

DATA MANIPULATION AND REPORTING WITH POWER BI

Contents

Purpose of this lab.....	
Data transformation	
Change datatype.....	
Capitalize column	
Remove columns.....	
Rename columns.....	
Split columns.....	
Assign data category.....	
Merge queries.....	
Append queries.....	
Manage relationships	
Auto-detect relations.....	
Creating Custom Relationships.....	

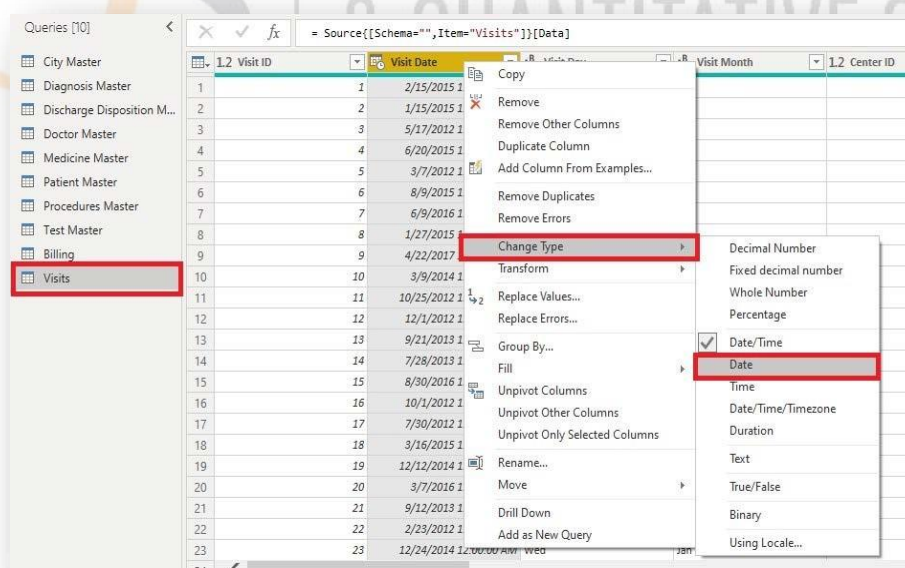
Purpose of this lab

After completing this lab, you should be able to apply different types of data transformations using PowerBI desktop so that the data inconsistencies can be removed. You also would be able to learn how to model your data, imported from various sources. In this section you'd learn how to create relationships between different tables either using the auto-detect option or by creating them manually.

Data transformation

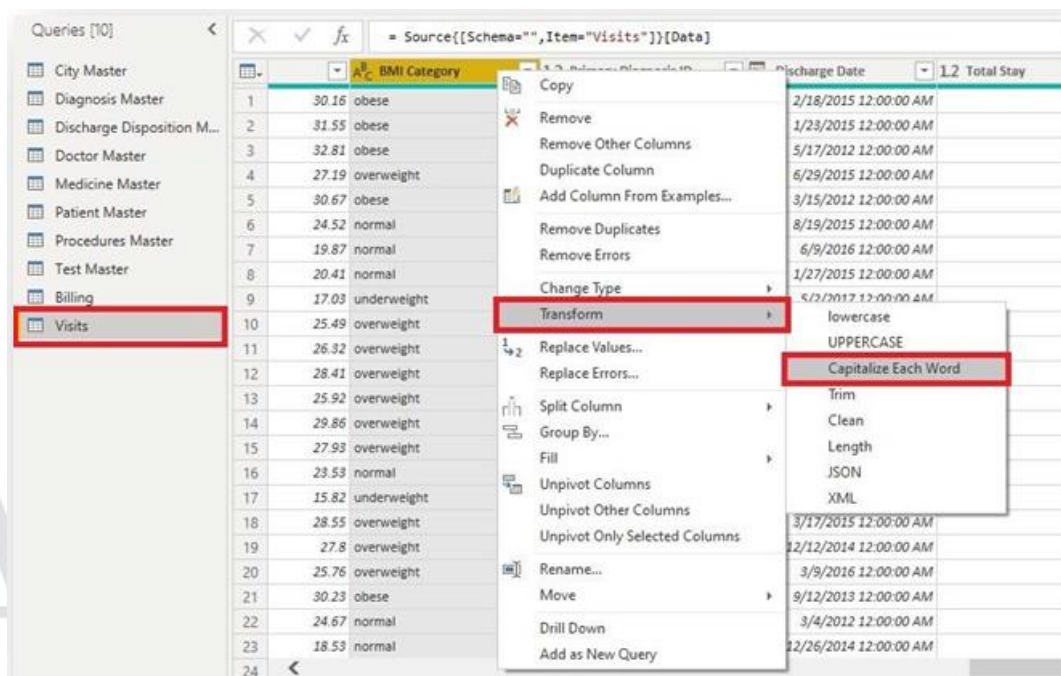
Change datatype

- 1) Launch PowerBI desktop.
- 2) Open the previously saved "Healthcare Dashboard" report.
- 3) From the "Home" menu, click on the "Edit Queries" icon.
- 4) In the "Query Editor" window, right click on the "Visit Date" column from "Visits" table and from the "Change Type" submenu, choose "Date". Repeat the same step for "Discharge Date".
- 5) Navigate to the "Patient Master" table, right click on the "DOB" column header and from the "Change Type" submenu, choose "Date". Ignore this step if the datatype is already set to "Date".



Capitalize column

- 1) Navigate to the “Visits” table, right click on the “BMI Category” column, navigate to “Transform” and select “Capitalize each word” option.
- 2) Navigate to the “Doctor Master” table and repeat the previous step for “Doctor Names” column.



Remove columns

Note: There are certain columns that contains incorrect data or a piece of information we don't require to satisfy the reporting requirements. So, let's remove such columns from the dataset.

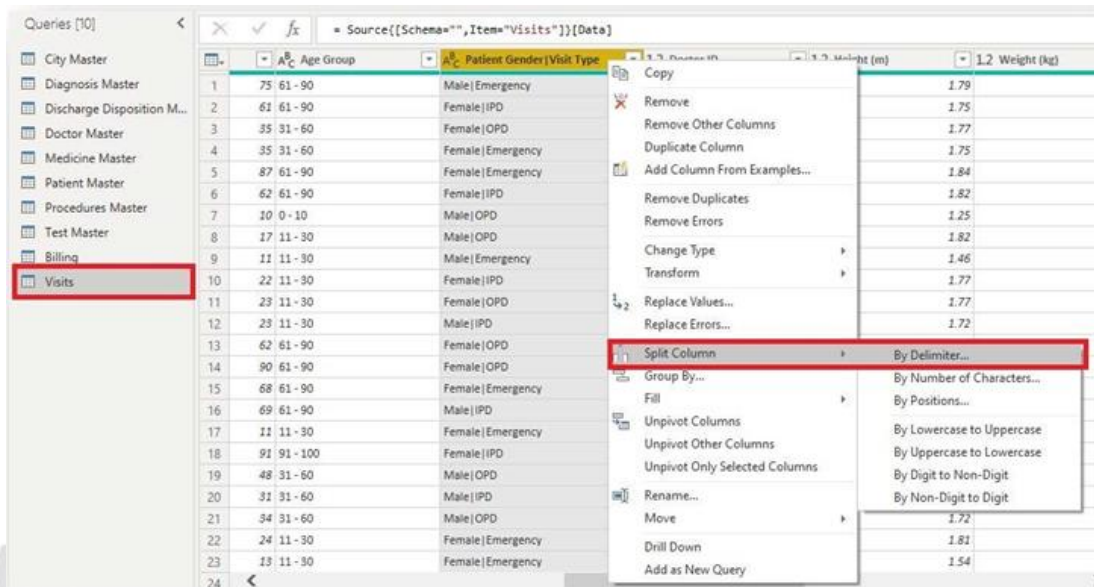
- 1) Navigate to the “Visits” table, right click on the “Visit Day” column and click on “Remove”. Repeat the same step for “Visit Month” column. These columns contain incorrect information.
- 2) Similarly navigate to the “Billing” table and remove the “Visit Date” and “Visit Year” columns as they're not required for satisfying any of the reporting requirements.

Rename columns

- 1) Navigate to the “Doctor Master” table, right click on the “Doctor Names” column and click on “Rename”. Rename the column as “Primary Physician” and hit Enter.

Split columns

- 1) In the “Visits” table right click on the “Patient Gender|Visit Type” column and from the “Split Column” submenu, select “By Delimiter”.



- 2) The system would automatically detect the detect the delimiter. However, in case it doesn't, select “Custom” from the list of delimiters.
- 3) Type | in the delimiter textbox.
- 4) Choose “Each occurrence of the delimiter” option.
- 5) Click on “Ok”.

Split Column by Delimiter

Specify the delimiter used to split the text column.

Select or enter delimiter
 --Custom--
 |

Split at
☐ Left-most delimiter
☐ Right-most delimiter
☒ Each occurrence of the delimiter

Advanced options
 Split into
☒ Columns
☐ Rows
 Number of columns to split into
 2
 Quote Character
 "
☐ Split using special characters
 Insert special character

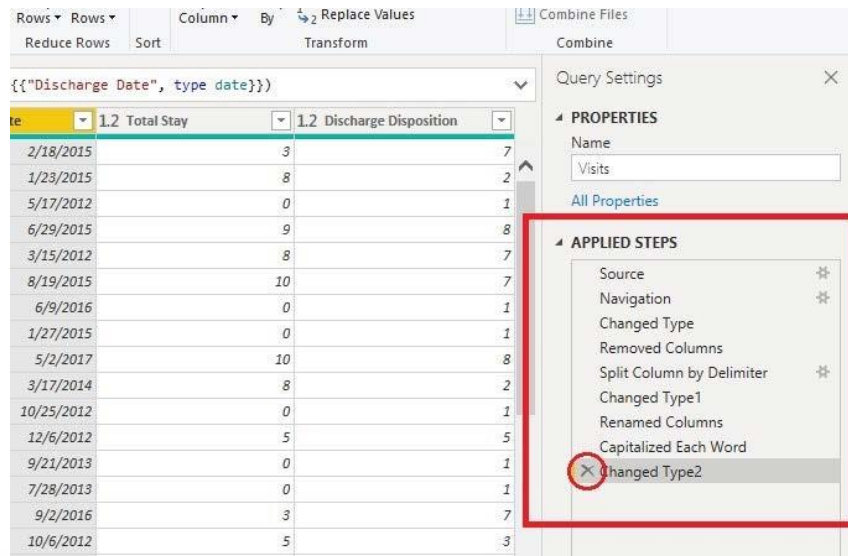
OK Cancel

- 6) After the column is split, right click on the "Patient Gender|Visit Type.1" column header and select "Rename". Rename the column as "Gender".
- 7) Right click on the "Patient Gender|Visit Type.2" column header and select "Rename". Rename the column as "Visit Type".
- 8) From the "Home" menu, click on "Close & Apply" button.
- 9) The changes would be applied on the model and the same can be viewed in the "Data View".

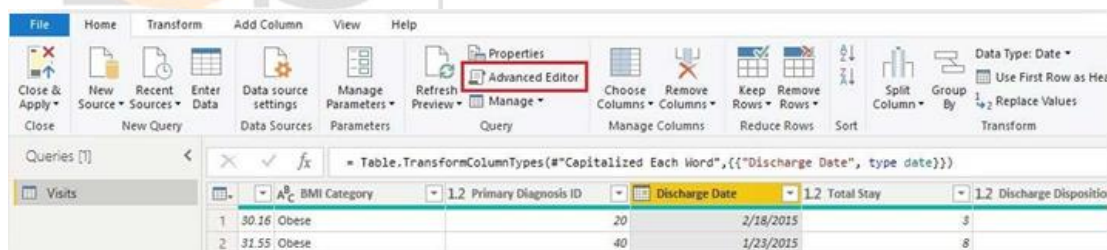
Table.TransformColumnTypes("#Split Column by Delimiter",{{"Patient Gender|Visit Type.1", type text}, {"Patient Gender|Visit Type.2", type text}}

id	A ^B _C Patient Gender Visit Type.1	A ^B _C Patient Gender Visit Type.2	1.2 Doctor ID	1.2 Height
	Male	Emergency		151
	Female	IPD		201
	Female	OPD		75
	Female	Emergency		59
	Female	Emergency		1
	Female	IPD		34
	Male	OPD		133
	Male	OPD		136
	Male	Emergency		84
	Female	IPD		186

Note: All the applied transformations can be viewed from the “Applied Steps” window. From this window you can selectively undo any of the applied transformations by clicking on the cross sign.



Similarly, when you apply a transformation using the “Edit Queries” screen, the system generates an MQuery code that can be copied and reused on another table. This code can be found in the “Advanced Editor” window, under the “Home” menu.



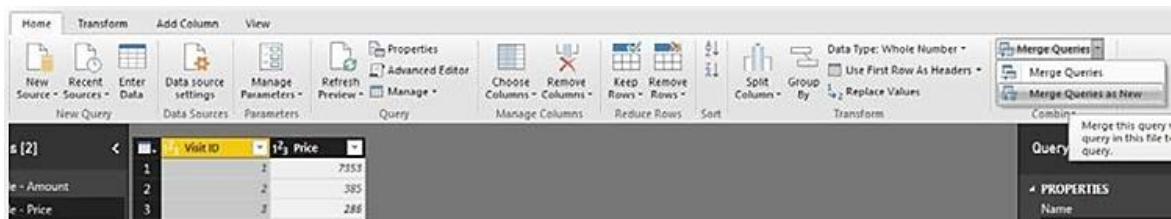


Assign data category

- 1) Navigate to the "Data View".
- 2) Expand the "City Master" table from the "Fields" panel on the right.
- 3) From the "City Master" table select the "City Name" field.
- 4) Navigate to the "Modeling" tab at the top and change the "Data Category" to "City".
- 5) The "City Name" should have a small globe like icon next to its name appearing under the "Fields" panel on the right. This shows that this column name is now treated as a geographic dimension (something that can be used with a Map visualization).
- 6) Save all the changes to file.

Merge queries

- 1) Launch PowerBI desktop.
- 2) Open the previously saved report where you've been doing some addhoc practice.
- 3) From the "Home" menu, click on "Get Data" button.
- 4) On the "Welcome screen", select "File" from the left-side list and "Excel" from the right-side list.
- 5) Click on "Connect".
- 6) Locate the "Merge Queries" excel file and click on "Open".
- 7) From the "Navigator" window, choose "Table – Amount" and "Table – Price" tables and click on "Edit".
- 8) Select the "Table – Price" table.
- 9) From the "Home" menu, expand the "Merge Queries" option and select "Merge Queries as New".



- 10) In the “Merge” screen, ensure “Table – Price” is selected as the first table and “Table – Amount” is selected as the second table.
- 11) Highlight the “Visit ID” column in both the selected tables. 12) Select the “Join Kind” as “Left Outer” and click on “Ok”.

Merge

Select tables and matching columns to create a merged table.

Table - Price

Visit ID	Price
1	7353
2	385
3	286
4	3322
5	444

Table - Amount

Visit ID	Amount
1	8280
2	778
3	772
4	3810
5	609

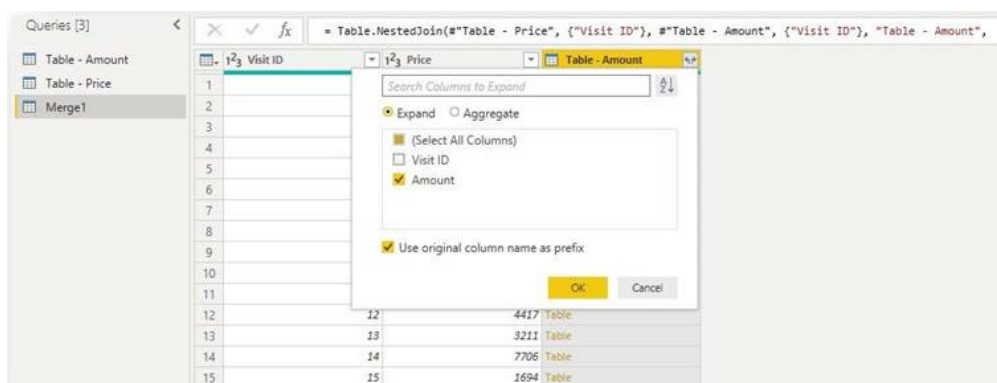
Join Kind

Left Outer (all from first, matching from second)

✓ The selection has matched 15 out of the first 15 rows.

OK Cancel

- 13) On the “Query Editor” window, select the new “Merge1” table that got created.
- 14) On the “NewColumn” column header, you’d find an icon that looks like two arrows pointing in opposite direction. Click on that icon and make sure that,
 - a. “Expand” radio button is selected.
 - b. “Visit ID” checkbox is unchecked.
 - c. “Amount” checkbox is checked.
 - d. “Use original column name as prefix” checkbox is checked.

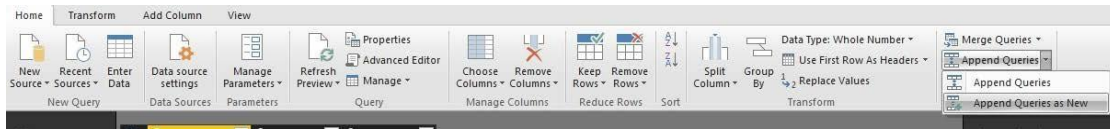


- 15) Click on “Ok”.
- 16) From the “Home” menu, click on “Close & Apply” button.
- 17) The new “Merge1” table should contain all the columns of “Table – Price” and “Table – Amount” tables.

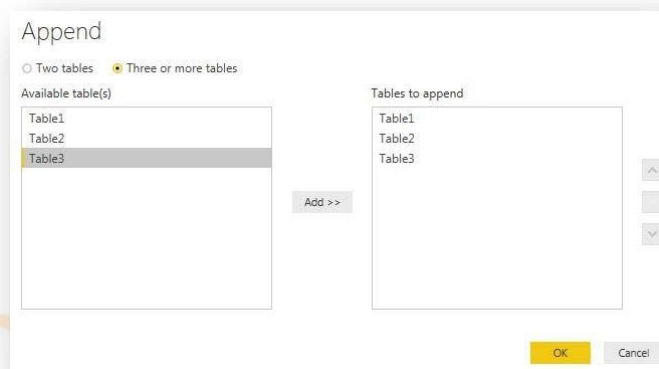
Visit ID	Price	Amount
1	7353	8280
2	385	778
3	286	772
4	3322	3810
5	444	609
6	671	1155
7	2136	2753
8	481	885
9	549	783
10	7837	8480
11	5620	6070
12	4417	2873
13	3211	7559
14	7706	5163
15	1694	7168

Append queries

- 1) Launch PowerBI desktop.
- 2) Open the previously saved report where you’ve been doing some addhoc practice.
- 3) From the “Home” menu, click on “Get Data” button.
- 4) On the “Welcome screen”, select “File” from the left-side list and “Excel” from the right-side list.
- 5) Click on “Connect”.
- 6) Locate the “Append Queries” excel file and click on “Open”.
- 7) From the “Navigator” window, choose “Table1”, “Table2 and “Table3” tables and click on “Edit”.
- 8) From the “Home” menu, expand the “Append Queries” option and select “Append Queries as New”.



- 9) On the “Append” screen choose, “Three or more tables” option.
 10) From the “Available Table(s)” list select, “Table1”, “Table2” and “Table3”. 11) Click on “Ok”.

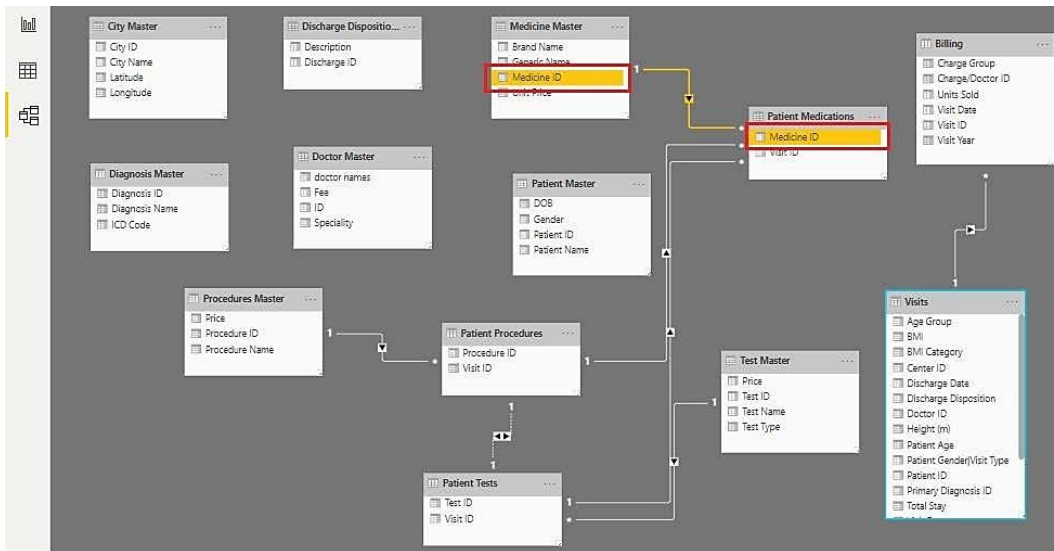


- 11) On the “Query Editor” a new table should be created called “Append1”. This table should contain all the records from all the tables.
 12) Click on “Close & Apply” to apply the changes.

Manage relationships

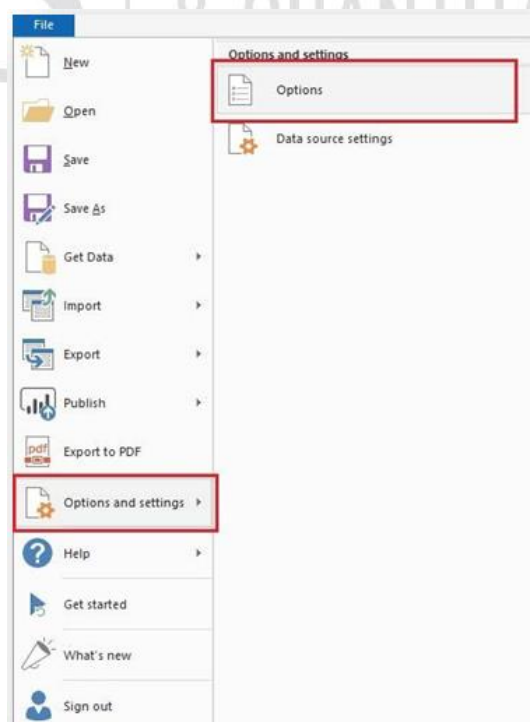
Auto-detect relations

PowerBI auto detects relationships between tables when you import them from a particular data source. The system essentially looks for columns with same name and datatype and automatically creates the relationships. While creating a relationship it also sets an appropriate cardinality and cross filter direction. Open the previously saved “Healthcare Dashboard” report and navigate to the “Model” view to see few relations being created automatically by the system.

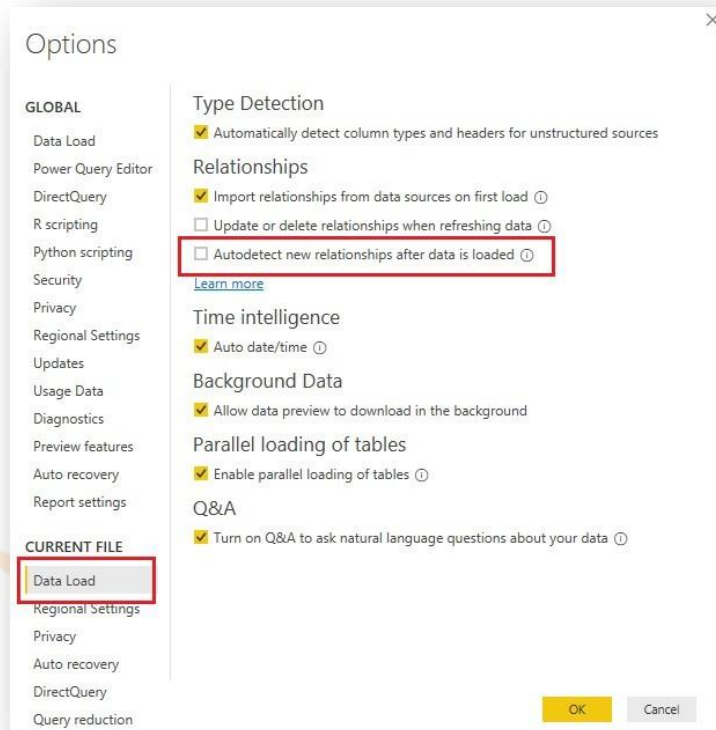


You can also turn off the auto detect option if you want, by following the below mentioned steps.

- 1) In the same “Healthcare Dashboard” report, navigate to the “File” menu and from “Options and Settings”, choose “Options”.

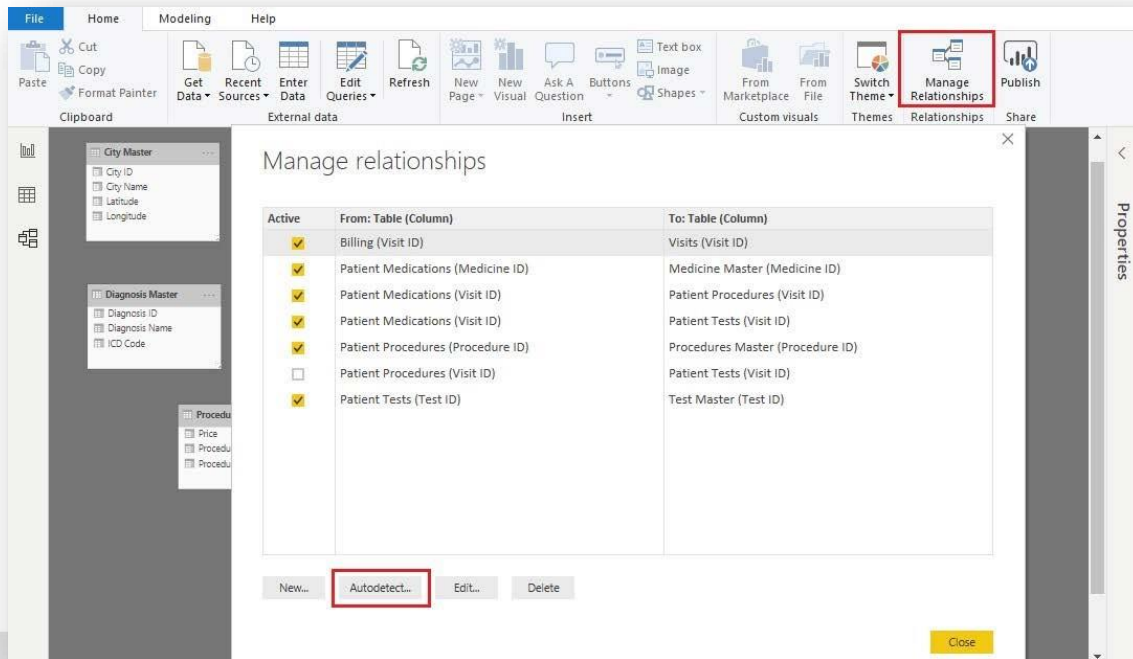


- 2) Navigate to the “Data Load” section that appears under “Current File”.
- 3) Uncheck the checkbox that reads, “Autodetect new relationships after data load”. 4) Click on Ok to save the changes.



You can also enforce the autodetect relationships manually, by following the below mentioned steps.

- 1) Open the “Healthcare Dashboard” report and navigate to the “Model” view.
- 2) Under the “Home” menu, click on “Manage Relationships”.
- 3) On the “Manage Relationships” screen, click on “Autodetect” and then click on “close”.
- 4) This should force the system to automatically detect and create more possible relationships in the model.



Creating Custom Relationships

In the “Healthcare Dashboard” report, to manually create your own relationships, follow the below mentioned steps. To have better control over your data model, you might want to delete the autodetected relationships first. To delete the existing relationships,

- 1) Navigate to the “Model” view and click on “Manage Relationships”.
- 2) Select all the relations by holding the shift key (so that they all get highlighted).
- 3) Click on the “Delete” button.

Manage relationships ×

Active	From: Table (Column)	To: Table (Column)
☒	Billing (Visit ID)	Visits (Visit ID)
☒	Patient Medications (Medicine ID)	Medicine Master (Medicine ID)
☒	Patient Medications (Visit ID)	Patient Procedures (Visit ID)
☒	Patient Medications (Visit ID)	Patient Tests (Visit ID)
☒	Patient Procedures (Procedure ID)	Procedures Master (Procedure ID)
☐	Patient Procedures (Visit ID)	Patient Tests (Visit ID)
☒	Patient Tests (Test ID)	Test Master (Test ID)

To create a new relationship,

- 1) Navigate to the "Model" view and click on the "Manage Relationships" button.
- 2) Click on the "New" button.
- 3) To create a relationship between the "Visits" and "Billing" table, select the "Visits" table from the top section.
- 4) Select the "Billing" table from the bottom section.
- 5) Highlight the "Visit ID" column in both the sections by clicking on the column header.
- 6) Make sure that the system sets the "Cardinality" as "One to Many" and the "Cross filter direction" as "Single".
- 7) "Make this relationship active" checkbox should be checked.
- 8) Click on Ok to save the relationship.

Create relationship

Select tables and columns that are related.

Visits

Visit ID	Visit Date	Visit Day	Visit Month	Center ID	Patient ID	Patient Age	Age Group
164	8/3/2017 12:00:00 AM	Fri	Jan	25	3417	78	61 - 90
322	1/20/2012 12:00:00 AM	Sun	Jan	9	13538	72	61 - 90
714	5/12/2012 12:00:00 AM	Sun	Jan	24	2855	81	61 - 90

Billing

Visit ID	Visit Date	Visit Year	Charge Group	Charge/Doctor ID	Units Sold
50	0/21/2013 12:00:00 AM	2013	Medicine	97	9
100	10/9/2013 12:00:00 AM	2013	Medicine	365	9
104	2/26/2013 12:00:00 AM	2013	Medicine	354	9

Cardinality: One to many (1:*)

Cross filter direction: Single

☒ Make this relationship active

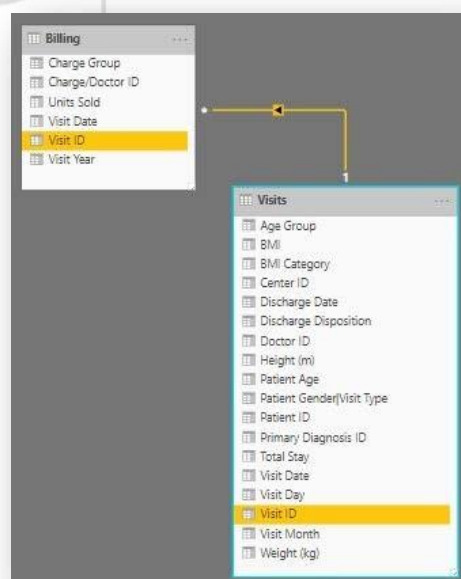
☐ Assume referential integrity

☐ Apply security filter in both directions

OK Cancel

10) Click on "Close" on the "Manage Relationships" screen.

11) When you get back to the "Relationships View", you see the newly created relationship being displayed using connector lines.



In the same way create the below mentioned relationships to configure the data model as per the reporting requirements.

Table From	Column Name	Table To	Column Name	Cardinality	Filter Direction	Active
Visits	Visit ID	Billing	Visit ID	One to many	Single	Yes
Visits	Center ID	City Master	City ID	Many to one	Single	Yes
Visits	Patient ID	Patient Master	Patient ID	Many to one	Single	Yes

Visits	Doctor ID	Doctor Master	ID	Many to one	Single	Yes
Visits	Primary Diagnosis ID	Diagnosis Master	Diagnosis ID	Many to one	Single	Yes
Visits	Discharge Disposition	Discharge Disposition Master	Discharge ID	Many to one	Single	Yes
Visits	Visit ID	Patient Medication	Visit ID	One to many	Single	Yes
Visits	Visit ID	Patient Procedures	Visit ID	One to one	Single	Yes
Visits	Visit ID	Patient Tests	Visit ID	One to one	Single	Yes
Billing	Charge/Doctor ID	Doctor Master	ID	Many to one	Single	Yes
Billing	Charge/Doctor ID	Medicine Master	Medicine ID	Many to one	Single	Yes
Billing	Charge/Doctor ID	Procedure Master	Procedure ID	Many to one	Single	Yes
Billing	Charge/Doctor ID	Test Master	Test ID	Many to one	Single	Yes

Once you create all the above-mentioned relationships, the relationship view should look like the screenshot shown below. Please note while creating the relationships, the system might make some of the relations inactive. This happens due to a circular reference.

While creating a relationship between two tables, if another indirect relationship is detected by the system (between the same set of tables), the current relation is then set as inactive. This is because you can have only one active relationship between two tables. Leave these settings to default as the

system controls the active / inactive checkbox.

