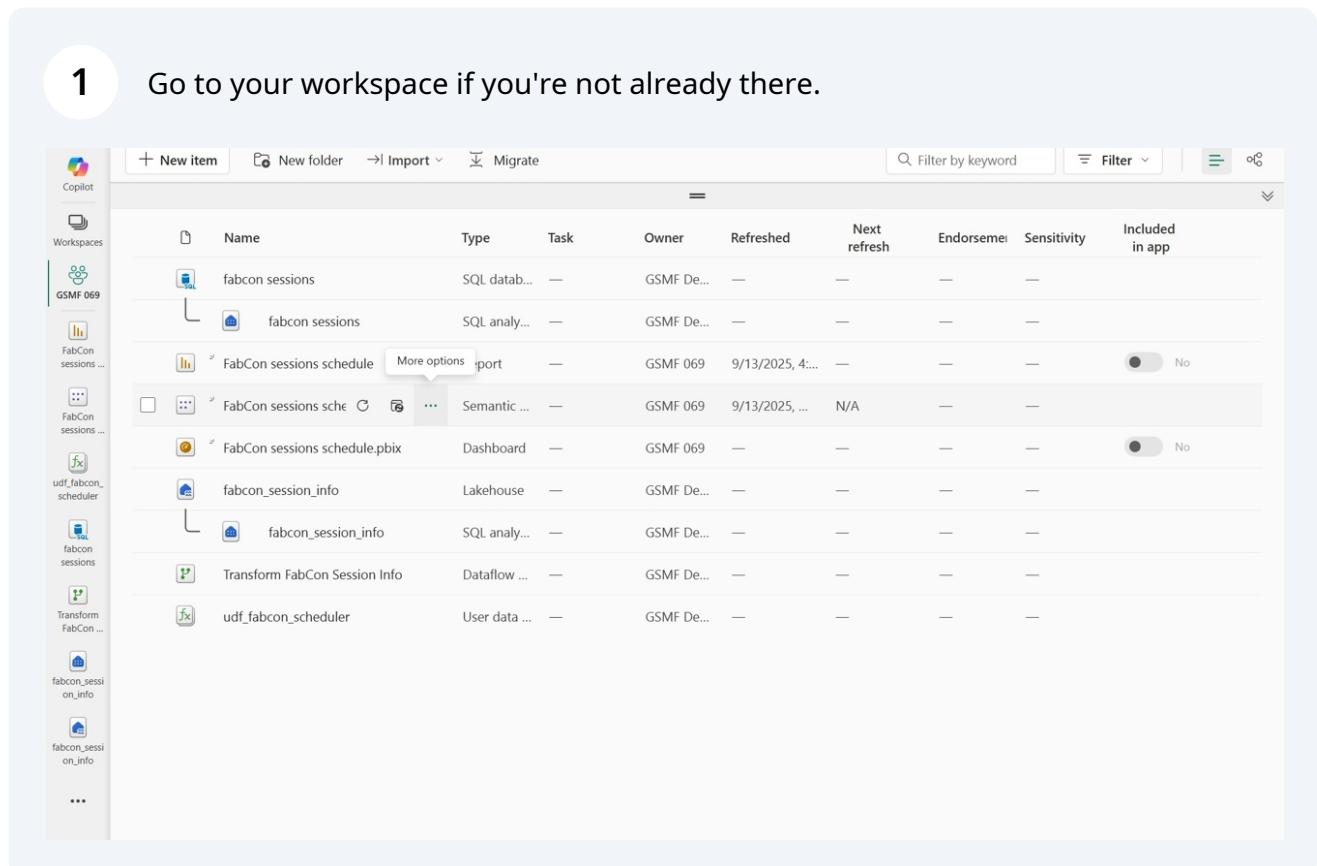


4b. Finish the semantic model in the service

Training. 

1

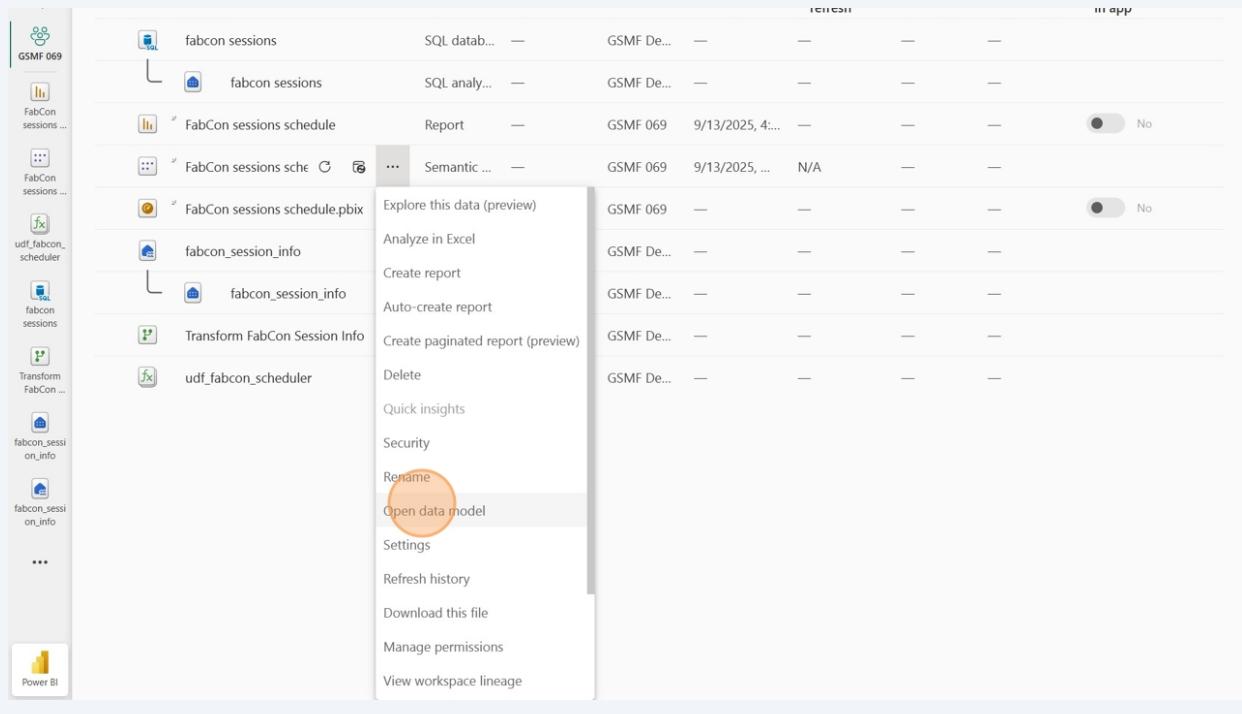
Go to your workspace if you're not already there.



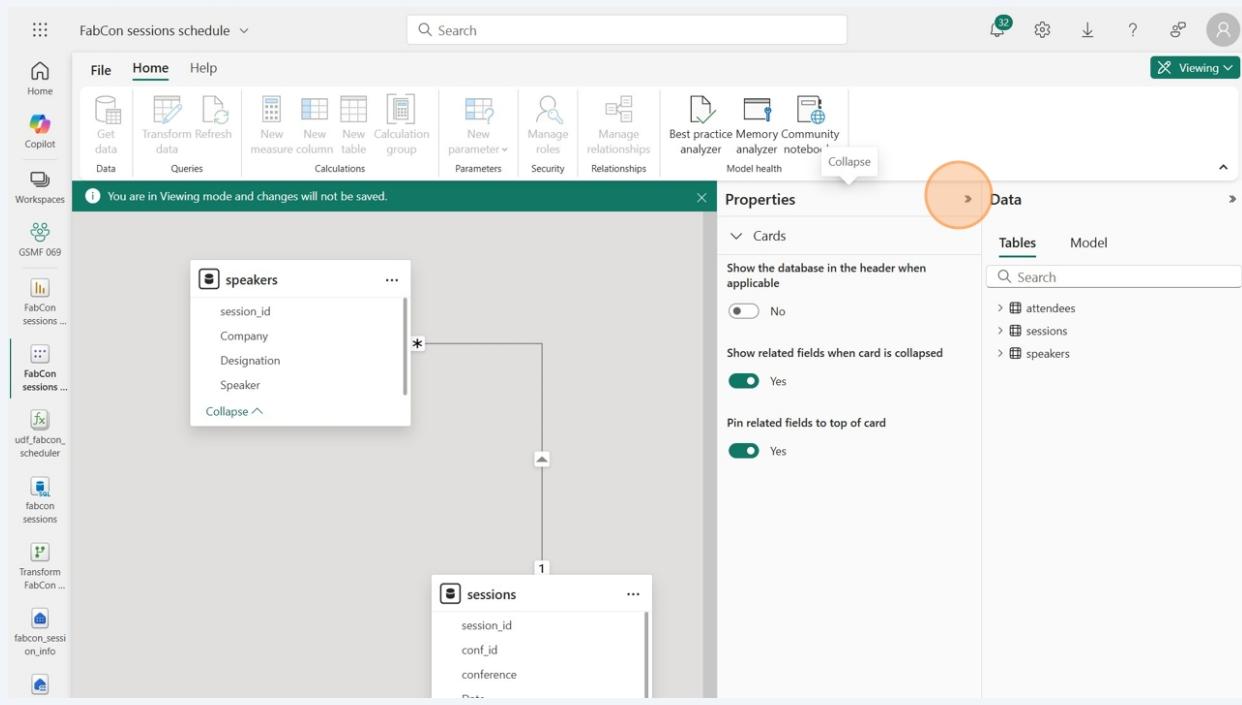
The screenshot shows a workspace interface with a sidebar on the left containing various project and file icons. The main area is a table view with the following columns: Name, Type, Task, Owner, Refreshed, Next refresh, Endorsement, Sensitivity, and Included in app. There are 10 rows of data:

Name	Type	Task	Owner	Refreshed	Next refresh	Endorsement	Sensitivity	Included in app
fabcon sessions	SQL database	—	GSMF Dev...	—	—	—	—	No
fabcon sessions	SQL analysis	—	GSMF Dev...	—	—	—	—	No
FabCon sessions schedule	More options Import	—	GSMF 069	9/13/2025, 4:00 AM	—	—	—	No
FabCon sessions schedule	Semantic...	—	GSMF 069	9/13/2025, 4:00 AM	N/A	—	—	No
FabCon sessions schedule.pbix	Dashboard	—	GSMF 069	—	—	—	—	No
fabcon_session_info	Lakehouse	—	GSMF Dev...	—	—	—	—	No
fabcon_session_info	SQL analysis	—	GSMF Dev...	—	—	—	—	No
Transform FabCon Session Info	Dataflow ...	—	GSMF Dev...	—	—	—	—	No
udf_fabcon_scheduler	User data ...	—	GSMF Dev...	—	—	—	—	No

- 2 Click "Open data model" for the semantic model.



- 3 This takes you to the semantic model editor. If your screen is small, you can minimize the Properties pane to see more of the model.



4

When you first open a semantic model, you're in "Viewing" mode. We need to be in "Editing" mode to make changes. Click the green "Viewing" dropdown in the top-right, and choose "Editing".

The screenshot shows the Power BI Semantic Model Editor interface. At the top, there's a navigation bar with 'File', 'Home' (selected), and 'Help'. Below the navigation bar is a toolbar with various icons for data management: 'Get data', 'Transform Refresh data', 'Queries', 'New measure column', 'New table', 'Calculation group', 'New parameter', 'Manage roles', 'Manage relationships', 'Parameters', 'Security', 'Relationships', 'Best practice analyzer', 'Memory analyzer', 'Community notebooks', and 'Model health'. A status bar at the top indicates: 'FabCon sessions schedule' and 'You are in Viewing mode and changes will not be saved.' On the left, a sidebar lists workspaces: 'FabCon sessions ...', 'FabCon sessions ...', 'udf_fabcon_scheduler', 'fabcon_sessions', 'Transform FabCon ...', 'fabcon_sessi on_info', and 'fabcon_sessi on_info'. The main area displays two tables: 'speakers' and 'sessions'. A relationship is shown between them, with 'session_id' in 'speakers' pointing to 'session_id' in 'sessions'. The 'Properties' pane on the right shows the relationship settings: 'Table speakers, Column session_id', 'Cardinality Many to one (*:1)', 'Table sessions, Column session_id', and 'Make this relationship active Yes'. The 'Cross-filter direction' is set to 'Single'. A button 'Apply changes' is present. The bottom of the screen shows tabs for 'All tables' (selected), 'Model view' (highlighted in blue), and 'DAX query view'. The status bar at the bottom right shows '100%'.

5

You will get a message that the model is being updated to the large semantic model storage format. Once it's finished, you can close the below message.

The screenshot shows the Power BI desktop application. On the left, there's a navigation pane with various workspace and data source icons. The main area displays two tables: 'speakers' and 'sessions'. A relationship is being configured between them. The 'Relationship' pane on the right shows the configuration for the 'speakers' to 'sessions' relationship. It specifies 'Table speakers' and 'Column session_id' as the primary key, and 'Table sessions' and 'Column session_id' as the foreign key. The cardinality is set to 'Many to one (*:1)'. A message at the top of the screen indicates that the model has been successfully converted to the large semantic model storage format.

Add relationships

6

We need to add two bi-directional relationships. One between speakers and sessions, and one between sessions and attendees. Both relationships will be on the session_id column in all the tables. You may already have one relationship that was automatically created. If so, you just need to make sure it's bi-directional.

7

Click the relationship between speakers and sessions if it already exists. Update the cross-filter direction to "Both."

The screenshot shows the Power BI Model view interface. On the left, there's a list of tables: 'FabCon sessions ...', 'FabCon sessions ...', 'udf_fabcon_scheduler', 'fabcon_sessions', 'Transform FabCon ...', 'fabcon_session_info', and 'fabcon_session_info'. Below this is a navigation bar with 'All tables' and a '+' button, followed by 'Model view' (which is selected) and 'DAX query view'. In the center, two tables are shown: 'speakers' and 'sessions'. A relationship line connects them, with a star symbol at the 'speakers' end and a one symbol at the 'sessions' end. To the right of the tables is the 'Relationship' pane. It shows the 'Table' as 'speakers' and 'Column' as 'session_id'. Under 'Cardinality', it says 'Many to one (*:1)'. Another entry shows 'Table' as 'sessions' and 'Column' as 'session_id'. A toggle switch labeled 'Make this relationship active' is set to 'Yes'. A dropdown menu for 'Cross-filter direction' is open, showing 'Single' (selected), 'Both', and 'None'. An orange circle highlights the 'Both' option. At the bottom of the pane are 'Apply changes' and 'Open relationship editor' buttons.

8

Click "Apply changes"

This screenshot is similar to the previous one but shows the changes made. The 'Cross-filter direction' dropdown now has 'Both' selected. The 'Apply changes' button is highlighted with an orange circle. The rest of the interface is identical to the first screenshot.

9

If you don't see the "attendees" table, scroll over until you find it. Click "session_id" in the attendees table and drag it to "session_id" in the sessions table to create a new relationship.

The screenshot shows the Power BI Model view interface. On the left, there's a tree view of tables: 'FabCon sessions ...', 'udf_fabcon_scheduler', 'fabcon_sessions', 'Transform FabCon ...', 'fabcon_session_info', and 'fabcon_session_info'. In the center, the 'sessions' table is selected, showing fields like session_id, conf_id, conference, Date, Day, DayNum, Description, End, enddt, and Collapse. An orange arrow points from the 'session_id' field in the 'sessions' table to the 'session_id' field in the 'attendees' table, which is shown in a preview window. On the right, the 'Model' tab is selected, showing settings for 'Cards': 'Show the database in the header when applicable' (No), 'Show related fields when card is collapsed' (Yes), and 'Pin related fields to top of card' (Yes). Below these are sections for 'Tables' (attendees, sessions, speakers) and 'Search'.

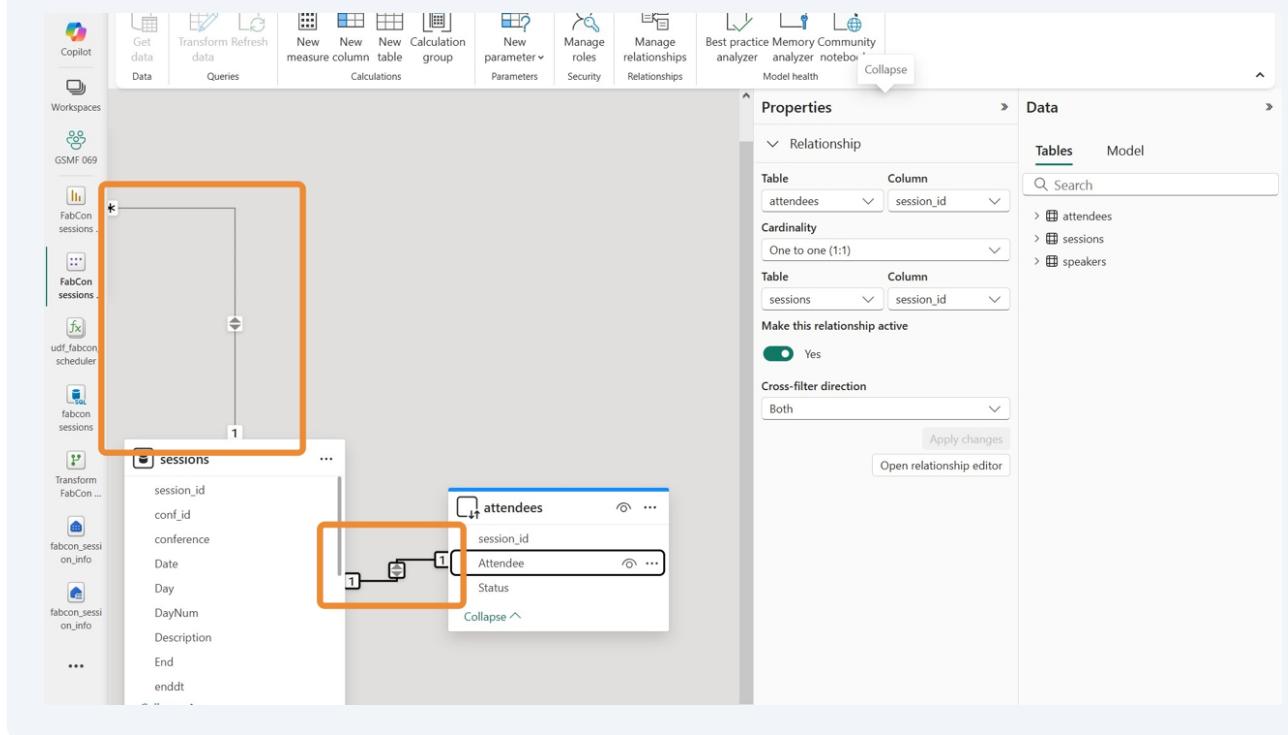
10

When you let go of the mouse button, the relationship manager will pop up. Make sure that the Cross-filter direction is set to "Both." Click "Save"

The screenshot shows the Power BI Model view with the relationship manager dialog open. The 'sessions' table is selected as the 'To table'. The 'Cardinality' is set to 'One to one (1:1)'. The 'Cross-filter direction' dropdown is highlighted with an orange box and set to 'Both'. There are two checkboxes at the bottom: 'Make this relationship active' (checked) and 'Assume referential integrity' (unchecked). At the bottom right of the dialog are 'Save' and 'Cancel' buttons, with 'Save' also highlighted with an orange circle.

11

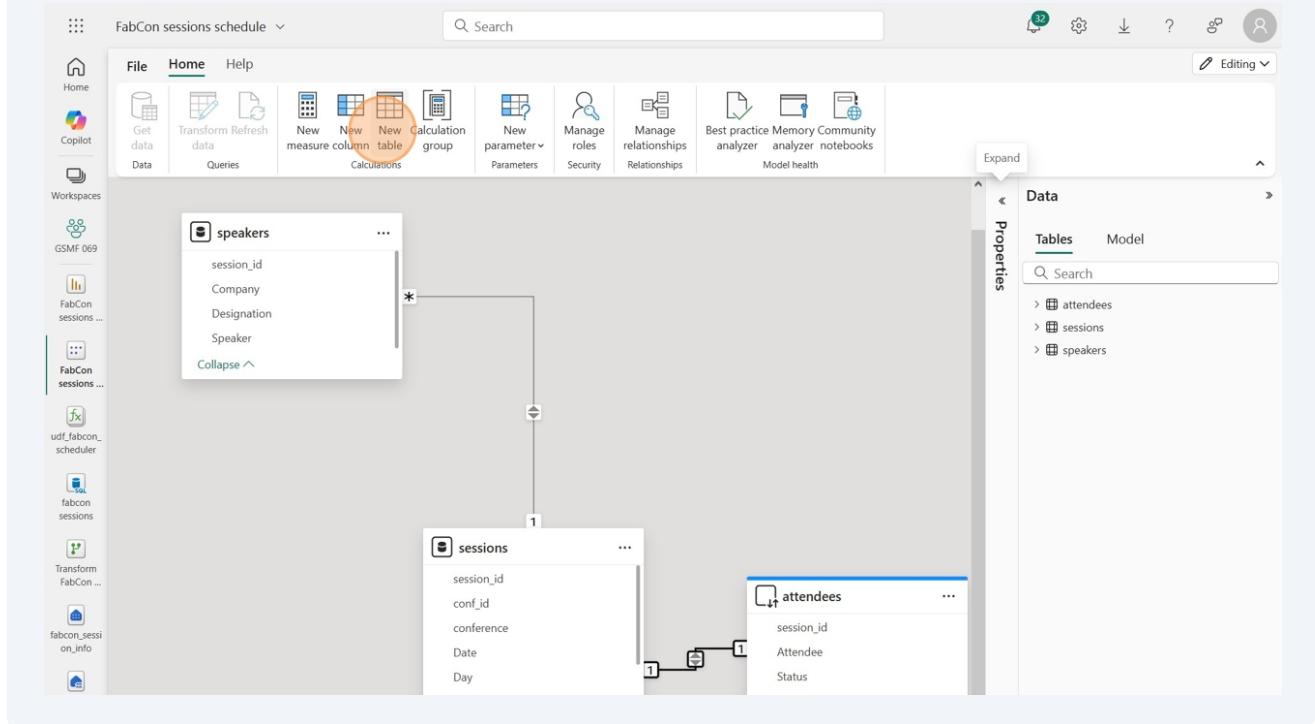
You should now see the two bi-directional relationships.



Create a table with DAX

12

We need one more table to populate a slicer for saving our session preferences. We want to be able to decide if we are "Definitely" or "Maybe" attending the sessions. So we'll create a table with just those two rows. No relationship will be required for this table. Click "New table" in the ribbon under the Home tab.



13

Copy and paste the following DAX into the formula bar to create the table and put the two rows into it.

```
attending status = DATATABLE (
    "status", STRING,
    {
        { "Definitely" },
        { "Maybe" }
    }
)
```

The screenshot shows the Power BI Data Model view. On the left, there's a list of workspaces and data models. In the center, a DAX code editor window is open with the following code:

```
attending status = DATATABLE (
    "status", STRING,
    {
        { "Definitely" },
        { "Maybe" }
    }
)
```

Below the code editor is a data model diagram. It shows two tables: 'sessions' and 'attendees'. The 'sessions' table has columns: session_id, conf_id, conference, Date, Day. The 'attendees' table has columns: session_id, Attendee, Status. A relationship is shown between sessions.session_id and attendees.session_id. The properties pane on the right shows the 'Tables' section with 'attendees', 'sessions', 'speakers', and 'Table' selected.

14

Click the green checkmark to the left of the formula bar to enter it.

The screenshot shows the Power BI Data Editor interface. In the center, there's a code editor window displaying the following DAX code:

```
1 attending status = DATATABLE (
2     "status", STRING,
3     {
4         { "Definitely" },
5         { "Maybe" }
6     }
7 )
8
```

The first line of the code, "attending status = DATATABLE (", has a green checkmark icon to its left, which is highlighted with a red oval. Below the code editor, there are two tables: "sessions" and "attendees". A relationship is visible between them. On the right side of the screen, there's a "Properties" pane and a "Data" pane. The "Tables" tab in the "Data" pane is selected, showing a list of tables including "attendees", "sessions", "speakers", and "Table".

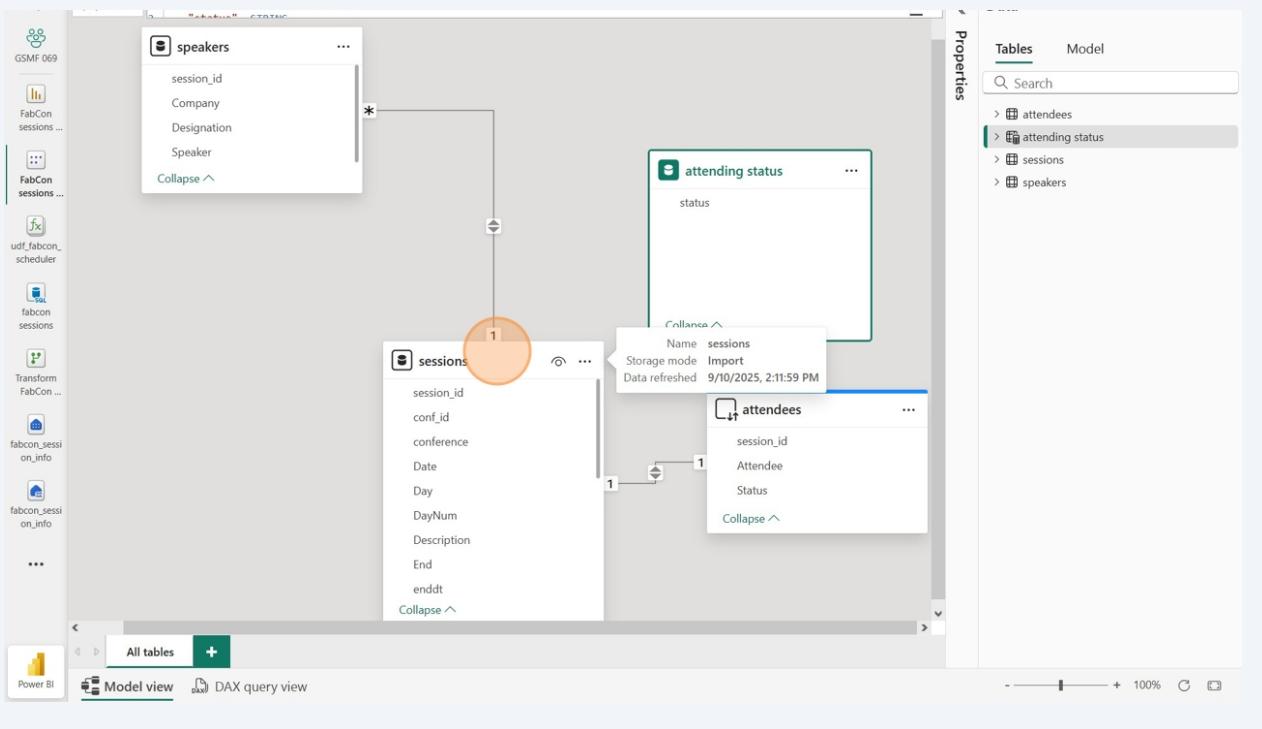
Create measures



Tip! We're building a pretty basic model here and we'll just drop our measures in the sessions table. Today's workshop isn't focused on semantic modeling best practices, but we encourage you to learn more about semantic modeling as well.

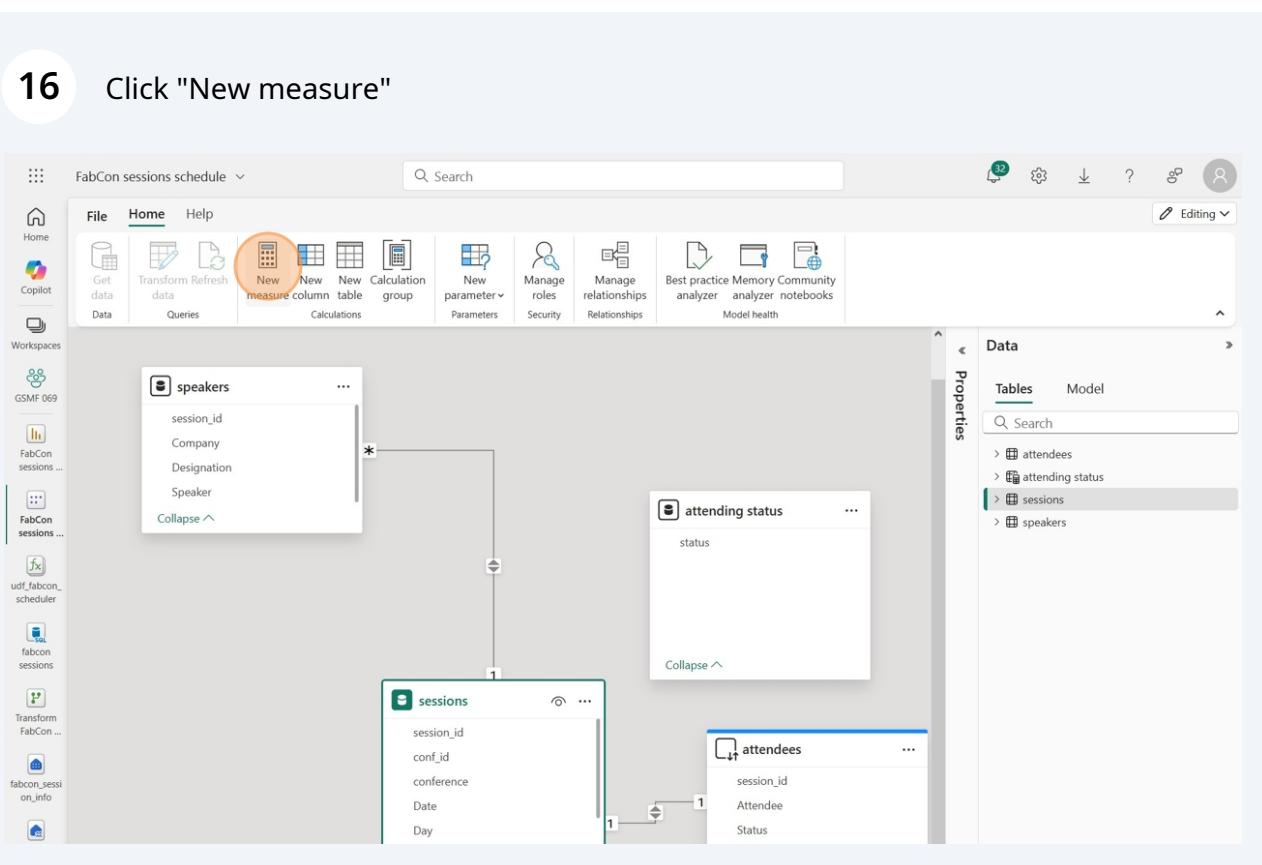
15

Click the "sessions" table to select it. We'll add a few measures to this table.



16

Click "New measure"



17

Replace the default text with the following measure:
User = USERPRINCIPALNAME()

This measure will return the UPN of the user running the report. We'll use it for the attendees updates.

The screenshot shows the Power BI Data Model view. In the center, there is a diagram of three tables: 'speakers', 'sessions', and 'attendees'. The 'speakers' table has columns 'session_id', 'Company', 'Designation', and 'Speaker'. It has a many-to-one relationship with the 'sessions' table, which has columns 'session_id', 'conf_id', 'conference', 'Date', and 'Day'. The 'sessions' table has a one-to-many relationship with the 'attendees' table, which has columns 'session_id', 'Attendee', and 'Status'. In the top navigation bar, the 'Queries' tab is selected. Below the navigation bar, a query named '1. User = USERPRINCIPALNAME()' is listed. On the right side, the 'Properties' pane is open, showing the 'Tables' section with 'Attendees' and 'Attending status' listed, and the 'Measure' section with 'session_id', 'Attendee', and 'Status' measures.

18

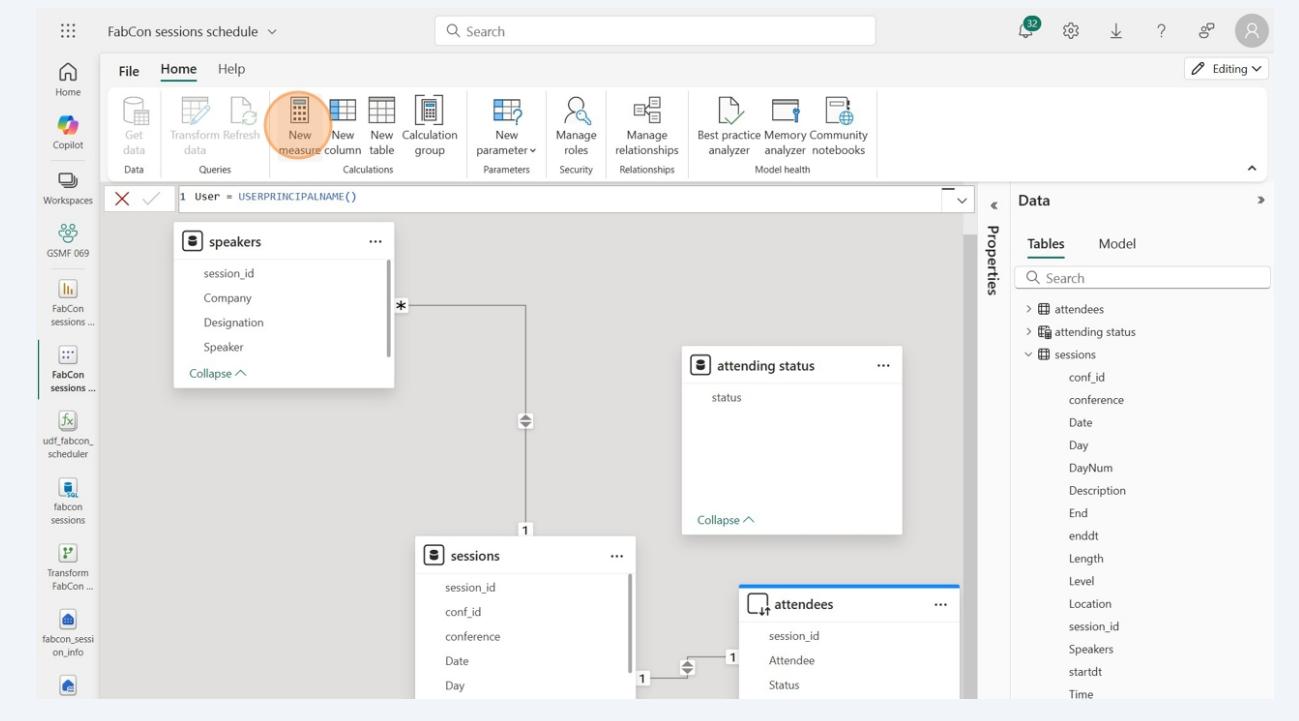
Click green checkmark to enter the measure.

The screenshot shows the Power BI Data Model view again. The setup is identical to the previous one, with the 'speakers', 'sessions', and 'attendees' tables in the center, a query '1. User = USERPRINCIPALNAME()' selected in the 'Queries' pane, and the 'Properties' pane showing the 'Tables' and 'Measure' sections. However, the 'Queries' tab in the ribbon is now highlighted, and the green checkmark icon next to the query name is circled in orange.

- 19** Do the same as above to create the following four measures:

```
[[Selected Session = SELECTEDVALUE(sessions[session_id])]]
[[Selected Status = SELECTEDVALUE(' attending status'[status])]]
[[Speaker count = DISTINCTCOUNTNOBLANK(speakers[Speaker])]]
[[Attendee Count = DISTINCTCOUNT(attendees[Attendee])]]
```

We'll use the selected session and selected status measures for the User Defined Function we created earlier.



20

You should now see all the measures you added in the sessions table. Go back to the workspace.

The screenshot shows the Power BI Data view interface. On the left, there's a sidebar with various workspace items like 'Copilot', 'FabCon sessions ...', and 'udf_fabcon_scheduler'. The main area displays a semantic model diagram with entities: 'speakers', 'attending status', 'sessions', and 'attendees'. A measure 'Attendee Count' is defined as `DISTINCTCOUNT(attendees[Attendee])`. The 'Properties' pane on the right shows the 'Tables' section with the 'sessions' table selected. Under 'Selected Session', 'Selected Status', and 'session id' are highlighted with orange boxes. The 'Model' section is also visible.

21

Click the semantic model.

The screenshot shows the Power BI workspace for 'GSMF 069'. The left sidebar lists workspaces including 'GSMF 069' (which is selected and highlighted with an orange circle). In the main content area, a table lists items: 'fabcon sessions', 'fabcon sessions schedule', 'FabCon sessions schedule.pbix', 'fabcon_session_info', 'Transform FabCon Session Info', and 'udf_fabcon_scheduler'. The 'fabcon sessions schedule' item is highlighted with an orange circle. The top navigation bar includes 'Create deployment pipeline', 'Create app', 'Manage access', and 'Workspace settings'.

22

This view of the semantic model lets you see what tables and columns are in the model, as well as related items.

The screenshot shows the Power BI Semantic Model view. On the left, there's a sidebar with various workspace items like 'Copilot', 'FabCon sessions ...', 'udf_fabcon_scheduler', 'fabcon_sessions', 'Transform FabCon ...', 'fabcon_session_info', and 'Power BI'. The main area has a title 'FabCon sessions schedule' with a subtitle 'GSMF 069'. It includes sections for 'Discover business insights' (with a 'Explore this data' button) and 'Share this data' (with a 'Share semantic model' button). Below these is a section titled 'See what already exists' showing items from the same data source. A 'Tables' pane on the right lists tables: 'attendees' (selected), 'attending status', 'sessions', and 'speakers'. A tooltip for 'attendees' says: 'To select more than one table, and view summarized data, create a paginated report.' A 'Create paginated report (preview)' button is also visible.

Details for FabCon sessions schedule
GSMF 069

+ Add description

Refreshed
9/13/25, 4:17:20 PM

Discover business insights

Explore this data to get insights fast or create an interactive report you can share. [Learn more](#)

[Explore this data](#)

Share this data

Give people access to the semantic model and set their permissions to work with it. [Learn more](#)

[Share semantic model](#)

See what already exists

These items use the same data source as FabCon sessions

Name	Type	Relation	Location	Refreshed	Endorsemen	Sensitivity
FabCon sessions s	Report	Downstream	GSMF 069	9/13/25, 4:1...	—	—
FabCon sessions s	Dashboard	Downstream	GSMF 069	—	—	—
fabcon_session_in	SQL analy...	Upstream	GSMF 069	—	—	—

Tables

Select a table and/or columns from this semantic model to view and export the underlying data. [Learn more](#)

To select more than one table, and view summarized data, create a paginated report.

Create paginated report (preview)

- attendees
- attending status
- sessions
- speakers