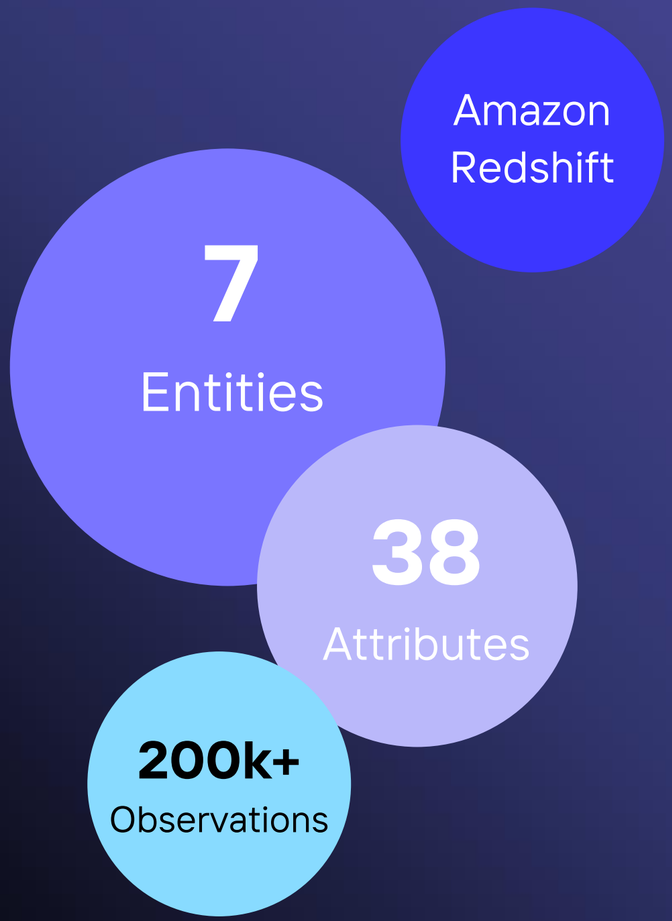
MetaData

Our data

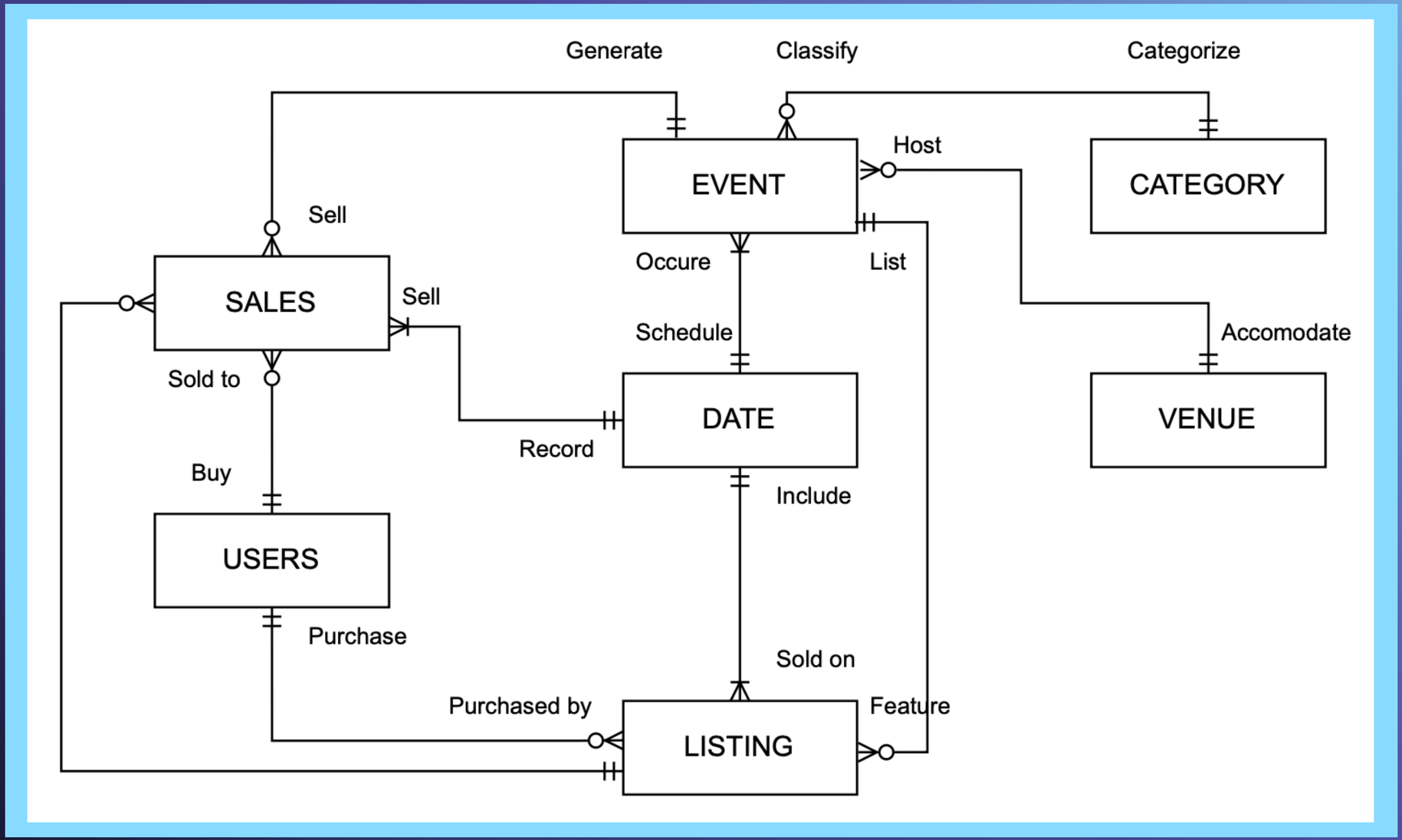


Entity	Description
CATEGORY	a classification of events based on the category, group, name, and description
DATE	an information related to specific dates that can be used for ticketing and event management
EVENT	an information related to a specific event
VENUE	an information related to a specific venue where events take place.
USERS	a person that have purchase or might purchase tickets
LISTING	an information related to the listing of events avaliable for sale
SALES	The transaction associated with the sale of tickets

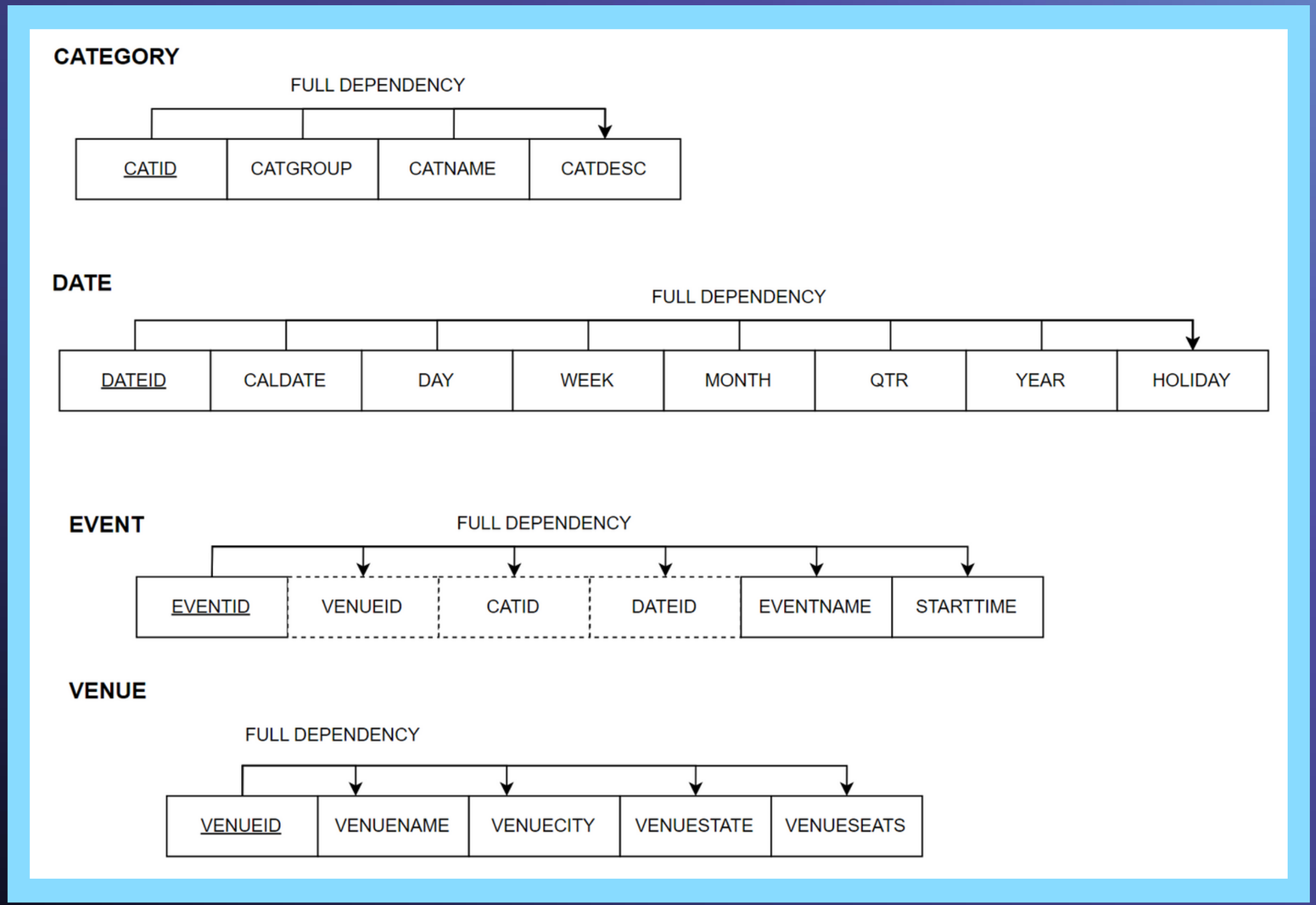
The Data Model

- On Each Date one or multiple events can be booked and one or more listings can be sold and have one or multiple sales.
- Each user can either generate zero or more sales. Similarly Each seller can sell no or more listings.
- Each listing can either be sold or bought by zero or more sellers and buyers respectively.
- Each event can generate zero or multiple sales and can be featured in zero or more listings to be sold.
- Each Category and venue can have zero or multiple events.

ERD diagram


Generate
Classify
Categorize
EVENT
CATEGORY
Host
Sell
SALES
Sell
Occure
List
Schedule
Accomodate
Sold to
DATE
VENUE
Record
Buy
Include
USERS
Purchase
Sold on
Purchased by
Feature
LISTING

As-Is Diagram

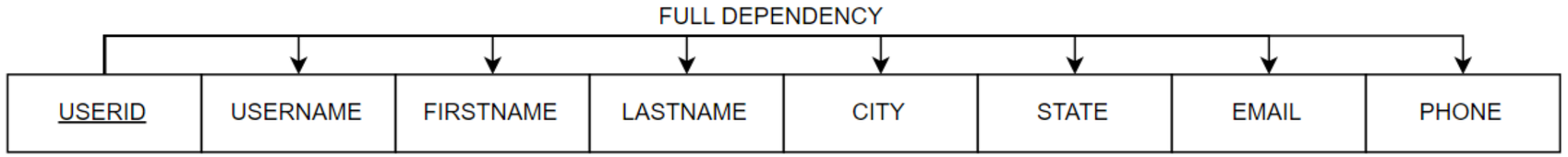


Our data contains **only full dependencies, and is 3rd normal form**

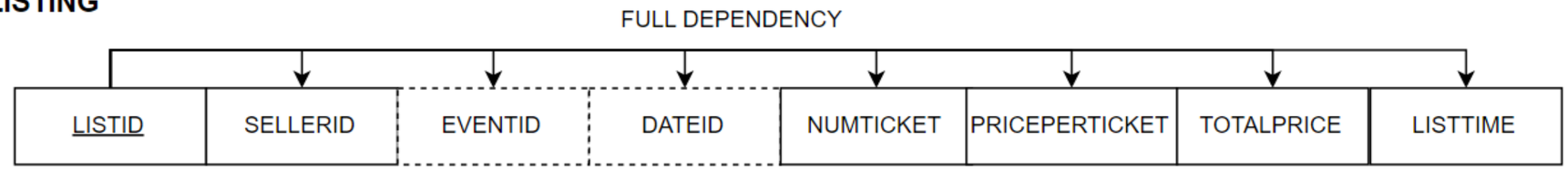
No convertion needed

As-Is Diagram

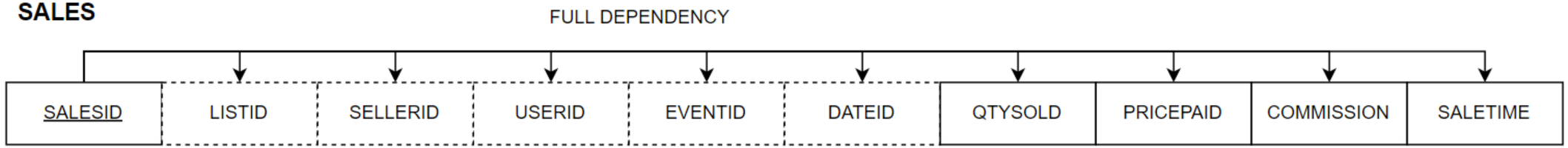
USERS



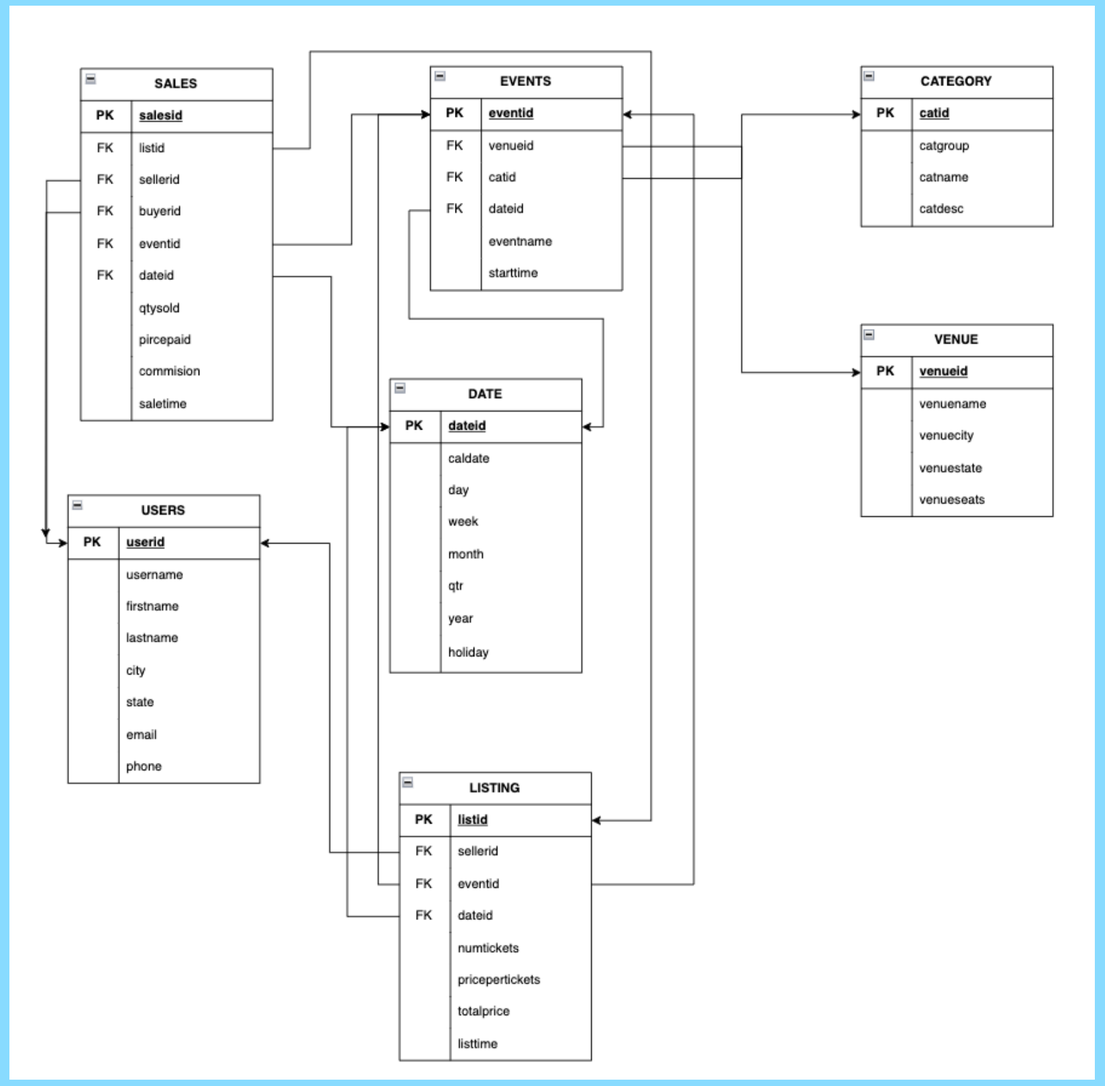
LISTING



SALES



Relational Data Model



- **Sales table has the more dependencies** because of its most foreign keys.
- **Events and listings come next (in terms of dependencises)**, with details about of when to list a event on a particular date.

Top 10 Events by Ticket Sales and Gross Revenue(Queries)

```sql
-- Top 10 Best selling events:
SELECT
    e.EVENTID,
    e.EVENTNAME,
    SUM(s.QTYSOLD) AS TOTAL_TICKETS_SOLD
FROM Events e
JOIN Sales S
ON e.EVENTID = s.EVENTID
GROUP BY e.EVENTID, e.EVENTNAME
ORDER BY TOTAL_TICKETS_SOLD DESC
LIMIT 10;
```

```sql
-- Top 10 best grossing events:
SELECT
    E.EVENTID,
    E.EVENTNAME,
    SUM(S.QTYSOLD * L.PRICEPERTICKET) AS TOTAL_REVENUE
FROM Events E
JOIN Sales S ON E.EVENTID = S.EVENTID
JOIN Listings L ON S.LISTID = L.LISTID
GROUP BY E.EVENTID, E.EVENTNAME
ORDER BY TOTAL_REVENUE DESC
LIMIT 10;
```

Top 10 Events by Ticket Sales and Gross Revenue

EVENTID	EVENTNAME	TOTAL_TICKETS_SOLD
1602	Phantom of the Opera	122
2079	Glengarry Glen Ross	106
851	The Little Mermaid	105
6693	Michael McDonald	103
6375	Greg Kihn Band	101
7745	Bloc Party	101
7895	Janet Jackson	100
6845	Gordon Lightfoot	98
8601	Built To Spill	97
8033	Smashing Pumpkins	97

EVENTID	EVENTNAME	TOTAL_REVENUE
289	Adriana Lecouvreur	51846
7895	Janet Jackson	51049
1602	Phantom of the Opera	50301
851	The Little Mermaid	49956
7315	Citizen Cope	49823
6471	Sevendust	48020
2118	Electra	47883
984	Mary Poppins	46780
7851	Live	46661
5638	Paul Weller	46280

- The venues listed on both tables are not the same, indicating a discrepancy between the top-selling venues based on ticket sales and those generating the highest revenue.
- Analyzing the variations between ticket sales and revenue figures can provide insights into pricing strategies, audience preferences, and the profitability of different productions or events.
- Event organizers and venue operators can use this information to assess the balance between ticket sales and revenue generation, identify opportunities for growth, and optimize their pricing and marketing strategies.

Top 10 Venues by Ticket Sales and Gross Revenue(Queries)

```sql
-- Top 10 best selling venues:

SELECT
    V.VENUEID,
    V.VENUENAME,
    SUM(S.QTYSOLD) AS TOTAL_TICKETS_SOLD
FROM Venue V
JOIN Events E ON V.VENUEID = E.VENUEID
JOIN Sales S ON E.EVENTID = S.EVENTID
GROUP BY V.VENUEID, V.VENUENAME
ORDER BY TOTAL_TICKETS_SOLD DESC
LIMIT 10;
```

```sql
-- Top 10 best grossing venues:

SELECT
    V.VENUEID,
    V.VENUENAME,
    SUM(S.QTYSOLD * L.PRICEPERTICKET) AS TOTAL_REVENUE
FROM Venue V
JOIN Events E ON V.VENUEID = E.VENUEID
JOIN Sales S ON E.EVENTID = S.EVENTID
JOIN Listings L ON S.LISTID = L.LISTID
GROUP BY V.VENUEID, V.VENUENAME
ORDER BY TOTAL_REVENUE DESC
LIMIT 10;
```

Top 10 Venues by Ticket Sales and Gross Revenue

VENUEID	VENUENAME	TOTAL_TICKETS_SOLD
220	Lunt-Fontanne Theatre	3326
203	August Wilson Theatre	3187
217	Hilton Theatre	2999
216	Helen Hayes Theatre	2948
226	Nederlander Theatre	2934
238	Winter Garden Theatre	2838
205	Ethel Barrymore Theatre	2828
243	Pasadena Playhouse	2739
218	Imperial Theatre	2702
208	Biltmore Theatre	2629

VENUEID	VENUENAME	TOTAL_REVENUE
220	Lunt-Fontanne Theatre	1115182
203	August Wilson Theatre	1032156
216	Helen Hayes Theatre	978765
238	Winter Garden Theatre	939257
226	Nederlander Theatre	936312
222	Majestic Theatre	894275
205	Ethel Barrymore Theatre	891172
217	Hilton Theatre	885686
218	Imperial Theatre	877993
248	Charles Playhouse	857031

- While there are many common venues in both the tables, there are a feew differences in the list of venues, indicating different in pricing and the possibility of venues having additional revenue streams apart from ticket sales.
- The comparison between the two tables shows Theatre as the most famous venue types, Organizers can use this info compare the market dynamics to generate more revenues.

Quarterly Performance by Ticket Sales and Gross Revenue(Queries)

```sql
-- Quarter performance Tickets sold:

SELECT
    D.YEAR,
    D.QTR,
    SUM(S.QTYSOLD) AS TOTAL_TICKETS_SOLD
FROM Date D
JOIN Events E ON D.DATEID = E.DATEID
JOIN Sales S ON E.EVENTID = S.EVENTID
GROUP BY D.YEAR, D.QTR
ORDER BY TOTAL_TICKETS_SOLD DESC;
```

```sql
-- Quarter performance by gross revenue:
SELECT
    D.YEAR,
    D.QTR,
    FORMAT(SUM(S.QTYSOLD * L.PRICEPERTICKET), 0) AS TOTAL_REVENUE
FROM Date D
JOIN Events E ON D.DATEID = E.DATEID
JOIN Sales S ON E.EVENTID = S.EVENTID
JOIN Listings L ON S.LISTID = L.LISTID
GROUP BY D.YEAR, D.QTR
ORDER BY TOTAL_REVENUE DESC;
```

Quarterly Performance by Ticket Sales and Gross Revenue

	YEAR	QTR	TOTAL_TICKETS_SOLD
▶	2008	3	97810
	2008	4	96826
	2008	2	94846
	2008	1	55867

	YEAR	QTR	TOTAL_REVENUE
▶	2008	3	31,123,584
	2008	2	30,912,322
	2008	4	30,844,098
	2008	1	17,885,427

- Quarterly performance by gross revenues and ticket sales shows consistent and robust trends throughout the year.
- The third and fourth quarters stand out as particularly strong periods, indicating sustained demand for events during the latter half of the year.
- Factors such as holiday seasons, increased cultural activities, and vacation periods might have contributed to the strong performance during these quarters.

Top 10 cities from where users have paid the most amount (Query)

```sql
-- Top 10 Cities with the higest amount paid by users:
SELECT
    U.CITY,
    SUM(S.PRICEPAID) AS TOTAL_AMOUNT_PAID
FROM Users U
JOIN Sales S ON U.USERID = S.BUYERID
GROUP BY U.CITY
ORDER BY TOTAL_AMOUNT_PAID DESC
LIMIT 10;
```

Top 10 cities from where users have paid the most amount?

CITY	TOTAL_AMOUNT_PAID
Richmond	343688
Columbia	296155
Charleston	276707
Springfield	247404
Hartford	238616
Concord	234245
Dover	221750
Jackson	207179
Columbus	204231
Medford	200996

- Cities like Richmond, Columbia, Charleston etc. indicate a strong market for events and entertainment. This suggests a potential opportunity for businesses to focus their efforts on attracting and engaging users.
- Dover, Jackson, Columbus, and Medford, although lower in terms of total amount paid by users compared to the top cities, still show a considerable level of spending. This implies a potential market for events and a willingness among users in these cities to invest in experiences.

Ranking Event Categories by Booking Success Rate (Query)

```sql
-- Success Rate For Each category
WITH SuccessRateByCategory AS (
    SELECT
        E.CATID,
        COUNT(*) AS TOTAL_LISTINGS,
        COUNT(CASE WHEN S.SALETIME IS NOT NULL THEN 1 END) AS SUCCESSFUL_LISTINGS,
        (COUNT(CASE WHEN S.SALETIME IS NOT NULL THEN 1 END) / COUNT(*)) * 100 AS SUCCESS_RATE
    FROM Listings L
    LEFT JOIN Sales S ON L.LISTID = S.LISTID
    JOIN Events E ON L.EVENTID = E.EVENTID
    GROUP BY E.CATID
)
SELECT
    RANK() OVER (ORDER BY SR.SUCCESS_RATE DESC) AS RANKING,
    C.CATNAME,
    SR.TOTAL_LISTINGS AS BOOKINGS_LISTED,
    SR.SUCCESSFUL_LISTINGS AS BOOKINGS_SUCCESSFUL,
    CONCAT(FORMAT(SR.SUCCESS_RATE, 2), '%') AS SUCCESS_RATE
FROM SuccessRateByCategory SR
JOIN Category C ON SR.CATID = C.CATID
ORDER BY RANKING;
```

Ranking Event Categories by Booking Success Rate

	RANKING	CATNAME	BOOKINGS_LISTED	BOOKINGS_SUCCESSFUL	SUCCESS_RATE
▶	1	Musicals	39327	25737	65.44%
	2	Pop	149257	97582	65.38%
	3	Opera	15180	9914	65.31%
	4	Plays	60105	39223	65.26%

- Musicals, Pop, Opera, and Plays are the top four categories based on the number of bookings listed.
- The success rate for all categories is relatively high, ranging from 65.26% to 65.44%.
- Despite the variations in the number of bookings listed, the success rates for all categories are similar, suggesting a consistent level of demand and audience interest across different genres.

Limitations

- We have data for only a single year – 2008. Having data for multiple years would be required to generate better insights
- Only 4 categories have data in other entities, narrowing the scope of many queries and analysis.

Key Insights

- Discrepancies between top-selling venues based on ticket sales and revenue indicate the need for pricing strategy optimization & understanding audience preferences.
- **Theatre venues** emerge as the most popular type, highlighting the importance of considering market dynamics to maximize revenue.
- The **third and fourth quarters** exhibit solid performance, indicating the importance of seasonality.
- **Richmond, Columbia, and Charleston** signify strong markets for events and entertainment. **Dover, Jackson, Columbus, and Medford** show significant future spending potential.
- **Musicals, Pop, Opera, and Plays** are the top four categories. All categories exhibit relatively high booking success rates across different genres.