

PROJECT FILE-3

Data Visualization Through Power BI

Submitted by:

RAJESH BAGHEL

9319652493

Connecting and Shaping Data

We have learned about the basics of the Power BI interface. However, to truly unlock the power of Power BI, we need to expand our data model. In order to do that, we need to learn about the query editor and how to relate multiple tables of data to one another and

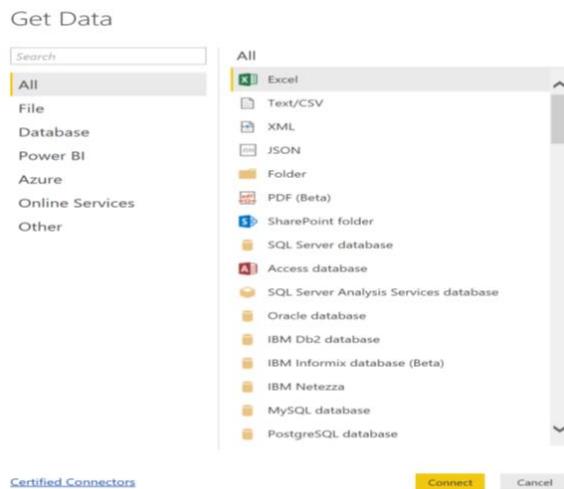
create a more complex data model. Every good visual report starts with a good data model, so it is imperative that we learn how to properly ingest, transform, and load our data into Power BI.

The following topics will be covered in this chapter:

- Getting data
- Transforming data
- Merging, copying, and appending queries
- Verifying and loading data

To create a query, we will implement the following steps:

1. Choose **Get Data** from the **Home** tab of the ribbon. Note the default list of potential data sources and select **More...** at the bottom of the list:



There are over 100 connectors available for ingesting data. These connectors are broken down into a number of categories:

All: This lists all of the available connectors.

File: File connectors, including Excel, Text/CSV, XML, JSON, Folder, PDF, and SharePoint folders.

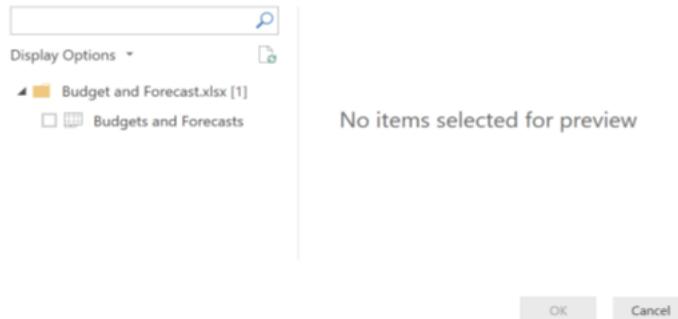
Database: The database lists sources such as SQL Server, Access, Oracle, IBM DB2, IBM Informix, IBM Netezza, MySQL, PostgreSQL, Sybase, Teradata, SAP, Impala, Google BigQuery, Vertica, Snowflake, Essbase, and AtScale.

Power BI: Power BI includes Power BI datasets and dataflows.

2. Click back on the **All** category and select the first item in the list, **Excel**.

3. Browse to your Budgets and Forecasts.xlsx file and click **Open**. This dialog is known as the **Navigator**

Navigator



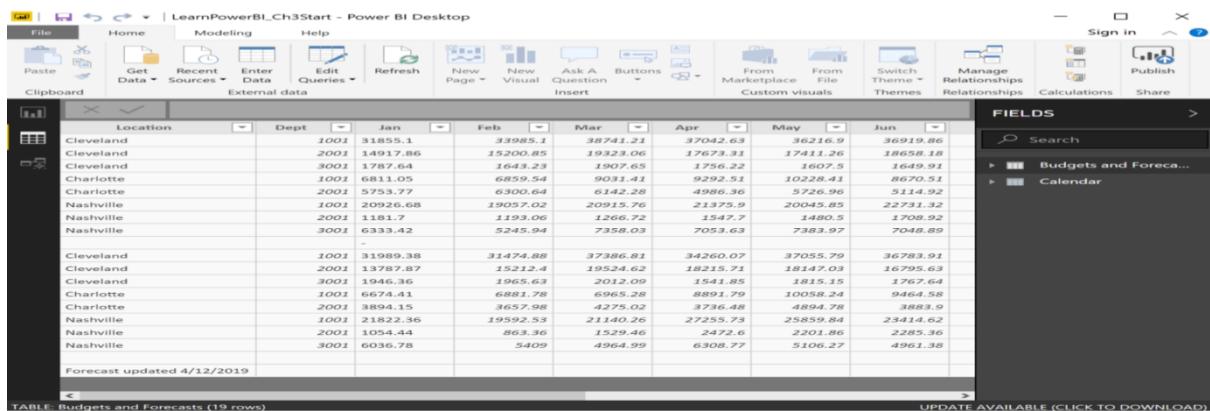
For Excel files, the **Navigator** dialog will list all the pages and tables within the Excel file.

- Click the checkbox next to **Budgets and Forecasts** and observe that a preview of the data is loaded in the right-hand pane:



Note that three options are available: **Load**, **Edit** or **Transform Data**, and **Cancel**. Depending on your version, the **Edit** button may read **Transform Data**.

- Click the **Load** button. Once the loading dialog completes, click on the **Data** view and then the **Budgets and Forecast** table to see this data loaded into the data model:



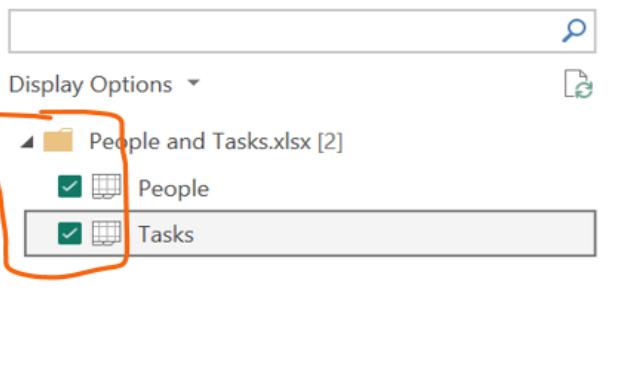
Getting additional data

Let's get some additional data. Since we are tracking PTO for people, we need to have some data about the employees at the company, as well as the tasks that they are performing. In order to get this data, perform the following actions:

1. From the **Home** tab of the ribbon, choose **Get Data** and then **Excel**.
2. Choose the People and Tasks.xlsx file and click the **Open** button.
3. Check both the **People** and **Tasks** pages and click the **Load** button.

Exploring this data back in the **Data** view of Power BI Desktop, we can see that we now have two additional tables, **People** and **Tasks**.

Navigator



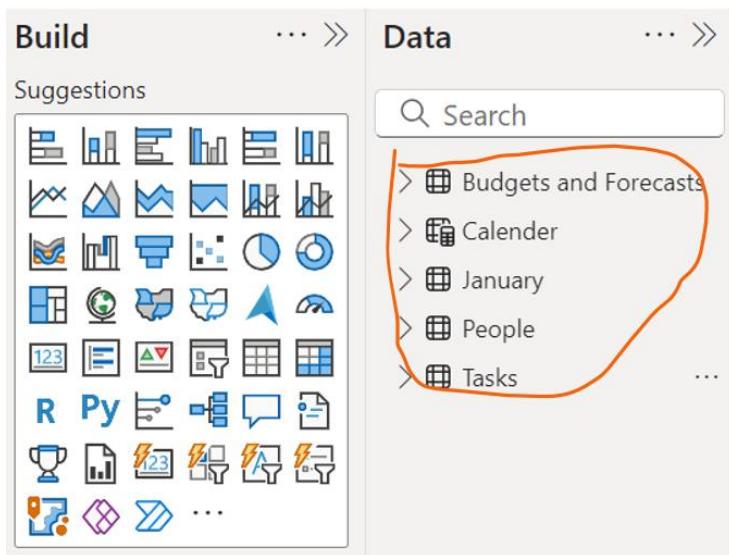
The screenshot shows the Power BI Navigator dialog. On the left, under 'Display Options', there is a tree view of the 'People and Tasks.xlsx' file. Two items are selected: 'People' and 'Tasks', which are highlighted with a red box. On the right, a preview of the 'Tasks' table is shown with the following data:

Column1	Column2
TaskID	Category
PTO	PTO
INTERNAL	Int Admin
1001TM	Billable
1001TB	Billable
2001TM	Billable
2001TV	Billable

The **People** table contains information about the employees of the company. Looking at the **People** table, we can see that we have information such as **ID**, **Name**, **Title**, **Employee Type**, **TermDate**, **HireDate**, and **Location**.

The final data that we need is the actual hours that employees are working. To do this, execute the following tasks:

1. Click on the **Home** tab of the ribbon, choose **Get Data**, and then **Excel**.
2. This time, choose the Hours.xlsx file and click the **Open** button.
3. In the **Navigator** dialog, this time, just choose **January** and click **Load**.

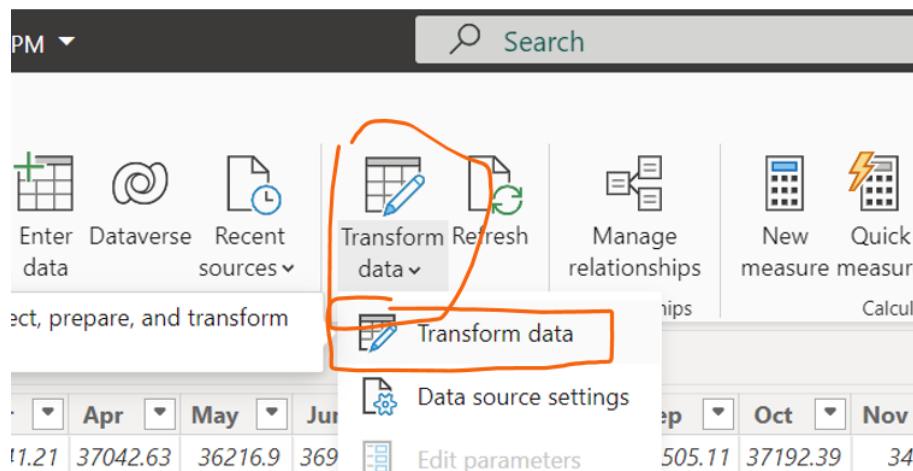


Transforming data

While Power BI did a good job of automatically identifying and categorizing our data, the data is not entirely in the format we will require for analysis. Therefore, we need to modify how the data gets loaded into the model. In other words, we need to transform the data. To do that, we will use a powerful sub-application: the **Power Query Editor**.

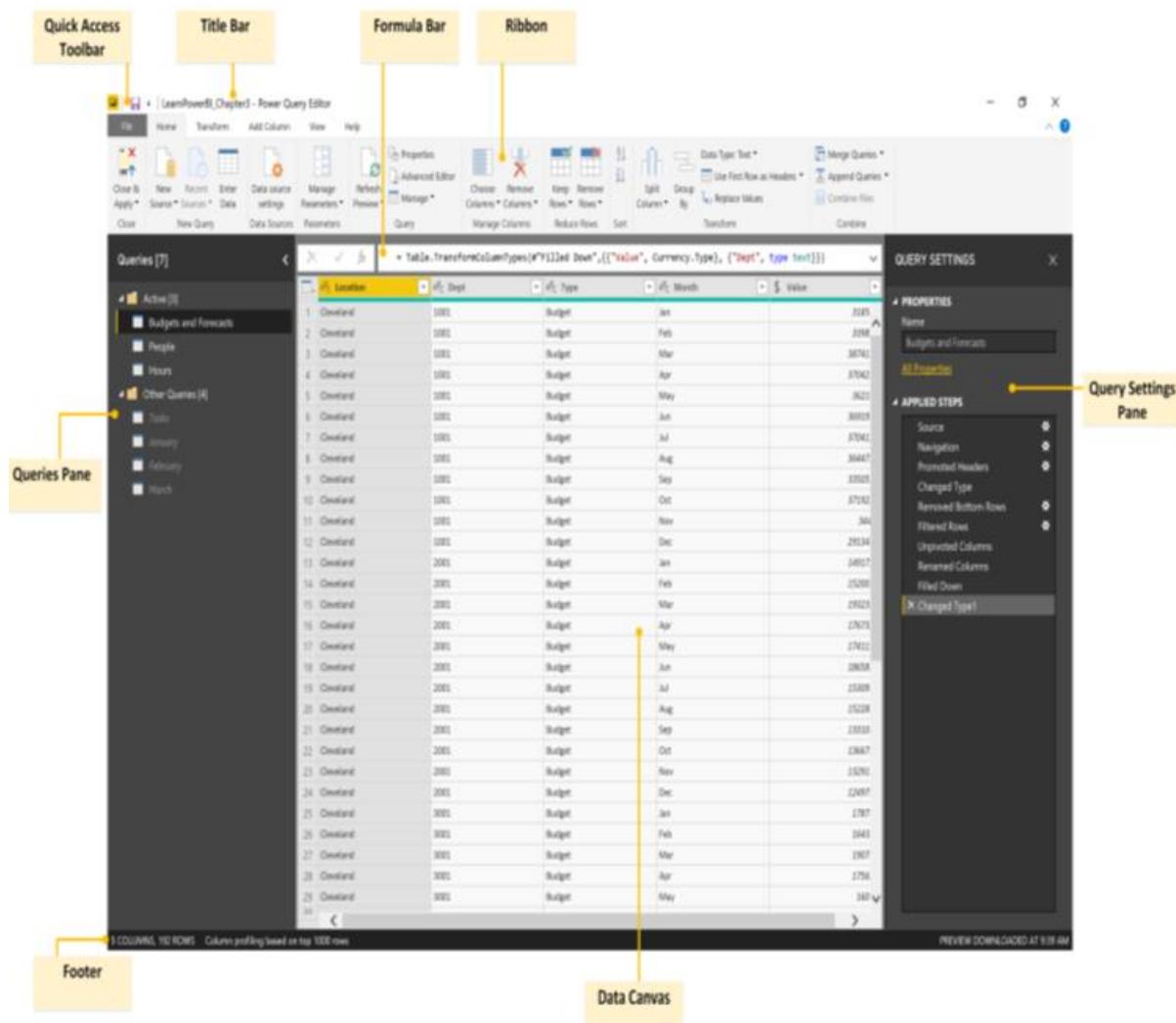
Touring the Power Query Editor

The **Power Query Editor** is launched from the **Home** tab of the ribbon by clicking Transform



Once launched, the following screen will be displayed:

As you might expect, the **Power Query Editor** interface is similar to, and shares common elements with, the Desktop. The **Power Query Editor** user interface comprises eight areas. We will cover these eight areas in brief in the sections that follow. You can refer to the preceding screenshot while going through the eight areas.



Transforming budget and forecast data

1. In the Power Query window, select **Budgets and Forecasts** in the **Queries** pane on the left. Looking at the data, we can see that we have a blank row in the middle and several extraneous rows at the end. Looking at the column headers, we notice that Power BI has identified that our first row contains column names and has already categorized our columns as text (**ABC**), whole numbers (**123**), and decimal (**1.2**). The table contains **Location**, **Department** and then a column for each month. As we scroll to the end, we see a column called **Type**.

2. In the **Query Settings** pane, under **Applied Steps**, we see that some steps have already been applied in our query. These steps were automatically created by Power BI.

A query is really just a collection of applied steps. If we click on these steps, we can see how our query changes our data table at each step.

3. Clicking on the **Navigation** step, we can see that our column headers are labelled **Column1**, **Column2**, and so on.

4. When we click on the **Promoted Headers** step, we can see Or use (Home – Use first row as header) that the first row has been promoted to be column headers, but that all of our columns are labelled ABC123.

5. Clicking on the **Changed Type** step, we can see that this is where our columns are categorized according to their data types.

Cleaning up extraneous bottom rows

Let's clean up those extraneous rows at the end. To do that, follow these steps:

1. Ensure that the last step, **Changed Type**, is selected.

2. From the ribbon, select the **Home** tab and then the **Remove Rows** button in the **Reduce Rows** section. Select **Remove Bottom Rows**

The screenshot shows the Power Query Editor interface. The ribbon is visible at the top with tabs for File, Home, Transform, Add Column, View, and Help. The Home tab is selected. On the far right of the ribbon, there is a 'Sort' dropdown menu with options A↓, Z↓, A↑, and Z↑. Below the ribbon, the 'Queries [4]' pane on the left lists 'Budgets and Forecasts', 'People', and 'Tasks'. The main area displays a table with three columns: 'Location' (containing 'Cleveland'), 'Dept' (containing '1001'), and a third column (containing '2001'). Above the table, the formula bar shows '= Table.PromoteHeaders(#"Budgets")'. To the right of the table, there is a 'Remove Rows' button with a dropdown menu. The menu includes options: Remove Top Rows, Remove Bottom Rows (which is highlighted in yellow), Remove Alternate Rows, Remove Duplicates, Remove Blank Rows, and Remove Errors. The 'Remove Bottom Rows' option is located in the second row of the dropdown menu.

3. In the following dialog, type **2** and then click the **OK** button:

Remove Bottom Rows

Specify how many rows to remove from the bottom.

Number of rows

2

OK **Cancel**

Now we can observe that the last two rows have been removed and that an additional step, **Removed Bottom Rows**, has been added to the bottom of our query steps in the **Applied Steps** area

Filtering rows:

Now, let's clean up that mostly blank row in the middle, row 9. To do this, follow these steps:

1. In the Location column header, click the drop-down arrow and you will notice that there are a number of options, including sorting, text filters, and a search bar. The Text Filters area presents a number of useful text filtering options, including filters such as Equals, Does Not Equal, Begins With, Does Not Begin With, Ends With, Does Not End With, Contains, and Does Not Contain. Also, note that all of the distinct values that appear in the column are listed, including (null), Charlotte, Cleveland, and Nashville:

The screenshot shows the Power Query Editor interface with the following details:

- File** tab is selected.
- Home** tab is active.
- Transform** tab is visible.
- Add Column**, **View**, and **Help** tabs are also visible.
- Queries [4]** pane shows four queries.
- Location** column is selected, indicated by a yellow header.
- Text Filters** dropdown menu is open, showing filter options for the column.
- Search** input field is present.
- Filter list** shows:
 - (Select All)
 - (null)
 - Charlotte (selected)
 - Cleveland (selected)
 - Nashville (selected)
- OK** and **Cancel** buttons are at the bottom of the filter dialog.
- Preview** pane shows the data with the last two rows removed.

Uncheck the box next to **(null)** and then click the **OK** button. Notice that the row of null values in the data table has been removed and that a new step has been added to the query, **Filtered Rows**. Also, notice that a small funnel icon appears in the **Location** column's header button, indicating that this column has been filtered.

Unpivoting data

Let's put this data into a better format for analysis. Since the numeric data that we will need to analyze appears in multiple columns (the month columns), this will make analysis difficult. It would be much easier to analyze if we transformed this data so that all of the numeric data was in a single column. We can do this by unpivoting our month columns. To unpivot our month columns, perform the following steps:

1. Start by selecting the column header for **Jan**.
2. Scroll to the right until you see the last column, **Type**.
3. Hold down the *Shift* key and then select the column header labeled **Dec**. Now, all of the columns between and including **Jan** through **Dec** will be highlighted in yellow and selected.
4. Click on the **Transform** tab and choose **Unpivot columns**:

The screenshot shows the Power BI Query Editor interface. On the left is a data grid with three columns: Type, Attribute, and Value. The data consists of 13 rows, starting with 'Budget' and ending with 'null'. The 'Attribute' column contains months from Jan to Dec. On the right is the 'QUERY SETTINGS' pane. Under 'APPLIED STEPS', the 'Unpivot Other Columns' step is listed with a gear icon, indicating it is active. The 'Properties' section shows the name 'Budgets and Forecasts'.

Type	Attribute	Value
Budget	Jan	31855.1
Budget	Feb	33985.1
Budget	Mar	38741.21
Budget	Apr	37042.63
Budget	May	36216.9
Budget	Jun	36919.86
Budget	Jul	37041.17
Budget	Aug	36447.85
Budget	Sep	33505.11
Budget	Oct	37192.39
Budget	Nov	34401
Budget	Dec	29134.96
null	Jan	14917.86
null	Feb	15200.85
null	Mar	19323.06
null	Apr	17673.31
null	May	17411.26
null	Jun	18658.18
null	Jul	15309.93
null	Aug	15228.32

Using Fill

We now want to fix our **Type** data. Notice that the **Type** column contains spotty information. The first twelve rows contain the word **Budget**, and then there is a gap until row 97, where the next twelve rows contain the word **Forecast**, and then another gap. We want all of these rows to contain either **Budget** or **Forecast**. To fix the data, we will use the Fill functionality by performing the following steps:

1. Select the **Type** column, then right-click and choose **Fill**, and then **Down**.
2. Notice that our **Type** column now contains a value for each row, and a **Filled Down** step appears in our **Applied Steps** area. The Fill operation takes the latest value found in the column and replaces any blank or null values with that value until a new value is found and the operation repeats.

As shown in the preceding screenshot, our month columns have been transformed into two columns, **Attribute** and **Value**. Our former column names, **Jan**, **Feb**, **Mar**, **Apr**, **May**, and so on, now appear as row values under the column called **Attribute**, and our numeric values appear in a column called **Values**. The step, **Unpivoted Columns**, appears in our **Applied Steps** area.

5. Double-click the **Attribute** column header and rename this column to Month. A **Renamed Columns** step will appear in the **Applied Steps** area.

Changing data types

The **ABC123** label means that the values in the column can be either text or numeric. To fix the data type, perform the following steps:

1. Click the **ABC123** icon and choose **Fixed decimal number**. The **ABC123** icon is replaced by a **\$** and a **Changed Type1** step is added to **Applied Steps**.
2. Similarly, click the **123** icon in the **Dept** header and change this to **Text**. Notice that the **123** icon is replaced by an **ABC** icon and that the values in the column are no longer italicized and are left-justified versus right-justified. While our department codes may be numeric, we do not want to ever sum, average, or otherwise aggregate these values numerically.

Transforming people, tasks, and January data

. We will perform similar operations on the **People**, **Tasks**, and **January** queries, like we did in the *Transforming budget and forecast data* section.

To transform the **People** query, performing the following steps:

1. Click on the **People** query in the **Queries** pane. Notice that four steps have already been created in this query, that is, **Source**, **Navigation**, **Promoted Headers**, and **Changed Type**. Power BI has automatically added a number of query steps to make the first row the column names and identify the data types of the columns. The first four columns, that is, **ID**, **Name**, **Title**, and **Employee Type**, have all been identified as text (**ABC**). The next two columns,

	Title	EmployeeType	TermDate	HireDate	Location
1	CONSULTANT	CONSULTANT	01-01-1900	01-01-1900	Charlotte
2	CONSULTANT	SALARY	15-05-2015	01-01-1900	Charlotte
3	SALES	ADMINISTRATION	01-01-1900	01-01-1900	Charlotte
4	ADMIN	ADMINISTRATION	01-01-1900	01-01-1900	Charlotte
5	ADMIN	ADMINISTRATION	29-11-2018	01-01-1900	Charlotte
6	CONSULTANT	SALARY	01-01-1900	01-01-1900	Charlotte
7	CONSULTANT	CONSULTANT	18-11-2005	17-04-2000	Charlotte
8	CONSULTANT	SALARY	08-03-2019	08-10-2001	Charlotte
9	ADMIN	ADMINISTRATION	28-01-2011	09-06-2003	Charlotte
10	ADMIN	ADMINISTRATION	01-01-9999	01-01-2004	Charlotte
11	ADMIN	ADMINISTRATION	18-08-2005	16-04-2004	Charlotte
12	CONSULTANT	CONSULTANT	03-01-2006	02-08-2004	Charlotte
13	SALES	ADMINISTRATION	21-02-2007	25-10-2004	Charlotte
14	CONSULTANT	CONSULTANT	01-01-9999	28-10-2004	Charlotte
15	CONSULTANT	SALARY	01-01-9999	04-01-2005	Charlotte
16	CONSULTANT	SUB-CONTRACTOR	01-01-1900	28-02-2005	Charlotte
17	SALES	ADMINISTRATION	01-01-9999	28-02-2005	Charlotte
18	CONSULTANT	CONSULTANT	14-09-2007	01-04-2005	Charlotte

TermDate and **HireDate**, have a calendar icon. These are date columns. The final column, **Location**, is a text column (ABC).

2. Ensure that **all of these columns have the correct data type**. If not, change their data type by either clicking on the data type icon in the column header or choosing the **Transform** tab of the ribbon and using the **Data Type** dropdown in the **Any Column** section. When you are finished, your data types should be the same as the ones in the following screenshot:

ID	Name	Title	EmployeeType	TermDate	HireDate	Location
1	KCMCAHON	Mcmahon, Karyn	CONSULTANT	CONSULTANT	1/1/1900	1/1/1900 Charlotte
2	CBRYANT	Bryant, Carolyn	CONSULTANT	SALARY	5/15/2015	1/1/1900 Charlotte
3	PLUCAS	Lucas, Pamela	SALES	ADMINISTRATION	1/1/1900	1/1/1900 Charlotte
4	ASHIELDS	Shields, Art	ADMIN	ADMINISTRATION	1/1/1900	1/1/1900 Charlotte
5	BFRANCO	Franco, Brenda	ADMIN	ADMINISTRATION	11/29/2018	1/1/1900 Charlotte
6	EBECKER	Becker, Eileen	CONSULTANT	SALARY	1/1/1900	1/1/1900 Charlotte
7	DVILLEGAS	Villegas, Desiree	CONSULTANT	CONSULTANT	11/18/2005	4/17/2000 Charlotte
8	MMONTES	Montes, Megan	CONSULTANT	SALARY	3/8/2019	10/8/2001 Charlotte
9	TPARRISH	Parrish, Tamera	ADMIN	ADMINISTRATION	1/28/2011	6/9/2003 Charlotte
10	LGARRISON	Garrison, Lenny	ADMIN	ADMINISTRATION	1/1/9999	1/1/2004 Charlotte
11	TBENSON	Benson, Terra	ADMIN	ADMINISTRATION	8/18/2005	4/16/2004 Charlotte
12	CVALDEZ	Valdez, Carl	CONSULTANT	CONSULTANT	1/3/2006	8/2/2004 Charlotte
13	EKAISER	Kaiser, Evangeline	SALES	ADMINISTRATION	2/21/2007	10/25/2004 Charlotte

Transforming the Tasks query

To transform the **Tasks** query, perform the following steps:

1. Click on the **Tasks** query in the **Queries** pane. Notice that there are only three **Applied Steps** in this query: **Source**, **Navigation**, and **Changed Type**. The two columns are named **Column1** and **Column2**. Also, note that the first row contains the values **TaskID** and **Category0**. We want the values in this first row to be the names of our columns.
2. Click the **Navigation** step in the **Applied Steps** area of the **Query Settings** pane.
3. Click the **Transform** tab and choose **Use First Row as Headers**, the second icon from the left. An **Insert Step** message will be displayed, asking you to confirm the insertion of a step into the query:

X

Insert Step

Are you sure you want to insert a step? Inserting an intermediate step may affect subsequent steps, which could cause your query to break.

Insert

Cancel

4. Click the **Insert** button. Notice that a **Promoted Headers** step has been inserted between the **Navigation** step and the **Changed Type** step. Our column headers are now **TaskID** and **Category**.

5. Click on the **Changed Type** step and notice that an error is displayed. This is because the **Changed Type** step was referring to the columns as **Column1** and **Column2**, but now these columns are called something different.

6. Remove the **Changed Type** step by clicking on the X icon to the left of the step name.

7. Now, change the data types of both columns to text.

Transforming the January query

Moving on to the **January** query, perform the following transformation steps:

1. Click on the **January** query. Note that four **Applied Steps** exist – **Source**, **Navigation**, **Promoted Headers**, and **Changed Type**.
2. The **EmployeeID**, **TaskID**, **JobID**, **Division**, **TimesheetBatchID**, **TimesheetID**, and **PayType** columns are all text columns (**ABC**).
3. The **Date**, **PeriodStartDate**, and **PeriodEndDate** columns are all Date columns (calendar icon).
4. The **Hours**, **HourlyCost**, **HourlyRate**, **TotalHoursBilled**, and **TotalHours** columns are all decimal number (1.2) columns.
5. Make sure that the **Changed Type** step is selected and then change **HourlyCost** and **HourlyRate** to **Fixed decimal number**. A **Change Column Type** prompt is displayed each time. Choose the **Replace current** button each time, as shown in the following screenshot:

X

Change Column Type

The selected column has an existing type conversion. Would you like to replace the existing conversion, or preserve the existing conversion and add the new conversion as a separate step?

Replace current

Add new step

Cancel

This finishes the basic transformation work on our queries. However, we have not yet finished transforming our data. We will now move on to more complex data transformations.

Merging, copying, and appending queries

We will merge the two queries together by doing the following:

1. Start by selecting the January query in the Queries pane.
2. Select the Home tab from the ribbon and then, in the Combine section of the ribbon, choose Merge Queries.

3. The Merge dialog is displayed, as shown in the following screenshot: Figure 14: Merge queries

Merge

Select a table and matching columns to create a merged table.

January

EmployeeID	Date	Hours	HourlyCost	HourlyRate	TaskID	JobID	Division	PeriodStart
GBRANCH	1/1/2019	8	136.972	0	PTO	1001TECSOL	1001 Technology	12
GBRANCH	1/2/2019	8	136.972	0	INTERNAL	1001TECSOL	1001 Technology	12
GBRANCH	1/3/2019	8	136.972	0	INTERNAL	1001TECSOL	1001 Technology	12
GBRANCH	1/4/2019	8	136.972	0	INTERNAL	1001TECSOL	1001 Technology	12
BREESE	1/2/2019	7	104.368	125	1001TM	CLEV003201	1001 Technology	12

Tasks

TaskID	Category
PTO	PTO
INTERNAL	Int Admin
1001TM	Billable
1001TB	Billable
2001TM	Billable

Join Kind

Left Outer (all from first, matching from second)

Use fuzzy matching to perform the merge

Fuzzy merge options

Similarity threshold (optional)

Ignore case

Match by combining text parts

Maximum number of matches (optional)

Transformation table (optional)

The selection has matched 10991 out of the first 10991 rows.

OK Cancel

4. The January query is listed at the top of the dialog, along with a preview of the columns and rows returned by that query. In the drop-down box just below this table, use the drop-down arrow and choose the Tasks query, as shown in the preceding screenshot. Again, a preview of columns and rows returned by the Tasks query is displayed. This is the query that we will be merging with our January query.

5. Select the TaskID column from both tables and note that the OK button becomes active, as shown in the preceding screenshot.

6. Also, be aware that, at the bottom of the dialog, as shown in the preceding screenshot, information is displayed indicating The selection has matched 10991 out of the first 10991 rows.

7. The Join Kind dialog displays Left Outer (all from first, matching from second). Other types of joins are listed in the dropdown, including the following: Left Outer (all from first table, matching from second) Right Outer (all from second table, matching from first) Full Outer (all rows from both) Inner (only matching rows) Left Anti (rows only in first) Right Anti (rows only in second)

8. Leave this as Left Outer (all from first, matching from second).

9. Below the Join Kind, there is a checkbox for Use fuzzy matching to perform the merge and an area for Fuzzy merge options.

A fuzzy merge allows similar but not identical items to be matched during a merge.

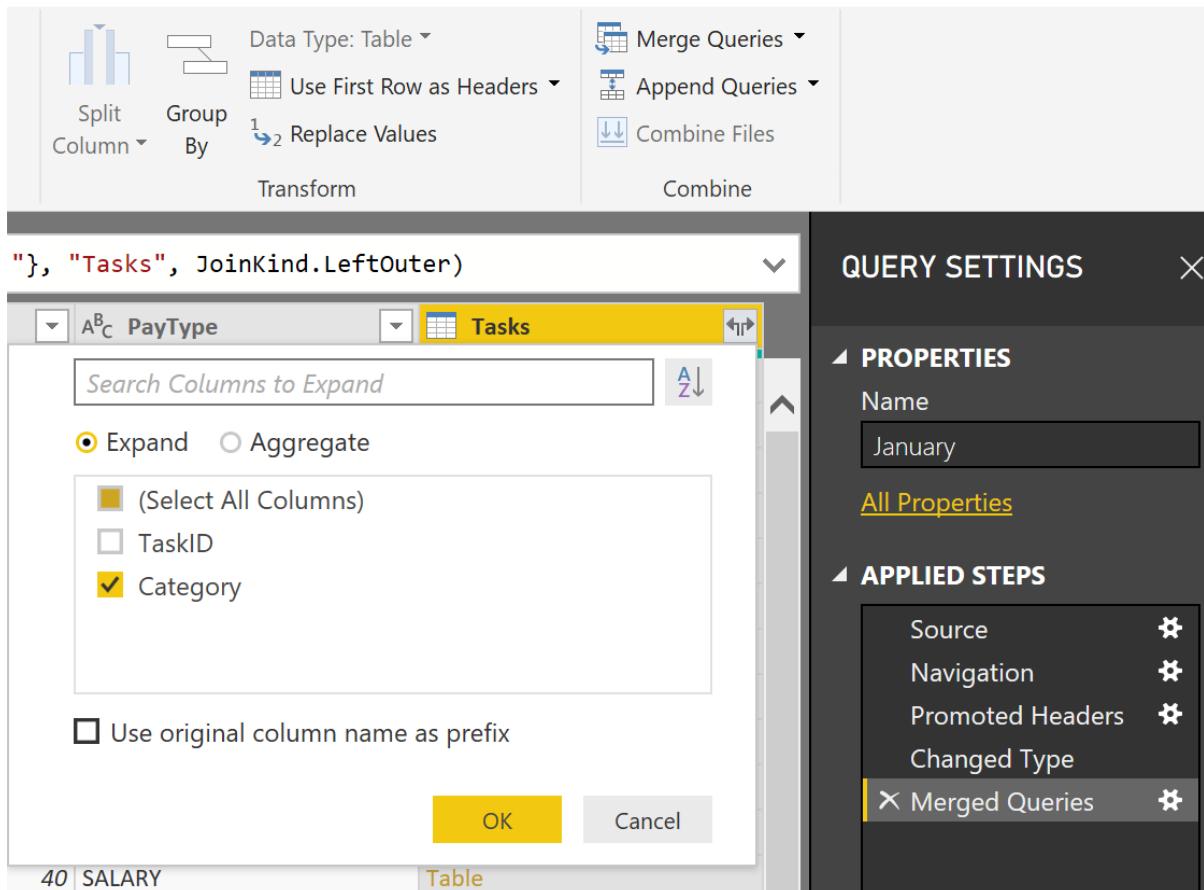
- Options include a Similarity threshold, which is optional. The Similarity threshold is a number between 0.00 and 1.00. A value of 0.00 will cause all values to match, while a value of 1.00 will cause only exact values to match. The default is a value of 0.80. Additional options include the ability to Ignore case as well as the ability to Match by combining text parts. For example, by ignoring case, mlcrSoft could match Microsoft, and by combining text parts, Micro and soft could be combined to match Microsoft.
- When performing fuzzy merges, it is possible to have multiple values match. You can use the optional setting, Maximum number of matches, to control how many matching rows are returned for each input row. This is a number that can range from 1 to 2,147,483,647 (the default).
- Finally, there is an option to use a Transformation table. This allows you to specify a table of values with a From and To column that can be used during the merge process. For example, the merge table might contain a value in the From column for USA that maps to a To column of United States.

10. Leave the Use fuzzy matching to perform the merge checkbox unchecked and press the OK button to exit the Merge dialog and perform the merge.

Expanding tables:

1. Click the diverging arrows icon on the far right of the column header for the Tasks column. This icon is used to expand the columns that contain table information.

2. Uncheck the checkbox next to TaskID and uncheck the checkbox for Use original column name as prefix, as shown in the following screenshot:



3. Click the OK button.

4. Note that the Tasks column changes to Category and that our task category values are listed in each row. Also, a step has been added to our query: Expanded Tasks.

Disabling the loading of queries

Now that we have merged our Tasks query with our January query, we really have no need to load the Tasks query into a separate table in our data model. To disable loading the Tasks query, perform these steps:

1. Select the **Tasks** query in the **Queries** pane.
2. Right-click on the **Tasks** query and note that a number of options are presented.
3. Uncheck **Enable load**.
4. A warning dialog appears, informing us that this operation will not load data from this query into its own table and actually remove the table from our data model if it exists. This warning dialog is shown in the following screenshot:



Possible Data Loss Warning

Disabling load will remove the table from the report, and any visuals that use its columns will be broken.

Continue

Cancel

5. Click the Continue button.

Back in the Power Query Editor, note that the **Tasks** query name is now italicized

Copying queries

Now that we have our **January** query modified to merge in our **Tasks** information, we still need to bring in February and March data. We can do this easily by copying our **January** query and then changing the source. To copy the **January** query, do the following:

1. Select the **January** query in the **Queries** pane.
2. Right-click the **January** query and choose **Copy**.
3. Right-click in a blank section of the **Queries** pane and choose **Paste**. Two new queries will be created, **January (2)** and **Tasks (2)**.
4. Right-click in a blank section of the **Queries** pane again and choose **Paste**. Two additional queries will be created, **January (3)** and **Tasks (3)**.
5. Click on the **January (2)** query in the **Queries** pane. Note that the **Applied Steps** are identical to our original **January** query, including the **Merged Queries** and **Expanded Tasks** steps.
6. Click the gear icon next to the **Merged Queries** step.
7. The **Merge** dialog will be displayed. Note that the two tables being merged are **January (2)** and the original **Tasks** (**change Task1 table to task table and select task ID**) and **do the similar task for January 3 and task tables**. Therefore, we do not need the **Tasks (2)** or **Tasks (3)** queries that were created.
8. Click the **Cancel** button.
9. Delete the **Tasks (2)** and **Tasks (3)** queries by right-clicking each query and choosing **Delete**.
10. In the **Delete Query** dialog, choose the **Delete** button each time.

Changing sources

Now that we have copies of our January query, we want to change the data sources for our **January (2)** and **January (3)** queries to instead load data for February and March, respectively. To do this, perform the following operations:

1. Click on the **January (2)** query.
2. In the **Applied Steps** area of the **Query settings** pane, click the gear icon next to the **Navigation** step.
3. In the **Navigation** dialog, double click on **February**.
4. Click the **OK** button.

Note that there is a yellow warning icon next to the **January (2)** query in the **Queries** pane. Also note that two additional steps have been added to our query, just below the **Navigation** step: **Promoted Headers1** and **Changed Type1**. If we click on the **Expanded Tasks** step in our query, we will see that an error message has been displayed. To fix this, execute the following steps:

1. Use the **X** icon to delete the **Promoted Headers1** and **Changed Type1** steps of our query.
2. In the **Delete Step** dialog that is displayed, choose the **Delete** button each time. After deleting these steps, all warnings and errors will be removed.
3. Finally, under the **PROPERTIES** section of the **Query Settings** pane, change the **Name** of the query from **January (2)** to February.

The preceding method is one way of changing the source for a query. But there is another way to change the source: by editing the underlying **M** code. To use this method, execute the following steps:

1. Click on the **January (3)** query in the **Queries** pane.
2. Click the **Navigation** step.

This time, edit the formula bar and replace the word **January** with the word **March** so that the formula for this step appears as follows:

```
= Source{[Item="March",Kind="Sheet"]}[Data]
```

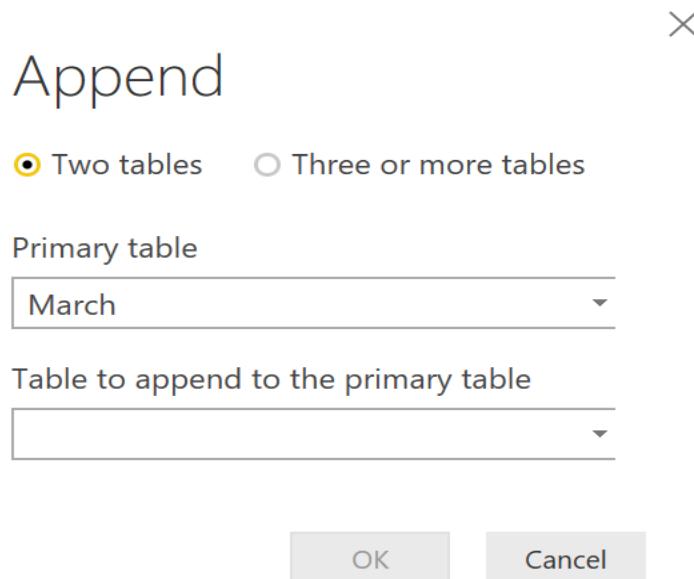
4. Press the *Enter* key to complete the formula. This time, no additional steps were added to our query.
5. Rename the query to March.

Append queries

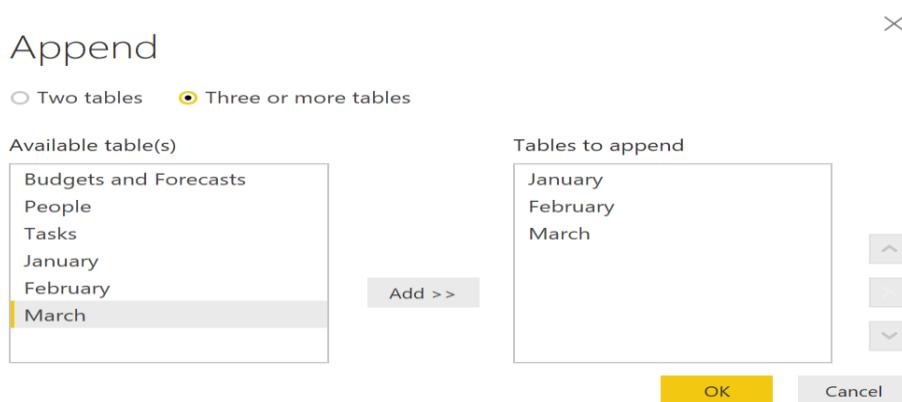
Appending queries:

We now have three separate tables for hours data reported by employees, one of each for January, February, and March. However, what we really need is for all of this data to reside in a single table. We can accomplish this by using an **Append** query, as follows:

1. Start by clicking on the **Home** tab of the ribbon.
2. In the **Combine** section at the far right of the ribbon, choose the **Append Queries** dropdown and choose **Append Queries as New**. This displays the **Append** queries dialog, as shown here:



3. In the Append queries dialog, select the radio button for Three or more tables.
4. Use the Add >> button to add the January, February, and March queries to Tables to append. The order does not matter:



5. When finished, click the **OK** button.

Table. Combine({January, February, March})

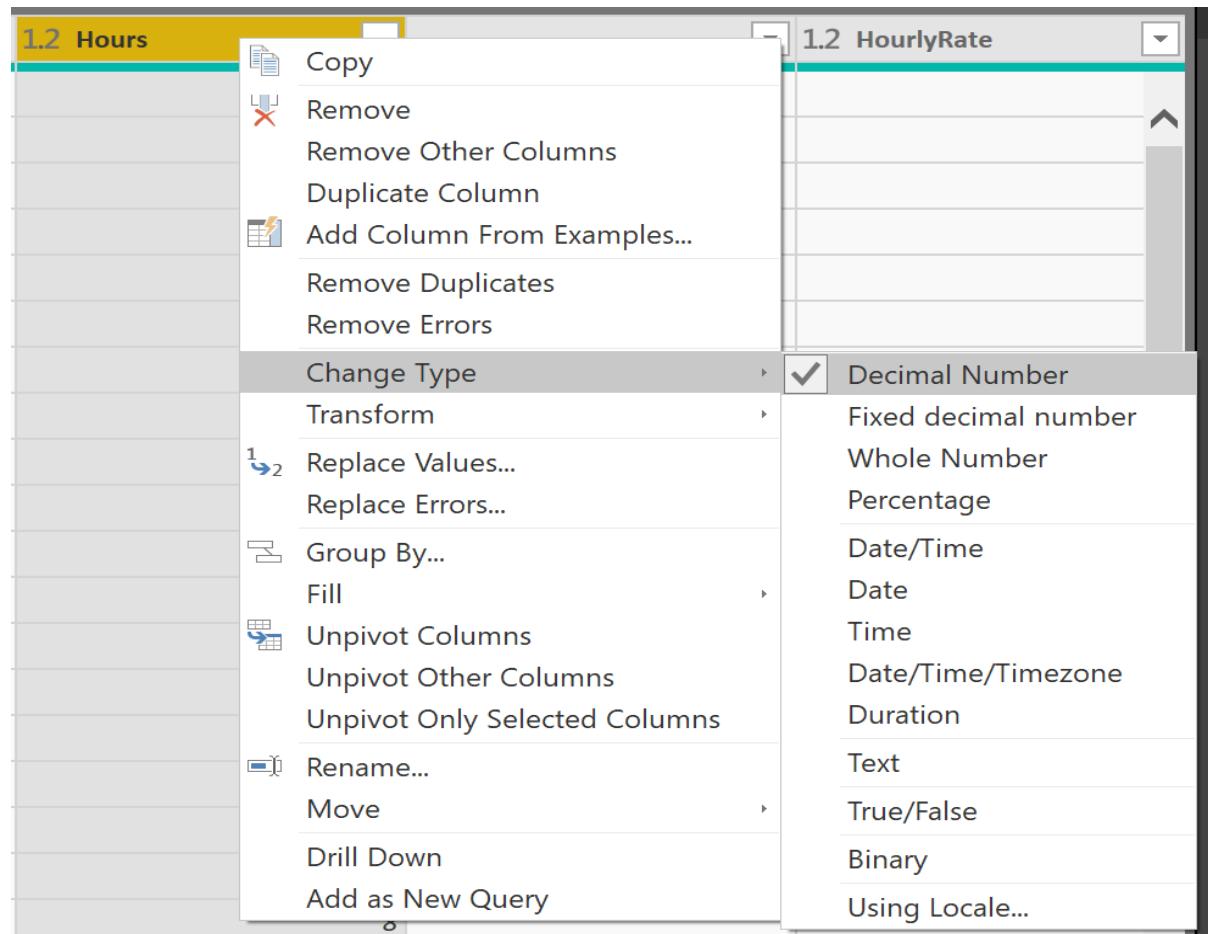
We will now put some finishing touches on this query by performing a few additional steps, as follows:

1. Select the query called **Append1** in the **Queries** pane.
2. Right-click the query and choose **Rename**. Rename this query to Hours.

The **Hours** query now contains all of the information from the **January**, **February**, and **March** queries. This means that we can disable loads on each of these queries. To do this, perform the following steps:

1. Right-click on each of these queries and uncheck **Enable load**.
2. A warning dialog appears, informing us that this operation will not load data from this query into its own table and will actually remove the table from our data model if it exists. Click the **Continue** button.
3. Finally, locate the **Hours** column in the **Hours** query. Right-click the header for

this column and choose **Change Type** and then **Decimal Number**. This ensures that **Hours** comes in as a **Decimal Number** and not a **Fixed Decimal Number** or currency. This operation is shown in the following screenshot:



Queries [7]

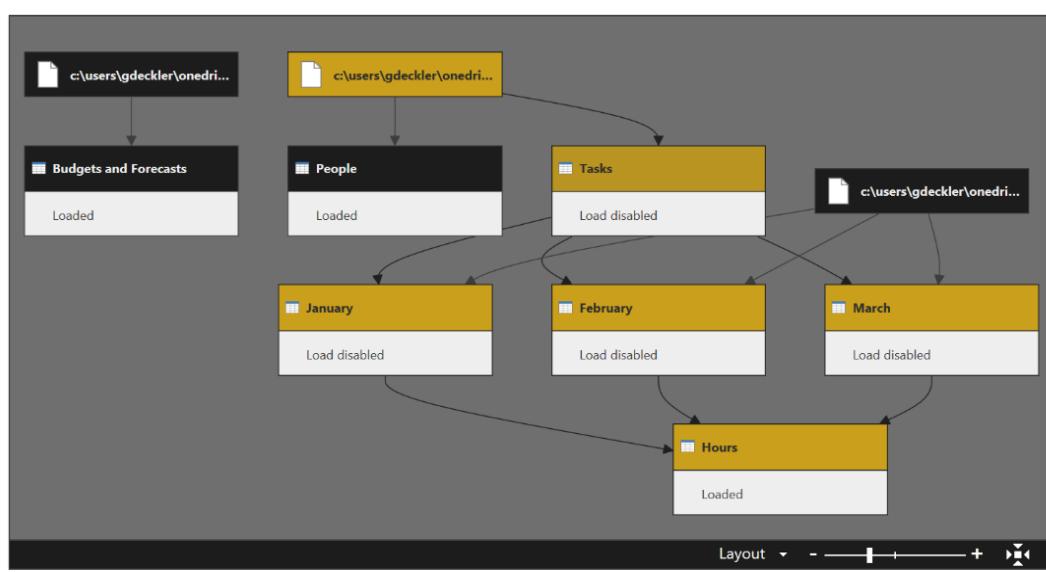
- Budgets and Forecasts
- People
- Tasks
- January
- February
- March
- Hours

	EmployeeID
1	GBRANCH
2	FMAYNARD
3	ACERVANTES
4	EMORSE
5	GBRANCH
6	GBRANCH
7	GBRANCH

We can view how our sources and queries are related to one another by viewing the query dependencies. We can do this by performing the following steps:

1. First, click on the **View** tab of the ribbon.
2. Then, click the **Query Dependencies** button in the **Dependencies** area of the ribbon. This displays the **Query Dependencies** window, as shown in the following screenshot:

Query Dependencies



3. Click on the Tasks query to highlight the sources and other queries related to the Tasks query.

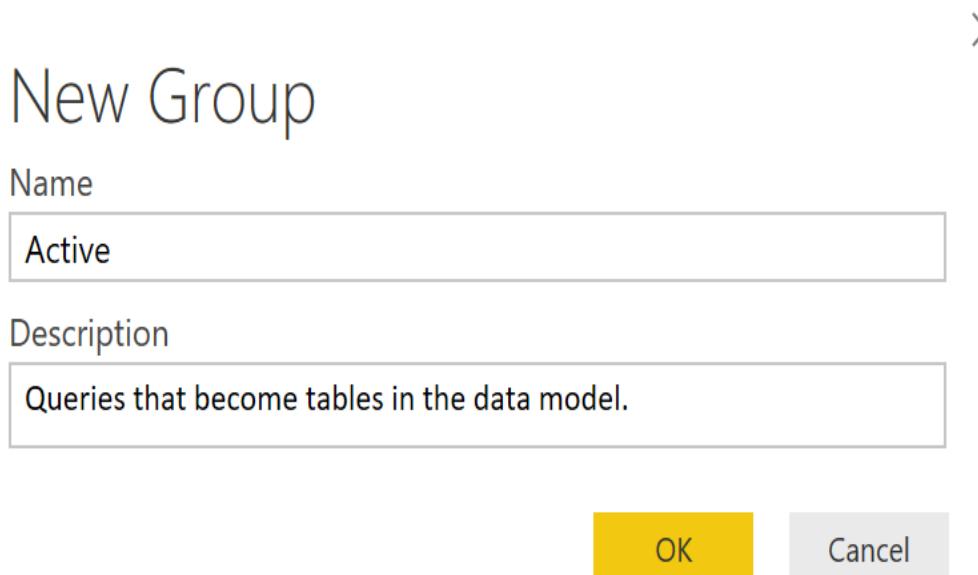
4. When you have finished exploring, click the Close button.

Organizing queries

Now, let's organize our queries by grouping queries together. In this case, we will create groups for our active and intermediate queries.

To do this, we will use the following steps:

1. Right-click the Hours query and select Move To Group and then New Group. The New Group dialog will be displayed, as shown in the following screenshot:



2. For **Name**, enter Active and, in the **Description**, enter Queries that become tables in the data model.
3. Click the **OK** button. Note that two folders are created, an **Active** folder and an **Other Queries** folder.
4. Select the **Budgets and Forecasts** query and right-click the query.
5. Choose **Move to Group** and then **Active**.
6. Repeat this process for the **People** query.

We now have our queries organized into **Active** queries and **Other Queries**.

Checking column quality, distribution, and profiles

As a final check before loading our data, the Power Query Editor includes powerful tools that allow us to understand the quality of our data. These tools can be found on the **View** tab of the ribbon in the **Data Preview** section

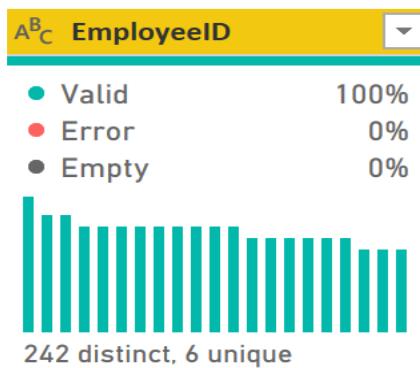
To use these features, we will perform the following steps:

1. Click on the **Hours** query in the **Active** group in the **Queries** pane.
2. Click the **View** tab of the ribbon and check the box next to **Column Quality** in the **Data Preview** section:



Note that, under the column headers, information is displayed regarding the quality of data in each column. This information includes what percentages of row values are **Valid**, **Error**, or **Empty**.

3. Click the checkbox next to **Column distribution** in the ribbon. This same area now displays information regarding the distinct and unique values that were found in the rows of each column. This information is based on the first 1,000 rows returned by the query:



Finally, we can also view additional statistical information about the values in each column.

4. Click the checkbox next to **Column profile** in the ribbon. Note that two additional areas are displayed, that is, **Column statistics** and **Column**

This completes the transformation process for our data. Our final step is to finally load the data into the data model.

Creating Data Models and Calculations

Creating a data model:

The concept of a data model is fundamental to Power BI. In short, a data model is defined by the tables that are created from Power Query queries, as well as the relationships that are defined. These are needed to connect individual tables to one another, as well as the **metadata** (data about data) regarding the columns within the tables.

In Power BI, the data model is stored within a SQL Server Analysis Services tabular cube. It is the creation of this data model that enables self-service analytics and reporting.

Touring the Model view

So far, we have explored the overall Power BI Desktop and Power Query Editor. We will now explore the **Model** view within Power BI Desktop. To switch to the **Model** view, click the bottom icon in the **Views Bar** of Power BI Desktop.

This is the **Model** view, as shown in the following screenshot

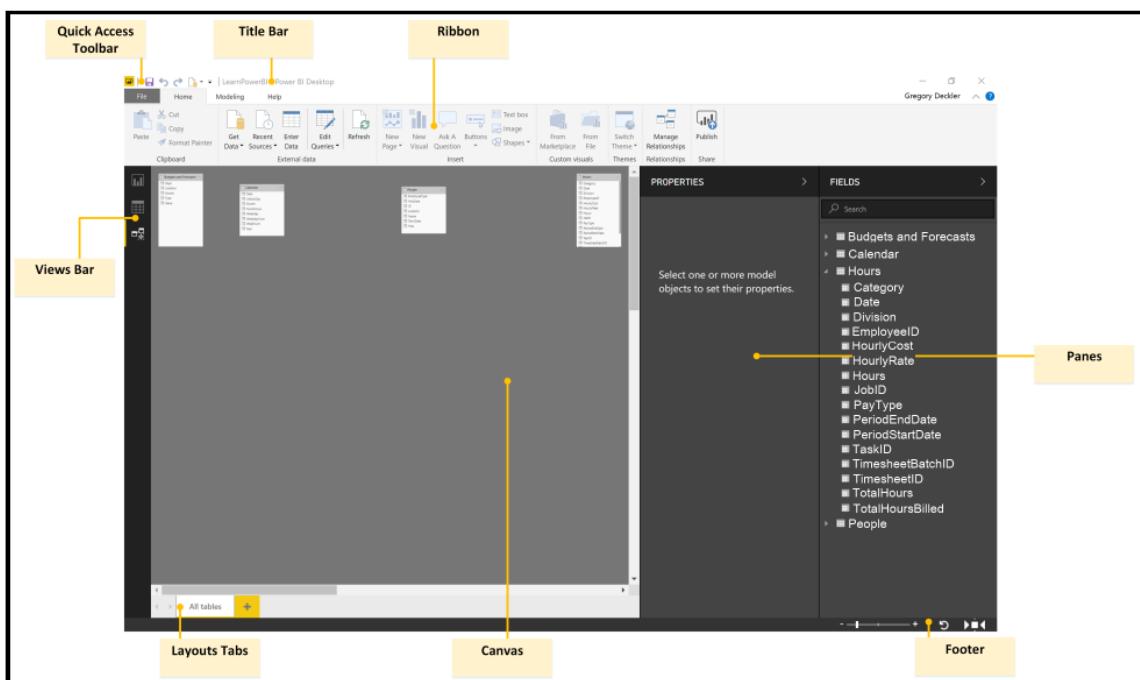


Figure 1 – Model view

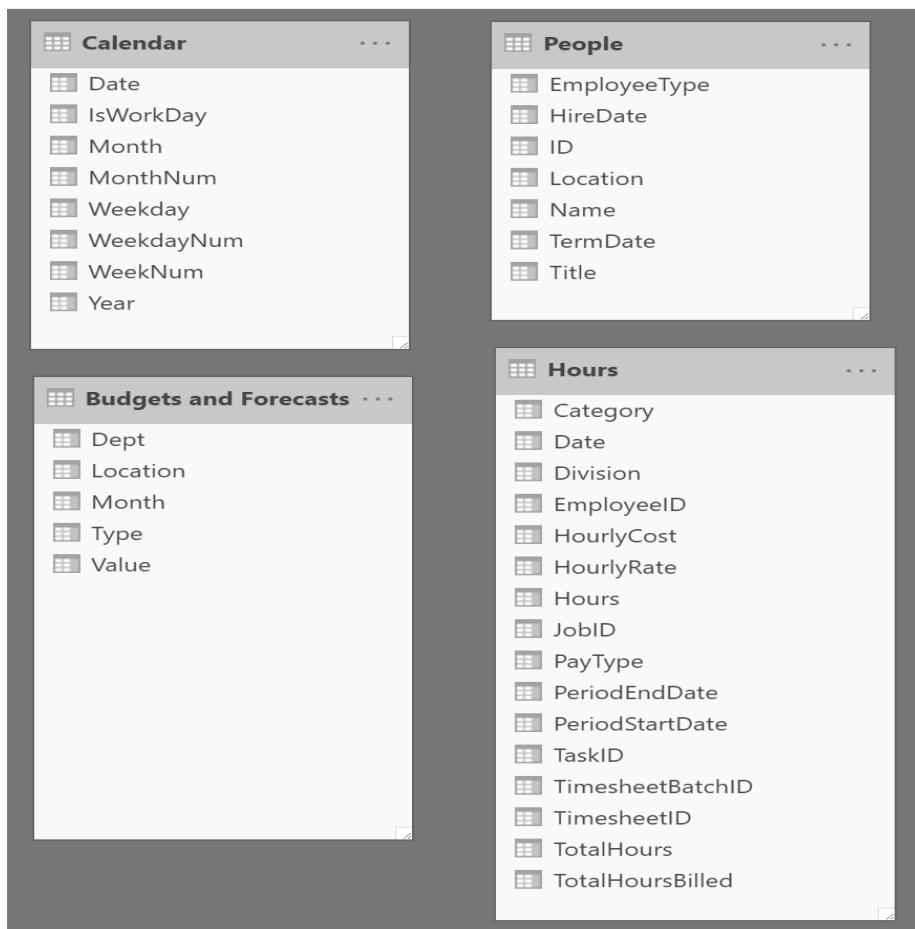
The **Model** view provides an interface for building our data model. It does this by creating relationships between tables as well as defining metadata for tables and columns. We can even create multiple layouts for our data model. As you might expect, the **Model** view interface is similar to and shares common elements with Desktop.

Modifying the layout

In the **Model** view, we have a default **All tables** layout, which was created for us automatically by Power BI.

To modify this layout, follow these steps:

1. Minimize the **FIELDS** and **PROPERTIES** panes by clicking on the arrow icon in the pane headers (>).
2. Click on the tables and drag them closer together. Use the **Fit to screen** icon in the footer to zoom in on the table layout. You should now be able to clearly see the table names and columns in the tables.
3. Move the **Calendar** and **People** tables to the center, at the top, and place the **Budgets and Forecasts** and **Hours** tables underneath these two tables. Use the **Fit to screen** icon in the footer to zoom in on the tables. Note that we cannot see all of the columns in the **Hours** table.
4. Use the sizing handle at the bottom-right corner of the table to adjust the size so that we can see all of the columns in the table. When finished, your **Canvas** should look similar to this:



. Creating and understanding relationships

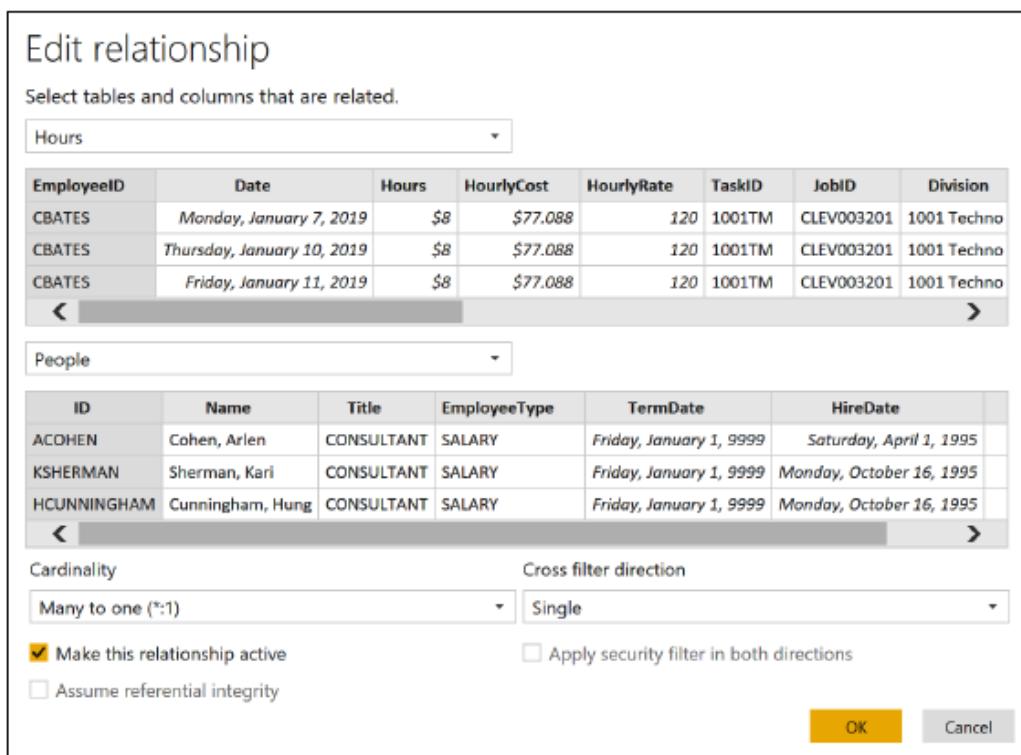
Now that we can clearly see our tables and columns, we can create relationships between our tables. Creating relationships between tables allows calculations and aggregations to work across tables so that multiple columns can be used from separate, related tables.

Note that the line has a **1** next to the **People** table and a ***** next to the **Hours** table. This means that this relationship is one-to-many or many-to-one.

In other words, there are unique row values in the **People** table that match multiple rows in the **Hours** table. This makes sense since each employee would submit an hours report for every day. The designation of a **1** (unique) or a ***** (many) defines the **cardinality** of the relationship between the tables.

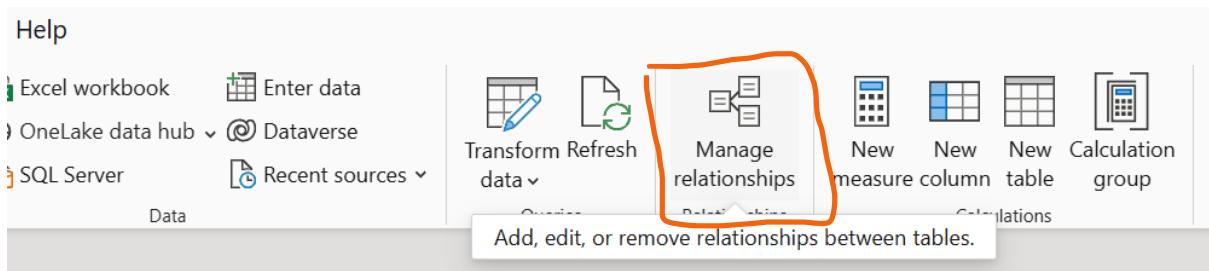
There are actually four different cardinalities for relationships in Power BI:

- **One-to-one:** This means that there are unique values in each table.
- **Many-to-one:** This means that there are unique values in one table that match multiple rows in the other table.
- **One-to-many:** This means that there are unique values in one table that match multiple rows in the other table.
- **Many-to-many:** This means that neither table has unique values for rows. It is generally good practice to avoid these types of relationships because of their complexity and the amount of processing and resources required.
- We can view and modify the relationship definition by double-clicking on the



relationship line.

- In the following screenshot, we can see the **Edit relationship** dialog:
- Note : Note that, since the **Hours** table is defined first and the **People** table is defined second, the **Cardinality** of our relationship is **Many to one (*:1)**. Also, note that the relationship is active and that the **Cross filter direction** is **Single**.
-
- **1. Click on Manage relationships button. The Manage relationships dialog will be displayed:**



The screenshot shows the 'Manage relationships' dialog. At the top, there's a green button labeled '+ New relationship' with a red box around it. Below it is an 'Autodetect' button. The main area lists existing relationships:

From: table (column)	Relationship	To: table (column)	Status
<input checked="" type="checkbox"/> Hours (Date)	$*$ — \square — 1	<input type="checkbox"/> Calender (Date)	Active
<input type="checkbox"/> Hours (EmployeeID)	$*$ — \square — 1	<input type="checkbox"/> People (ID)	Active

- Here, we can see our existing Active relationship between the Hours table and the People table, including the columns involved in the relationship in parentheses. From this dialog, we can Edit or Delete the relationship or have Power BI attempt to AutoDetect relationships between tables. Power BI can sometimes autodetect relationships between tables based on the column names and row values.
- **2. Select the New button. This displays the Create relationship dialog:**

Create relationship

Select tables and columns that are related.

Hours

EmployeeID	Date	Hours	HourlyCost	HourlyRate	TaskID	JobID	Division
CBATES	Monday, January 7, 2019	\$8	\$77.088	120	1001TM	CLEV003201	1001 Techno
CBATES	Thursday, January 10, 2019	\$8	\$77.088	120	1001TM	CLEV003201	1001 Techno
CBATES	Friday, January 11, 2019	\$8	\$77.088	120	1001TM	CLEV003201	1001 Techno

Calendar

Date	Month	Year	MonthNum	WeekNum	Weekday	WeekdayNum	IsWorkDay
1/1/2017	January	2017	1	1	Sunday	7	0
1/2/2017	January	2017	1	1	Monday	1	1
1/3/2017	January	2017	1	1	Tuesday	2	1

Cardinality: Many to one (*:1)

Cross filter direction: Single

Make this relationship active

Apply security filter in both directions

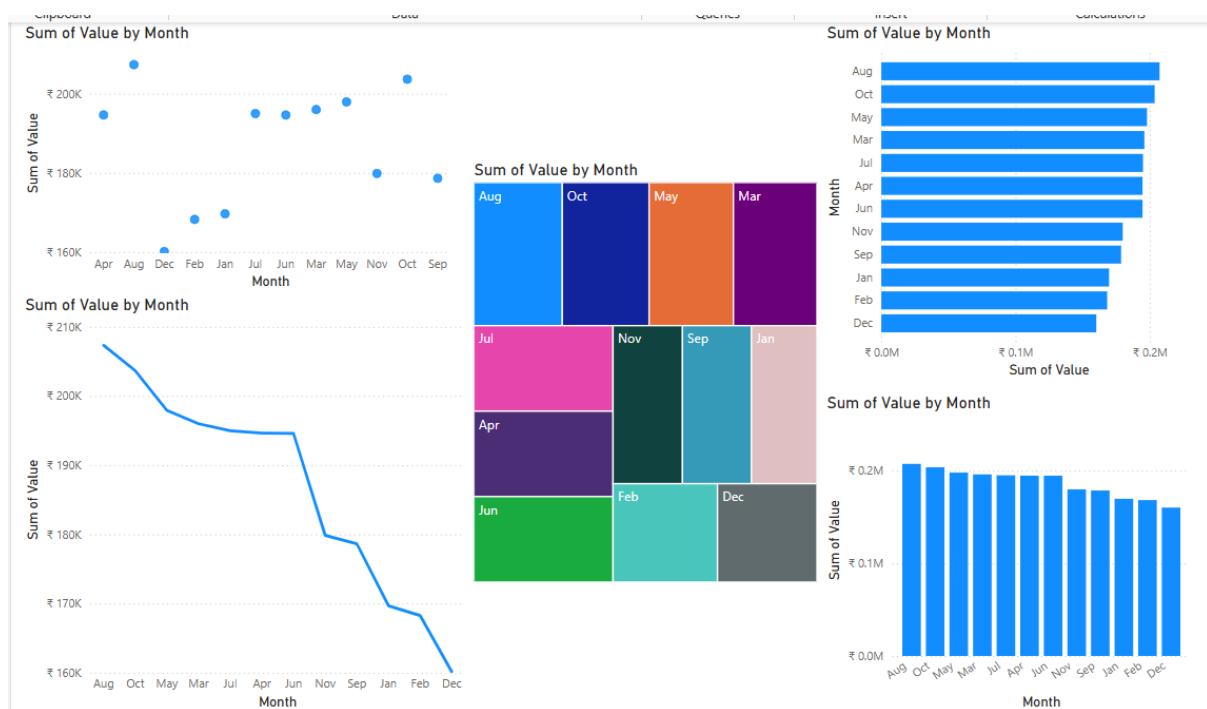
Assume referential integrity

OK Cancel

- 3. In the first drop-down menu, choose the **Hours** table. A preview of the table will be displayed. Then, click the **Date** column.
- 4. Choose **Calendar** in the second drop-down menu. A preview of this table will be displayed. Choose the **Date** column from this table. Power BI detects the appropriate **Cardinality** and **Cross filter direction**. Since there are no other relationships between these tables, Power BI checks the **Make this relationship active** checkbox.
- 5. Click the **OK** button to create this relationship. Note that this new relationship now appears in the **Manage relationships** dialog.
- 6. Click the **Close** button to close the **Manage relationships** dialog. In the canvas, there will now be a relationship line linking the **Calendar** table to the **Hours** table.
-
- Congratulations! You have successfully linked the three separate tables to create a data model! For now, we won't link the **Budgets and Forecasts** table within our data model.
- Instead, we will start to create visualizations and calculations so that we can analyze our data.

Objective: Exploring the data model

1. Start by clicking on the Report view in the Views pane.
2. At the bottom of the report canvas, click the plus (+) icon next to Page 1. This creates a new blank page, Page 2. Double-click Page 2, change its name to Utilization, and then press the *Enter* key. This changes the name of the page to Utilization.
3. Expand the People table in the FIELDS pane by clicking the small arrow to the left of the People table. Check the box next to the Name field. This creates a Table visualization on our report canvas with the names of employees. Use the sizing handles for this visualization to resize the table to take up the entire page.
4. Expand the Hours table. Make sure that our table is selected and then check the box next to the Hours field. The number of hours reported by each employee is shown next to their name. This occurs because of the relationship between our People table and our Hours table. Because these tables are joined by a relationship based on the ID of each employee, we can use fields from both tables in visualizations. Due to this, the rows in the tables are automatically filtered based on this relationship.
5. With the table selected, click on the **Matrix** visualization in the **VISUALIZATIONS** pane. This icon is to the immediate right of the highlighted **Table** visualization. Note that the **FIELDS** pane changes from just displaying a **Values** area to now containing **Rows**, **Columns**, and **Values** areas. Our **Name** field is now under **Rows**, while our **Hours** field is under **Values**.



6. From the **Hours** table, click on **Category** and drag and drop this field into the **Columns** area. We can now see a breakdown of the hours for each employee by **Category**.

Name	Hours	Category
Adkins, Rita	8	Billable
Allen, Susana	2	Billable
Allen, Susana	3	Billable
Allen, Susana	4	Billable
Allen, Susana	4	Billable
Allen, Susana	5	Billable
Anderson, Pearlie	1	Other
Anderson, Pearlie	1	Sales Support
Anderson, Pearlie	2	Sales Support
Anderson, Pearlie	2	Sales Support
Anderson, Pearlie	3	Sales Support
Anderson, Pearlie	3	Sales Support
Anderson, Pearlie	4	Billable
Anderson, Pearlie	4	Sales Support

Creating calculations the organization knows that, to remain profitable, the target utilization must be 80%. Therefore, we must create this calculation

Calculated columns

Calculated columns are additional columns that are created in data model tables through the use of the **Data Analysis Expressions (DAX)** formula language.

Understanding context for calculated columns

Follow these steps to create a calculated column:

1. Click on the **Data** view in the **Views** pane.
2. Click on the **Hours** table in the **FIELDS** pane. From the ribbon, choose the **Modeling** tab and then **New Column** in the **Calculations** section. The formula bar is populated by **Column =**, and a new field called **Column** will appear in the **FIELDS** pane. Click the formula bar and enter the following formula:

The screenshot shows the Power BI Data Editor interface. At the top, there are tabs for 'Structure' and 'Formatting'. In the 'Formatting' tab, there are sections for 'Name' (set to 'Column'), 'Data type' (set to 'Whole number'), 'Format' (set to 'General'), and 'Summarization' (checkbox checked). Below the tabs, a message says 'Auto recovery contains some recovered files that haven't been opened.' A preview pane shows a table with columns: HourlyRate, TaskID, JobID, Division, PeriodStartDate, and PeriodEndDate. The preview shows three identical rows where the sum of hours is 177997.15.

HourlyRate	TaskID	JobID	Division	PeriodStartDate	PeriodEndDate
₹ 120	1001TM	CLEV003201	1001 Technology	02 March 2019	08 March 2019
₹ 120	1001TM	CLEV003201	1001 Technology	02 March 2019	08 March 2019
₹ 120	1001TM	CLEV003201	1001 Technology	02 March 2019	08 March 2019

In case above error comes that means we need to change the value of hours col.

A quick look at the results shows us that something is clearly not correct. The same number appears for every row in the table! We can see that this is the case by clicking on the dropdown arrow in the header for **Column**. Note that only a single value appears in the filter area, that is, **177997.15**, as shown in the following screenshot

To see row context in action, create another new calculated column and enter the following

formula:

Column2 = [Hours] / [TotalHours]

In this calculation, the TotalHours column contains the total number of hours that were reported during the entire reporting period that was defined by the PeriodStartDate and PeriodEndDate columns.

Because no functions have been used that change the evaluation context from row to filter, only the current row is considered during the evaluation of the formula.

Hence, this formula simply divides the **Hours** column in each row by the **TotalHours** column in each row. Now, if we click on the drop-down arrow for Column 2, we will see that there are many different values.

Creating calculated columns for utilization

For the purposes of calculating utilization, we want the total hours billed by an employee divided by the total hours reported by an employee.

In order to get the first part of this, that is, the total hours billed, we need to create a third calculated column using the following formula:

Column 3 = SUMX(FILTER(ALL('Hours'),[Category] = "Billable" && [EmployeeID] = EARLIER([EmployeeID])),[Hours])

1. Take the entire table (ALL) and filter this table down so that it only has rows where the Category column has a value of Billable and rows where the EmployeeID column matches the current row value for EmployeeID.
2. Second, sum the Hours column for this table and return the result. Hence, each row for an employee should return the exact same numeric value, which is the total hours that were reported by that employee that were billable hours.

The screenshot shows the Power BI ribbon with the 'Modeling' tab selected. Below the ribbon is a table view with the following data:

EmployeeID	Date	Hours	HourlyCost	HourlyRate
CBATES	04 March 2019	8	₹ 77,088	
CBATES	05 March 2019	8	₹ 77,088	
CBATES	06 March 2019	8	₹ 77,088	
CBATES	07 March 2019	0	₹ 77,000	

For the second part, that is, the total number of hours reported by an employee, we simply need to create another new column with a slight modification to remove the filter for only billable hours. This column can be written with the following formula:

Column 4 = SUMX(FILTER(ALL('Hours'),[EmployeeID] = EARLIER([EmployeeID])),[Hours])

Finally, to create our utilization calculation, create a new calculated column that utilizes row context:

Column 5 = DIVIDE([Column 3], [Column 4],0)

Note: While the preceding formula could have been written as Column 5 = [Column 3] / [Column 4], it is a good idea to use the DIVIDE function instead of the divide (/) operator. The DIVIDE function takes a third parameter, which allows for the return of an alternate number instead of an error in the event of an attempt to divide by zero.

We can now select Column 5 by clicking on its header and change its formatting to be a percentage. To change the column to a percentage, we need to go to the Formatting section of the Modeling tab of the ribbon. From here, select the % icon.

The screenshot shows the Power BI Data view. At the top, there are settings for 'Name' (Column 5), 'Data type' (Decimal number), and 'Format' (Percentage). Below these are 'Structure' and 'Formatting' tabs. A message in the status bar says 'Auto recovery contains some recovered files that haven't been opened.' The main area displays a table with four columns: PeriodStartDate, PeriodEndDate, TotalHoursBilled, and TimesheetBatchID. The table has three rows, each with the same values: 02 March 2019, 08 March 2019, 40, and WB030219.

To perform this cleanup, execute the following steps:

- Right-click the header for Column and choose Delete.
 - Confirm the deletion of this column in the Delete column dialog box by clicking Delete.
 - Repeat this procedure for Column 2.
 - Double-click the TotalHours column and rename this column TotalHoursInPeriod.
 - Rename Column 3 to TotalBillableHours, Column 4 to TotalHours, and Column 5 to % Utilization.
1. Click on the Report view in the Views pane.
 2. Shrink the existing table visualization horizontally to create space on the report canvas.
 3. Click on an empty portion of the report canvas to deselect the existing table.
 4. Click Name from the People table to create a new table visualization on the blank area of the report canvas.
 5. With this new table selected, click on TotalBillableHours, TotalHours, and % Utilization from the Hours table to add these to this second table visualization.



Note that the values that are returned in this second table are really large! This is because the default aggregation for numeric fields is sum. Since all of the rows for each employee contain our desired number, summing these values doesn't make sense. In the **VALUES** area of the **VISUALIZATIONS** pane, select the drop-down arrow next to TotalBillableHours and choose **Average** instead of **Sum** as the aggregation.

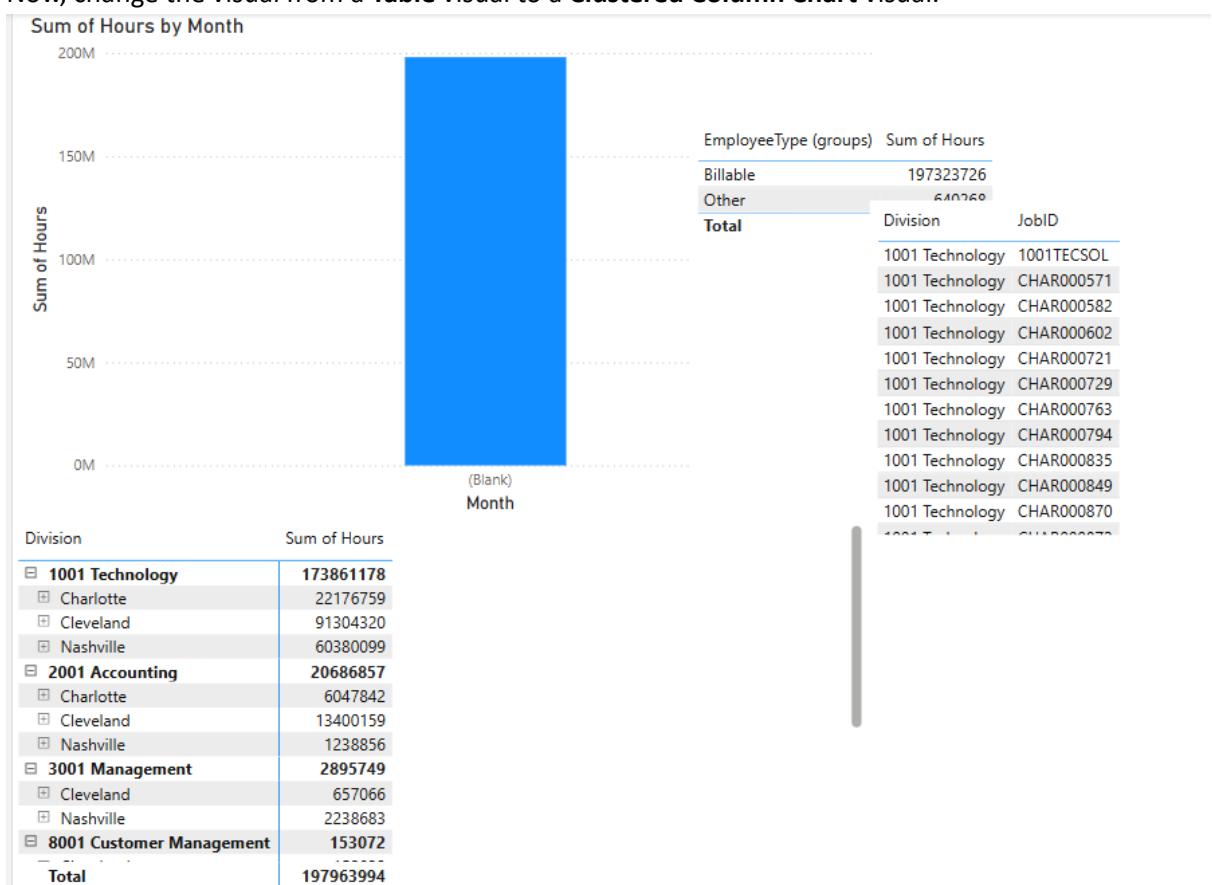
1. Repeat this procedure for TotalHours and % Utilization.

Objective: Segmenting data, using report navigating features, Advanced visualization techniques

Creating groups

The easiest way to see grouping in Power BI is to simply use visualizations to summarize data. We do see this by following these steps:

1. Create a new page, **Page 3**, in Power BI Desktop on the **Report** view.
2. While on this page, expand the **Calendar** table and choose the **Month** column.
3. Next, add the **Hours** column from the **Hours** table.
4. Now, change the visual from a **Table** visual to a **Clustered Column Chart** visual.



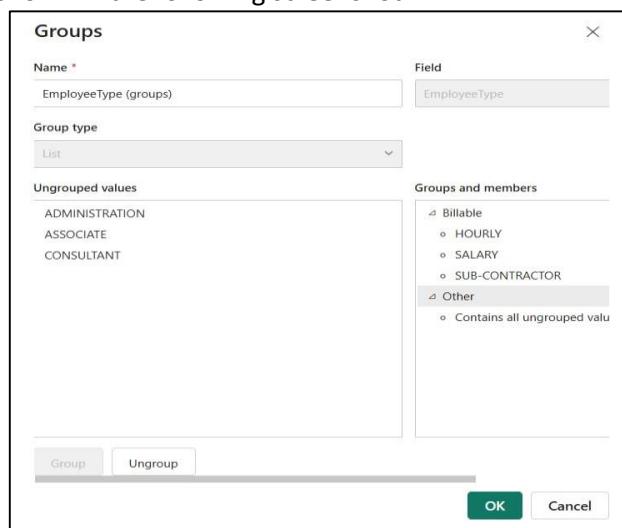
After following these steps, we will have created an ad hoc grouping of our hours based on the month in which those hours were reported.

However, there is another way to use groups within Power BI. Instead of using ad hoc grouping, we can define groupings of information as part of the data model itself and then use those defined groups within visualizations.

Refer to the following steps to see how this is done:

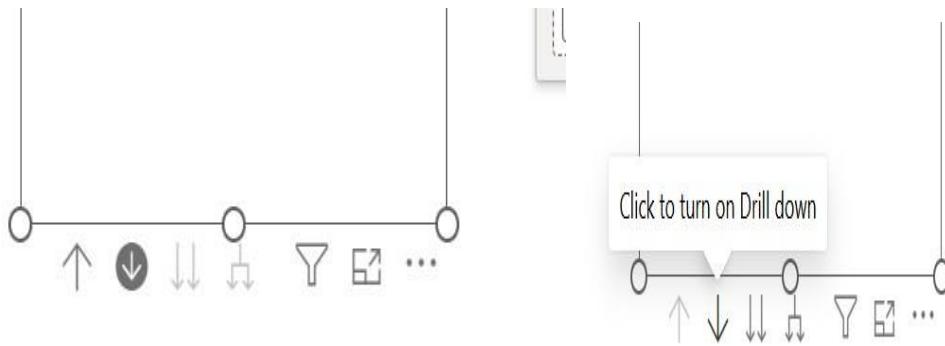
1. Expand the **People** table. Right-click EmployeeType and choose **New Group**. The **Groups** dialog will open. Here, we can define a name for our groups, as well as define the groups themselves. Leave the name of the groups as EmployeeType (groups).
2. In the **Ungrouped values** area, select **HOURLY**, **SALARY**, and **SUBCONTRACTOR** by first clicking on **HOURLY** and then holding down the *Ctrl* key while clicking on **SALARY** and **SUB-CONTRACTOR**. Now, click the **Group** button. A new group will be created in the **Groups and members** section. Rename this group to Billable.
3. Check the **Include Other group** checkbox and then click **OK**.

The **Groups** dialog is shown in the following screenshot:



Creating hierarchies : Hierarchies are powerful features of Power BI that allow data to be summarized and report viewers to drill down into the data to obtain additional details. Similar to groups, we can also create ad hoc hierarchies and define hierarchies within our data model. The easiest way to see an ad hoc hierarchy is to use a Matrix visualization, as follows:

1. Click on a blank area of the canvas and choose Location from the People table.
2. Switch the visualization from a Table visualization to a Matrix visualization. Some visuals work well with hierarchies, while others do not. Table visualizations are not particularly suited for use with hierarchies, while Matrix visualizations are.
3. Drag the **Division** column from the **Hours** table into the **Rows** area in the **VISUALIZATIONS** pane. Note that several arrow icons appear above/below our **Matrix** visualization.



Drag **Hours** from the **Hours** table into **Matrix** visualization. Use the forked arrow above our **Matrix** visualization to **Expand all down one level in the hierarchy**. We can now see the reported hours broken down by **Location** and **Division** in our matrix.

- Some steps to see hierarchies:
- Click on drill down arrow
 - Now Click Expand all down one level in the hierarchy

Location	Count of Hours
Charlotte	4145
1001 Technology	2131
2001 Accounting	1935
9001 Sales	79
Cleveland	15483
1001 Technology	8969
2001 Accounting	6007
3001 Management	381
8001 Customer Management	95
Marketing	23
Quality	2
Shared Services	6
Nashville	8241
1001 Technology	5185
2001 Accounting	372
3001 Management	2629
8001 Customer Management	55
Total	27869

- Expand the Hours table.
 - Right-click Division and then choose Create hierarchy. A Division Hierarchy element will be created within the Hours table, just below our Division column.
- Note that Division is the first element within our new hierarchy.
- Click on the JobID column – Click add to hierarchy ----Select Division hierarchy--onto the Division Hierarchy. JobID will be added below Division within our hierarchy.

Follow these steps to see our new hierarchy in action:

- Click on the blank area of the canvas and then click on **Division Hierarchy**.
- Switch the created visual from a **Table** visualization to a **Matrix** visualization.
- Add **Hours** to the **Values** area.
- Use the forked arrow icon for this new **Matrix** visualization to **Expand all down one level in the hierarchy**. We can now see our hours broken down by **Division** and **JobID**.

Defining hierarchies within the data model is a convenient way to save time. It does this by creating reusable hierarchies of information that can be used within multiple visualizations.

Exploring hierarchies

So far, we have used the forked arrow icon to Expand all down one level in the hierarchy. Now, let's take a look at the functionality of the other arrow icons:

1. In the latest Matrix visualization that we created using our Division Hierarchy, use the up arrow icon to Drill Up within our hierarchy.
2. Switch the visualization to a Clustered column chart.
3. Click on the down arrow to turn on Drill Down. Note that the down arrow icon is now surrounded by a gray circle.



4. Click on the 3001 Management column to drill down into the hierarchy for just the 3001 Management data. We are now at the JobID level of the hierarchy, just within the 3001 Management leaf of our hierarchy.
5. Click on the down arrow icon again to turn off Drill Down and then click on the up arrow icon to drill back up to the top level of the hierarchy.
6. This time, click on the double down arrow icon. This icon drills down into the second level of the hierarchy across all of the top-level leaves of our hierarchy.
7. Use the up arrow icon to drill back up to the top level of the hierarchy. The same four arrow icons for expanding/collapsing and drilling down and up through hierarchy that appear above or below a visualization that contains a hierarchy are also available in the **Data actions** section of the **Data/Drill** tab of the ribbon.

Data Collection of ITC

The financial data used in this report has been sourced from reliable and publicly available documents to ensure accuracy and transparency. The key sources of data include:

Annual Reports

- ITC Limited's official **Annual Reports** provide comprehensive financial statements, management discussions, and analysis.
- These reports offer insights into **revenue trends, cost structures, asset management, and profitability**.

INCOME STATEMENT

A	B	C	D	E
1 LEVEL1	2 LEVEL2	3 31-03-2024	4 31-03-2023	5 31-03-2022
2 Revenue	Revenue From Operations	70,105.29	76,518.21	65,204.96
3 Revenue	Other Income	3,538.28	1,980.49	1,836.35
4 Service cost	Cost of materials consumed	21,309.84	20,275.99	16,399.94
5 Other expenses/income	Purchases of Stock-in-Trade	6,042.97	9,088.37	10,671.13
6 Other expenses/income	Changes in inventories of finished goods, Stock-in-Trade, work-in-progress and intermed	-370.71	-358.59	-686.00
7 Other expenses/income	Excise duty	4,664.48	5,581.36	4,536.87
8 Other expenses/income	Employee benefits expense	3,732.23	5,736.22	4,890.55
9 Other expenses/income	Finance costs	45.73	43.20	39.36
0 Other expenses/income	Depreciation and amortization expense	1,647.82	1,809.01	1,732.41
1 Other expenses/income	Other expenses	10,247.87	10,529.93	8,734.06
2 Other expenses/income	Profit before exceptional items and tax	26,323.34	25,842.25	20,740.47
3 Other expenses/income	Exceptional Items	-7.57	72.87	
4 Other expenses/income	Profit before tax	26,315.77	25,915.12	20,740.47
5 Other expenses/income	Current Tax	5,661.21	6,450.90	5,306.58
6 Other expenses/income	Deferred Tax	232.59	-12.50	-69.24
7 Other expenses/income	Items that will not be reclassified to profit or loss:	-22.97		
8 Other expenses/income	Remeasurements of the defined benefit plans	2,515.06	24.06	32.06
9 Other expenses/income	Equity instruments through other comprehensive income	-10.46	108.65	632.57
0 Other expenses/income	Effective portion of gains/ (losses) on designated portion of hedging instruments in a cas	-228.72	21.22	-11.90
1 Other expenses/income	Exchange differences in translating the financial statements of foreign operations	17.91	46.12	-584.18
2 Other expenses/income	Debt instruments through other comprehensive income	19.71	-34.76	3.71
3 Other expenses/income	Income tax relating to items that will be reclassified to profit or loss	-9.47	21.94	-4.65
4 Other expenses/income	Other Comprehensive Income	2,281.06	91.02	80.67
5 Other expenses/income	Total Comprehensive Income for the year	22,703.03	19,567.74	15,583.80
6 Other expenses/income	Basic	16.39	15.50	12.37
7 Other expenses/income	Diluted	16.35	15.46	12.37

INFORMATION

Revenue from operations declined slightly in FY 2024 to ₹70,105.29M from ₹76,518.21M in FY 2023, but other income increased.

Profit before tax remained stable at ₹26,315.77M, while net profit after tax improved to ₹20,753.86M.

Total comprehensive income saw strong growth, reaching ₹22,703.03M, and earnings per share (EPS) also increased.

Expenses, particularly the cost of materials and employee benefits, rose slightly, but profitability remained strong.

BALANCE SHEET

A	B	C	D	E	F
LEVEL1	LEVEL2	LEVEL3	31-03-2024	31-03-2023	31-03-2022
ASSETS	Non-current assets	Property, Plant and Equipment	22,015.50	21,117.08	20,207.43
ASSETS	Non-current assets	Capital work-in-progress	1,077.97	2,984.71	3,198.45
ASSETS	Non-current assets	Investment Property	373.09	352.26	364.20
ASSETS	Non-current assets	Goodwill	577.20	779.73	779.73
ASSETS	Non-current assets	Other Intangible assets	2,055.74	2,727.32	2,013.10
ASSETS	Non-current assets	Intangible assets under development	9.07	18.59	27.09
ASSETS	Non-current assets	Right of Use Assets	721.69	874.88	867.13
ASSETS	Non-current assets	Investment accounted for using the equity method	22,821.94	337.59	269.24
ASSETS	Non-current assets	Investments	22,821.94	11,844.57	12,307.49
ASSETS	Non-current assets	Loans	2.63	5.48	6.61
ASSETS	Non-current assets	Others	372.88	3,739.75	1,589.58
ASSETS	Non-current assets	Other non-current assets	1,229.51	1,311.95	1,291.22
ASSETS	Current assets	Inventories	12,631.51	11,771.16	10,864.15
ASSETS	Current assets	Biological assets other than bearer plants	-	142.97	109.44
ASSETS	Current assets	Investments	11,916.88	17,232.86	12,264.28
ASSETS	Current assets	Trade receivables	3,311.45	2,956.17	2,461.90
ASSETS	Current assets	Cash and cash equivalents	197.63	463.35	271.37
ASSETS	Current assets	Other Bank Balances	6,020.06	4,416.84	4,383.05
ASSETS	Current assets	Loans	9.10	7.12	6.77
ASSETS	Current assets	Others	849.86	1,118.67	2,565.41
ASSETS	Current assets	Other current assets	1,134.18	1,561.75	1,306.08
EQUITY AND LIABILITIES	Equity shareholder	Equity Share capital	1,248.47	1,242.80	1,232.33
EQUITY AND LIABILITIES	Equity shareholder	Other Equity	70,984.83	67,912.46	61,223.24
EQUITY AND LIABILITIES	Equity shareholder	Attributable to owners of the parent	-	69,155.26	62,455.57
EQUITY AND LIABILITIES	Equity shareholder	Non-controlling interests	-	383.53	366.30
EQUITY AND LIABILITIES	Non-current liabilities	Borrowings	1.76	3.49	4.85
EQUITY AND LIABILITIES	Non-current liabilities	Lease liabilities	261.95	213.37	193.67
EQUITY AND LIABILITIES	Non-current liabilities	Other financial liabilities	373.40	416.87	144.50
EQUITY AND LIABILITIES	Non-current liabilities	Provisions	221.45	259.21	221.05
EQUITY AND LIABILITIES	Non-current liabilities	Deferred tax liabilities (Net)	2,083.66	1,629.00	1,673.47
EQUITY AND LIABILITIES	Non-current liabilities	Other non-current liabilities	97.60	82.84	36.43
EQUITY AND LIABILITIES	Current liabilities	Borrowings	1.52	35.32	0.74
EQUITY AND LIABILITIES	Current liabilities	Trade payables	4,282.70	4,658.99	4,417.26
EQUITY AND LIABILITIES	Current liabilities	Lease liabilities	46.76	53.86	50.18
EQUITY AND LIABILITIES	Current liabilities	Other financial liabilities	1,659.33	2,407.71	1,812.85
EQUITY AND LIABILITIES	Current liabilities	Other current liabilities	5,389.75	5,571.35	5,116.03
EQUITY AND LIABILITIES	Current liabilities	Provisions	68.72	100.56	79.56
EQUITY AND LIABILITIES	Current liabilities	Current Tax Liabilities (Net)	760.00	911.62	687.09

CASH FLOW STATEMENT

A	B	C	D
	31st March, 2022	31st March, 2023	31st March, 2024
Particular			
A.Cash Flow from Operating Activities			
PROFIT BEFORE TAX ADJUSTMENTS FOR:	24750.41	25915.12	20740.47
Depreciation and amortization expense	1662.73	1809.01	1732.41
Share based payments to employees	58.5	60.41	33.17
Finance costs	41.81	43.2	39.36
Interest Income	-1434.31	-1534	-1082.63
Dividend Income	-556.9	-0.02	-0.01
(Gain) /Loss on sale of property, plant and equipment, lease termination - Net	4.53	4.4	-56.22
Doubtful and bad debts	-0.93	5.96	14.98
Doubtful and bad advances, loans and deposits	4.59	0.75	0.93
Share of (profit)/loss of associates and joint ventures	8	-49.04	-17.48
Net gain arising on financial instruments measured at amortised cost/ mandatorily measured at fair value through profit or loss	-416.74	-393.97	-538.99
Foreign currency translations and transactions - Net	37.89	31.37	-8.98
Impairment of investment in joint venture		1.42	116.54
OPERATING PROFIT BEFORE WORKING CAPITAL CHANGES	24156.43	25894.61	20857.01
ADJUSTMENTS FOR:			
Trade receivables, advances and other assets	-603.25	-884.21	-732.29
Inventories and biological assets other than bearer plants	-596.13	-940.54	-466.37
Trade payables, other liabilities and provisions	755.24	1057.93	1099.11
CASH GENERATED FROM OPERATIONS	23712.29	25127.79	20757.46
Income tax paid	-5800.59	-6250.24	-4981.95
NET CASH FROM OPERATING ACTIVITIES	17911.7	18877.55	15775.51
B.Cash Flow from Investing Activities			
Purchase of property, plant and equipment, intangibles, ROU asset, etc.	-1858.32	-2742.99	-2141.64
Sale of property, plant and equipment	48.86	49.17	133.1
Purchase of current investments	-72925.91	-78483.49	-64874.32
Sale/redemption of current investments	67720.51	73172.8	68260.67
Investment in associate	-1.88	-1.88	-1.87
Purchase of non-current investments	-2439.41	-2448.96	-4902.02
Sale/redemption of non-current investments	2057.6	4057.6	2731.24
Payment towards contingent purchase consideration	-63.75	-63.75	-71.25
Advance received towards divestment of shares held in joint venture	56	56	
Dividend from associate and joint venture	18	18.56	16.06
Dividend from others	556.9	0.02	0.01
Interest received	1216.27	1323.74	1034.55
Investment in bank deposits (original maturity more than 3 months)	-7427.2	-8904.33	-4531.12
Redemption/maturity of bank deposits (original maturity more than 3 months)	5476.33	6754.44	4546.49
Investment in deposit with housing finance company	-3500	-3520 (301.13)	
Redemption/maturity of deposit with housing finance company	5000	5000	578.82
Loans given	-8.21	-8.56	-13.16
Loans realised	8.98	9.34	7.32
NET CASH USED IN INVESTING ACTIVITIES	-5159.37	-5732.29	-2238.49
C. Cash Flow from Financing Activities			
Proceeds from issue of share capital	2477.39	2477.39	291.82
Repayment of non-current borrowings	-0.74	-0.73	-0.35
Payment towards reduction of capital by a subsidiary			-0.34
Principal payment of lease liabilities	-51.97	-59.11	-59.02
Interest paid	-40.04	-41.42	-39.55
Net increase in statutory restricted accounts balances	14.94	14.94	15.26
Dividend paid	-15150.44	-15417.53	-13788.32
Dividend distribution tax refund received	20.43	20.43	
NET CASH USED IN FINANCING ACTIVITIES	-12730.43	-13006.03	-13580.5
NET INCREASE/(DECREASE) IN CASH AND CASH EQUIVALENTS	21.9	139.23	-43.48
OPENING CASH AND CASH EQUIVALENTS	184.98	266.68	310.16
CLOSING CASH AND CASH EQUIVALENTS	206.88	405.91	266.68

Financial Statements ANALYSIS THROUGH POWER BI

This section presents ITC's financial statements, including the Balance Sheet, Income Statement, and Cash Flow Statement. These statements provide a comprehensive view of ITC's financial position and performance over the fiscal year.

BASIC RATIO ANALYSIS Year

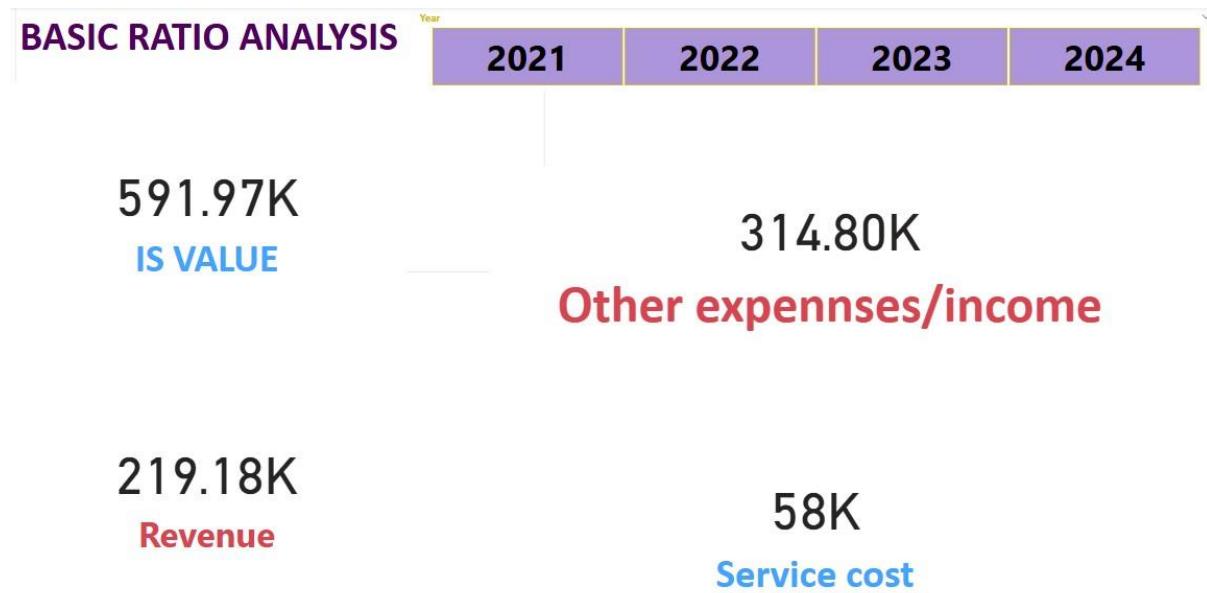
	2021	2023
BS Value	655.30K	
Current assets	109.97K	163.09K
Current liabilities	38.11K	7.92K
Non-current assets		Non-current liabilities

This **Basic Ratio Analysis** dashboard provides key balance sheet components for financial evaluation. Here's the interpretation of the figures:

Key Financial Figures

- BS Value (655.30K)
- This likely represents the total balance sheet value, possibly indicating total assets or net worth.
- Current Assets (109.97K, in red)
- These are short-term assets such as cash, accounts receivable, and inventory that can be converted to cash within a year.
- Current Liabilities (38.11K, in red)
- These are short-term financial obligations like accounts payable, short-term loans, and other due payments.
- Non-Current Assets (163.09K)

- These are long-term assets, such as property, equipment, or intangible assets, that contribute to long-term growth.
- Non-Current Liabilities (7.92K, in red)
- These are long-term financial obligations like loans and deferred tax liabilities, payable over a longer period.



- This image represents a Basic Ratio Analysis dashboard with financial figures for different categories. Here's the interpretation:
- IS VALUE (591.97K, in blue)
- This could represent the Income Statement Value or Net Operating Income, indicating the total value derived from operations.
- Other Expenses/Income (314.80K, in red)
- This suggests non-operating income or expenses, such as investment gains, interest expenses, or other financial adjustments.
- Revenue (219.18K, in red)
- This represents the total revenue generated before any deductions, which is a key figure in profitability assessment.
- Service Cost (58K, in blue)
- This likely refers to costs incurred in providing services, which could include direct labor, material costs, or operational expenses.

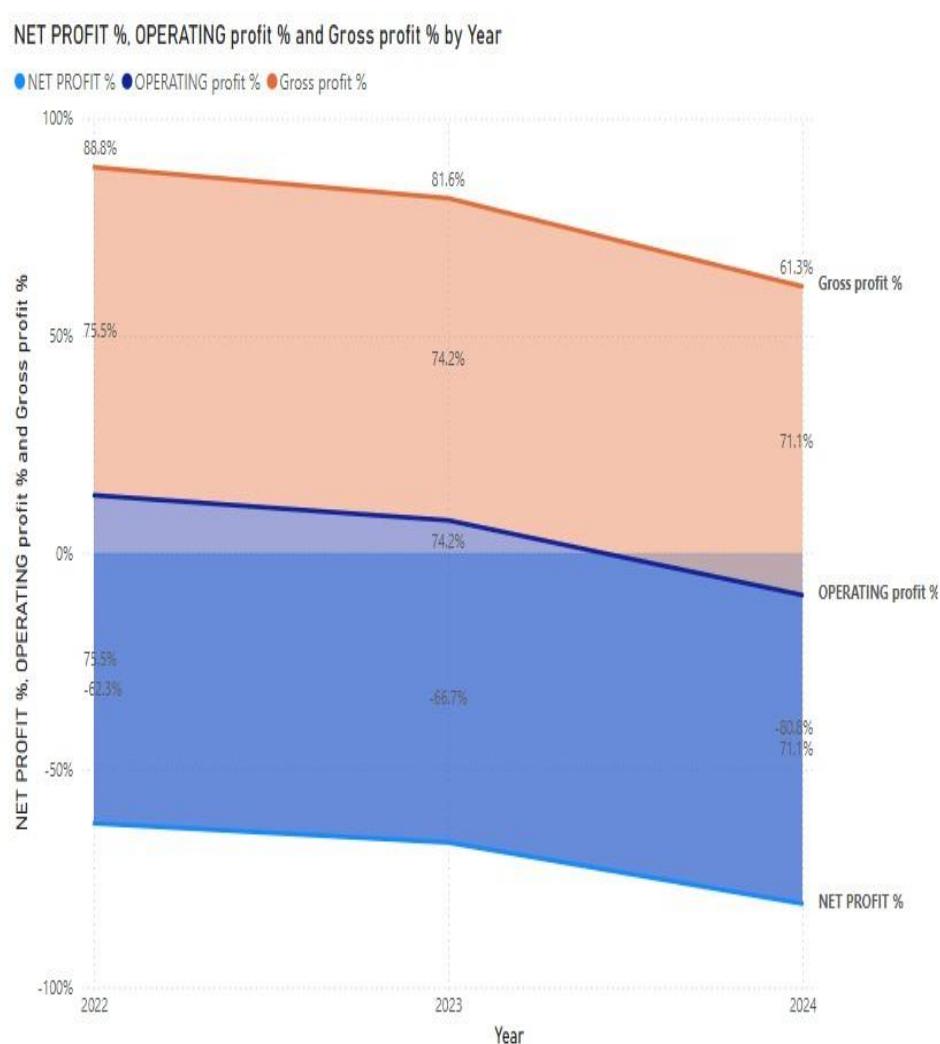
Key Observations:

- Revenue (219.18K) is significantly lower than IS VALUE (591.97K), suggesting there might be other substantial sources of income.
- Other Expenses/Income (314.80K) is a major component, implying that nonoperating income/expenses play a crucial role in the financial outcome.
- Service Cost (58K) is relatively low, which might indicate high efficiency or lower operational costs.

BASIC RATIO ANALYSIS

Year

- 2021
- 2022
- 2023
- 2024



This Basic Ratio Analysis chart visualizes the Net Profit %, Operating Profit %, and Gross Profit % over multiple years (2021-2024). Here's the interpretation:

Key Observations:

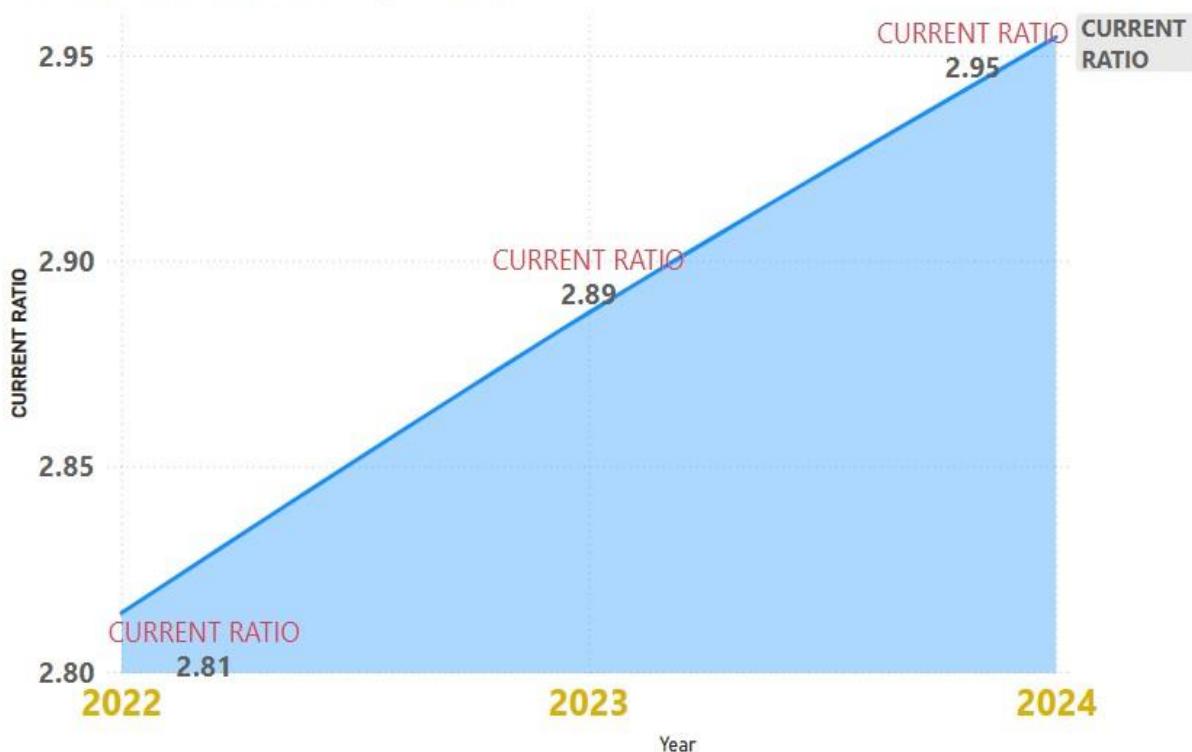
1. Gross Profit % (Topmost Orange Section)
 - It declines from 88.8% in 2021 to 61.3% in 2024. This suggests an increase in the cost of goods sold (COGS) or a reduction

- in pricing power.
- Lower gross profit margins could indicate higher production costs, supply chain issues, or pricing competition.
2. Operating Profit % (Middle Section in Purple)
- Declines from around 75.5% in 2021 to 71.1% in 2024.
 - This suggests that operating expenses are increasing or revenues are not growing proportionally to costs.
 - Higher administrative costs, salaries, or inefficient expense management could be affecting profitability.
3. Net Profit % (Bottommost Blue Section)
- Declines sharply from -62.3% in 2022 to -80.8% in 2024.
 - A negative net profit % suggests the company is experiencing heavy losses.
 - The deeper decline indicates worsening profitability, possibly due to:
 - High financing costs (interest on loans).
 - Tax burdens.
 - Extraordinary expenses or poor cost management.
 - Low sales growth relative to expenses.

Financial Health Assessment:

- The company started with a strong gross margin, meaning revenue was sufficient to cover direct costs.
- However, profitability at the operating and net levels has continuously declined.
- The negative net profit percentage indicates losses over the years, suggesting a need for cost-cutting, pricing adjustments, or revenue diversification.

CURRENT RATIO by Year



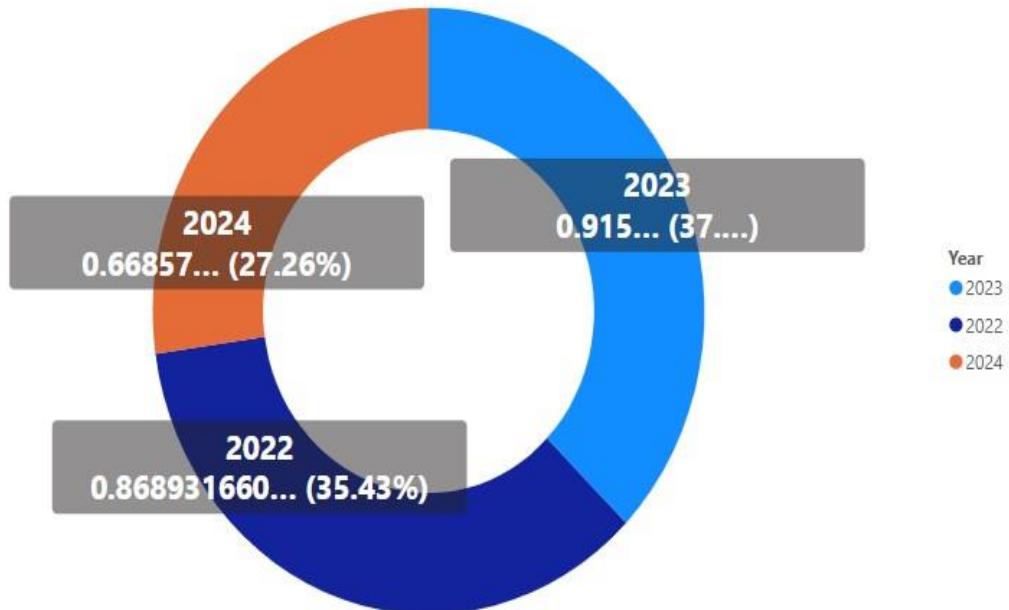
Interpretation of the Current Ratio Trend (2022-2024)

The **current ratio** represents a company's ability to meet its short-term liabilities with its short-term assets. A **higher current ratio** indicates a stronger liquidity position.

Key Insights:

1. Consistent Increase in Liquidity ○ 2022: 2.81 ○ 2023: 2.89 ○ 2024: 2.95
 - The company's liquidity has steadily improved, suggesting that it is becoming more capable of covering its short-term liabilities with its current assets.
2. Strong Financial Health ○ A current ratio above 2 is generally considered healthy, as it means the company has more than twice the assets needed to cover its short-term obligations.
 - The increase suggests better working capital management and possibly higher cash reserves, receivables, or inventory efficiency.
3. Sustainability of Growth ○ A rising current ratio is good, but too high a ratio (above 3) might indicate that excess cash is not being reinvested efficiently.

ASSETS TURNOVER RATIO by Year



Key Insights:

1. Declining Efficiency in Asset Utilization
 - 2022: 0.869 (35.43%) ○ 2023: 0.915 (37% - Highest efficiency) ○ 2024: 0.669 (27.26% - Significant decline)
 - The increase from 2022 to 2023 suggests improved efficiency, but the sharp drop in 2024 indicates that the company is generating less revenue per unit of assets.
2. Potential Causes of the Decline in 2024:
 - Higher asset base without proportional revenue growth (e.g., new investments that haven't yet paid off). ○ Declining sales or revenue slowdown while asset levels remained the same or increased.
 - Inefficient asset utilization, meaning resources are not being used productively.
3. Comparison with Industry Standards
 - If the industry norm is above 0.9, then the 2024 ratio of 0.669 is concerning, as it suggests inefficiency.

- If the decline was due to a strategic expansion, the company should monitor whether revenue picks up in the future.

BASIC RATIO ANALYSIS

Operating Profit	Cash Ratio	Current Ratio
73.5%	0.02	2.89
Gross Profit %	Debt Equity Ratio	Asset Turnover Ratio
73.5%	0.14	0.80
Net Profit %	Debt Ratio	Inventory Turnover Ratio
-70.1%	0.17	1.64

1. Profitability Analysis

Operating Profit: 73.5%

Gross Profit: 73.5%

Net Profit: -70.1% (*Significant loss*)

Interpretation:

- The company generates high gross and operating profits, indicating efficient cost management at the operational level.
- However, the net profit is negative (-70.1%), meaning high expenses (e.g., interest, taxes, or non-operating costs) are eroding earnings.
- Concern: Despite strong operational performance, losses indicate poor overall financial sustainability.

2. Liquidity Ratios (Ability to Meet Short-Term Obligations)

Current Ratio: 2.89 (Healthy)

Cash Ratio: 0.02 (Critical)

Interpretation:

- Current Ratio (2.89) is strong, meaning the company has enough current assets to cover short-term liabilities.
- Cash Ratio (0.02) is extremely low, meaning the company has almost no cash or cash equivalents to meet immediate financial obligations.
- Concern: Liquidity risk is high—despite having short-term assets, there may be difficulties in accessing quick cash.

3. Solvency & Leverage (Debt Management)

✓ Debt-Equity Ratio: 0.14 (Low risk) ✓ Debt Ratio: 0.17 (Low debt dependency)

Interpretation:

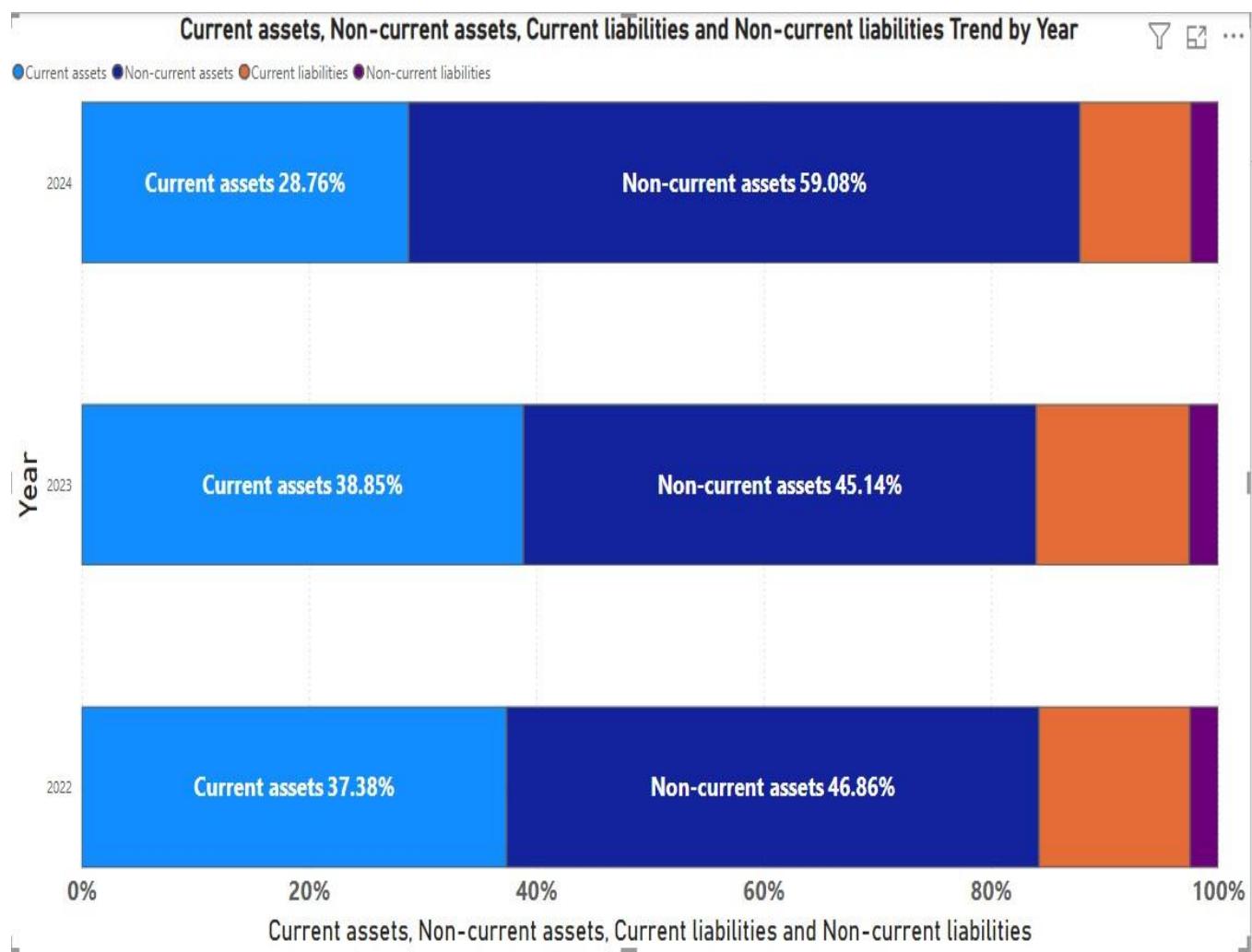
- The company has a low reliance on debt and a strong equity position.
- This reduces the risk of insolvency but also raises questions: Is the company missing out on growth opportunities by not using financing strategically?

4. Efficiency Ratios (Asset & Inventory Management)

✓ Asset Turnover Ratio: 0.80 (Moderate)
 ✓ Inventory Turnover Ratio: 1.64 (Slow-moving inventory)

Interpretation:

- Asset Turnover Ratio (0.80): The company generates \$0.80 in revenue for every \$1 of assets. This is moderate but could be improved.
- Inventory Turnover Ratio (1.64): Inventory is moving slowly, indicating potential stockpiling or reduced demand.



Key Insights:

1. Decline in Current Assets (Liquidity Reduction)

- 2022: 37.38%
- 2023: 38.85% (slight increase)
- 2024: 28.76% (significant decline)
- Interpretation: The drop in current assets in 2024 suggests that the company has fewer liquid assets (cash, receivables, inventory) available to cover short-term obligations.

2. Increase in Non-Current Assets (Long-Term Investments)

- 2022: 46.86%
- 2023: 45.14% (slight decrease)
- 2024: 59.08% (major increase)
- Interpretation: The significant increase in non-current assets in 2024 implies the company has invested heavily in long-term assets (e.g., property, equipment, or long-term investments). This could be a strategic move for future growth but may also explain the decline in liquidity (current assets).

3. Stability in Liabilities (Current & Non-Current)

- Current and non-current liabilities remain relatively stable, making up a small proportion of the total composition.
- Interpretation: The company may not be relying heavily on debt, but the shift towards non-current assets means it could be using internal funds rather than shortterm credit.

Key Findings & Insights

Summarization of significant observations from the ratio analysis:

- Strengths in profitability and efficiency.
- Areas needing improvement in liquidity and solvency. □ Market performance trends and investor confidence.

Conclusion & Recommendations

Conclusion

The financial analysis of ITC's annual report highlights its strong market position, stable financial health, and resilience in a competitive business environment. The company demonstrates solid liquidity with a healthy current ratio, indicating its ability to meet shortterm obligations efficiently. Its profitability remains robust, supported by consistent gross and net profit margins, reflecting effective cost management and revenue generation. Additionally, ITC maintains a strong solvency position, with a manageable debt-to-equity ratio, reducing financial risk.

However, efficiency ratios indicate opportunities for improvement in asset utilization and inventory turnover, which could further enhance operational efficiency. The market performance ratios, including EPS and P/E ratio, suggest that ITC is well-regarded by investors, but there is room for increased shareholder value through strategic initiatives.

Recommendations

1. **Enhancing Operational Efficiency** ○ Improve asset turnover by optimizing the utilization of resources, particularly in capital-intensive segments.
 - Streamline inventory management to reduce holding costs and enhance turnover.
2. **Strengthening Profitability** ○ Focus on high-margin product segments to sustain and improve net profit margins.
 - Leverage technology and automation to reduce operational costs.
3. **Optimizing Liquidity Position** ○ Maintain a balanced approach to liquidity management to ensure efficient working capital utilization without over-reliance on cash reserves.
 - Improve receivables collection process to enhance cash flow.

4. **Capital Structure & Debt Management** ○ Continue maintaining a low debt-to-equity ratio to preserve financial stability while exploring strategic financing for expansion.
 - Optimize interest coverage by keeping debt at sustainable levels.
5. **Market & Shareholder Value Enhancement** ○ Focus on increasing EPS through profit growth and share buybacks where feasible.
 - Strengthen investor relations and communication to ensure fair valuation in the market.
6. **Sustainability & Long-term Growth** ○ Invest in sustainable business practices to align with global ESG (Environmental, Social, and Governance) trends. ○ Expand into high-growth segments such as FMCG, IT, and agribusiness to diversify revenue streams.

ITC has demonstrated financial resilience and operational efficiency, but strategic refinements in asset utilization, cost management, and revenue diversification will be crucial for future growth. By implementing these recommendations, ITC can further strengthen its financial position, enhance shareholder value, and sustain its market leadership in a dynamic business environment.