

AZURE SQL

```
SELECT TOP 20 concat(LastName, ', ', FirstName) as Name, SUM(i.Total) as Sales
FROM chinook.[Customer] c
JOIN chinook.[Invoice] i on c.CustomerId = i.CustomerId
GROUP BY concat(LastName, ', ', FirstName)
ORDER BY SUM(i.Total) DESC
```

The screenshot shows the Azure Data Studio interface. The top toolbar includes buttons for Run, Cancel, Disconnect, Change Connection, and Estimated Plan. The query editor contains the following SQL code:

```
1 SELECT TOP 20 concat(LastName, ', ', FirstName) as Name, SUM(i.Total) as Sales
2 FROM chinook.[Customer] c
3 JOIN chinook.[Invoice] i on c.CustomerId = i.CustomerId
4 GROUP BY concat(LastName, ', ', FirstName)
5 ORDER BY SUM(i.Total) DESC
6
```

The Results pane displays the output of the query as a table with two columns: Name and Sales. The table contains 20 rows of data, sorted by total sales in descending order.

	Name	Sales
1	Holg, Helena	49.62
2	Cunningham, Richard	47.62
3	Rojas, Luis	46.62
4	Kovács, Ladislav	45.62
5	O'Reilly, Hugh	45.62
6	Barnett, Julia	43.62
7	Ralston, Frank	43.62
8	Zimmermann, Fynn	43.62
9	Gruber, Astrid	42.62
10	Stevens, Victor	42.62
11	Hämäläinen, Terhi	41.62
12	Mercier, Isabelle	40.62
13	Van der Berg, Johannes	40.62
14	Nichterlová, František	40.62
15	Hansen, Bjørn	39.62
16	Fernandes, João	39.62
17	Gonçalves, Luis	39.62
18	Girard, Wyatt	39.62
19	Leacock, Heather	39.62
20	Miller, Dan	39.62

The status bar at the bottom indicates the current position in the document: Ln 6, Col 27, Spaces: 4, UTF-8, CRLF, 20 rows, MSSQL, 00:00:00, and the connection details: akshita22-sql-database.windows.net : akshita22-db.

AZURE FOR MYSQL

SELECT GENRE.NAME, SUM(INVOICELINE.UNITPRICE*INVOICELINE.QUANTITY) AS SALES

FROM GENRE

JOIN TRACK ON TRACK.GENREID=GENRE.GENREID

JOIN INVOICELINE ON TRACK.TRACKID=TRACK.TRACKID

GROUP BY GENRE.NAME

The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
1 SELECT GENRE.NAME, SUM(INVOICELINE.UNITPRICE*INVOICELINE.QUANTITY) AS SALES
2 FROM GENRE
3 JOIN TRACK ON TRACK.GENREID=GENRE.GENREID
4 JOIN INVOICELINE ON TRACK.TRACKID=TRACK.TRACKID
5 GROUP BY GENRE.NAME
6
```

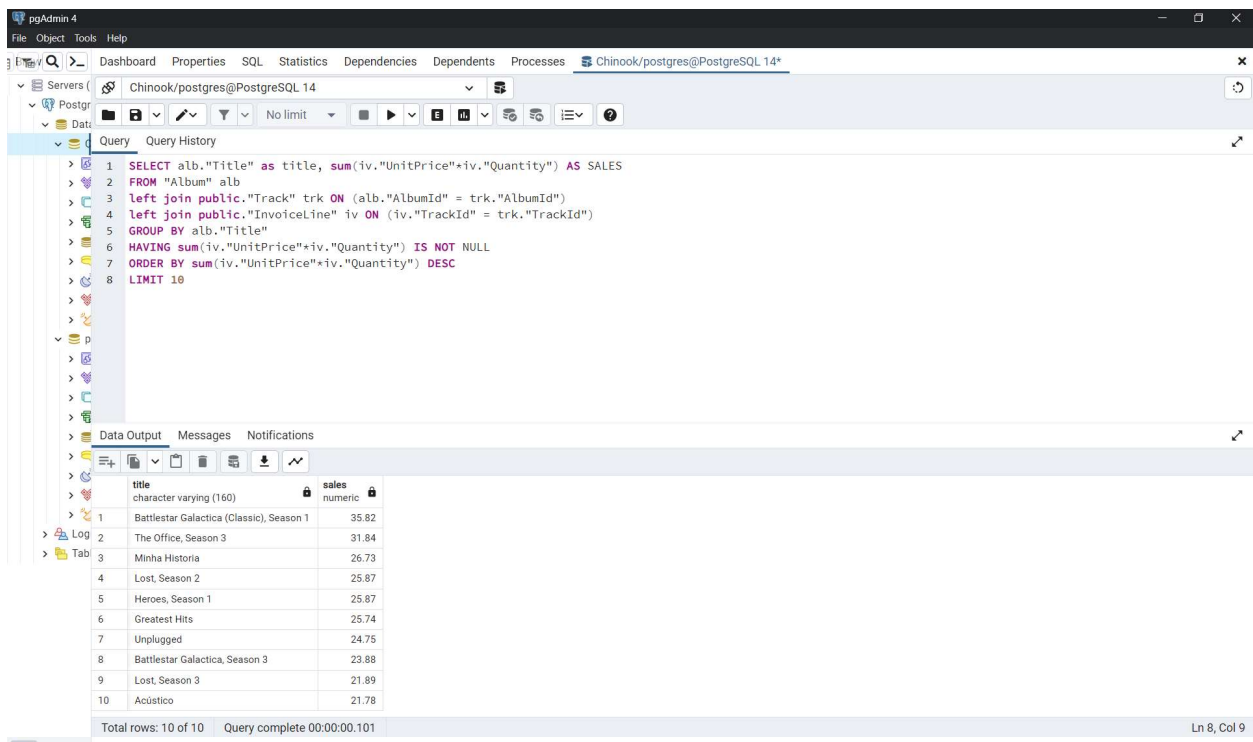
The Results window displays the following data:

NAME	SALES
Opera	2338.40
Classical	172316.40
Alternative	93144.00
Comedy	39586.20
Drama	149030.40
Sci Fi & Fantasy	60543.60
TV Shows	216559.80
Science Fiction	30271.80
Hip Hop/Rap	81501.00
World	65200.80
Electronica/D...	69858.00
R&B/Soul	142044.60
Heavy Metal	65200.80
Easy Listening	59886.40
Bossa Nova	34929.00
Soundtrack	100129.80
Pop	111772.80
Reggae	135058.80
Latin	1348259.40
Blues	188616.60
Rock And Roll	27943.20
Alternative & ...	773095.20
Metal	870596.40

The interface also shows a sidebar with navigation options like MANAGEMENT, INSTANCE, and PERFORMANCE. A message on the right states: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

AZURE FOR POSTGRE

```
SELECT alb."Title" as title, sum(iv."UnitPrice"*iv."Quantity") AS SALES
FROM "Album" alb
left join public."Track" trk ON (alb."AlbumId" = trk."AlbumId")
left join public."InvoiceLine" iv ON (iv."TrackId" = trk."TrackId")
GROUP BY alb."Title"
HAVING sum(iv."UnitPrice"*iv."Quantity") IS NOT NULL
ORDER BY sum(iv."UnitPrice"*iv."Quantity") DESC
LIMIT 10
```



The screenshot shows the pgAdmin 4 interface with a SQL query executed in the 'Query' tab. The query is a complex join and aggregation. The 'Data Output' tab shows the results of the query, which are 10 rows of album titles and their total sales. The status bar at the bottom indicates 'Total rows: 10 of 10' and 'Query complete 00:00:00.101'.

	title	sales
1	Battlestar Galactica (Classic), Season 1	35.82
2	The Office, Season 3	31.84
3	Minha Historia	26.73
4	Lost, Season 2	25.87
5	Heroes, Season 1	25.87
6	Greatest Hits	25.74
7	Unplugged	24.75
8	Battlestar Galactica, Season 3	23.88
9	Lost, Season 3	21.89
10	Acústico	21.78

Total rows: 10 of 10 Query complete 00:00:00.101 Ln 8, Col 9