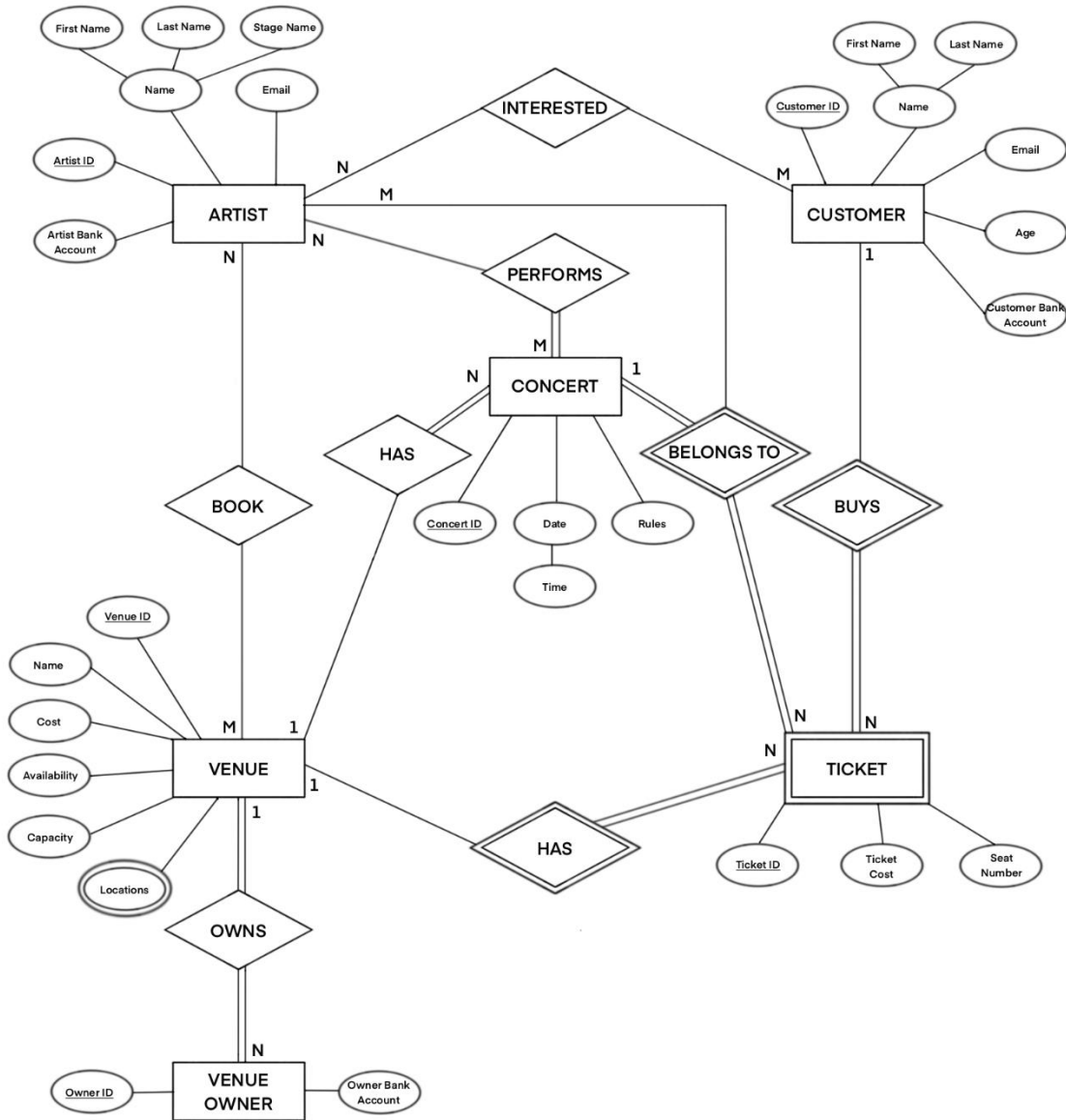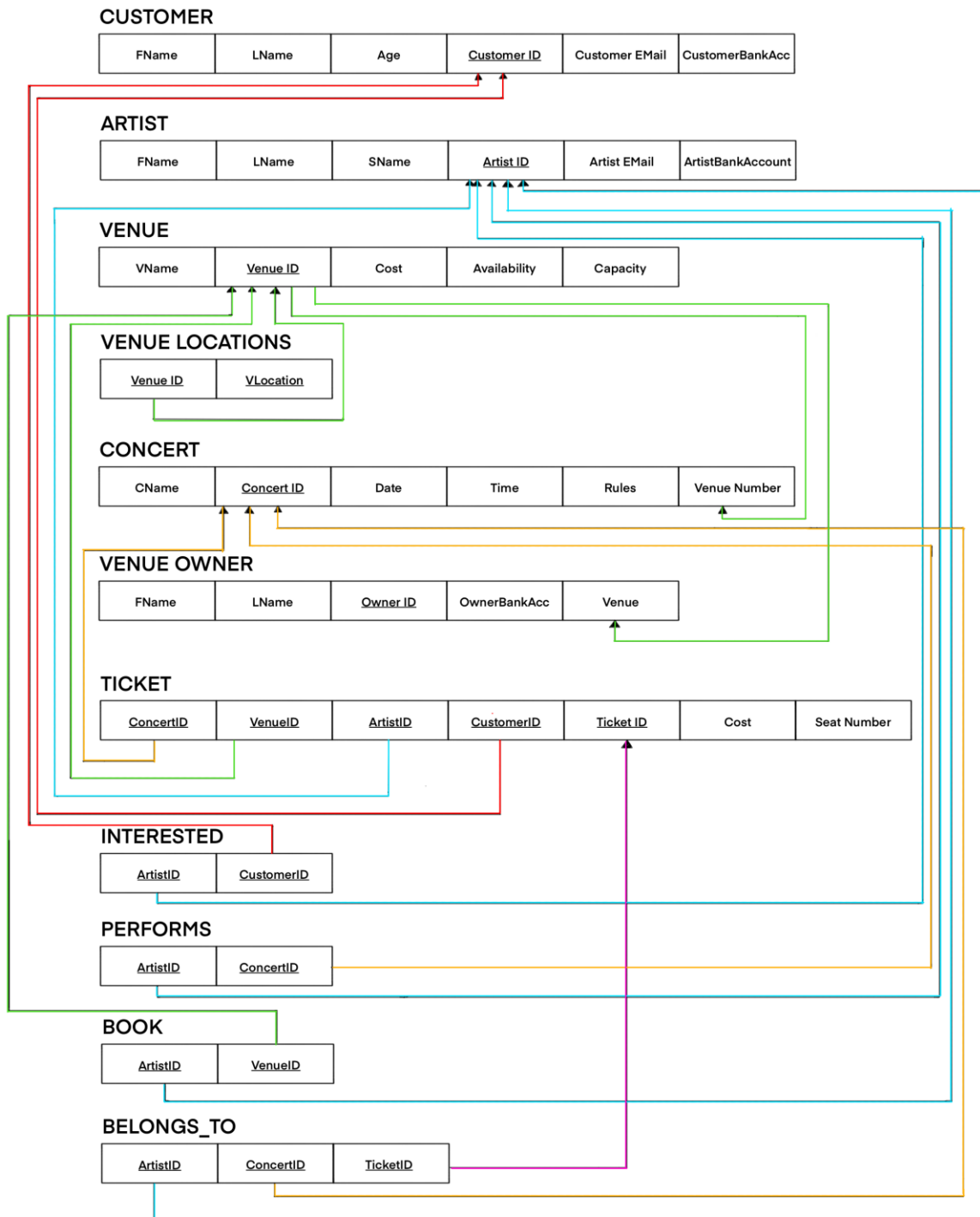ERD :

Schema :

CUSTOMER

FName	LName	Age	Customer ID	Customer EMail	CustomerBankAcc

ARTIST

FName	LName	SName	Artist ID	Artist EMail	ArtistBankAccount

VENUE

VName	Venue ID	Cost	Availability	Capacity

VENUE LOCATIONS

Venue ID	VLocation

CONCERT

CName	Concert ID	Date	Time	Rules	Venue Number

VENUE OWNER

FName	LName	Owner ID	OwnerBankAcc	Venue

TICKET

ConcertID	VenueID	ArtistID	CustomerID	Ticket ID	Cost	Seat Number

INTERESTED

ArtistID	CustomerID

PERFORMS

ArtistID	ConcertID

BOOK

ArtistID	VenueID

BELONGS_TO

ArtistID	ConcertID	TicketID

CUSTOMER

FName	LName	Age	Customer ID	Customer EMail	CustomerBankAcc

ARTIST

FName	LName	SName	Artist ID	Artist EMail	ArtistBankAccount

VENUE

VName	Venue ID	Cost	Availability	Capacity

VENUE LOCATIONS

Venue ID	VLocation

CONCERT

CName	Concert ID	Date	Time	Rules	Venue Number

VENUE OWNER

FName	LName	Owner ID	OwnerBankAcc	Venue

TICKET

ConcertID	VenueID	ArtistID	CustomerID	Ticket ID	Cost	Seat Number

INTERESTED

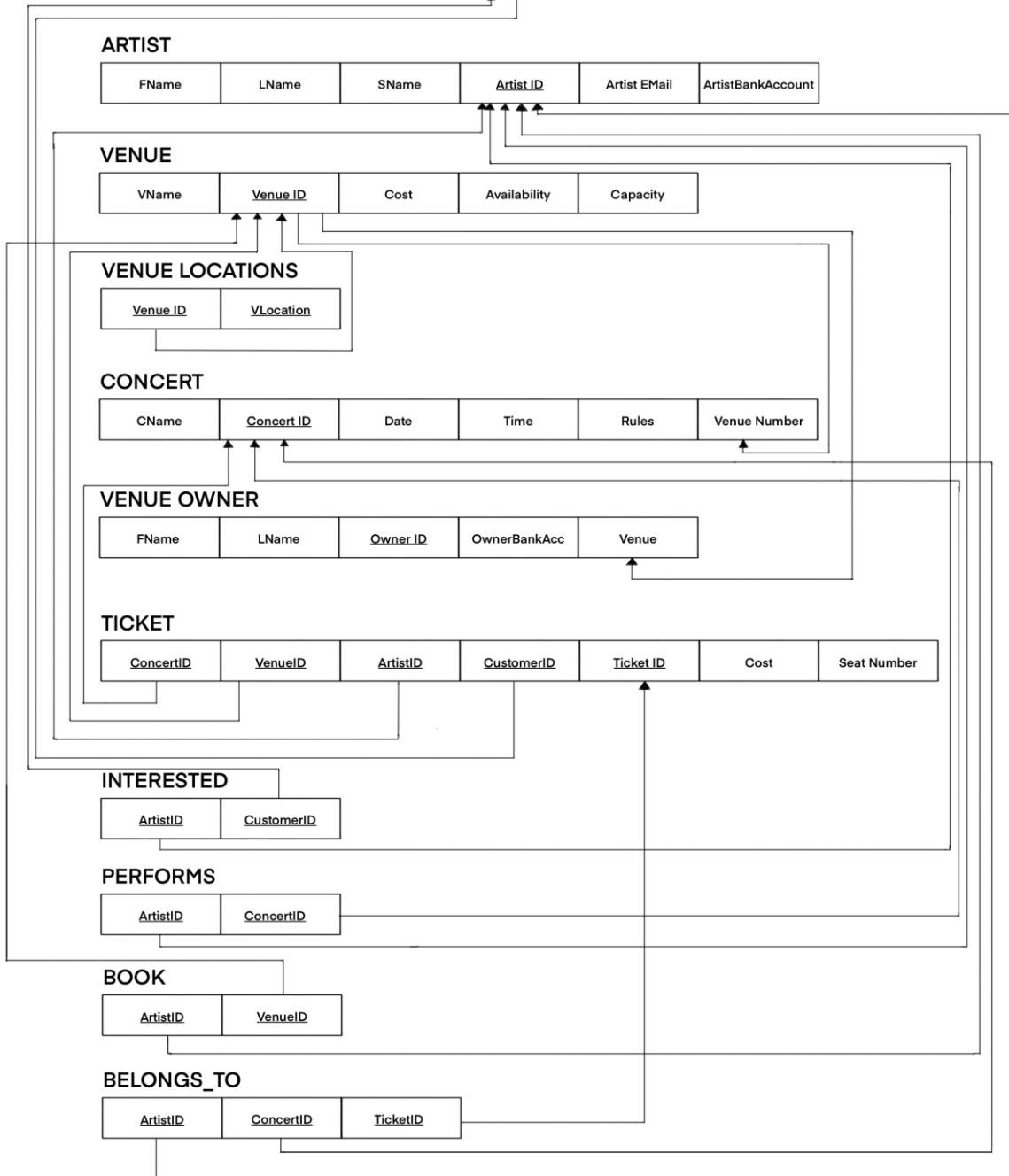| ArtistID	CustomerID

PERFORMS

ArtistID	ConcertID

BOOK

ArtistID	VenueID

BELONGS_TO

ArtistID	ConcertID	TicketID

SQL:

```sql
CREATE TABLE Customer (
    custID INT PRIMARY KEY,
    Fname VARCHAR(50),
    Lname VARCHAR(50),
    CustomerEmail VARCHAR(50),
    custAge INT,
    custBankAcc VARCHAR(50)
);

CREATE TABLE Artist (
    artistID INT PRIMARY KEY,
    Fname VARCHAR(50),
    Lname VARCHAR(50),
    Sname VARCHAR(50),
    artistEmail VARCHAR(50),
    artistBankAcc VARCHAR(50)
);

CREATE TABLE VenueOwner (
    OwnerID INT PRIMARY KEY,
    FName VARCHAR(50),
    LName VARCHAR(50),
    OwnerBankAcc VARCHAR(50)
);
```

```sql
CREATE TABLE Venue (

    venueID INT PRIMARY KEY,

    venueName VARCHAR(50),

    cost DECIMAL(10,2),

    availability BOOLEAN,

    capacity INT,

    ownerID INT,

    FOREIGN KEY (ownerID) REFERENCES VenueOwner(OwnerID)

);



CREATE TABLE VenueLocations (

    VenueID INT,

    VLocation VARCHAR(100),

    PRIMARY KEY (VenueID, VLocation),

    FOREIGN KEY (VenueID) REFERENCES Venue(venueID)

);


CREATE TABLE Concert (

    concertID INT PRIMARY KEY,

    concertName VARCHAR(100),

    date DATE,

    time TIME,

    rules VARCHAR(255),

    venueID INT,

    FOREIGN KEY (venueID) REFERENCES Venue(venueID)

);
```

```sql
CREATE TABLE Ticket (

    ticketID INT PRIMARY KEY,

    price DECIMAL(10,2),

    seatNumber INT,

    custID INT,

    concertID INT,

    venueID INT,

    artistID INT,

    FOREIGN KEY (custID) REFERENCES Customer(custID),

    FOREIGN KEY (concertID) REFERENCES Concert(concertID),

    FOREIGN KEY (venueID) REFERENCES Venue(venueID),

    FOREIGN KEY (artistID) REFERENCES Artist(artistID)

);


CREATE TABLE Interested (

    artistID INT,

    custID INT,

    PRIMARY KEY (artistID, custID),

    FOREIGN KEY (artistID) REFERENCES Artist(artistID),

    FOREIGN KEY (custID) REFERENCES Customer(custID)

);
```

```sql
CREATE TABLE Performs (

    artistID INT,

    concertID INT,

    PRIMARY KEY (artistID, concertID),

    FOREIGN KEY (artistID) REFERENCES Artist(artistID),

    FOREIGN KEY (concertID) REFERENCES Concert(concertID)

);


CREATE TABLE Book (

    artistID INT,

    venueID INT,

    concertID INT,

    PRIMARY KEY (artistID, venueID, concertID),

    FOREIGN KEY (artistID) REFERENCES Artist(artistID),

    FOREIGN KEY (venueID) REFERENCES Venue(venueID),

    FOREIGN KEY (concertID) REFERENCES Concert(concertID)

);

INSERT INTO Customer (custID, Fname, Lname, CustomerEmail, custAge, custBankAcc) VALUES

    (1, 'John', 'Doe', 'john.doe@example.com', 30, '1234567890'),

    (2, 'Jane', 'Smith', 'jane.smith@example.com', 25, '0987654321');

INSERT INTO Artist (artistID, Fname, Lname, Sname, artistEmail, artistBankAcc) VALUES

    (1, 'Alice', 'Johnson', 'AJ', 'alice.johnson@example.com', '1122334455'),

    (2, 'Bob', 'Brown', 'BB', 'bob.brown@example.com', '2233445566');
```

INSERT INTO VenueOwner (OwnerID, FName, LName, OwnerBankAcc) VALUES

    (1, 'Charlie', 'Davis', '3344556677'),

    (2, 'Diana', 'Evans', '4455667788');

INSERT INTO Venue (venueID, venueName, cost, availability, capacity, ownerID) VALUES

    (1, 'Stadium A', 5000.00, TRUE, 20000, 1),

    (2, 'Arena B', 3000.00, FALSE, 15000, 2);

INSERT INTO VenueLocations (VenueID, VLocation) VALUES

    (1, 'New York'),

    (2, 'Los Angeles');

INSERT INTO Concert (concertID, concertName, date, time, rules, venueID) VALUES

    (1, 'Rock Fest', '2023-12-01', '18:00:00', 'No outside food or drinks', 1),

    (2, 'Jazz Night', '2023-12-15', '20:00:00', 'Formal attire required', 2);

INSERT INTO Ticket (ticketID, price, seatNumber, custID, concertID, venueID, artistID) VALUES

    (1, 100.00, 1, 1, 1, 1, 1),

    (2, 150.00, 2, 2, 2, 2, 2);

INSERT INTO Interested (artistID, custID) VALUES

    (1, 1),

    (2, 2);

INSERT INTO Performs (artistID, concertID) VALUES

    (1, 1),

    (2, 2);

```
INSERT INTO Book (artistID, venueID, concertID) VALUES
    (1, 1, 1),
    (2, 2, 2);
```

GIThub link :

https://github.com/TechKing1/venyou