

.Net Programming using C#

Unit IV

ADO.NET Programming

Architecture of ADO.NET

.Net Framework Data Provider

Connection

Transaction

Command

Parameters

DataReader

Data Adapter

SelectCommand

InsertCommand

UpdateCommand

DeleteCommand

DataSet

DataTableCollection

DataTable

DataRowCollection

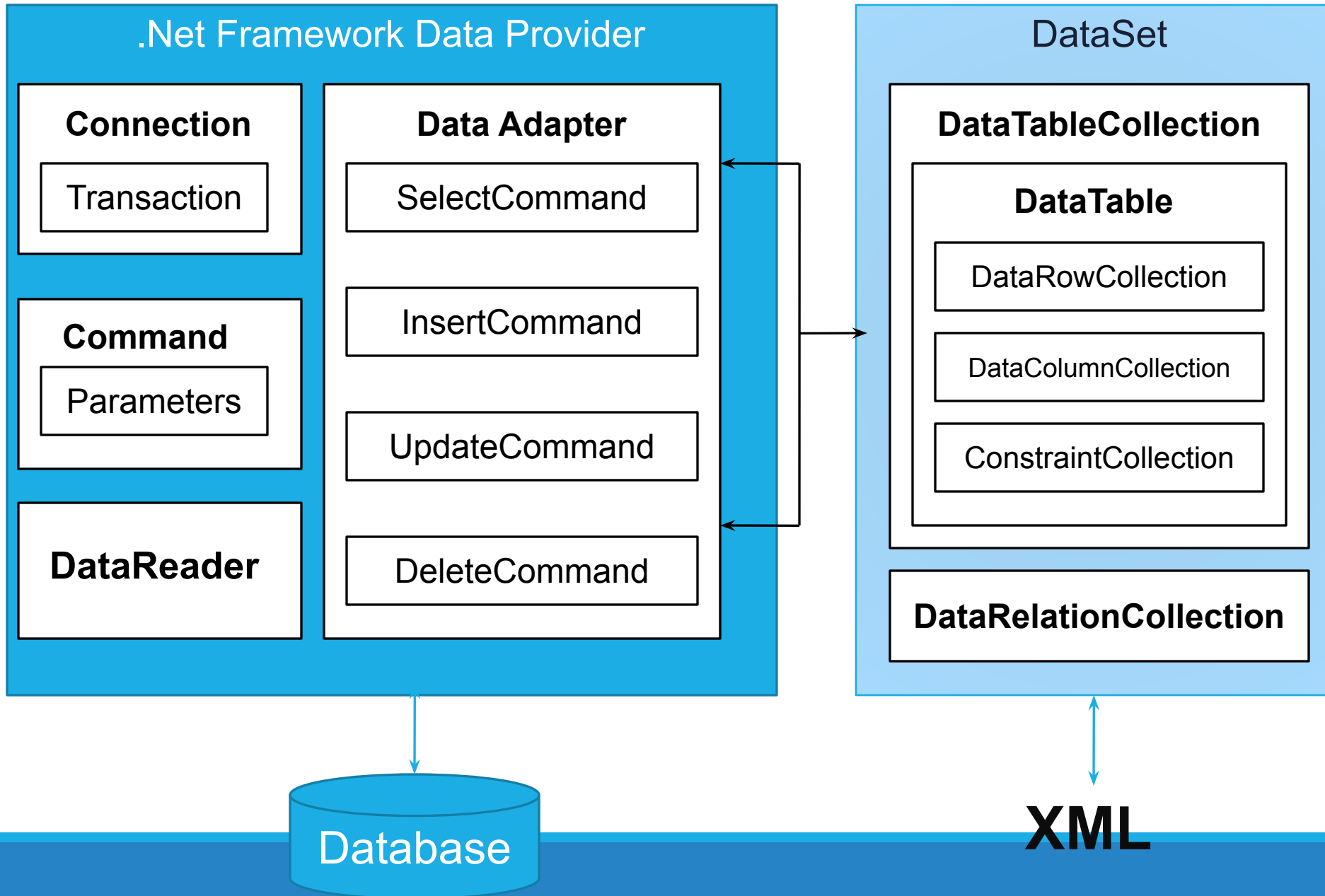
DataColumnCollection

ConstraintCollection

DataRelationCollection

Database

XML



Connected Environment

SqlConnection

SqlCommand

SqlDataReader

SqlConnection Class

Constructor

- SqlConnection(string connectionString)

Properties

- ConnectionString
- State
- ConnectionTimeout

Methods

- Open()
- Close()

SqlCommand Class

Constructor

- `SqlCommand(String cmdText, SqlConnection conn)`

Properties

- `CommandText`
- `Connection`

Methods

- **`SqlDataReader`** `ExecuteReader()`
- `int ExecuteNonQuery()`
- `object ExecuteScalar()`

SqlDataReader Class

Properties

- HasRow
- FieldCount
- this[int]
- this[string]

Methods

- bool Read()
- void Close()

CRUD

CRUD represents an acronym for the database **operations** Create, Read, Update, and Delete.

The communication between two layers could be in the form of ad hoc **SQL** statements such as INSERT, SELECT, UPDATE, and DELETE.

Disconnected Environment

SqlDataAdapter

Properties

- InsertCommand
- DeleteCommand
- SelectCommand
- UpdateCommand

Methods

- Fill(DataSet ds)

- Fill(DataSet ds, String name)
- Fill(DataTable table)
- Fill(DataRow[] rows)

- Update(DataSet ds)
- Update(DataSet ds, String name)
- Update(DataTable table)
- Update(DataRow[] rows)

DataSet

Properties

- Tables
 - Tables.Count
 - Tables[0].TableName
 - Tables[0].Rows.Count
 - Tables[0].Columns.Count

Methods

- `void Clear()`
- `DataRow Tables[0].NewRow()`
- `DataRow[] Tables[0].Select(string filter)`
- `void Tables[0].Rows.Add(DataRow row)`
- `void Tables[0].Rows.Remove(DataRow row)`
- `void Tables[0].Rows.RemoveAt(int index)`

DataTable

Properties

- Rows.Count
 - Columns.Count
 - TableName
-

Methods

- void Clear()
- DataRow[] Select(string filter)
- DataRow NewRow()
- void Rows.Add(DataRow row)
- void Rows.Remove(DataRow row)
- void Rows.RemoveAt(int index)

DataRow

Properties

- `this[int]`
- `this[string]`

Methods

- `public void BeginEdit()`
- `public void EndEdit()`
- `public void Delete()`

DataColumn

Properties

- ColumnName
- DataType

DataGrid View Control

Properties

- Columns
 - Add(string ColumnName, string HeaderText)
 - Remove(string ColumnName)
 - RemoveAt(int index)
- Rows
 - Add(params object[] value)
 - Remove(DataGridViewRow row)
 - RemoveAt(int index)
- DataSource

Events

- CellContentClick
- SelectionChanged

Report in .CSV

- CSV stands for "**comma-separated values/ character-separated values**". Its data fields are most often separated, or delimited, by a comma.
- **CSV** is a simple file format used to store tabular data, such as a spreadsheet or database.

- For example, let's say you had a spreadsheet containing the following data.

Name	Class	Dorm	Room	GPA
Sally Whittaker	2018	McCarren House	312	3.75
Belinda Jameson	2017	Cushing House	148	3.52
Jeff Smith	2018	Prescott House	17-D	3.20
Sandy Allen	2019	Oliver House	108	3.48

Sally Whittaker,2018,McCarren House,312,3.75

Belinda Jameson,2017,Cushing House,148,3.52

Jeff Smith,2018,Prescott House,17-D,3.20

Sandy Allen,2019,Oliver House,108,3.48

```
BindReport(ReportDocument crystalReport)
{
    crystalReport.Load("CustomerReport.rpt");
    SqlDataAdapter adp = new
    SqlDataAdapter("select * from      stud",
    conn);
    adp.Fill(ds);
    crystalReport.SetDataSource(ds);
    CrystalReportViewer1.ReportSource =
    crystalReport;
}
```

```
switch (formate_selection)
{
    case "Word":
        formatType = ExportFormatType.WordForWindows;
        break;
    case "PDF":
        formatType = ExportFormatType.PortableDocFormat;
        break;
    case "CSV":
        formatType =
ExportFormatType.CharacterSeparatedValues;
        break;
}
    crystalReport.ExportToHttpResponse(formatType,
Response, true, "Crystal");
}
```