

Use Power BI's automatic calendar tables and time intelligence functions in DAX measures

Open the **Music Tours - Basic Time Intelligence.pbix** file in the exercise's folder. On **Page 1** you'll find a matrix containing the dates of shows and the tickets sold and revenue generated.

Artist, Tour name	Year	Tickets Sold	Show Revenue
AC/DC	1981		
Bad Bunny	Qtr 3		
Beyoncé	September		
Bon Jovi	25	90,782	\$1,429,817
Britney Spears	26	90,782	\$1,429,817

You can use the slicer to choose different artists and tours.

Add a measure to create a monthly running total of tickets sold using the **TOTALMTD** function.

*Remember to reference the **Date** column of the automatic calendar table that is related to the **Show date** field, rather than just the **Show date** field itself.*

Add the measure to the matrix.

Year	Tickets Sold	Show Revenue	Monthly Running Total Tickets Sold
1981			
Qtr 3			
September			
25	90,782	\$1,429,817	90,782
26	90,782	\$1,429,817	181,564
27	75,000	\$1,125,000	256,564
28			256,564
29			256,564

Annoyingly, the running total makes dates on which no tickets were sold appear in the matrix.

To hide the dates on which no tickets were sold, update the measure to include an **IF** function which checks if the sum of tickets sold is blank.

Year	Tickets Sold	Show Revenue	Monthly Running Total Tickets Sold
1981			
Qtr 3			
September			
25	90,782	\$1,429,817	90,782
26	90,782	\$1,429,817	181,564
27	75,000	\$1,125,000	256,564
Qtr 4			
October			
3	60,000	\$960,000	60,000
4	60,000	\$960,000	120,000

The matrix should now hide any dates with no ticket sales.

Add another measure to calculate a monthly running total of show revenue. Add this measure to the matrix and make sure that it doesn't cause dates with no ticket sales to appear.

Year	Tickets Sold	Show Revenue	Monthly Running Total Tickets Sold	Monthly Running Total Show Revenue
1981				
Qtr 3				
September				
25	90,782	\$1,429,817	90,782	\$1,429,817
26	90,782	\$1,429,817	181,564	\$2,859,633
27	75,000	\$1,125,000	256,564	\$3,984,633

On **Page 2** you'll find a matrix showing the total length of tracks released as singles, grouped by year and month.

Year	Sum Single Length (secs)
1980	8,757
January	523
February	202

Create a measure which compares the sum of single length with the same period in the previous year. You can use a combination of the **CALCULATE** and **SAMEPERIODLASTYEAR** functions to do this. Add the measure to the matrix.

Year	Sum Single Length (secs)	Sum Single Length vs. Same Dates Last Year
1983	10,422	4,899
January	634	401
February	257	-386

You could add conditional formatting to highlight positive and negative numbers.

Add a measure which compares the sum of single length with the same value from one month ago. You can use the **CALCULATE** and **DATEADD** functions to do this. Add the measure to the matrix.

Year	Sum Single Length (secs)	Sum Single Length vs. Same Dates Last Year	Sum Single Length vs. Previous Month
December	436	436	-685
1984	9,520	-902	-62
January	878	244	442
February	548	291	-330

Again, conditional formatting might make it easier to read the results.

When comparing with the previous month, it doesn't make sense to display a value for the year. Try modifying the measure so that it only shows a result when the **Month** column is in scope.

*You can use the **ISINSCOPE** function to check if the **Month** column of the **Single release date** field is in scope.*

Save and close the report.