There are several ways to work with Excel, no one method is proper for all situations which is important to know. In short why use Excel automation to read cells while OleDb data provider gives you what is needed for most operations such as reading, edits, adds and deletes. At the same time a simple operation such as removal of a WorkSheet is not supported with OleDb data provider so this time using Excel automation, a third party library or Open Office is the way to go.

I am focusing on working with OleDb and provide examples to get the job done or for you to have a base to work through more complex operations. Once you understand the basics it will be easy to move forward. Below I am providing overviews for each project but in several there is much more going on to learn from.

Note: Each project has a MS-Word document which gives an overview for what the project is demonstrating.

Note: I make no attempt at writing in depth on everything done in this article as that would a) be an extremely long article b) break down the article into many parts.

The first thing to go over is OleDbHelper class project. This project has a class named Connections which is responsible for setting up connections to Excel files which is used in the majority of the form projects. If you get the connection wrong than down the road things will not work or present data improperly. A good example is we have a sheet that the first row is data but we configure the connection to see the first row has column names. If this happens we basically lose the first row of data. In reverse if we configure the connection for the first row as column names and the first row is data we lose a row just the same. Next up is IMEX portion of the connection string. If IMEX is not set properly than modifications to the underlying sheet or sheets will fail and the exception/error message might point to a invalid SQL statement which is not the case. Instead of going on and on about each setting it is best to work through the working examples to learn about the settings. The second class has methods to retrieve sheet names from a file and also a function to see if a sheet exists or not.

In the first project, Demo1 I show how to open a worksheet from a collection of sheets and control if the first row is data or column headers.

Demo2 project shows how to selectively populate a DataTable with all column or selected columns from a CheckedListBox where the sheet is hard coded to a sheet that has the first row populated with column names.

Demo3 project shows how to alias column names from reading a sheet where there are no column names. In this case each column is known as Fn were n in the ordinal index of the column. The majority of code in this project is validating the alias process that the user has entered somewhat valid column names and as protection we use a) brackets around the names b) wrap the reading of data in aTry-Catch statement.

Demo4 project is to show reading a single cell's data and also a range of cells data. The DataGridView has pre-defined columns and these columns have their data property set to a column in the DataTable. Note the sheet has a connection for no header and this sheet has headers but it does not matter as we are only dealing with specific cells.

Demo5 project is a glimmer at working with mixed data types in a single column. By no means is this the end all for working with mixed types but does show some hoops one may need to leap through for filtering out data.

Demo6 project demonstrates how to create new sheet via OleDb and populate it. Please note you cannot remove a sheet via OleDb but can using Office automation and since this solution revolves around OleDb I am not including a Office automation solution but if you need it feel free to ask.

Demo7, although not the best way to update an Excel cell or cells this project shows how to update a cell based on the contents of another cell.

**Note:** In this project module1.vb physically resides in the project Demo6. It was added here via add -> link (selected via the drop-down of the add button)

WithAdapter project shows the basics of working with a DataAdapter. There are many complications with using an Adapter coupled with no primary key so I would advise not going this route but decided to include this as some will wonder "how can I do this?"

Also note this project was done years before the other projects so the style of code will be slightly different from the other projects in this solution but works just fine.