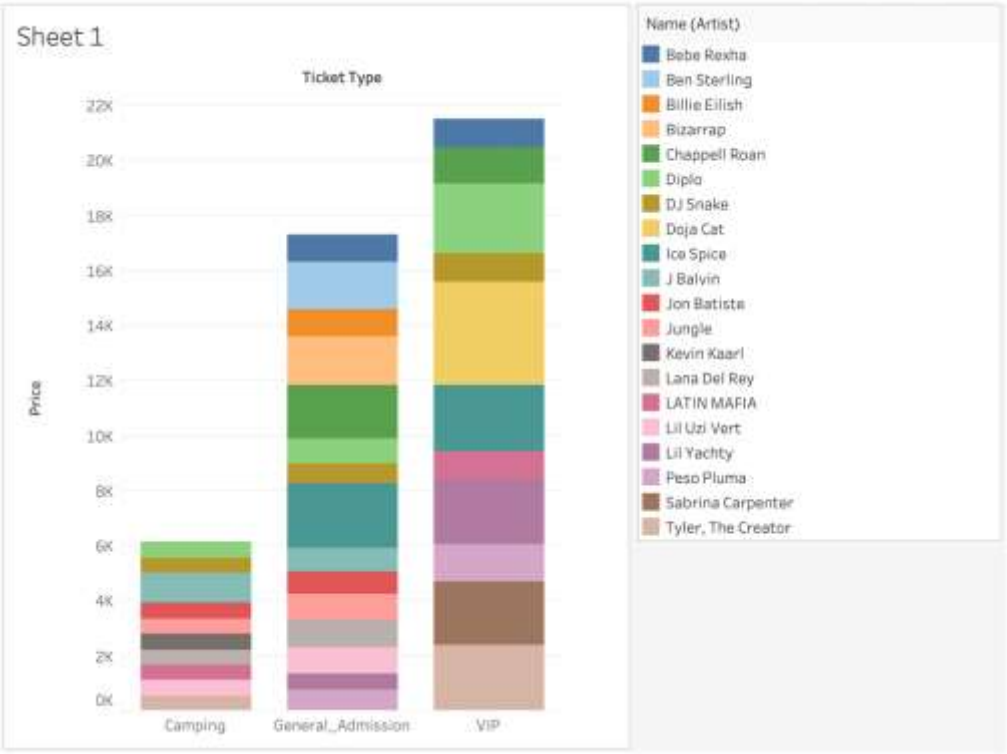
Coachella Visuals and Findings with SQL Code

QUERY 1: Total Revenue by Ticket Type

Query 1 gives us valuable information on *the total revenue by ticket type*. This can help us evaluate our marketing strategy for each ticket type and even perform an opportunity-cost analysis on the different ticket prices.

```
SELECT Ticket_Type, SUM(Price) AS total_revenue
FROM Ticket
GROUP BY Ticket_Type;
```

	Ticket_Ty...	total_revenue
1	Camping	5069.00
2	General_Ad...	18385.00
3	VIP	22607.00

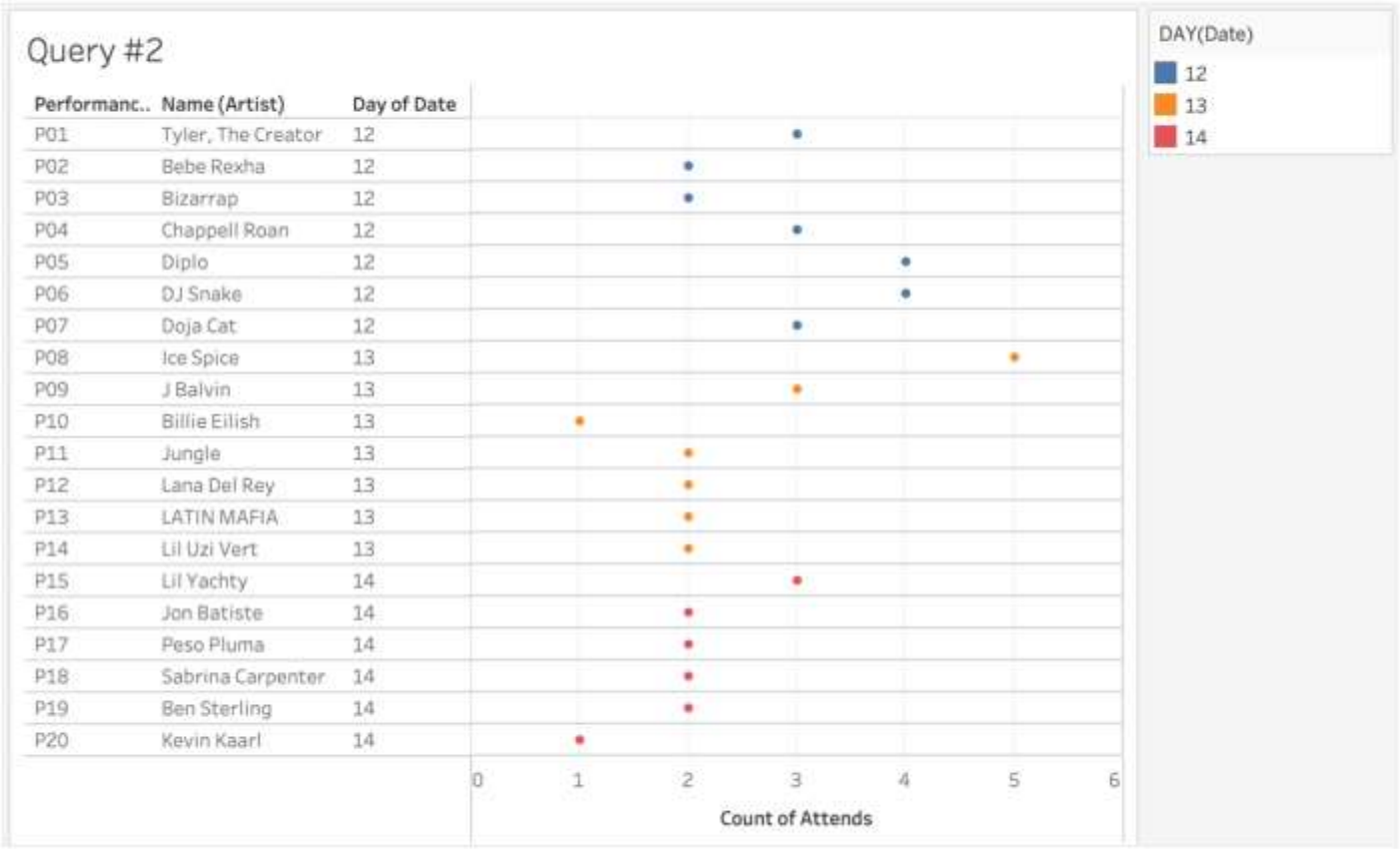


QUERY 2: The Most Popular Performance

Query 2 informs decisions *about artist selection, venue capacity, and targeted marketing* for successful events.

```
SELECT
  Performance.Performance_ID,
  Artist.Name AS Artist_Name,
  Performance.Date AS Performance_Date,
  COUNT(Attends.Buyer_ID) AS Total_Attendees
FROM
  Performance
JOIN Artist ON Performance.Artist_ID = Artist.Artist_ID
JOIN Attends ON Performance.Performance_ID = Attends.Performance_ID
GROUP BY
  Performance.Performance_ID, Artist.Name, Performance.Date
ORDER BY
  Total_Attendees DESC;
```

	Performa...	Artist_Name	Performan...	Total_Attendees
1	P08	Ice Spice	2024-04-13	5
2	P05	Diplo	2024-04-12	4
3	P06	DJ Snake	2024-04-12	4
4	P15	Lil Yachty	2024-04-14	3
5	P01	Tyler, The Creator	2024-04-12	3
6	P04	Chappell Roan	2024-04-12	3
7	P09	J Balvin	2024-04-13	3
8	P07	Doja Cat	2024-04-12	3
9	P12	Lana Del Rey	2024-04-13	2
10	P17	Peso Pluma	2024-04-14	2



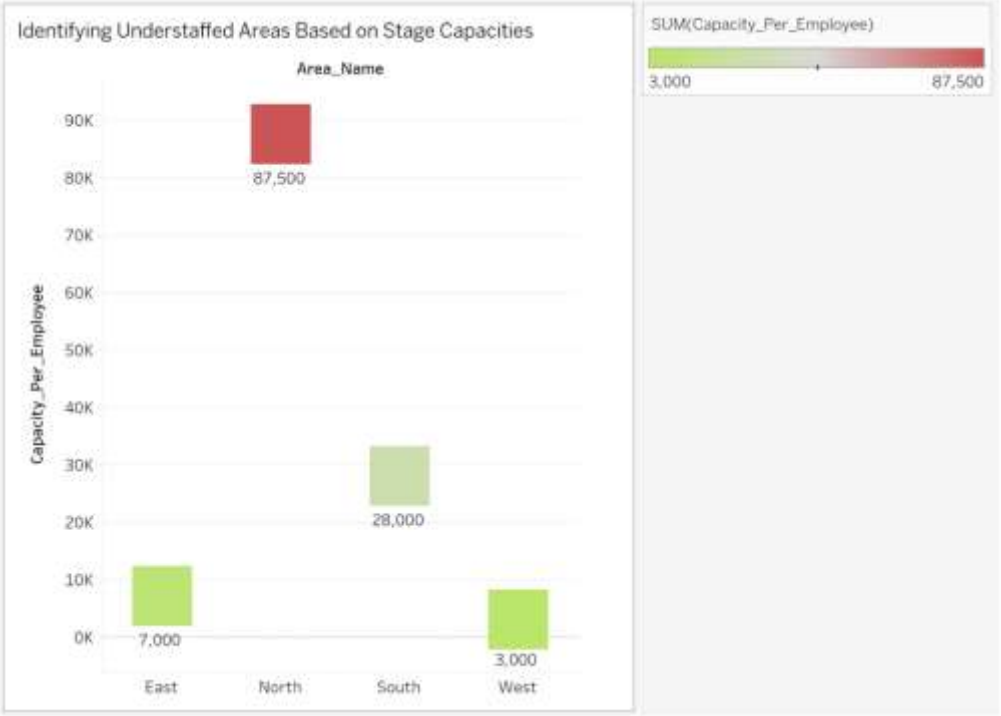
QUERY 3: Identifying Understaffed Areas: Employee Coverage vs. Stage Capacity

Query 3 shows us *which areas are understaffed* based on employee assignments relative to stage capacity. This can help in reallocating employees to other areas to ensure smooth operations.

```
SELECT
  A.Area_Name,
  SUM(S.Capacity) AS Total_Stage_Capacity,
  COUNT(E.Employee_ID) AS Employees_Assigned,
  (SUM(S.Capacity) / COUNT(E.Employee_ID)) AS Capacity_Per_Employee
FROM Area A
JOIN Stage S ON A.Area_ID = S.Area_ID
LEFT JOIN Employees E ON A.Area_ID = E.Area_ID
GROUP BY A.Area_Name
ORDER BY Capacity_Per_Employee DESC;
```

```
SELECT
    A.Area_Name,
    SUM(S.Capacity) AS Total_Stage_Capacity,
    COUNT(E.Employee_ID) AS Employees_Assigned,
    (SUM(S.Capacity) / COUNT(E.Employee_ID)) AS Capacity_Per_Employee
FROM Area A
JOIN Stage S ON A.Area_ID = S.Area_ID
LEFT JOIN Employees E ON A.Area_ID = E.Area_ID
GROUP BY A.Area_Name
ORDER BY Capacity_Per_Employee DESC;
```

	Area_Name	Total_Stage_Capacity	Employees_Assigned	Capacity_Per_Employee
1	North	875000	10	87500.0000
2	South	280000	10	28000.0000
3	East	84000	12	7000.0000
4	West	18000	6	3000.0000



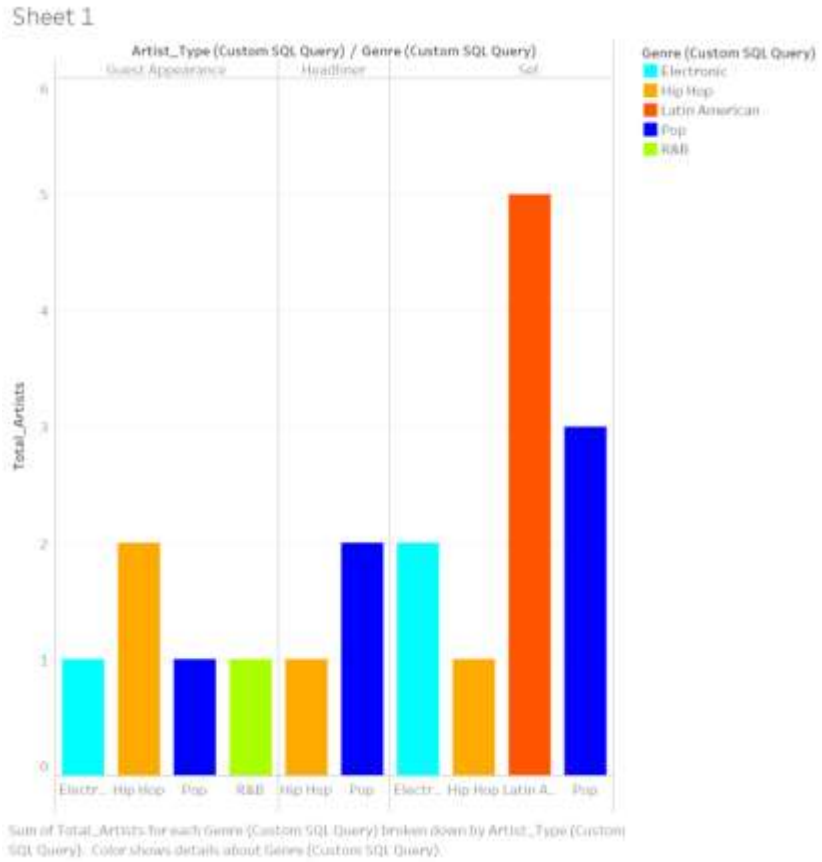
QUERY 4: Counting the number of artists by their type and genre combination

Query 4 provides a breakdown of *the number of artists based on their type* (e.g. Headliner, Set, Guest Appearance) and genre (eg. Hip Hop, Pop, Electronic). This is useful for analyzing the diversity of artist types and genres represented in the event.

```
SELECT
  Artist_Type,
  Genre,
  COUNT(Artist_ID) AS Total_Artists
FROM
  Artist
GROUP BY
  Artist_Type, Genre
ORDER BY
  Artist_Type ASC, Genre ASC;
```

```sql
SELECT
    Artist_Type,
    Genre,
    COUNT(Artist_ID) AS Total_Artists
FROM
    Artist
GROUP BY
    Artist_Type, Genre
ORDER BY
    Artist_Type ASC, Genre ASC;
```

	Artist_Type	Genre	Total_Artists
1	Guest Appe...	Electronic	1
2	Guest Appe...	Hip Hop	2
3	Guest Appe...	Pop	1
4	Guest Appe...	R&B	1
5	Headliner	Hip Hop	1
6	Headliner	Pop	2
7	Set	Electronic	2
8	Set	Hip Hop	1
9	Set	Latin American	5
10	Set	Pop	3



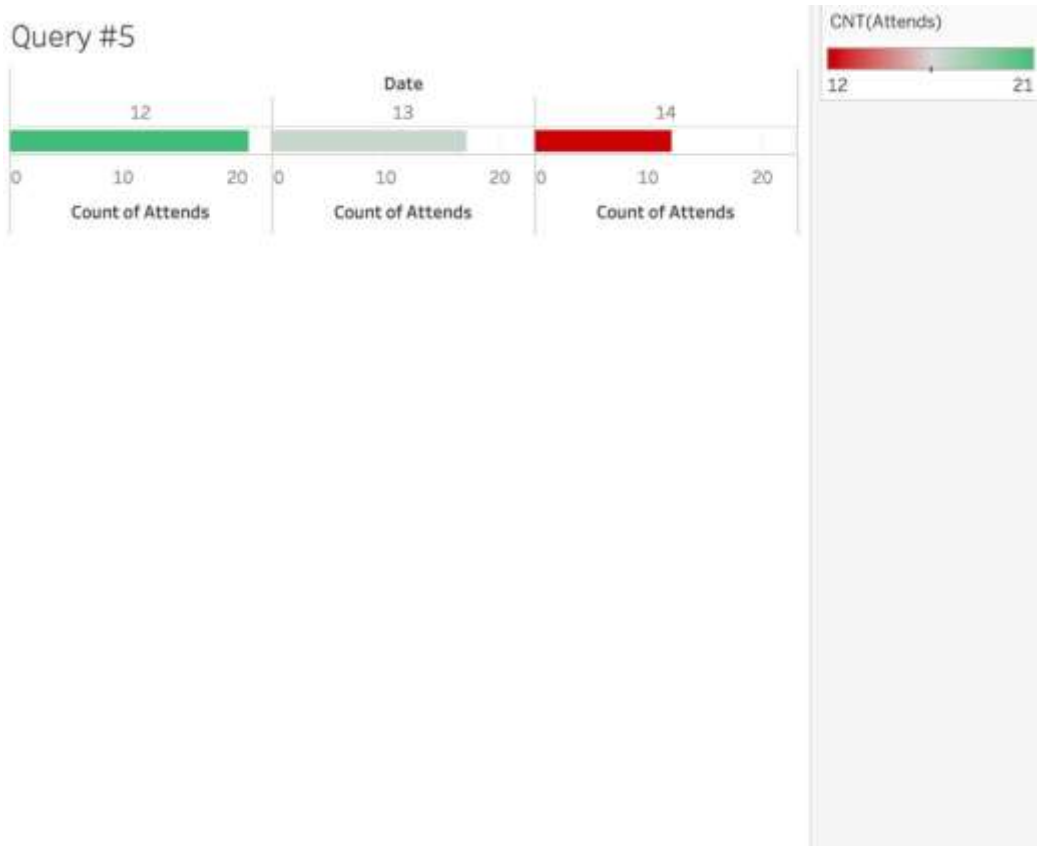
QUERY 5: The Most Crowded Day

Query 5 helps with *event logistics, crowd management, and identifying peak days* for future planning.

```
SELECT
  Performance.Date AS Event_Date,
  COUNT(Attends.Buyer_ID) AS Total_Attendees
FROM
  Performance
JOIN
  Attends ON Performance.Performance_ID = Attends.Performance_ID
GROUP BY
  Performance.Date
ORDER BY
  Total_Attendees DESC
LIMIT 1;
```

```sql
SELECT
    Performance.Date AS Event_Date,
    COUNT(Attends.Buyer_ID) AS Total_Attendees
FROM
    Performance
JOIN
    Attends ON Performance.Performance_ID = Attends.Performance_ID
GROUP BY
    Performance.Date
ORDER BY
    Total_Attendees DESC
LIMIT 1;
```

	Event_Date	Total_Attendees
1	2024-04-12	21



QUERY 6: The Most Popular Genre

Query 6 offers insights into *market trends and helps curate relevant events, recruit talent, and align marketing efforts* with audience preferences.

SELECT Genre, COUNT(*) AS Artist_Count
FROM Artist
GROUP BY Genre
ORDER BY Artist_Count DESC
LIMIT 1;

```sql
SELECT Genre, COUNT(*) AS Artist_Count
FROM Artist
GROUP BY Genre
ORDER BY Artist_Count DESC
LIMIT 1;
```

	Genre	Artist_Count
1	Pop	6