



Lila

Event Organization



Ankara Yıldırım Beyazıt University

MIS204 Project Assignments *Lila Event Database System* ***Project Members;***

- *Aliye Nur Bulut - 21030411042*
- *Salih Eren Çakar - 210304110*
- *Mervenur Güler - 22030411057*
- *Esin Nur Karcı - 21030411044*
- *Hasan Feyzi Kertmen - 22030411047*
- *Fatma Uslu - 22030411053*

Project Introduction:

Overview:

In the contemporary context of an ever-increasing number of events, the effective organisation and management of events has become paramount for ensuring a smooth and satisfactory user experience. A well-structured database system is imperative for the effective organisation of events, ensuring the accurate and efficient storage, access, and management of information. Such a system must encompass all pertinent details, including ticketing, attendee information, and venue data, facilitating the seamless execution of events ranging from concerts and workshops to conferences and festivals.

The project under discussion introduces a custom-built Event Database System, the purpose of which is to handle the complete lifecycle of events. The system facilitates the browsing of events, the purchase of tickets, and the monitoring of participation, while enabling the administrator (the event organiser) to create and manage events from a centralised structure.

Objective:

The main objective of this project is to design and implement a relational database system that efficiently:

- Organize and store detailed information about different cultural events.
- Manage interactions between different entities such as participants, artists, and sales agents.
- Enable efficient retrieval and manipulation of event data to enhance the planning and execution of events.

Business Requirements:

The requirements for this project were developed with the aim of enabling more detailed and comprehensive presentation of data in the event of establishing a potential large-scale event firm. Consequently, this would allow for the compilation and display of events catering to different audiences with various styles on different days. The identified key requirements are as follows:

Event Management: Capture essential details about events, including event name, date & time, and the associated location. Each event should be linked to specific rules and categorized by ticket types.

Location Management: Maintain detailed information about event venues, including the venue name, address, and seating capacity.

Artist Information Management: Store information about artists involved in events, such as their names and contact details, and track their participation in specific events.

Participant Management: Record and manage data about event participants, including personal details and the tickets they have purchased.

Ticket Management: Track ticket sales for events, including details about ticket prices, seat numbers, and sales agents.

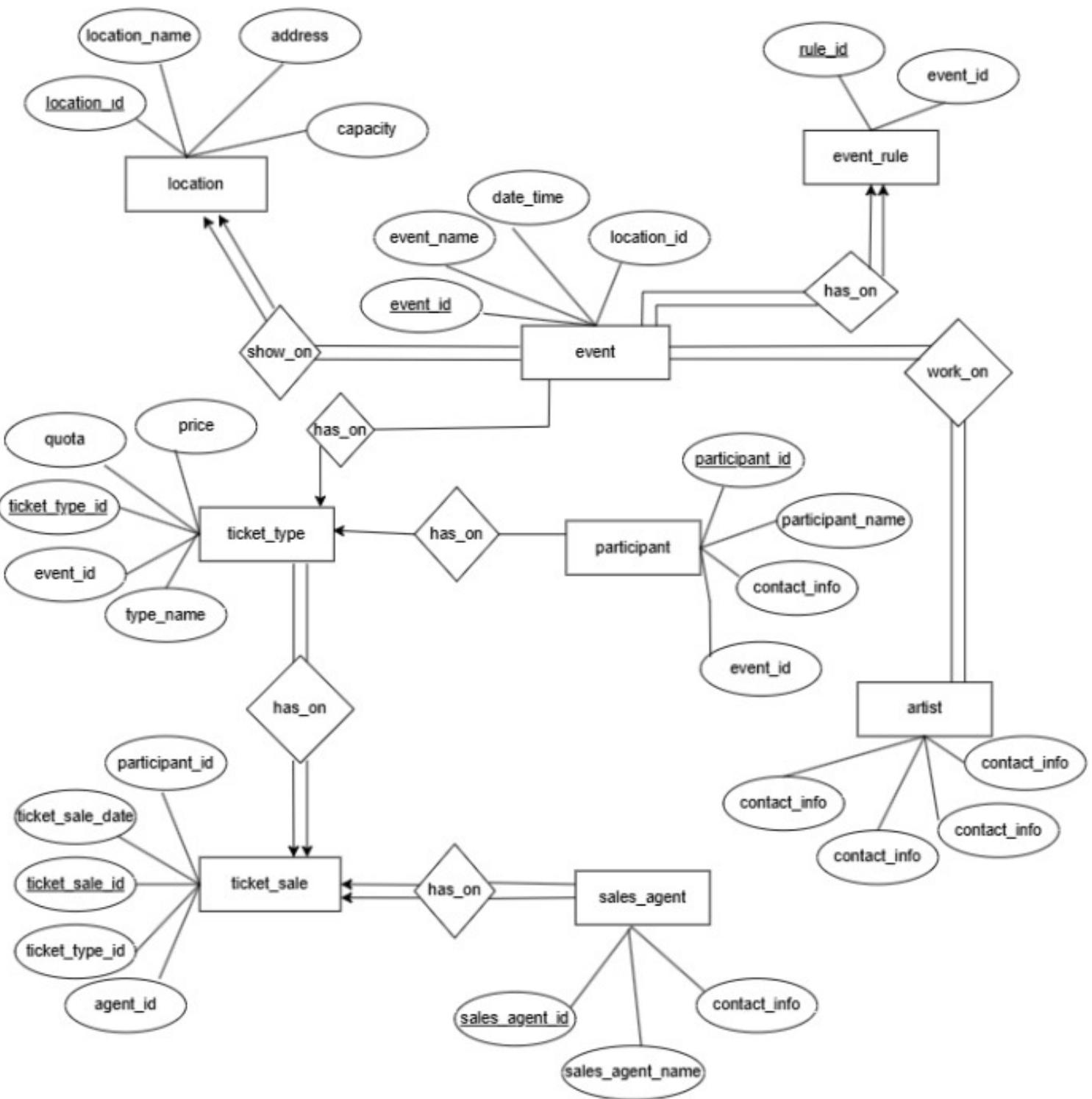
Sales Agent Management: Store contact information and names of sales agents who facilitate ticket sales for events.

Event Rule Definition: Associate specific rules or conditions with events (e.g., age limits, entry times) and allow rule-based queries or filtering.

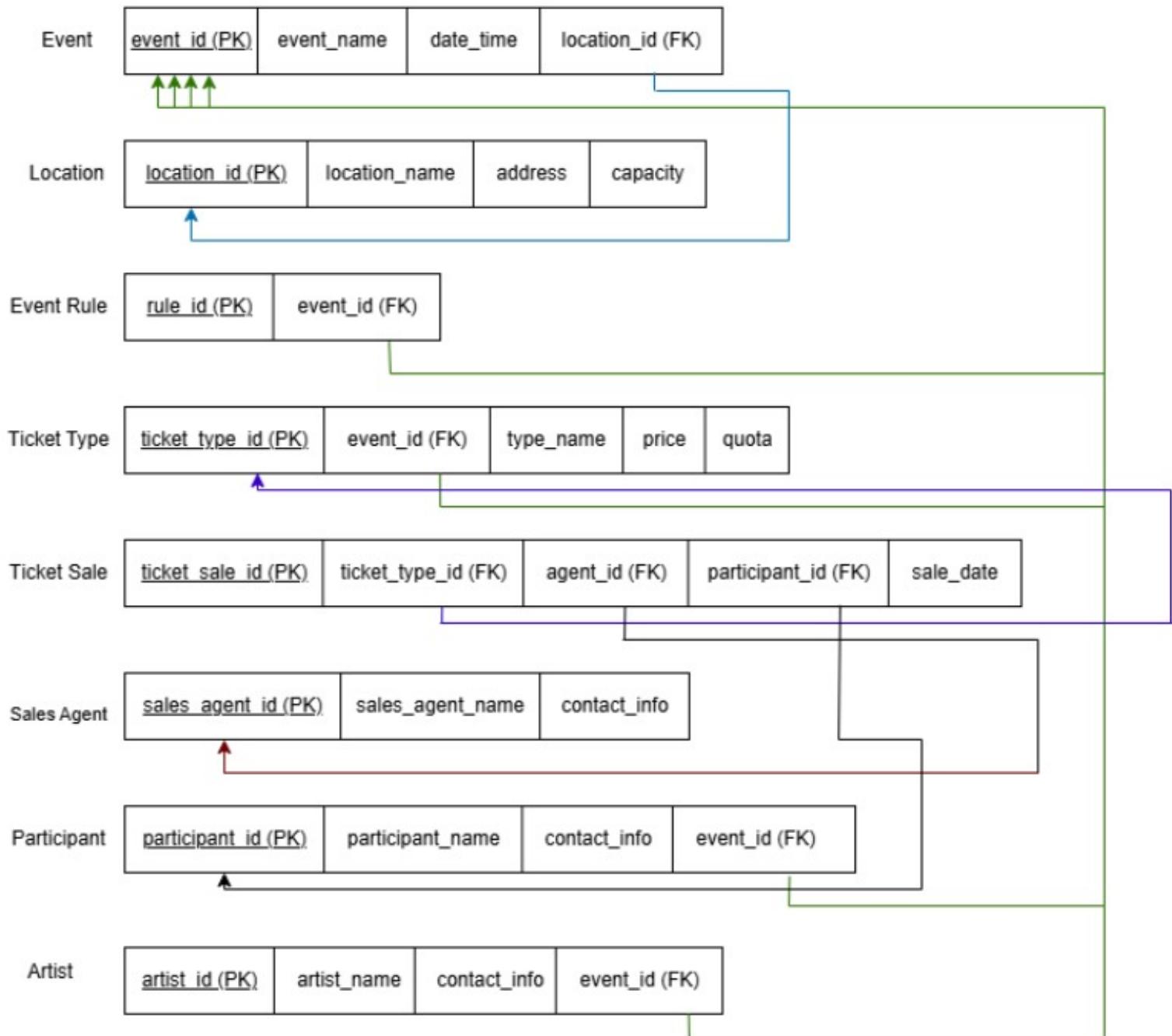
Search and Reporting Functionality: Allow for event searches by date, location, or type, and provide detailed sales reports based on events and ticket types.

Data Relationships and Integrity: Ensure that relationships among entities (event, location, artist, participant, ticket, etc.) are clearly established and data integrity is preserved.

ER Diagram:



Relational Mapping:



Creating Tables and Inserting Data by Using SQL:

```
1 • |create database lila;
2
3 • |CREATE TABLE Locations (
4 |  Location_ID INT AUTO_INCREMENT,
5 |  Location_Name VARCHAR(100),
6 |  Address VARCHAR(255),
7 |  Capacity INT,
8 |  PRIMARY KEY (Location_ID)
9
10 );
11
12 • |CREATE TABLE Events (
13 |  Event_ID INT AUTO_INCREMENT,
14 |  Event_Name VARCHAR(100),
15 |  Date_Time DATETIME,
16 |  Location_ID INT,
17 |  PRIMARY KEY (Event_ID),
18 |  FOREIGN KEY (Location_ID) REFERENCES Locations(Location_ID)
19
20 );
21
22 • |CREATE TABLE Sales_Agents (
23 |  Agent_ID INT AUTO_INCREMENT,
24 |  NAME VARCHAR(100),
25 |  Contact_Info VARCHAR(100),
26 |  PRIMARY KEY (Agent_ID)
27
28 );
29
30 • |CREATE TABLE Participants (
31 |  Participant_ID INT AUTO_INCREMENT,
32 |  Event_ID INT,
33 |  Name VARCHAR(100),
34 |  Contact_Info VARCHAR(100),
35 |  PRIMARY KEY (Participant_ID),
36 |  FOREIGN KEY (Event_ID) REFERENCES Events(Event_ID)
37
38 );
39
40 • |CREATE TABLE Artists (
41 |  Artist_ID INT AUTO_INCREMENT,
42 |  Event_ID INT,
43 |  Contact_Info VARCHAR(100),
44 |  Name VARCHAR(100),
45 |  PRIMARY KEY (Artist_ID),
46 |  FOREIGN KEY (Event_ID) REFERENCES Events(Event_ID)
47
48 );
49
50 • |CREATE TABLE Tickets_Type (
51 |  Ticket_ID INT AUTO_INCREMENT,
52 |  Event_ID INT,
53 |  Participant_ID INT,
54 |  Ticket_Type_ID INT,
55 |  Price DECIMAL(10,2),
56 |  Seat_Number INT,
57 |  PRIMARY KEY (Ticket_ID),
58 |  FOREIGN KEY (Event_ID) REFERENCES Events(Event_ID),
59 |  FOREIGN KEY (Participant_ID) REFERENCES Participants(Participant_ID)
60
61 );
62
```

```

61     );
62
63 • CREATE TABLE Ticket_Sales (
64     Sale_ID INT AUTO_INCREMENT,
65     Ticket_ID INT,
66     Agent_ID INT,
67     Sale_Date DATETIME,
68     Participant_ID INT,
69     PRIMARY KEY (Sale_ID),
70     FOREIGN KEY (Ticket_ID) REFERENCES Tickets_Type(Ticket_ID),
71     FOREIGN KEY (Agent_ID) REFERENCES Sales_Agents(Agent_ID),
72     FOREIGN KEY (Participant_ID) REFERENCES Participants(Participant_ID)
73 );
74
75
76 • CREATE TABLE Event_Rule (
77     Event_Rule_ID INT AUTO_INCREMENT,
78     Event_ID INT,
79     Rule_Description VARCHAR(100),
80     PRIMARY KEY (Event_Rule_ID),
81     FOREIGN KEY (Event_ID) REFERENCES Events(Event_ID)
82 );
83
84
85 • INSERT INTO Locations (Location_Name, Address, Capacity) VALUES
86     ('Lila Arena', '123 Sunset Blvd, Los Angeles, CA', 15000),
87     ('Echo Hall', '456 Harmony St, New York, NY', 8000),
88     ('Skyline Stage', '789 Skyline Ave, Chicago, IL', 12000),
89     ('Moonlight Park', '321 Moon Rd, Austin, TX', 10000),
90     ('Nova Center', '654 Star Ln, Seattle, WA', 9000),
91     ('Pulse Dome', '987 Beat Blvd, Miami, FL', 14000);
92
93 -----
94
95 • INSERT INTO Events (Event_Name, Date_Time, Location_ID ) VALUES
96     ('Annual Music Festival', '2025-08-15 19:00:00', 1),
97     ('Local Theater Night', '2025-09-20 20:00:00', 2),
98     ('Tech Conference 2025', '2025-10-05 09:00:00', 3);
99
100 -----
101
102 • INSERT INTO Participants (Event_ID, Name, Contact_Info) VALUES
103     (1, 'Alice Johnson', 'alice.johnson@example.com'),
104     (2, 'Bob Smith', 'bob.smith@example.com'),
105     (3, 'Carol White', 'carol.white@example.com'),
106     (1, 'Dave Brown', 'dave.brown@example.com'),
107     (2, 'Eve Black', 'eve.black@example.com'),
108     (1, 'Frank Grey', 'frank.grey@example.com'),
109     (2, 'Grace Red', 'grace.red@example.com'),
110     (3, 'Harry Blue', 'harry.blue@example.com'),
111     (1, 'Nina Green', 'nina.green@example.com'),
112     (2, 'Oscar Silver', 'oscar.silver@example.com');
113
114 -----
115
116
117 • INSERT INTO Sales_Agents (Name, Contact_Info) VALUES
118     ('Sam Green', 'sam.green@salesagency.com'),
119     ('Judy Blue', 'judy.blue@salesagency.com'),
120     ('Mark Orange', 'mark.orange@salesagency.com'),
121     ('Lily Purple', 'lily.purple@salesagency.com'),
122     ('Tom Yellow', 'tom.yellow@salesagency.com');
123
124 -----
125

```

```
124 -----  
125  
126 • INSERT INTO Artists (Event_ID, Name ,Contact_Info) VALUES  
127 (1, 'Liam Harmony', 'liam.harmony@example.com'),  
128 (1, 'Ava Storm', 'ava.storm@example.com'),  
129 (2, 'Noah Blaze', 'noah.blaze@example.com'),  
130 (2, 'Mia Skye', 'mia.skye@example.com'),  
131 (3, 'Ethan Phoenix', 'ethan.phoenix@example.com'),  
132 (3, 'Luna Ray', 'luna.ray@example.com'),  
133 (3, 'Oliver Vibe', 'oliver.vibe@example.com'),  
134 (1, 'Isla Moon', 'isla.moon@example.com'),  
135 (1, 'Elijah Sound', 'elijah.sound@example.com'),  
136 (2, 'Aria Flame', 'aria.flame@example.com'),  
137 (2, 'James Echo', 'james.echo@example.com'),  
138 (3, 'Sophie Beat', 'sophie.beat@example.com'),  
139 (3, 'Lucas Nova', 'lucas.nova@example.com'),  
140 (1, 'Zoe Pulse', 'zoe.pulse@example.com'),  
141 (2, 'Leo Drift', 'leo.drift@example.com');  
142 -----  
143  
144 -----  
145 • INSERT INTO Tickets_Type (Event_ID, Participant_ID, Ticket_Type_ID, Price, Seat_Number) VALUES  
146 (1, 3, 1, 150.00, 101),  
147 (2, 5, 2, 200.00, 202),  
148 (1, 1, 3, 150.00, 102),  
149 (3, 4, 1, 175.00, 303),  
150 (2, 2, 2, 200.00, 203),  
151 (3, 6, 3, 120.00, 404),  
152 (1, 7, 1, 150.00, 103),  
153 (3, 8, 2, 175.00, 304),  
154 (2, 9, 3, 200.00, 204),  
155 (2, 10, 1, 120.00, 405);  
156  
157 -----  
158  
159 • INSERT INTO Ticket_Sales (Ticket_ID, Agent_ID, Participant_ID, Sale_Date) VALUES  
160 (1, 2, 5, '2025-04-10'),  
161 (2, 1, 3, '2025-04-11'),  
162 (3, 2, 5, '2025-04-12'),  
163 (1, 1, 2, '2025-04-13'),  
164 (4, 2, 3, '2025-04-14'),  
165 (2, 2, 1, '2025-04-15'),  
166 (3, 1, 4, '2025-04-16'),  
167 (5, 2, 2, '2025-04-17'),  
168 (1, 1, 3, '2025-04-17'),  
169 (2, 2, 4, '2025-04-18');  
170  
171 -----  
172  
173 • INSERT INTO Event_Rule (Rule_Description, Event_ID) VALUES  
174 ('No outside food or drinks allowed.', 1),  
175 ('Attendees must be 18 or older.', 2),  
176 ('Photography is prohibited.', 3),  
177 ('Face masks are required at all times.', 1),  
178 ('Tickets are non-refundable.', 1),  
179 ('No re-entry after leaving the venue.', 2);
```

Normalization Process for Ticket Table:

Before Normalization:

Ticket_ID	Event_ID	Participant_ID	Price	Seat_Number
1	3	1	150.00	101
2	5	2	200.00	202
1	1	3	150.00	102
3	4	1	175.00	303
2	2	2	200.00	203
3	6	3	120.00	404
1	7	1	150.00	103
1	8	2	175.00	304
2	9	3	200.00	204
2	10	1	120.00	405

First Normal Form of Ticket Table:

Ticket_ID	Event_ID	Participant_ID	Price	Seat_Number
1	3	1	150.00	101
2	5	2	200.00	202
3	1	3	150.00	102
4	4	1	175.00	303
5	2	2	200.00	203
6	6	3	120.00	404
7	7	1	150.00	103
8	8	2	175.00	304
9	9	3	200.00	204
10	10	1	120.00	405

Second Normal Form of Ticket Table:

Ticket_ID	Event_ID	Participant_ID	Seat_Number
1	3	1	101
2	5	2	202
3	1	3	102
4	4	1	303
5	2	2	203
6	6	3	404
7	7	1	103
8	8	2	304
9	9	3	204
10	10	1	405

Ticket_ID	Event_ID	Participant_ID
1	3	1
2	2	2
3	4	3
4	1	1
5	3	2
6	3	3
7	7	1
8	3	2
9	3	3
10	2	1

Event_ID	Price
1	150.00
2	200.00
3	150.00
4	175.00
5	200.00
6	120.00
7	150.00
8	150.00
9	200.00
10	120.00

Normalization Process for Ticket Sales Table:

Before Normalization:

Ticket_ID	Agent_ID	Participant_ID	Sale_Date
1	2	5	2025-04-10
2	1	3	2025-04-11
3	2	5	2025-04-12
1	1	2	2025-04-13
4	2	3	2025-04-14
2	2	1	2025-04-15
3	1	4	2025-04-16
5	2	2	2025-04-17
1	1	3	2025-04-17
2	2	4	2025-04-18

First Normal Form of Ticket Sales Table:

Ticket_ID	Agent_ID	Participant_ID	Sale_Date
1	2	5	2025-04-10
2	1	3	2025-04-11
3	2	5	2025-04-12
4	1	2	2025-04-13
5	2	3	2025-04-14
6	2	1	2025-04-15
7	1	4	2025-04-16
8	2	2	2025-04-17
9	1	3	2025-04-17
10	2	4	2025-04-18

Normalization Process for Location Table:

Location_Name	Address	Capacity
Lila Arena	123 Sunset Blvd, Los Angeles, CA	15000
Echo Hall	456 Harmony St, New York, NY	8000
Skyline Stage	789 Skyline Ave, Chicago, IL	12000
Moonlight Park	321 Moon RD, Austin, TX	10000
Nova Center	654 Star LN, Seattle, WA	9000
Pulse Dome	987 Beat Blvd, Miami, FL	14000

Normalization Process for Event Table:

Event_Name	Date_Time	Location_ID
Annual Music Festival	2025-08-15 19:00:00	1
Local Theatre Night	2025-09-20 20:00:00	2
Tech Conference 2025	2025-10-05 09:00:00	3

Normalization Process for Participation Table:

Before Normalization:

Event_ID	Name	Contact_Info
1	Alice Johnson	alice.johnson@example.com
2	Bob Smith	bob.smith@example.com
3	Carol White	carol.white@example.com
1	Dave Brown	dave.brown@example.com
2	Eve Black	eve.black@example.com
1	Frank Grey	frank.grey@example.com
2	Grace Red	grace.red@example.com
3	Harry Blue	harry.blue@example.com
1	Nina Green	nina.green@example.com
2	Oscar Silver	oscar.silver@example.com

Initial Participation Table

First Normal Form of Participation Table:

Participant_ID	Event_ID	Name	Contact_Info
1	1	Alice Johnson	alice.johnson@example.com
2	2	Bob Smith	bob.smith@example.com
3	3	Carol White	carol.white@example.com
4	1	Dave Brown	dave.brown@example.com
5	2	Eve Black	eve.black@example.com
6	1	Frank Grey	frank.grey@example.com
7	2	Grace Red	grace.red@example.com
8	3	Harry Blue	harry.blue@example.com
9	1	Nina Green	nina.green@example.com
10	2	Oscar Silver	oscar.silver@example.com

Normalization Process for Sales Agent Table:

Before Normalization:

Name	Contact_Info
Sam Green	sam.green@salesagency.com
Judy Blue	judy.blue@salesagency.com
Mark Orange	mark.orange@salesagency.com
Lily Purple	lily.purple@salesagency.com
Tom Yellow	tom.yellow@salesagency.com

First Normal Form of Sales Agent Table:

Agent_ID	Name	Contact_Info
1	Sam Green	sam.green@salesagency.com
2	Judy Blue	judy.blue@salesagency.com
3	Mark Orange	mark.orange@salesagency.com
4	Lily Purple	lily.purple@salesagency.com
5	Tom Yellow	tom.yellow@salesagency.com

Normalization Process for Artist Table:

Event_ID	Name	Contact_Info
1	Liam Harmony	liam.harmony@example.com
1	Ava Storm	ava.storm@example.com
2	Noah Blaze	noah.blaze@example.com
2	Mia Skye	mia.skye@example.com
3	Ethan Phoenix	ethan.phoenix@example.com
3	Luna Ray	luna.ray@example.com
3	Oliver Vibe	oliver.vibe@example.com
1	Isla Moon	isla.moon@example.com
1	Elijah Sound	elijah.sound@example.com
2	Aria Flame	aria.flame@example.com
2	James Echo	james.echo@example.com
3	Sophie Beat	sophie.beat@example.com
3	Lucas Nova	lucas.nova@example.com
1	Zoe Pulse	zoe.pulse@example.com
2	Leo Drift	leo.drift@example.com

Normalization Process for Event Rule Table:

Rule_Description	Event_ID
No outside food or drinks allowed	1
Attendees must be 18 or older	2
Photography is prohibited	3
Face masks are required at all times	1
Tickets are non-refundable	1
No re-entry after leaving the venue	2

SQL QUERIES:

Code for Artist Names Starting With L:

```
1 •   SELECT *
2     FROM Artists
3      WHERE Name LIKE 'L%';|
4
```

Code for Artist Names Starting With L:

Artist_ID	Event_ID	Contact_Info	Name
1	1	liam.harmony@example.com	Liam Harmony
6	3	luna.ray@example.com	Luna Ray
13	3	lucas.nova@example.com	Lucas Nova
15	2	leo.drift@example.com	Leo Drift
16	1	liam.harmony@example.com	Liam Harmony
21	3	luna.ray@example.com	Luna Ray
28	3	lucas.nova@example.com	Lucas Nova
30	2	leo.drift@example.com	Leo Drift
NULL	NULL	NULL	NULL

Code for List How Many Rules There are for Each Event in the Event Rule :

```
1 •  SELECT Events.Event_Name, COUNT(Event_Rule.Event_Rule_ID) AS Rule_Count  
2   FROM Event_Rule  
3   JOIN Events ON Event_Rule.Event_ID = Events.Event_ID  
4   GROUP BY Events.Event_Name;
```

Table for List How Many Rules There are for Each Event in the Event Rule :

Event_Name	Rule_Count
Annual Music Festival	3
Local Theater Night	2
Tech Conference 2025	1

Code for List the Participants Name Starting with A:

```
1 •  SELECT *
2   FROM Participants
3 WHERE Name LIKE 'A%';
4 |
```

Table for List the Participants Name Starting with A:

Participant_ID	Event_ID	Name	Contact_Info
1	1	Alice Johnson	alice.johnson@example.com
NULL	NULL	NULL	NULL

Code for Events that Will Take Place after September 1, 2025:

```
1 •  SELECT *
2   FROM Events
3 WHERE Date_Time > '2025-09-01';
4
```

Table for Events that Will Take Place after September 1, 2025:

Event_ID	Event_Name	Date_Time	Location_ID
2	Local Theater Night	2025-09-20 20:00:00	2
3	Tech Conference 2025	2025-10-05 09:00:00	3
NULL	NULL	NULL	NULL

Code for List the Sales Representatives whose Name is “Blue”:

```
1 •  SELECT *
2   FROM Sales_Agents
3 WHERE Name LIKE '%Blue%';
4 |
```

Table for List the Sales Representatives whose Name is “Blue”:

Agent_ID	Name	Contact_Info
2	Judy Blue	judy.blue@salesagency.com
7	Judy Blue	judy.blue@salesagency.com
12	Judy Blue	judy.blue@salesagency.com
NONE	NONE	NONE

Code for List Venues with a Capacity Greater than 10,000:

```
1 •   SELECT *
2     FROM Locations
3    WHERE Capacity > 10000;
4
```

Table for List Venues with a Capacity Greater than 10,000:

Location_ID	Location_Name	Address	Capacity
1	Lila Arena	123 Sunset Blvd, Los Angeles, CA	15000
3	Skyline Stage	789 Skyline Ave, Chicago, IL	12000
6	Pulse Dome	987 Beat Blvd, Miami, FL	14000
NULL	NULL	NULL	NULL

Code for List Events that Will Take Place after September 1, 2025 :

```
1 •  SELECT *
2   FROM Events
3 WHERE Date_Time > '2025-09-01';
4
```

Table for List Events that Will Take Place after September 1, 2025 :

Event_ID	Event_Name	Date_Time	Location_ID
2	Local Theater Night	2025-09-20 20:00:00	2
3	Tech Conference 2025	2025-10-05 09:00:00	3
HULL	HULL	HULL	HULL

Code for Max Price of Tickets:

```
1 •  SELECT * FROM lila.Tickets_Type;  
2 •  SELECT  
3      MAX(Price) AS Max_Ticket_Price  
4  FROM  
5      Tickets_Type;
```

Table for Max Price of Tickets:

Max_Ticket_Price
200.00

Code for Min Price of Tickets:

```
1 •  SELECT * FROM lila.Tickets_Type;  
2 •  SELECT  
3      MIN(Price) AS Min_Ticket_Price  
4  FROM  
5    Tickets_Type;
```

Table for Min Price of Tickets:

Min_Ticket_Pri...
120.00

Code of Sorting Price:

```
1 •  SELECT * FROM lila.Tickets_Type;  
2 •  SELECT  
3     Event_ID,  
4     Price  
5  FROM Tickets_Type ORDER BY  
6     Price DESC;
```

Table for Sorting Price:

Event_ID	Price
2	200.00
2	200.00
2	200.00
3	175.00
3	175.00
1	150.00
1	150.00
1	150.00
3	120.00
2	120.00

DATA DICTIONARY:

Entities	Attributes	Content	Types	Formats
EVENTS	Event-ID	id of event	INT	PK
	Event_Name	name of event	VARCHAR(100)	
	Date_Name	date of event	DATETIME	
	Location_ID	id of location	INT	FK
	Event_Type_ID	id of event type	INT	FK
LOCATIONS	Location-ID	id of location	INT	PK
	Location_Name	name of location	VARCHAR(100)	
	Address	address of event	VARCHAR(255)	
	Capacity	capacity of area	INT	
ARTIST	Artist-ID	id of artist	INT	PK
	Event_ID	id of event	INT	FK
	Name	name of artist	VARCHAR(100)	
	Contact_Info	contact information of artist	VARCHAR(100)	
EVENT_TYPES	Event-Type-ID	id of event type	INT	PK
	Type_Description	event type	VARCHAR(100)	
TICKETS	Ticket-ID	id of ticket	INT	PK
	Event_ID	id of event	INT	FK
	Participant_ID	id of participants	INT	FK
	Price	price of ticket	DECIMAL(10, 2)	
	Seat_Number	number of seat	INT	
TICKET_SALES	Ticket-ID	id of ticket	INT	PK
	Agent_ID	id of agent	INT	PK-FK
	Sale_Date	date of sale	DATETIME	
SALES_AGENTS	Agent-ID	id of agent	INT	PK
	Name	name of agent	VARCHAR(100)	
	Contact_Info	contact information of agent	VARCHAR(100)	
PARTICIPANTS	Participant-ID	id of participant	INT	PK
	Name	name of participant	VARCHAR(100)	
	Event_ID	id of event	INT	FK
	Contact_Info	contact information	VARCHAR(100)	