✳ ADO . NET ✳

Front End Application:

*   The Application With Which End user Will Interact is Known as front End Application.

Ex: 1. Banking Application
2. Restaurant Application
3. ATM Application
4. SHIGGY Application.

front End tools :

The software that is used to develop front End Application is Known as front End tools.

Ex: 1. fox pro
2. MS access
3. Visual basic
4. Power builder
5. .net
6. Java
7. Phython

Back END Application :-

*   The Application Where orginal data is stored is Known as Back End Application.

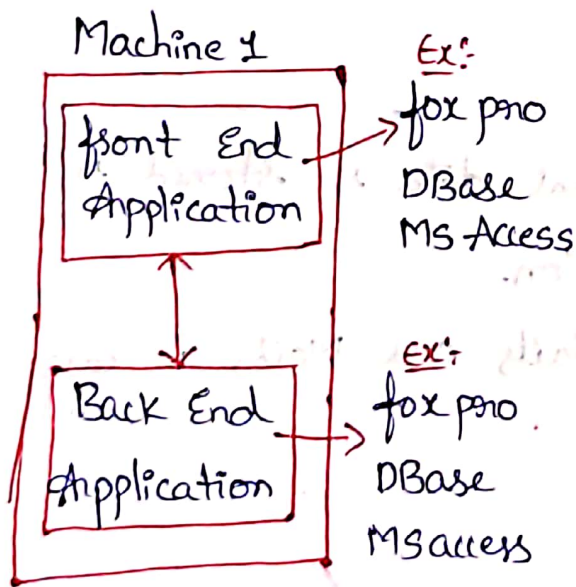Ex: Customer data, user details and Mails in case of Mailing Application.

Back End Tool:-

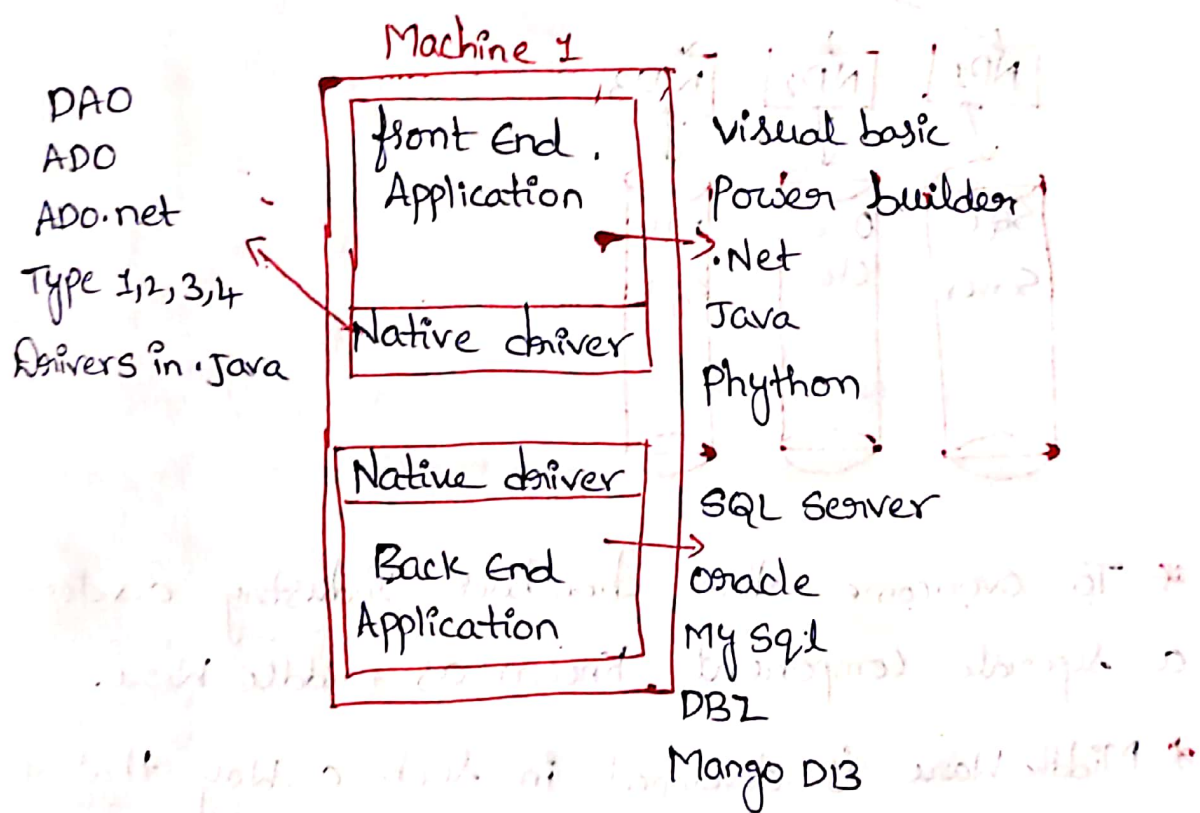The software that are used to develop back end Application is known as back end tools.

Ex:-
1. data base
2. fox pro
3. Ms-access
4. Sql Server
5. Oracle
6. My sql etc...

* In Early days of the Application development both frontend and back End Application were developed using same tool and is present in single machine. As both front End and back End Application were develop in same technology in single Machine no seperate translation Medium is required.

* front and backend were able to communicate with Each other Easily.

Machine 1

front End Application

Back End Application

Ex:-
fox pro
DBase
Ms Access

Ex:-
fox pro
DBase
Msaccess

* In the above scenario there was a drawback that when back End Application was required to connect from multiple front End Application of multiple client Machines. it was not possible. to overcome this drawback, so, both front End and back End Application were developed using different technologies and are present in two different Machines.
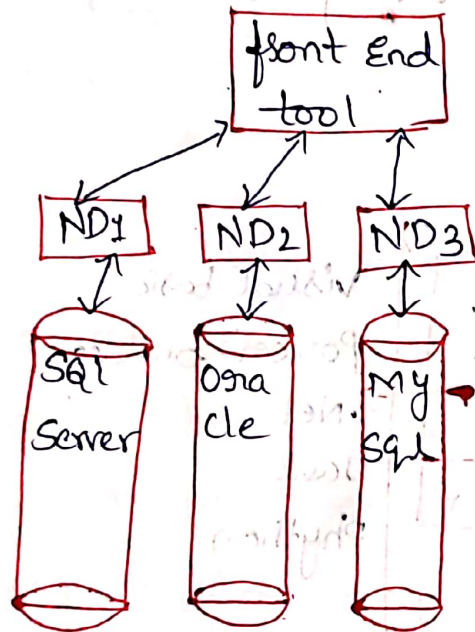
DAO
ADO
ADO.net

Type 1,2,3,4
Drivers in Java

Machine 1

| front End Application |
| Native driver |
| Native driver |
| Back End Application |

Visual basic
Power builder
>.Net
Java
Phython

SQL Server
Oracle
My Sql
DB2
Mango DB

* In this method both front End and backend Application are developed using different technologies front End Can't understand back End and backend Can't understand front End. So, for this purpose native drivers are developed as a translatory Medium.
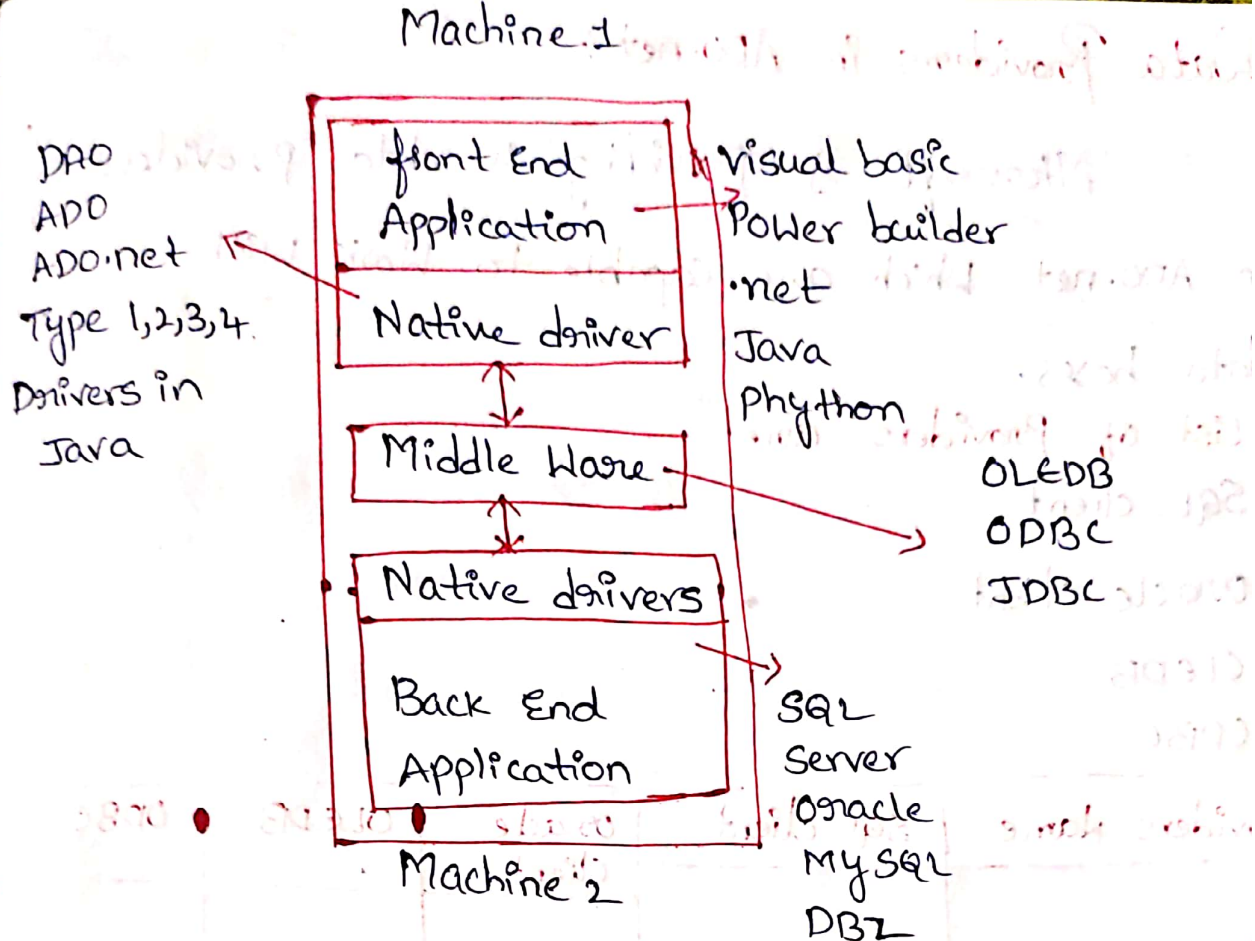
* But, there was a drawback faced with Native drivers that Every front End tool should have separate set of native drivers for Every back End tool similarly Every back End tool should have one separate set of native drivers to Interact with Every front End tool like:

* To overcome this drawback Industry developed a separate Component Known as middle Ware.

* Middle Ware is developed in such a way that it Can understand any front End tool and any back End tool.

Machine 1

DAO
ADO
ADO.net
Type 1,2,3,4.
Drivers in
Java

Front End Application	Visual basic
Native driver	Power builder
	.net
	Java
	Phython

Middle Ware

Native drivers

Back End
Application

Machine 2

OLEDB
ODBC
JDBC

SQL
Server
Oracle
My sql
DB2

ADO.net :-

* ADO.net is the Native driver developed by microsoft for dotnet. Which is used to interact with database of middle ware.

* ADO.net Means Active-x data active object for .net.

DAO → data active object

ADO → Active-x data object

There are

* The name space for .net is system. data.

Data Providers in ADO.net :

Microsoft is providing 4 data providers in ADO.net which one.. capable to work with data bases.

List of Providers are

1. SQL client
2. Oracle client
3. OLEDB
4. ODBC

Providers Name	SQL client	Oracle Client	OLEDB	ODBC
Name space →	system.data. SQL client	system. data. Oracle client	system. data.OLEDB	system.data ODBC!
Data base Connectivity →	Will Work with SQL server only	Will Work with oracle only	Will Work with Any database	Will Work with Any data base.
Code mechanism used →	Managed Code	Managed Code	unmanaged code	unmanaged Code
Execution →	faster	faster	slowest	slowest
Internal Architecture →	Easy	Easy	Complex	Most complex

Object Name :	Class(es) Names
1. Connection →	SQL Connection	Oracle Connection	OLEDB Connection	ODBC Connection.
2. Command →	SQL Command	Oracle Command	OLEDB Command	ODBC Command.
3. Data Reader →	SQL Data Reader	Oracle Data Reader	OLEDB Data Reader	ODBC Data Reader.
4. Data Adaptor →	SQL data Adopter	Oracle data Adopter	OLEDB data Adopter	ODBC data Adopter.
5. Transaction →	SQL Transaction	Oracle Transaction	OLEDB Transaction	ODBC Transaction.
6. Command Builder →	SQL Command builder	Oracle command builder	OLEDB command builder	ODBC Command Builder.
7. Parameter →	SQL parameter	Oracle parameter	OLEDB parameter	ODBC parameter.
8. Data set →	Data set
9. Connection string builder	

Working With Connection Object :-

* Connection object is responsible to Establish the connection with database.

* Other than Establishing connection to database. We can't use connection object for other purposes.

Methods Working With Connection object :-

(i) Close()

(ii) Change data base (string DBName)

(iii) Open()

(i) close :-

This Method is used to close and opened Connection to the data base

(ii) change :-

This Method is used to switch to other database after Establishing the Connection.

(iii) open :-

This Method is used to send a request from front end Application to database and used to Establish the Connection to the database.

Properties With Connection Object :-

Connection string

Connection Time out

Data Source

Data base

State

Connection string :-

Used to set or get required connection string value which is used to Establish Connection to the data base.

Connection Time out :-

Used to set or get the time period in second that front End Application should Wait once request is send to data base.

default time out period is 15 Seconds.
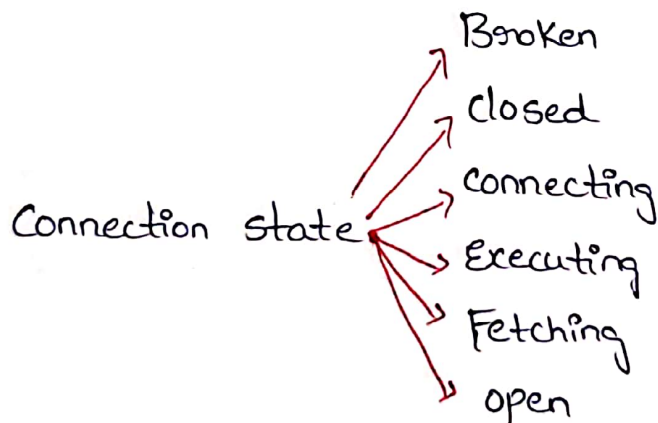
data Source :-

Used to set on get the data source name i.e. Server name to Which Connection is required to be Established.

data base :-

used to set or get the required database name to Which Connection is Established.

static :-

Used to get to Know What is the current state of the connection. all these states are available in an Enumeration Known as connection state like

Connection State → Broken
→ Closed
→ Connecting
→ Executing
→ Fetching
→ open

Example to Connect to sql Server data base :-

* Create a new Windows form application then design the form.

[Connect]

Write the following code:-

```
using.system.data.sql.client;

private void btn connect - click (object sender, Event
                                                    Args e)
{
    string sqlCon string = "server = SAI-BHANU\\SQL Server;
                            User Id = sa; password = abc@123;
                            database = Employee DB";
    sql connection.con = new sql connection (sql Con string);
    Con.open ();
        if (con.state == connection state.open)
            Message box.show ("connection is successful");
}
```

Connection string:-

Server = Server Name; user Id = User Name; password = user
Password; Database = database Name;