**Aim:- Perform the Extraction Transformation and Loading (ETL)**

**process to construct the database in the Sqlserver / Power BI.**

**1. Importing Data from OData Feed**

In this task, you'll bring in order data. This step represents connecting to a sales system. You

import data into Power BI Desktop from the sample Northwind OData feed at the following

URL, which you can copy (and then paste) in the steps below:

**http://services.odata.org/V3/Northwind/Northwind.svc/**

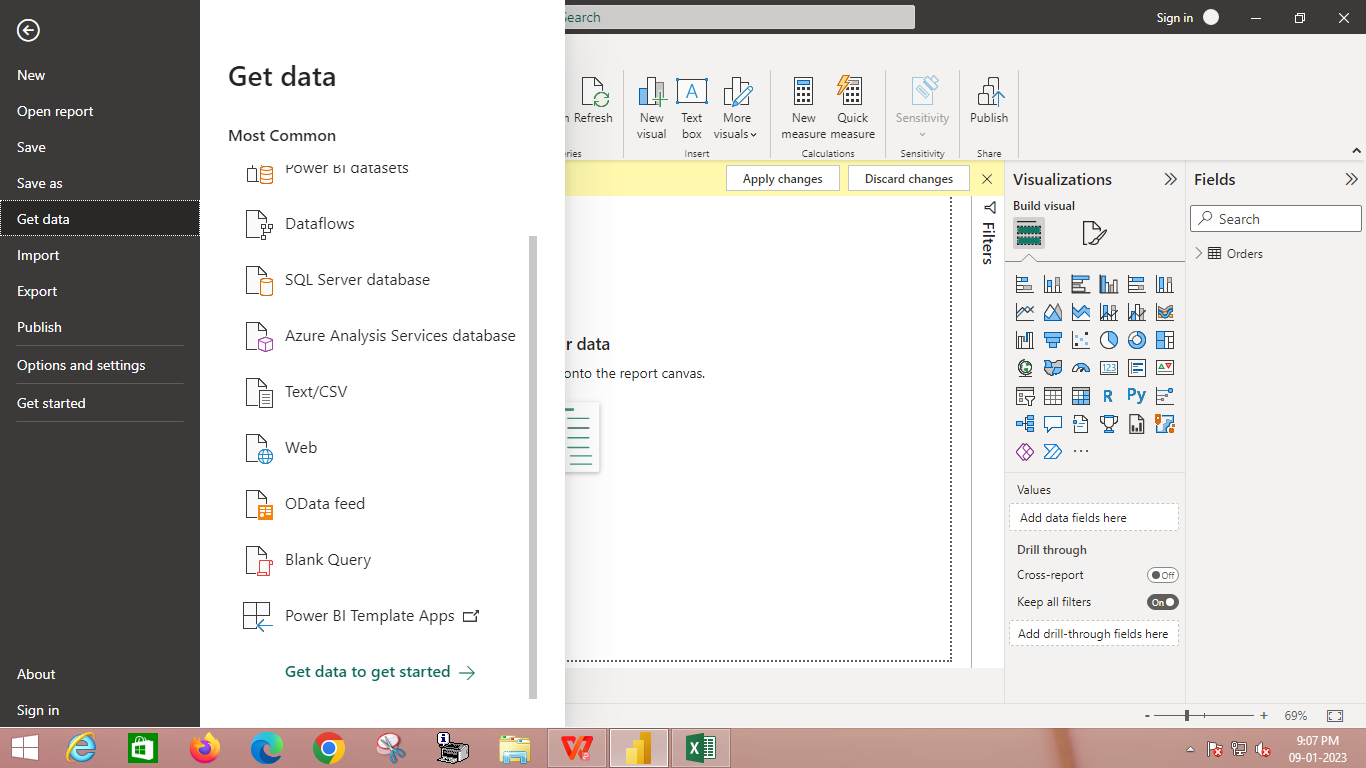
Connect to an OData feed:

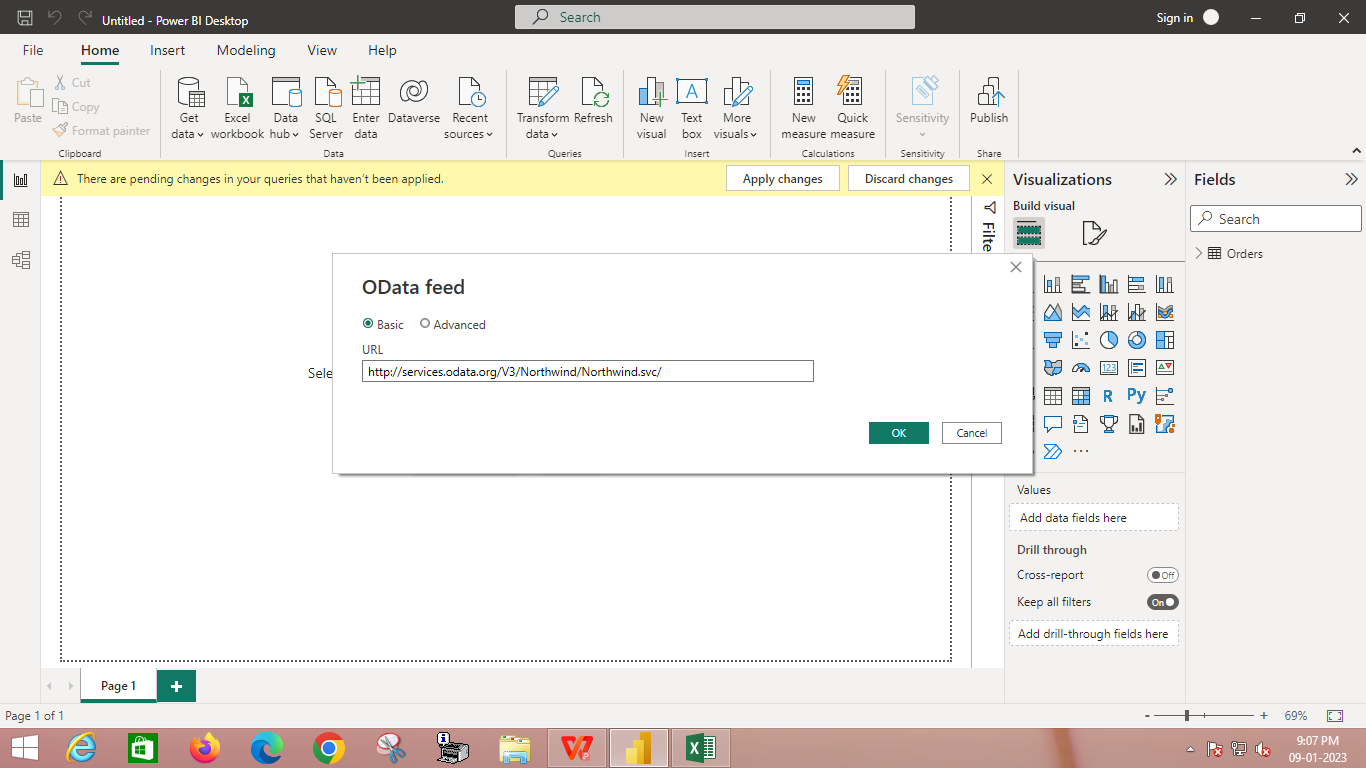
1) From the Home ribbon tab in Query Editor, select Get Data.

2) Browse to the OData Feed data source.

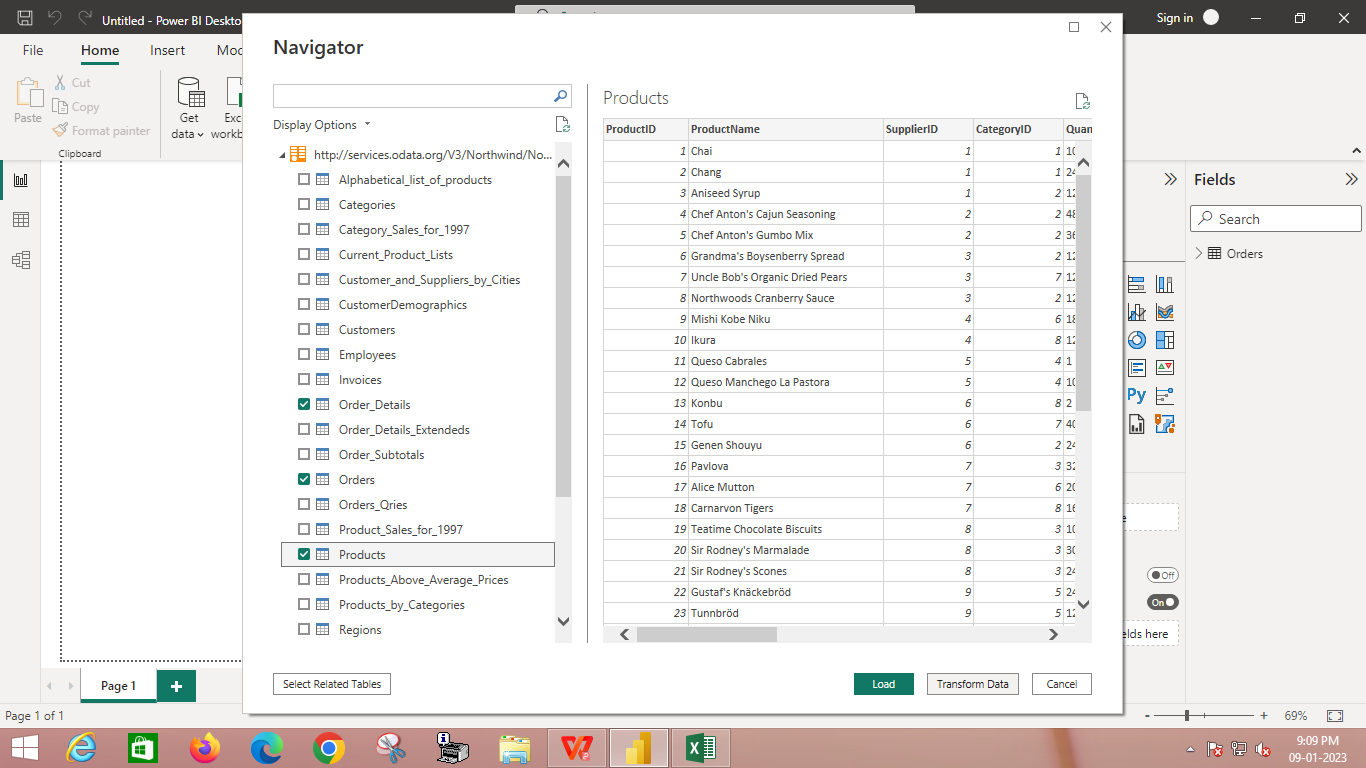
3) In the OData Feed dialog box, paste the URL for the Northwind OData feed.

4) Select OK.-

****

****

**2. Select the order details, order and product**

****

**3. ETL Process in Power BI**

**1) Remove other columns to only display columns of interest**

In this step you remove all columns except **ProductID**, **ProductName**, **UnitsInStock**, and

**QuantityPerUnit**

Power BI Desktop includes Query Editor, which is where you shape and transform your data

connections. Query Editor opens automatically when you select **Edit** from Navigator. You can

also open the Query Editor by selecting Edit Queries from the Home ribbon in Power BI

Desktop. The following steps are performed in Query Editor.

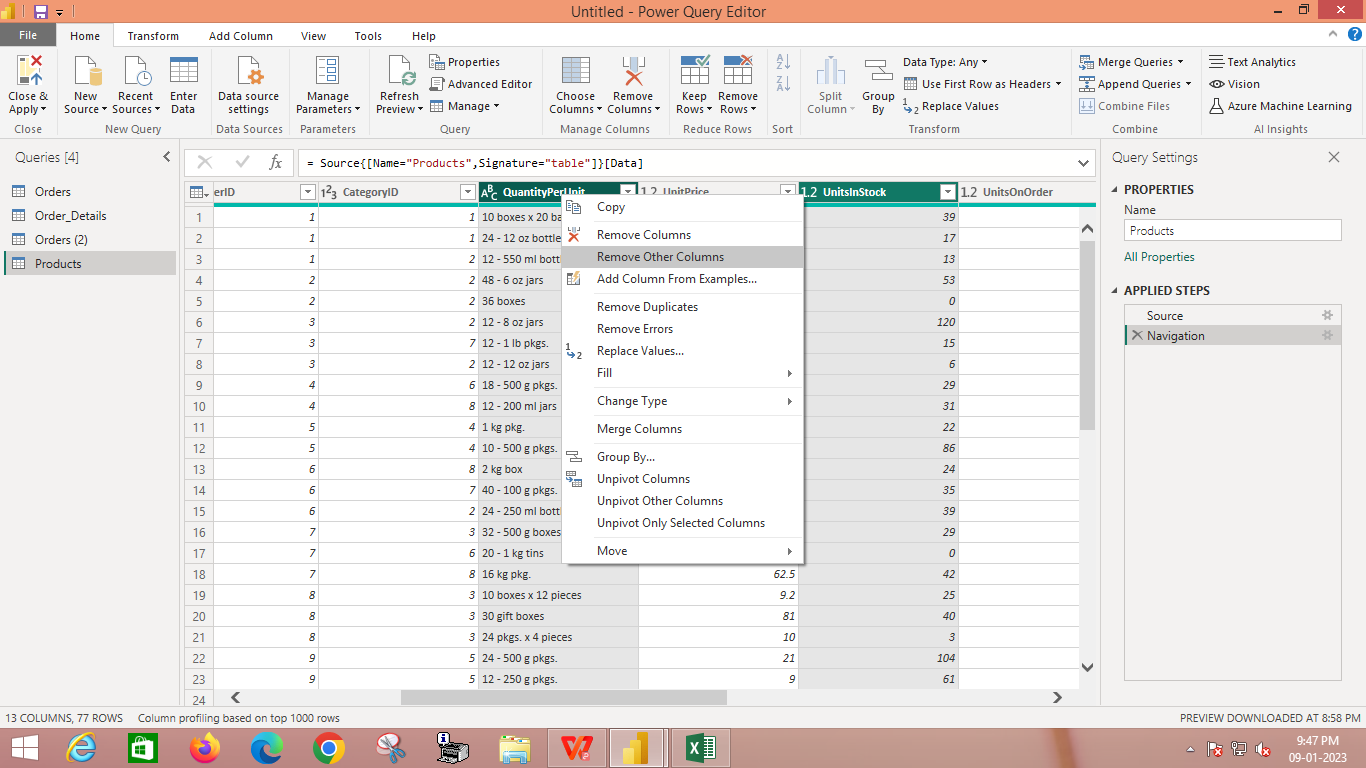
1. In **Query Editor**, select the **ProductID, ProductName**, **QuantityPerUnit,** and

**UnitsInStock** columns (use **Ctrl+Click** to select more than one column, or

**Shift+Click** to select columns that are beside each other).

2. Select **Remove Columns > Remove** Other Columns from the ribbon, or right-click

on a column header and click Remove Other Columns.

****

**4. Change the data type of the UnitsInStock column**

When Query Editor connects to data, it reviews each field and to determine the best data type.

For the Excel workbook, products in stock will always be a whole number, so in this step you

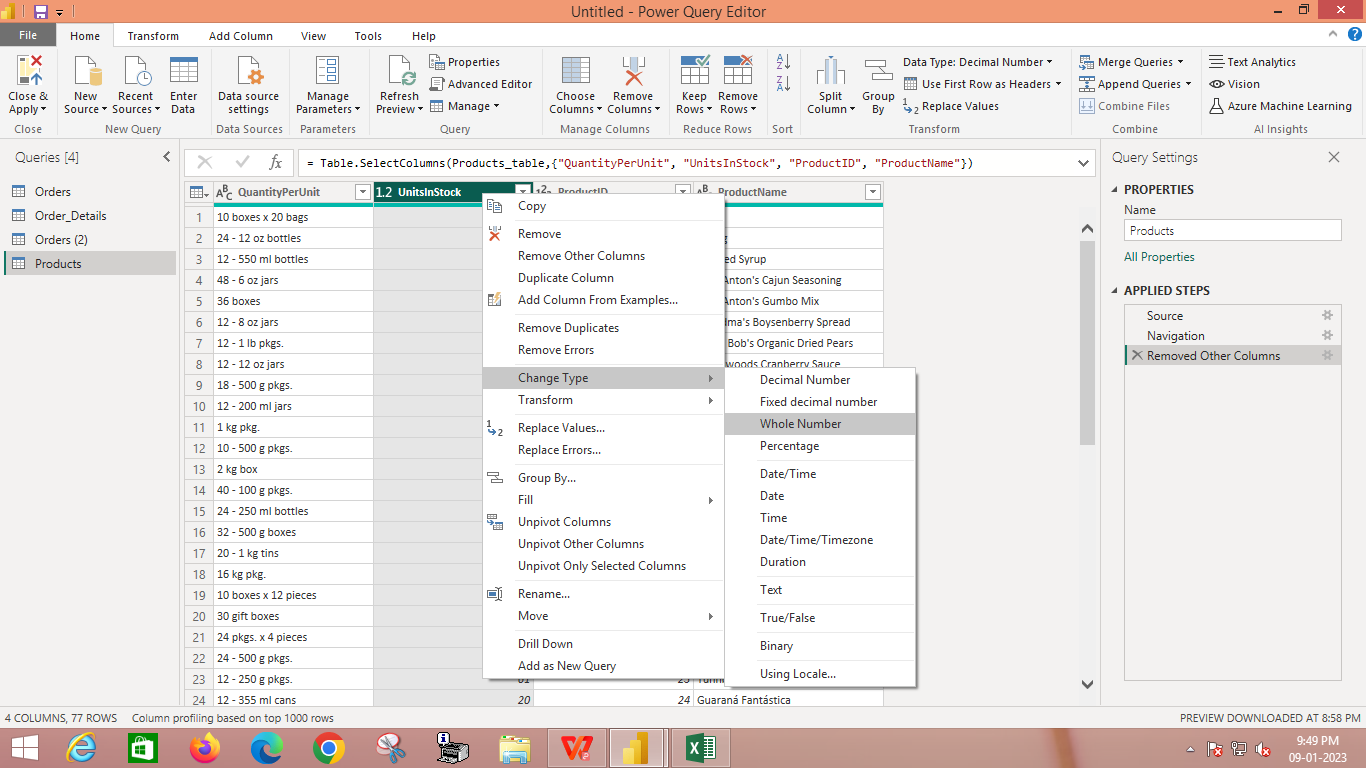
confirm the **UnitsInStock** column’s datatype is Whole Number.

1. Select the **UnitsInStock** column.

2. Select the **Data Type drop-down button** in the **Home ribbon**.

3. If not already a Whole Number, select **Whole Number** for data type from the drop down

(the Data Type: button also displays the data type for the current selection).

****

**5. Expand the Order\_Details table**

The Orders table contains a reference to a Details table, which contains the individual products

that were included in each Order. When you connect to data sources with multiples tables (such

as a relational database) you can use these references to build up your query

In this step, you expand the **Order\_Details** table that is related to the Orders table, to combine

the **ProductID**, **UnitPrice**, and **Quantity** columns from **Order\_Details** into the **Orders table**.

This is a representation of the data in these tables:

The Expand operation combines columns from a related table into a subject table. When the

query runs, rows from the related table (**Order\_Details**) are combined into rows from the

subject table (**Orders**).

After you expand the Order\_Details table, three new columns and additional rows are added

to the Orders table, one for each row in the nested or related table.

1. In the Query View, scroll to the Order\_Details column.

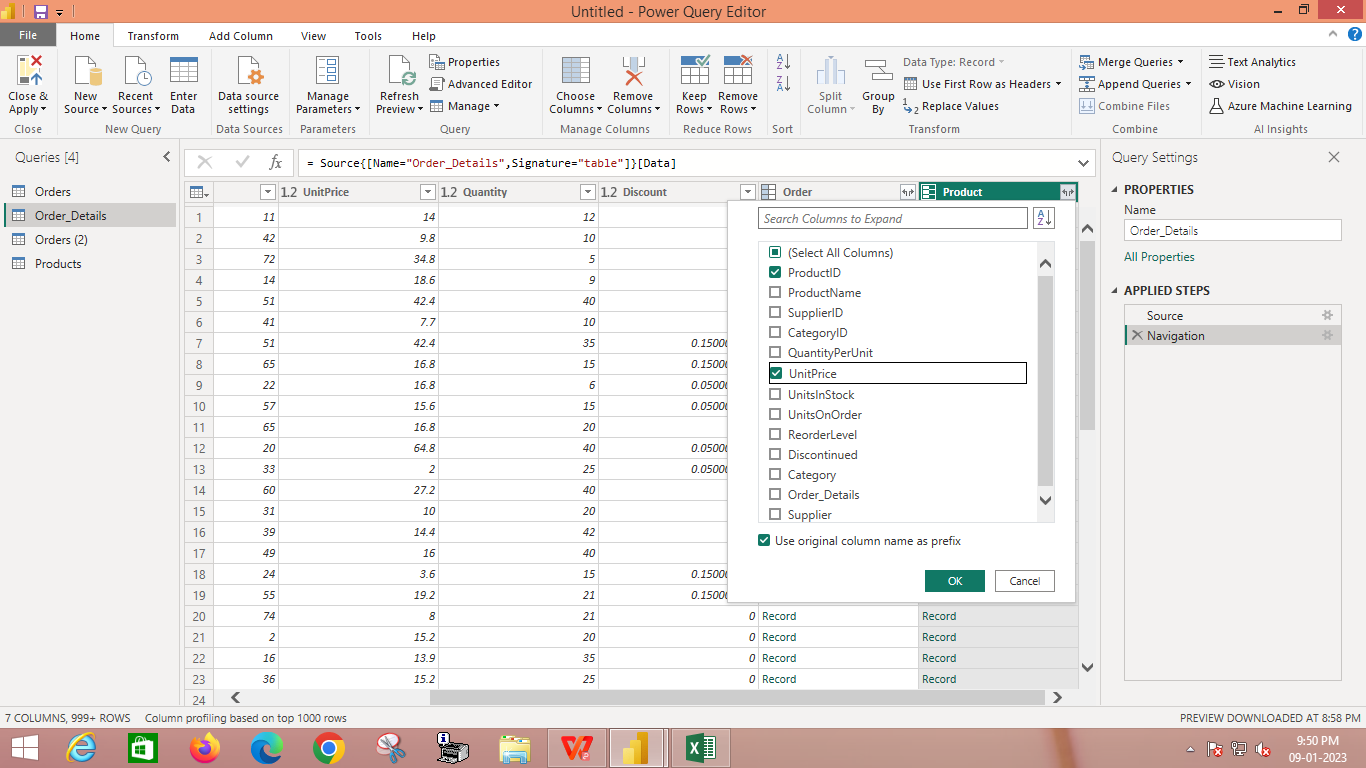
2. In the Order\_Details column, select the expand icon ( ).

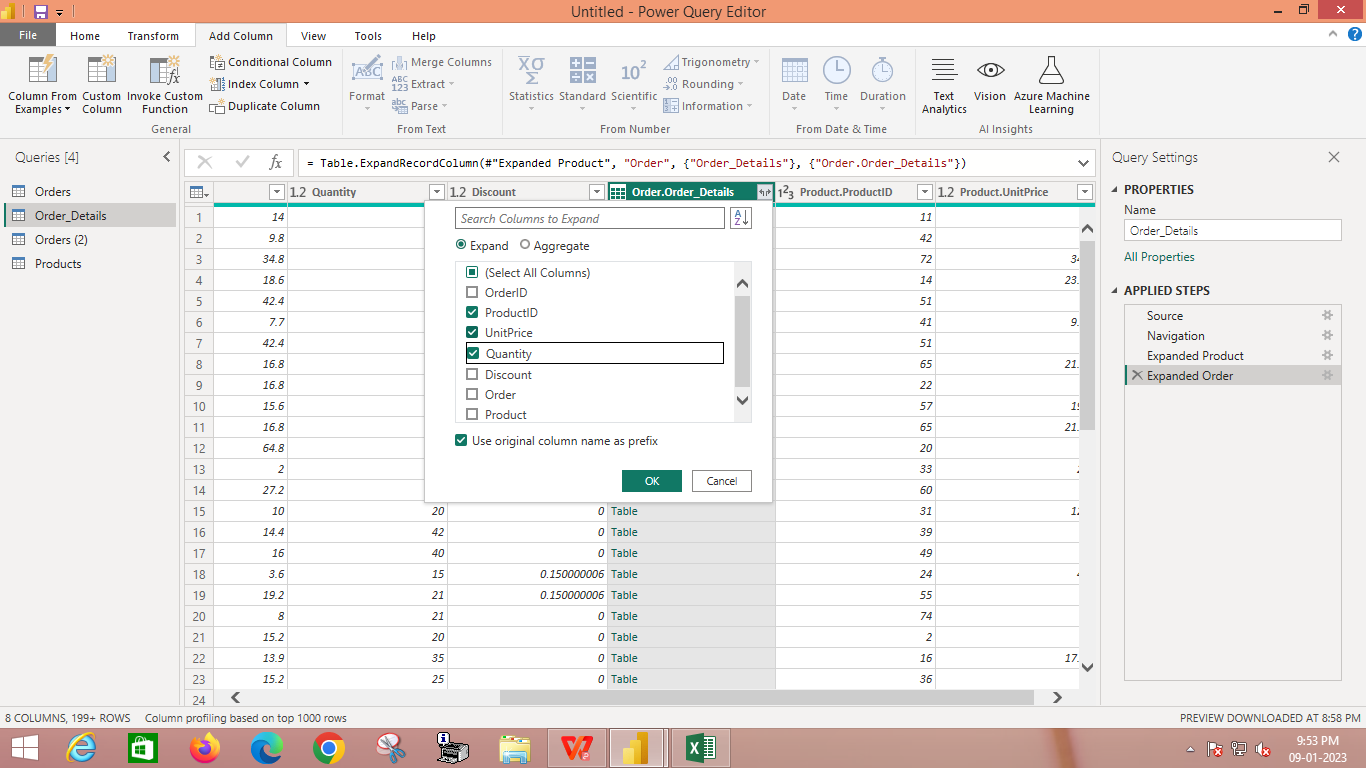
3. In the Expand drop-down:

a. Select (Select All Columns) to clear all columns.

b. Select ProductID, UnitPrice, and Quantity.

c. Click OK.

****

****

**6. Calculate the line total for each Order\_Details row**

Power BI Desktop lets you to create calculations based on the columns you are importing, so

you can enrich the data that you connect to. In this step, you create a Custom Column to

calculate the line total for each Order\_Details row.

Calculate the line total for each Order\_Details row:

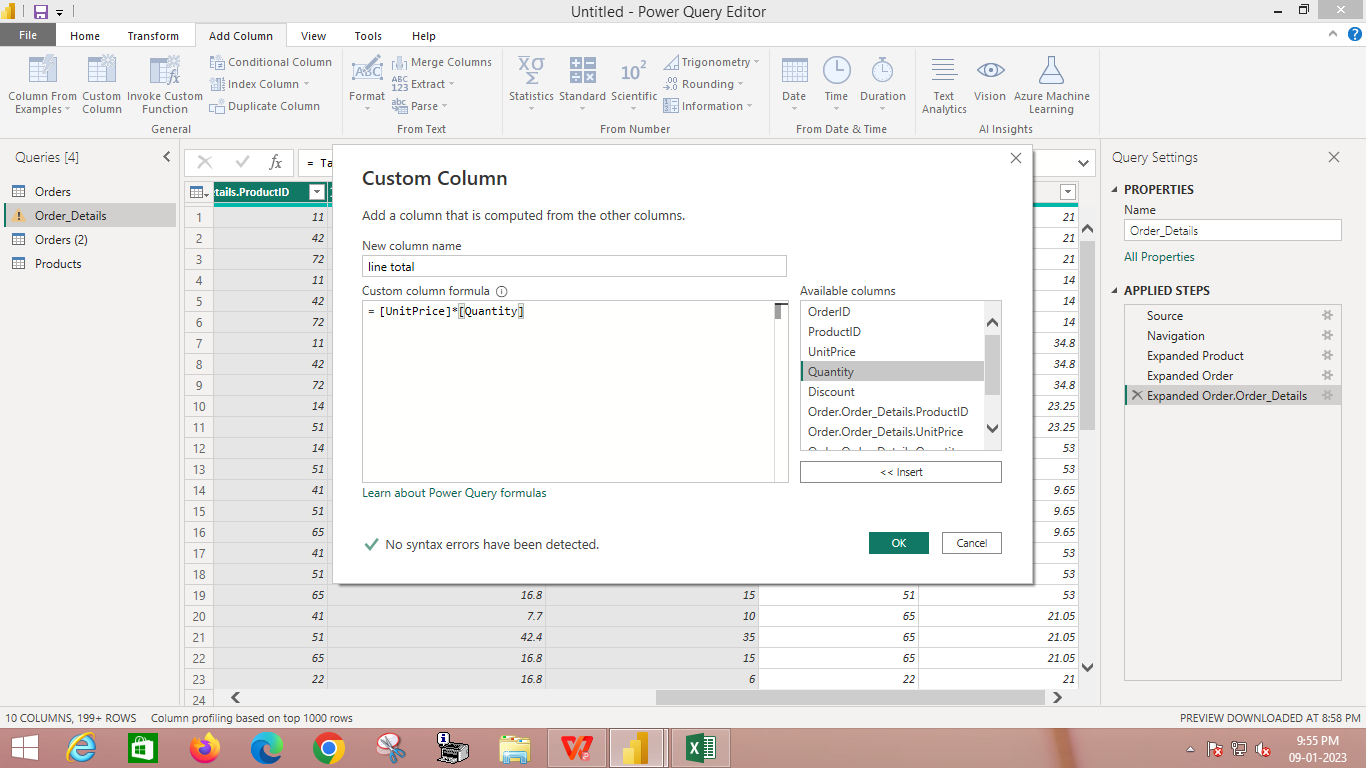
1. In the Add Column ribbon tab, click Add Custom Column.

2. In the Add Custom Column dialog box, in the Custom Column Formula textbox, enter

[Order\_Details.UnitPrice] \* [Order\_Details.Quantity].

3. In the New column name textbox, enter LineTotal.

4. Click OK.

****

**7. Combine the Products and Total Sales queries**

Power BI Desktop does not require you to combine queries to report on them. Instead, you

can create Relationships between datasets. These relationships can be created on any column

that is common to your datasets

we have Orders and Products data that share a common 'ProductID' field, so we need to

ensure there's a relationship between them in the model we're using with Power BI Desktop.

Simply specify in Power BI Desktop that the columns from each table are related (i.e.

columns that have the same values). Power BI Desktop works out the direction and

cardinality of the relationship for you. In some cases, it will even detect the relationships

automatically.

In this task, you confirm that a relationship is established in Power BI Desktop between the

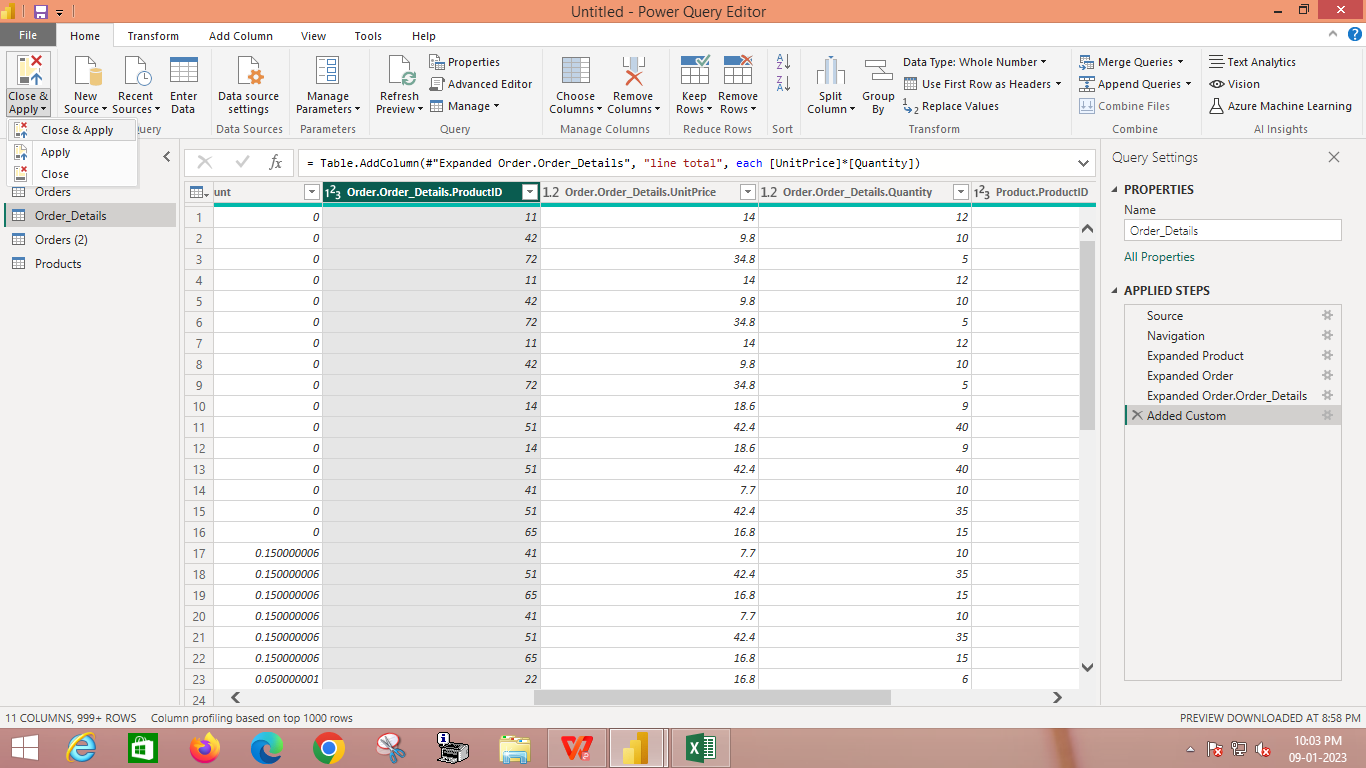
Products and Total Sales queries

Step 1:

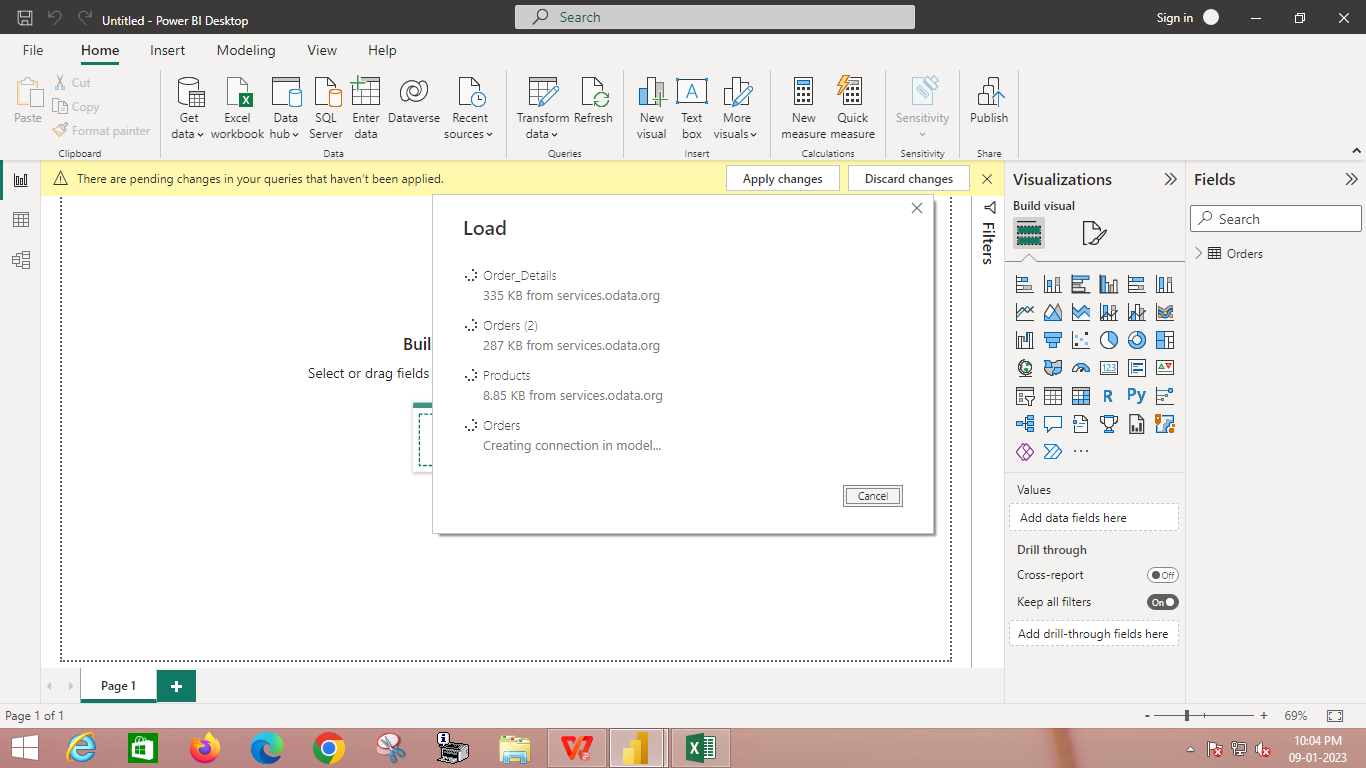
Confirm the relationship between Products and Total Sales

1. First, we need to load the model that we created in Query Editor into Power BI

Desktop. From the Home ribbon of Query Editor, select Close & Load.

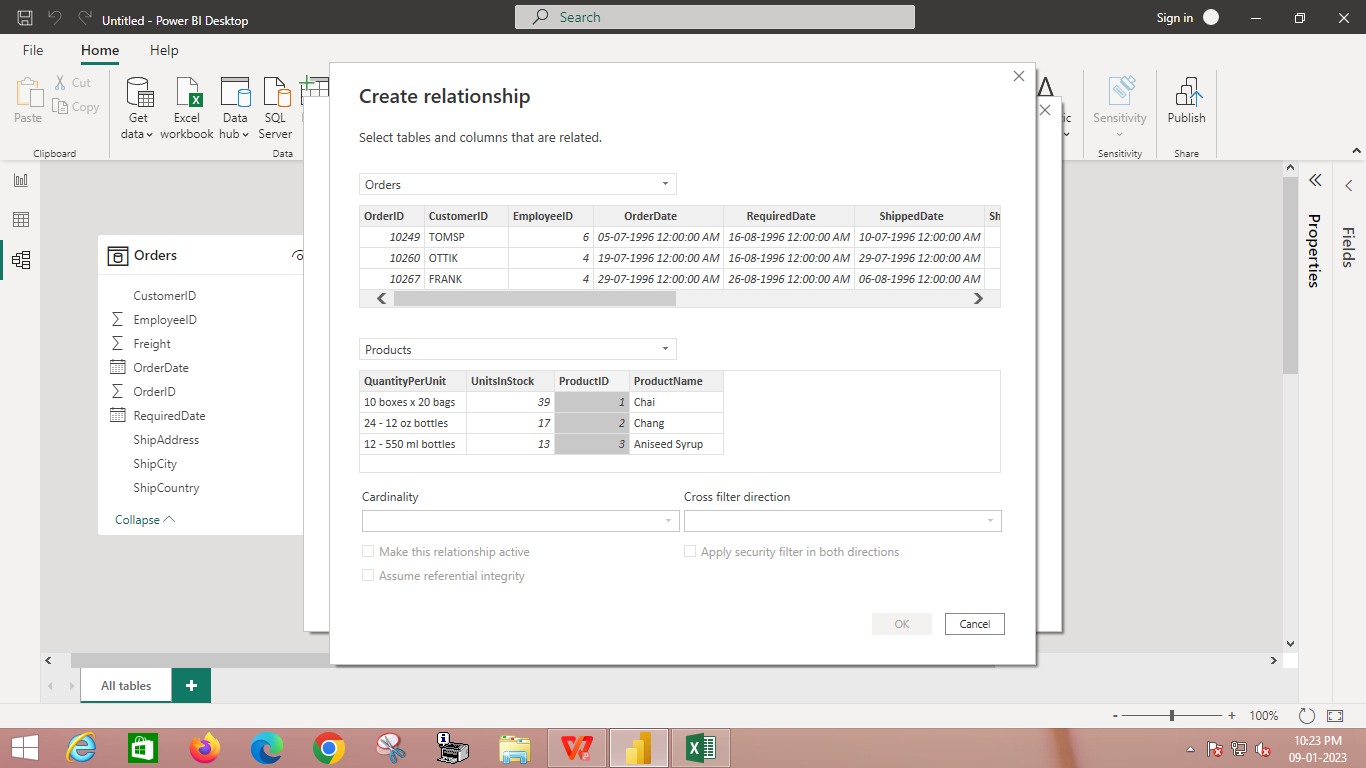
****

2. Power BI Desktop loads the data from the two queries.

****

3. Once the data is loaded, select the Manage Relationships button Home ribbon.

Select the New… button

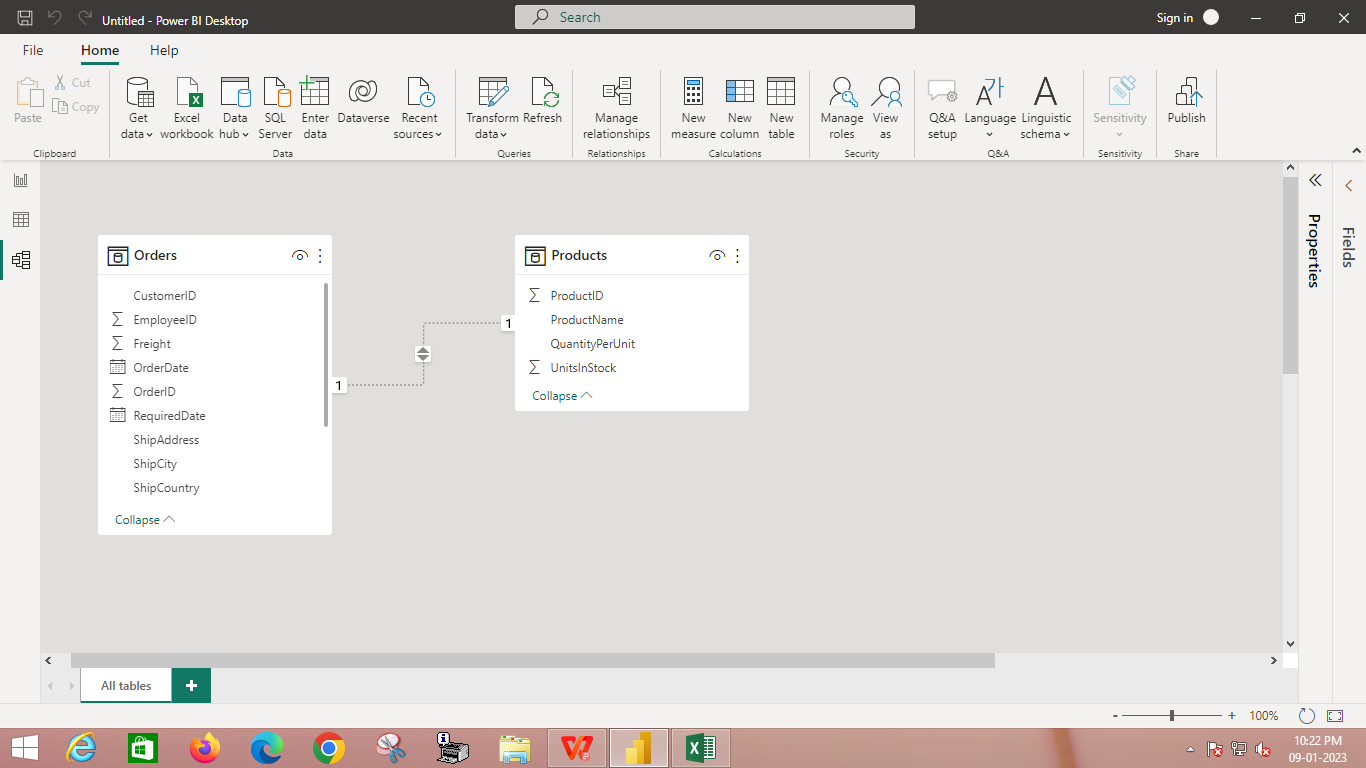


4.When we attempt to create the relationship, we see that one already exists! As shown in the

Create Relationship dialog (by the shaded columns), the ProductsID fields in each query

already have an established relationship.

Select Cancel, and then select Relationship view in Power BI Desktop.

****