single data source. It will be handled directly by the database.

For Example: We import 'System.Data.SQL client' namespace, if we need to perform data transaction using Sqlserver. Similarly, we import 'System.Data.Oracle client' namespace, if we are using Oracle database.

Distributed Transaction: If the user needs to perform a transaction across multiple data Servers like SQL Server, Oracle etc., he can use a distributed transaction.

Q #27) What is the difference between DataReader and DataSet?

Answer: **Differences are as follows:**

DataReader	DataSet
Performance is better	Provides lower performance
It cannot modify data	It can modify data
It supports connected Architecture	It supports disconnected architecture
It has Read-only access	It has Read/Write access

DataReader	DataSet
It has faster access to data	It has slower access to data
It must be manually coded	Visual studio has toolset to support it

Conclusion

Hope you found this tutorial on most frequently asked interview questions on 'ADO.net.Practice to be much helpful. If you are clear with these questions, then you can easily get through any ADO.net interview.

Best of luck with your interview!!!

Recommended Reading

- Interview Questions and Answers
- 25 Best Agile Testing Interview Questions and Answers
- Spock Interview Questions with Answers (Most Popular)
- 25 Popular Test QA Analyst Interview Questions and Answers
- ETL Testing Interview Questions and Answers
- 20 Most Popular TestNG Interview Questions and Answers
- Top 30+ Popular Cucumber Interview Questions and Answers
- Top 50 Most Popular CCNA Interview Questions and Answers

About SoftwareTestingHelp

Helping our community since 2006! Most popular portal for Software professionals with **100 million+ visits!** You will absolutely love our tutorials on Software Testing, Development, Software Reviews and much more!

Join Over 300,000+ Followers!

-
-
-
-
-
-

Top FREE Training Tutorials

- Load Testing Buyer's Guide
- Appium Testing Tutorial
- Free QA Training
- Selenium Tutorials
- QTP/UFT Tutorials
- Quality Center QC Tutorials
- LoadRunner Tutorials
- JMeter Tutorials

Join Our Team!

Please upgrade to a supported browser to get a reCAPTCHA challenge.

Why is this happening to me?