**Database Structure:**

/SQL Server

|── localdb

|── Databases

|── Client

|── Tables

|── dbo.CustomerTable

|── dbo.FlightScheduleTable

|── dbo.FlightSeatAvailabilityTable

|── dbo.FlightTable

|── dbo.GuestTable

|── dbo.SeatTypeTable

|── Programmability

|── System Stored Procedure

|── dbo.ADD\_CUSTOMER

|── dbo.ADD\_GUEST

|── dbo.GET\_AVAILABLE\_FLIGHT

|── dbo.GET\_CUSTOMER

|── dbo.GET\_FINAL\_BOOK\_DETAILS

|── dbo.GET\_FLIGHT\_BY\_ID

|── dbo.GET\_TOTAL\_BOOKING\_BY\_FLIGHTSEATAVAILABILITYID

**Tables:**

**dbo.CustomerTable**

CREATE TABLE [dbo].[CustomerTable] (

[Id] INT IDENTITY (1, 1) NOT NULL,

[Name] NVARCHAR (100) NOT NULL,

[City] NVARCHAR (50) NULL,

[Country] NVARCHAR (50) NULL,

[PostalCode] NVARCHAR (10) NULL,

[Phone] NVARCHAR (15) NULL,

[Email] NVARCHAR (255) NOT NULL,

[Password] NVARCHAR (255) NOT NULL,

PRIMARY KEY CLUSTERED ([Id] ASC),

UNIQUE NONCLUSTERED ([Email] ASC)

);

**dbo.FlightScheduleTable**

CREATE TABLE [dbo].[FlightScheduleTable] (

[FlightScheduleId] INT IDENTITY (1, 1) NOT NULL,

[FlightId] INT NOT NULL,

[FlightDate] DATETIME NOT NULL,

[TakeoffTime] DATETIME NOT NULL,

[LandingTime] DATETIME NOT NULL,

[Price] DECIMAL (10, 2) NOT NULL,

PRIMARY KEY CLUSTERED ([FlightScheduleId] ASC),

FOREIGN KEY ([FlightId]) REFERENCES [dbo].[FlightTable] ([FlightId]) ON DELETE CASCADE

);

**dbo.FlightSeatAvailabilityTable**

CREATE TABLE [dbo].[FlightSeatAvailabilityTable] (

[FlightSeatAvailabilityId] INT IDENTITY (1, 1) NOT NULL,

[FlightScheduleId] INT NOT NULL,

[SeatTypeId] INT NOT NULL,

[SeatAvailability] INT DEFAULT ((0)) NOT NULL,

PRIMARY KEY CLUSTERED ([FlightSeatAvailabilityId] ASC),

FOREIGN KEY ([FlightScheduleId]) REFERENCES [dbo].[FlightScheduleTable] ([FlightScheduleId]) ON DELETE CASCADE,

FOREIGN KEY ([SeatTypeId]) REFERENCES [dbo].[SeatTypeTable] ([SeatTypeId]) ON DELETE CASCADE

);

**dbo.FlightTable**

CREATE TABLE [dbo].[FlightTable] (

[FlightId] INT IDENTITY (1, 1) NOT NULL,

[FlightName] NVARCHAR (100) NOT NULL,

[Source] NVARCHAR (100) NOT NULL,

[Destination] NVARCHAR (100) NOT NULL,

[ImageUrl] NVARCHAR (500) NULL,

PRIMARY KEY CLUSTERED ([FlightId] ASC)

);

**dbo.GuestTable**

CREATE TABLE [dbo].[GuestTable] (

[GuestId] INT IDENTITY (1, 1) NOT NULL,

[FirstName] NVARCHAR (100) NOT NULL,

[LastName] NVARCHAR (100) NOT NULL,

[Gender] NVARCHAR (10) NOT NULL,

[Age] INT NOT NULL,

[Email] NVARCHAR (255) NOT NULL,

[Phone] NVARCHAR (15) NOT NULL,

[FlightSeatAvailabilityId] INT NOT NULL,

[Price] DECIMAL (10, 2) NOT NULL,

PRIMARY KEY CLUSTERED ([GuestId] ASC),

UNIQUE NONCLUSTERED ([Email] ASC),

FOREIGN KEY ([FlightSeatAvailabilityId]) REFERENCES [dbo].[FlightSeatAvailabilityTable] ([FlightSeatAvailabilityId]) ON DELETE CASCADE,

CHECK ([Gender]='Male' OR [Gender]='Female' OR [Gender]='Other')

);

**dbo.SeatTypeTable**

CREATE TABLE [dbo].[SeatTypeTable] (

[SeatTypeId] INT IDENTITY (1, 1) NOT NULL,

[SeatType] NVARCHAR (50) NOT NULL,

PRIMARY KEY CLUSTERED ([SeatTypeId] ASC),

UNIQUE NONCLUSTERED ([SeatType] ASC)

);

**Stored Procedures:**

**dbo.ADD\_CUSTOMER**

CREATE PROCEDURE ADD\_CUSTOMER

@Name NVARCHAR(100),

@City NVARCHAR(50),

@Country NVARCHAR(50),

@PostalCode NVARCHAR(10),

@Phone NVARCHAR(15),

@Email NVARCHAR(255),

@Password NVARCHAR(255)

AS

BEGIN

INSERT INTO CustomerTable (Name, City, Country, PostalCode, Phone, Email, Password)

VALUES (@Name, @City, @Country, @PostalCode, @Phone, @Email, @Password);

END

**dbo.ADD\_GUEST**

CREATE PROCEDURE ADD\_GUEST

@FirstName NVARCHAR(100),

@LastName NVARCHAR(100),

@Gender NVARCHAR(10),

@Age INT,

@Email NVARCHAR(255),

@Phone NVARCHAR(15),

@FlightSeatAvailabilityId INT,

@Price DECIMAL(10,2)

AS

BEGIN

SET NOCOUNT ON;

DECLARE @FlightScheduleId INT, @ActualPrice DECIMAL(10,2), @RowsAffected INT;

-- Get FlightScheduleId from FlightSeatAvailabilityTable

SELECT @FlightScheduleId = FlightScheduleId

FROM FlightSeatAvailabilityTable

WHERE FlightSeatAvailabilityId = @FlightSeatAvailabilityId;

-- Get Price from FlightScheduleTable using FlightScheduleId

SELECT @Price = Price

FROM FlightScheduleTable

WHERE FlightScheduleId = @FlightScheduleId;

-- Check if seat is available

IF (SELECT SeatAvailability FROM FlightSeatAvailabilityTable WHERE FlightSeatAvailabilityId = @FlightSeatAvailabilityId) > 0

BEGIN

-- Insert guest along with price

INSERT INTO GuestTable (FirstName, LastName, Gender, Age, Email, Phone, FlightSeatAvailabilityId, Price)

VALUES (@FirstName, @LastName, @Gender, @Age, @Email, @Phone, @FlightSeatAvailabilityId, @Price);

-- Reduce seat availability by 1

UPDATE FlightSeatAvailabilityTable

SET SeatAvailability = SeatAvailability - 1

WHERE FlightSeatAvailabilityId = @FlightSeatAvailabilityId;

SELECT @@ROWCOUNT AS RowsAffected;

END

ELSE

BEGIN

-- Return failure if no seats are available

RAISERROR ('No available seats for the selected FlightSeatAvailabilityId.', 16, 1);

RETURN;

END

END;

**dbo.GET\_AVAILABLE\_FLIGHT**

CREATE PROCEDURE GET\_AVAILABLE\_FLIGHT

@Source NVARCHAR(100),

@Destination NVARCHAR(100),

@FlightDate DATE

AS

BEGIN

SET NOCOUNT ON;

SELECT

f.FlightId,

f.FlightName,

f.Source,

f.Destination,

f.ImageUrl,

fs.FlightScheduleId,

--CONVERT(VARCHAR, fs.FlightDate, 101) AS FlightDate, -- MM/dd/yyyy format

FORMAT(fs.FlightDate, 'yyyy-MM-ddTHH:mm:ss') AS FlightDate,

--fs.TakeoffTime,

--fs.LandingTime,

FORMAT(fs.TakeoffTime, 'hh:mm:ss') AS TakeoffTime, -- Convert TimeSpan to string

FORMAT(fs.LandingTime, 'hh:mm:ss') AS LandingTime, -- Convert TimeSpan to string

fs.Price,

st.SeatType,

fsa.FlightSeatAvailabilityId,

fsa.SeatAvailability

FROM FlightTable f

INNER JOIN FlightScheduleTable fs ON f.FlightId = fs.FlightId

INNER JOIN FlightSeatAvailabilityTable fsa ON fs.FlightScheduleId = fsa.FlightScheduleId

INNER JOIN SeatTypeTable st ON fsa.SeatTypeId = st.SeatTypeId

WHERE f.Source = @Source

AND f.Destination = @Destination

AND fs.FlightDate = @FlightDate

AND fsa.SeatAvailability > 0 -- Only show flights with available seats

ORDER BY fs.TakeoffTime, st.SeatType;

END;

**dbo.GET\_CUSTOMER**

CREATE PROCEDURE GET\_CUSTOMER

@Email NVARCHAR(255),

@Password NVARCHAR(255)

AS

BEGIN

SELECT Id, Name, Email, Phone, City, Country, PostalCode

FROM CustomerTable

WHERE Email = @Email AND Password = @Password;

END;

**dbo.GET\_FINAL\_BOOK\_DETAILS**

CREATE PROCEDURE GET\_FINAL\_BOOK\_DETAILS

@GuestId INT

AS

BEGIN

SET NOCOUNT ON;

SELECT

g.GuestId,

g.FirstName,

g.LastName,

g.Gender,

g.Age,

g.Email,

g.Phone,

fsa.FlightSeatAvailabilityId,

fs.FlightScheduleId,

f.FlightId,

f.Source,

f.Destination,

f.FlightName,

fs.FlightDate,

fs.TakeoffTime,

fs.LandingTime,

fs.Price,

st.SeatTypeId,

st.SeatType

FROM GuestTable g

JOIN FlightSeatAvailabilityTable fsa ON g.FlightSeatAvailabilityId = fsa.FlightSeatAvailabilityId

JOIN FlightScheduleTable fs ON fsa.FlightScheduleId = fs.FlightScheduleId

JOIN FlightTable f ON fs.FlightId = f.FlightId

JOIN SeatTypeTable st ON fsa.SeatTypeId = st.SeatTypeId

WHERE g.GuestId = @GuestId;

END;

**dbo.GET\_FLIGHT\_BY\_ID**

CREATE PROCEDURE GET\_FLIGHT\_BY\_ID

@FlightId INT,

@FlightScheduleId INT,

@SeatType NVARCHAR (50)

AS

BEGIN

SET NOCOUNT ON;

SELECT

f.FlightId,

f.FlightName,

f.Source,

f.Destination,

f.ImageUrl,

fs.FlightScheduleId,

fs.FlightDate,

FORMAT(fs.TakeoffTime, 'hh:mm:ss') AS TakeoffTime,

FORMAT(fs.LandingTime, 'hh:mm:ss') AS LandingTime,

fs.Price,

st.SeatType,

fsa.FlightSeatAvailabilityId,

fsa.SeatAvailability

FROM FlightTable f

INNER JOIN FlightScheduleTable fs ON f.FlightId = fs.FlightId

INNER JOIN FlightSeatAvailabilityTable fsa ON fs.FlightScheduleId = fsa.FlightScheduleId

INNER JOIN SeatTypeTable st ON fsa.SeatTypeId = st.SeatTypeId

WHERE f.FlightId = @FlightId

AND fs.FlightScheduleId = @FlightScheduleId

AND st.SeatType = @SeatType;

END;

**dbo.GET\_TOTAL\_BOOKING\_BY\_FLIGHTSEATAVAILABILITYID**

CREATE PROCEDURE GET\_TOTAL\_BOOKING\_BY\_FLIGHTSEATAVAILABILITYID

@FlightSeatAvailabilityId INT

AS

BEGIN

SET NOCOUNT ON;

SELECT

g.FirstName,

g.LastName,

g.Gender,

g.Age,

g.Email,

g.Phone,

fsa.FlightSeatAvailabilityId,

fs.FlightScheduleId,

f.FlightId,

f.FlightName,

fs.Price,

st.SeatTypeId,

st.SeatType

FROM GuestTable g

JOIN FlightSeatAvailabilityTable fsa ON g.FlightSeatAvailabilityId = fsa.FlightSeatAvailabilityId

JOIN FlightScheduleTable fs ON fsa.FlightScheduleId = fs.FlightScheduleId

JOIN FlightTable f ON fs.FlightId = f.FlightId

JOIN SeatTypeTable st ON fsa.SeatTypeId = st.SeatTypeId

WHERE fsa.FlightSeatAvailabilityId = @FlightSeatAvailabilityId;

END;