Nagaon Education Society's
GANGAMAI COLLEGE OF ENGINEERING (DHULE)
Nagaon, Dist- 05.
Dept of Computer

Name……

Year:…….……Branch/Course:.…….……...

Roll No:…….Expt No:……..

Date of Performance:……Date of Completion……

Sign and grade:

EXPERIMENT NO : 2

Aim: Implementation of ASP.net Application with Sql Server as back-end .

1. Objective: Development and deployment of ASP.net Application software.

2. Background:

ASP.NET is an open-source,[2] server-side web-application framework designed for web development to produce dynamic web pages. It was developed by Microsoft to allow programmers to build dynamic web sites, applications and services.

It was first released in January 2002 with version 1.0 of the .NET Framework and is the successor to Microsoft's Active Server Pages (ASP) technology. ASP.NET is built on the Common Language Runtime (CLR), allowing programmers to write ASP.NET code using any supported .NET language. The ASP.NET SOAP extension framework allows ASP.NET components to process SOAP messages.

ASP.NET's successor is ASP.NET Core. It is a re-implementation of ASP.NET asa modular web framework, together with other frameworks like Entity Framework. The new framework uses the new open-source .NET Compiler Platform (codename "Roslyn") and is cross platform. ASP.NET MVC, ASP.NET Web API, and ASP.NET Web Pages (a platform using only Razor pages) have merged into a unified MVC

ASP.net and SqlServer (ADO.net)

ADO.NET provides a bridge between the front end controls and the back end database. The ADO.NET objects encapsulate all the data access operations and the controls interact with these objects to display data, thus hiding the details of movement of data.

The following figure shows the ADO.NET objects at a glance

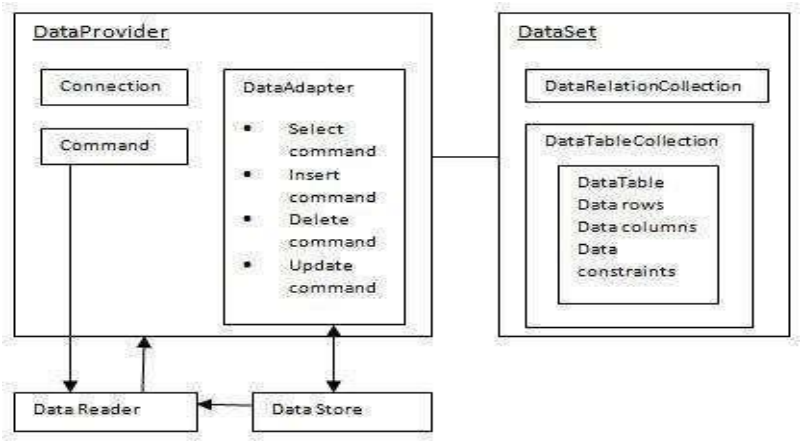


Fig :ADO.NET Objects

The DataSet Class

The dataset represents a subset of the database. It does not have a continuous connection to the database. To update the database a reconnection is required. The DataSet contains DataTable objects and DataRelation objects. The DataRelation objects represent the relationship between two tables.

Following table shows some important properties of the DataSet class:

Properties	Description
CaseSensitive	Indicates whether string comparisons within the datatables are case- sensitive.
Container	Gets the container for the component.
DataSetName	Gets or sets the name of the current data set.
DefaultViewManager	Returns a view of data in the data set.
DesignMode	Indicates whether the component is currently indesign mode.
EnforceConstraints	Indicates whether constraint rules are followed whenattempting any update operation.
Events	Gets the list of event handlers that are attached to thiscomponent.
ExtendedProperties	Gets the collection of customized user informationassociated with the DataSet.

HasErrors	Indicates if there are any errors.
IsInitialized	Indicates whether the DataSet is initialized.

Locale	Gets or sets the locale information used to compare stringswithin the table.
Namespace	Gets or sets the namespace of the DataSet.
Prefix	Gets or sets an XML prefix that aliases the namespace of theDataSet.
Relations	Returns the collection of DataRelation objects.
Tables	Returns the collection of DataTable objects.

The following table shows some important methods of the DataSet class:

Methods	Description
AcceptChanges	Accepts all changes made since the DataSet was loaded or this methodwas called.
BeginInit	Begins the initialization of the DataSet. The initialization occurs atrun time.
Clear	Clears data.
Clone	Copies the structure of the DataSet,including all DataTable schemas, relations, and constraints. Does not copy any data.
Copy	Copies both structure and data.
CreateDataReader()	Returns a DataTableReader with oneresult set per DataTable, in the samesequence as the tables appear in the Tables collection.
CreateDataReader(DataTable[])	Returns a DataTableReader with oneresult set per DataTable.

EndInit	Ends the initialization of the data set.
Equals(Object)	Determines whether the specified Object is equal to the current Object.
Finalize	Free resources and perform othercleanups.
GetChanges	Returns a copy of the DataSet with all changes made since it was loadedor the AcceptChanges method was called.
GetChanges(DataRowState)	Gets a copy of DataSet with all changes made since it was loaded or the AcceptChanges method was called,filtered by DataRowState.
GetDataSetSchema	Gets a copy of XmlSchemaSet forthe DataSet.
GetObjectData	Populates a serialization informationobject with the data needed to serialize the DataSet.
GetType	Gets the type of the current instance.
GetXML	Returns the XML representation of thedata.
GetXMLSchema	Returns the XSD schema for the XMLrepresentation of the data.
HasChanges()	Gets a value indicating whether the DataSet has changes, including new,deleted, or modified rows.
HasChanges(DataRowState)	Gets a value indicating whether the DataSet has changes, including new,deleted, or modified rows, filtered by DataRowState.

IsBinarySerialized	Inspects the format of the serializedrepresentation of the DataSet.
Load(IDataReader, LoadOption, DataTable[])	Fills a DataSet with values from a datasource using the supplied IDataReader,using an array of DataTable instances to supply the schema and namespace information.
Load(IDataReader, LoadOption, String[])	Fills a DataSet with values from a datasource using the supplied IDataReader,using an array of strings to supply the names for the tables within the DataSet.
Merge()	Merges the data with data from another DataSet. This method has different overloaded forms.
ReadXML()	Reads an XML schema and data into the DataSet. This method has differentoverloaded forms.
ReadXMLSchema(0)	Reads an XML schema into the DataSet. This method has different overloaded forms.
RejectChanges	Rolls back all changes made since thelast call to AcceptChanges.
WriteXML()	Writes an XML schema and data from the DataSet. This method has differentoverloaded forms.
WriteXMLSchema()	Writes the structure of the DataSet as an XML schema. This method has different overloaded forms.

The DataTable Class

The DataTable class represents the tables in the database. It has the following important properties; most of these properties are read only properties except thePrimaryKey property:

Properties	Description
ChildRelations	Returns the collection of child relationship.
Columns	Returns the Columns collection.
Constraints	Returns the Constraints collection.
DataSet	Returns the parent DataSet.
DefaultView	Returns a view of the table.
ParentRelations	Returns the ParentRelations collection.
PrimaryKey	Gets or sets an array of columns as the primary key for thetable.
Rows	Returns the Rows collection.

The following table shows some important methods of the DataTable class:

Methods	Description
AcceptChanges	Commits all changes since the last AcceptChanges.
Clear	Clears all data from the table.
GetChanges	Returns a copy of the DataTable with all changes made sincethe AcceptChanges method was called.
GetErrors	Returns an array of rows with errors.
ImportRows	Copies a new row into the table.
LoadDataRow	Finds and updates a specific row, or creates a new one, ifnot found any.
Merge	Merges the table with another DataTable.
NewRow	Creates a new DataRow.
RejectChanges	Rolls back all changes made since the last call toAcceptChanges.
Reset	Resets the table to its original state.
Select	Returns an array of DataRow objects.

The DataRow Class

The DataRow object represents a row in a table. It has the following important properties:

Properties	Description
HasErrors	Indicates if there are any errors.
Items	Gets or sets the data stored in a specific column.
ItemArrays	Gets or sets all the values for the row.
Table	Returns the parent table.

Methods	Description
AcceptChanges	Accepts all changes made since this method was called.
BeginEdit	Begins edit operation.
CancelEdit	Cancels edit operation.
Delete	Deletes the DataRow.
EndEdit	Ends the edit operation.
GetChildRows	Gets the child rows of this row.
GetParentRow	Gets the parent row.
GetParentRows	Gets parent rows of DataRow object.
RejectChanges	Rolls back all changes made since the last call to AcceptChanges.

The DataAdapter Object

The DataAdapter object acts as a mediator between the DataSet object and the database. This helps the Dataset to contain data from multiple databases or other data source.

The DataReader Object

The DataReader object is an alternative to the DataSet and DataAdapter combination. This object provides a connection oriented access to the data records in the database. These objects are suitable for read-only access, such as populating a list and then breaking the connection.

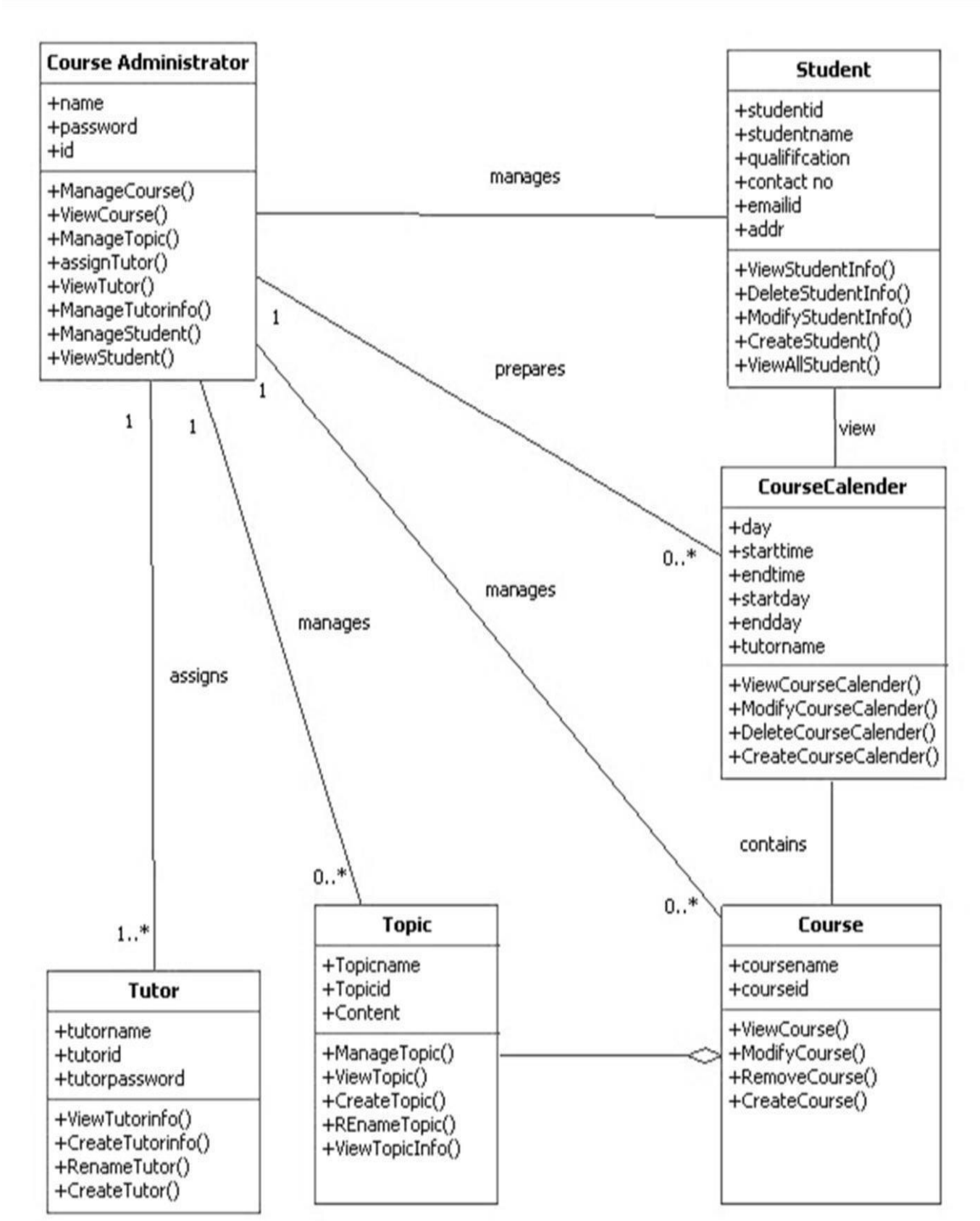
DbCommand and DbConnection Objects

The DbConnection object represents a connection to the data source. Theconnection could be shared among different command objects.

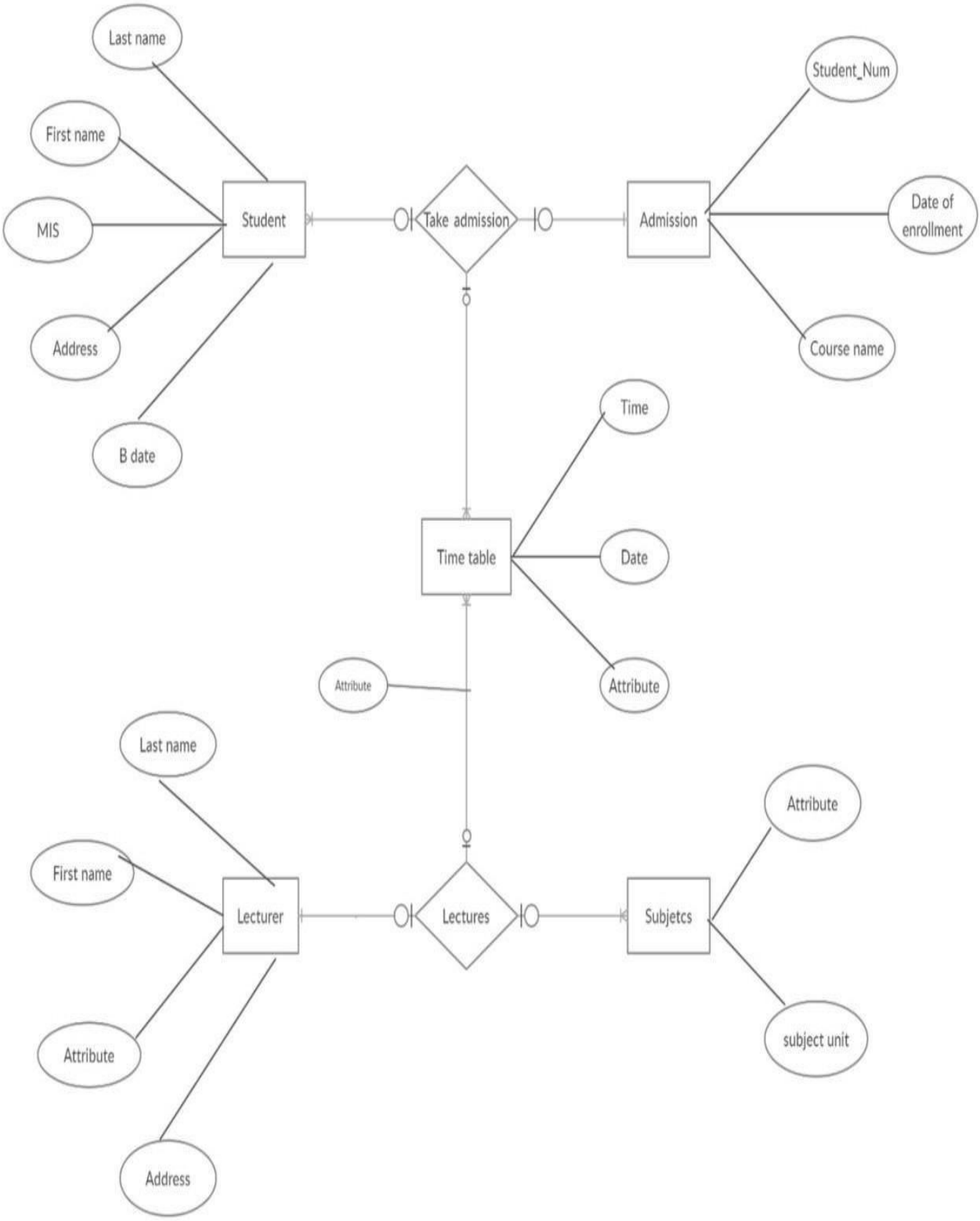
The DbCommand object represents the command or a stored procedure sent tothe database from retrieving or manipulating data.

3. Pre-lab Task

1 Use case diagram for college management system.



1.E-R diagram for College management system



2. Post Lab Task

3. Login Form

Login Form Asp.net Code:

Database

Output:

Database Output:

Outcomes:

1. Creates ASP.net Web Forms
2. Created ADO.net programs that use various library functions, and that manipulate database.

Questions:

1. What is ADO.NET?
2. What is the ADO.NET components?
3. How can you define data structure in ADO.NET?
4. What is connection pooling in ADO.NET?
5. What is the difference between data leader and data set?