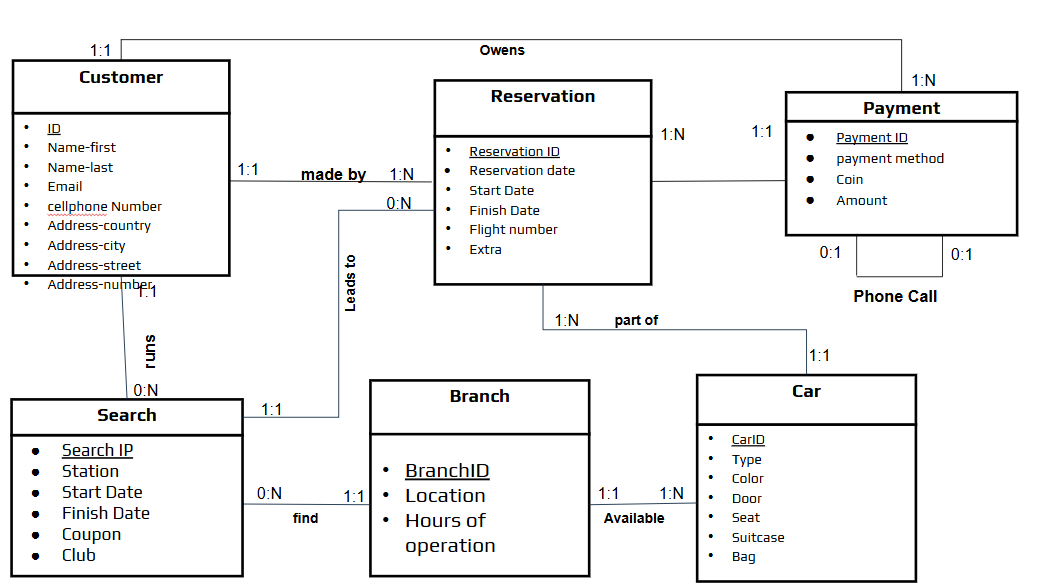
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| מס' קבוצה | אתר | | | תאריך הגשה |
| 20 | [השכרת רכב בחו"ל](https://www.avis.co.il/%D7%94%D7%A9%D7%9B%D7%A8%D7%AA-%D7%A8%D7%9B%D7%91-%D7%91%D7%97%D7%95%D7%9C/) | | | 26/12/24 |
| חברי הצוות – מספרי ת.ז | | | | |
| 322816885 | | 207476151 | 206364911 | |

## פרויקט בסיסי נתונים

## מטלה מקדימה – ה-ERD שהוגש בחלק א'

**התרשים בהגשה הקודמת:**



**ההנחות שנרשמו בהגשה הקודמת:**

* נניח כי ע"מ להיחשב כלקוח על המשתמש להזמין רכב.
* נניח כי בסניף יש לפחות מכונית אחת וכל מכונית שייכת לסניף אחד מסוים.

**ההערות שניתנו להגשה הקודמת**

1. הקשר בין Search ל- Reservation:  
קרדינליות לא הגיונית. האם חיפוש בודד יכול להוביל למס` הזמנות? יותר הגיוני שזה יהיה 0:1 - או שמצאתי בחיפוש אפשרות שמתאימה לי ואזמין אותה, או שלא.

2. קשר רקורסיבי בישות Payment:  
כדאי להוסיף הנחה שמסבירה מה הכוונה בPhone call- אם התכוונתם שזו האפשרות לבצע תשלום טלפוני - אפשר להוסיף פשוט שדה שמתאר את אופן ביצוע התשלום.

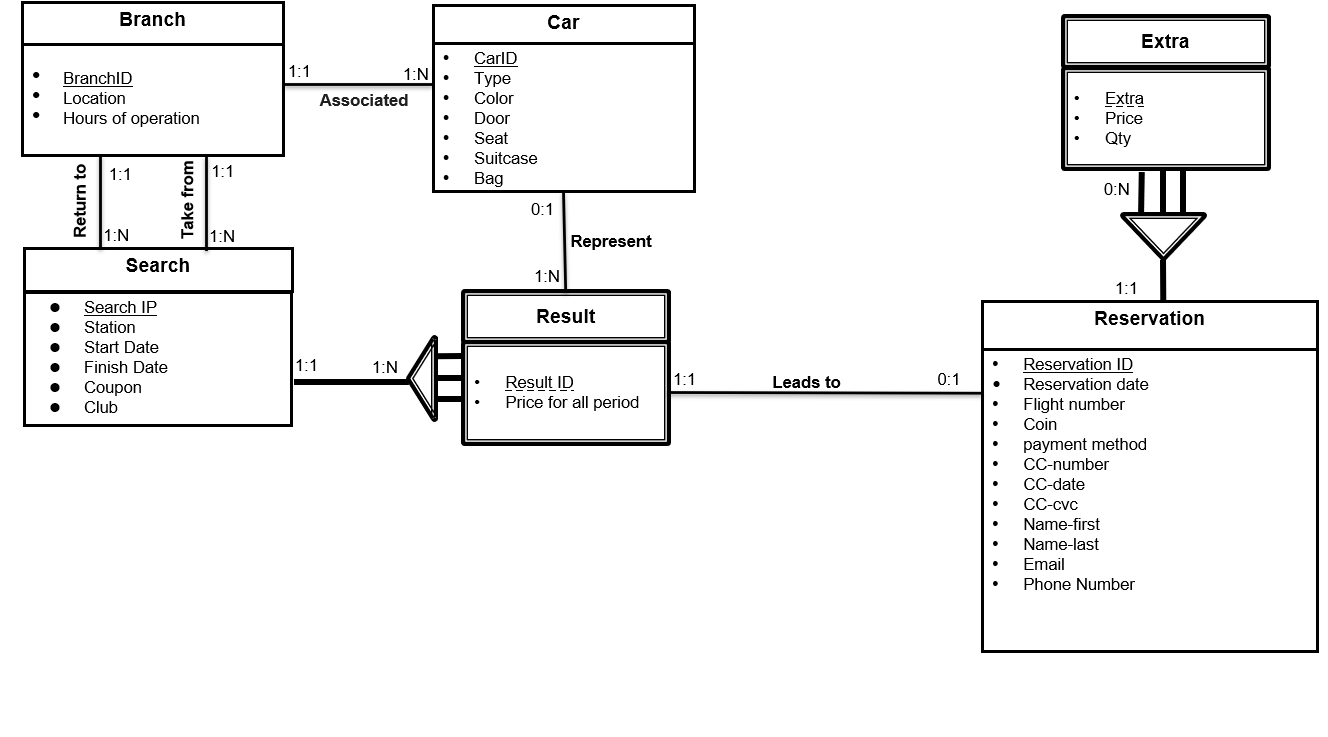
3. הישות Search:  
כפי שכתבתם למעלה, כשמבצעים חיפוש מזינים את המיקום שממנו רוצים לאסוף ואת המיקום שאליו רוצים להחזיר - המודל שלכם מתעלם מזה.

4. הקשר find בין Search ל-Branch:  
לא ברור מה הכוונה של הקשר FIND כשמריצים חיפוש, התוצאות שלו יהיו אוסף של רכבים שזמינים בתאריכים שרצינו, ולא אוסף של סניפים. שימו לב גם שיכול להיות שלא תתקבל אף תוצאה, למשל אם כל הרכבים תפוסים. הקרדינליות שלכם לא מאפשרת זאת.

5. הישות Search:  
איך אפשר לדעת איזה רכבים עלו בחיפוש? חסר כאן קשר חשוב מאוד. תחשבו גם מבחינה עסקית- נרצה לדעת אם יש חיפושים שאנשים מבצעים ולא מניבים אף תוצאה (כדי להעביר עוד מכוניות לסניף שממנו רצו, לדוגמה) או איזה תוצאות הם קיבלו כדי לדעת מה יותר מבוקש.

6. הישות Search וקשריה:  
זאת הנחה מאוד בעייתית שסותרת את ההיגיון העסקי. אם אני מנכ"ל AVIS, יעניין אותי מה החיפושים שאנשים מבצעים, ולא רק ההזמנה הסופית. לדוגמה, אם המון אנשים מנסים להזמין רכב באיטליה והחיפוש מעלה להם אופציות גרועות, הם ילכו למתחרים ולא יזמינו. אם לא נשמור את החיפושים שלהם, לא נוכל לדעת שיש לנו פה פער בהיצע.

## מטלה 1 (40%) – עיצוב קונספטואלי (תרשים ישויות-קשרים)



## הנחות מודל ה- ERD

|  |  |
| --- | --- |
| **מרכיב** | **הנחות** |
| Entity: Customer   * ID * Name-first * Name-last * Email * Phone Number * Address-country * Address-city * Address-street * Address-number | * לקוח נחשב רק מי שביצע הזמנה, מכיוון שפרטי המזמין נשמרים רק בביצוע ההזמנה, אין באתר אופציה להתחבר למשתמש קיים. |
| Entity: Extra   * Extra * Price * Qty | * בעת ההזמנה הלקוח יכול לבחור אם להוסיף תוספות לרכב בהתאם לסניף ולרכב שבחר, לכל תוספת מחיר שונה ויש אפשרות להזמין כמות מתוספת מסוימת (למשל 2 כיסאות ילדים). |
| הקשר בין "Branch" ל-"Car" | * בכל סניף יש לפחות מכונית אחת (סניף שאין בו רכבים פשוט לא יוצג בחיפוש של המשתמש באתר) וכל מכונית שייכת לסניף אחד מסוים הדואג לתחזוקה הכללית שלו.  לפיכך הקרדינאליות בין "Branch" ל-"Car" הוגדרה כ- 1:1 ו-N:1. |

## הנחות קודומות:

## הנחות מודל ה- ERD המעודכן:

|  |  |
| --- | --- |
| **מרכיב** | **הנחות** |
| Customer ביטול הישות | * האתר אינו שומר את פרטי הלקוח, ולכן הוחלט להכניס את השדות הקשורות ללקוח לישות Reservation |
| paymentביטול הישות | * האתר אינו שומר את פרטי התשלום של הלקוח, ולכן הוחלט להכניס את השדות הקשורות לתשלום לישות Reservation |
| Entity: Extra   * Extra * Price * Qty | * בעת ההזמנה הלקוח יכול לבחור אם להוסיף תוספות לרכב בהתאם לסניף ולרכב שבחר, לכל תוספת מחיר שונה ויש אפשרות להזמין כמות מתוספת מסוימת (למשל 2 כיסאות ילדים). |
| הקשר בין "Branch" ל-"Car" | * בכל סניף יש לפחות מכונית אחת (סניף שאין בו רכבים פשוט לא יוצג בחיפוש של המשתמש באתר) וכל מכונית שייכת לסניף אחד מסוים הדואג לתחזוקה הכללית שלו.  לפיכך הקרדינאליות בין "Branch" ל-"Car" הוגדרה כ- 1:1 ו-N:1. |
| הישות Result | * הישות מייצגת תוצאת חיפוש , ולכן אין תוצאה בלי חיפוש ולכן הגדרנו את הישות result כישות חלשה של search |
| הקשר בין הישות Result לישות Car | * הישות Result מייצגת תוצאה ספציפית ולכן היא יכולה להציג אפס מכוניות או מכונית אחת |

## מטלה 2 (30%) – עיצוב לוגי (מודל נתונים טבלאי)

* **BRANCHES** (BranchID, Location, HoursOfOperation)
* **CARS** (CarID, Car Type, Gear,Color, Door, Seat, Suitcase, Bag, BranchID(BRANCHES))
* **SEARCHES** (Search IP, Station, Start Date, Finish Date, Club, BranchT (BRANCHES), BranchR(BRANCHES))
* **RESULTS**(SearchID(SEARCHES), ResultID, Price for all period, CarID(CARS)))
* **CARDS**(CC-number, CC-date, CC-cvc)
* **RESERVATIONS (**ReservationID, Reservation date, Flight number, Coin, Payment-method, Name-first, Name-last, Email, Phone Number,CC-number(CARDS), CarID (CARS))  
   {ERSSearchID, ResultID}(RESULTS))
* **EXTRAS** (Reservation ID(RESERVATIONS), Extra, Price, Qty)

**הנחות והסברים נוספים:**

* **ביטוי קשר בין Result ל-Reservation:**

קשר יחיד ליחיד. על פי חוקי המעבר למודל הטבלאי יכולנו לבחור האם לבטא את מזהה הקשר כ-Result או כ-Reservation. בחירה ב-result מוסיפה לנו 2 שדות נוספים (מכיוון שהמזהה של result מורכב מ-2 שדות) נבהיר כי מבחינה לוגית לא כל תוצאת חיפוש מובילה להזמנה, ולכן אם נגדיר את ההזמנה להיות מפתח זר של התוצאה יהיו המון תוצאות שלא יהיה להם מזהה ונקבל ערכי NULL במפתח הזר ובכדי להימנע מכך נגדיר את התוצאה להיות מפתח זר של ההזמנה .

* **השדה Payment method בישות Reservation:**

**האתר מאפשר ללקוח לבצע תשלום בכרטיס אשראי באתר או בכרטיס אשראי בשיחה טלפונית.**

* **לקוח הוא מי שביצע הזמנה, כלומר נשמר במערכת של אוויס רק לאחר שמבצע הזמנה, אין מזהה מיוחד שמתבקש מהלקוח בשלב ביצוע ההזמנה- לפיכך החלטנו להוריד את ישות הלקוח .**
* **הטבלה CARD נובעת כתוצאה מנרמול מסדר שלישי של השדה CC-number , היות והשדה מגדיר בצורה ישירה את השדות CC-cvc, CC-date**

# פירוט שדות בסיס הנתונים והערכת גודלם:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table | Field | | Data Type | Size (Bytes) |
| **BRANCHES** | Branch ID | | Int | 5 |
| BLocation | | Varchar(40) | 40 |
| Hours of operation | | Varchar(11) | 11 |
| Total Record Size (Bytes) | | | 56 |
| **CARS** | CarID | | Varchar(11) | 11 |
| Type | | Varchar(30) | 30 |
| Gear | | Varchar(10) | 10 |
| Color | | Varchar(15) | 15 |
| Door | | Int | 1 |
| Seat | | Int | 2 |
| Suitcase | | Int | 2 |
| Bag | | Int | 2 |
| Branch ID (BRANCHES) | | Int | 5 |
| Total Record Size (Bytes) | | | 78 |
| **SEARCHES** | Search IP | | Varchar(20) | 20 |
| Station | | Varchar(100) | 100 |
| Start Date | | Date | 10 |
| Finish Date | | Date | 10 |
| Club | | Int | 10 |
| BranchT(BRANCHES) | | Int | 5 |
|  | | BranchR(BRANCHES) | Int | 5 |
| Total Record Size (Bytes) | | 160 |
| **RESULTS** | | SearchID | Varchar(20) | 20 |
| ResultID | Int | 50 |
| Price for all period | Decimal(10,2) | 9 |
| CarID(CARS) | Varchar(11) | 11 |
| Total Record Size (Bytes) | | 90 |
| **CARDS** | | CC-Number | Varchar(19) | 19 |
| CC-Cdate | Date | 10 |
| CC-cvc | Char(3) | 3 |
| Total Record Size (Bytes) | | 32 |
| **RESERVATIONS** | | ReservationID | Int | 20 |
| Reservation date | Date | 10 |
|  | Flight number | | Varchar(50) | 50 |
| Coin | | Varchar(10) | 10 |
| Payment-method | | Varchar(15) | 15 |
| Name-first | | Varchar(20) | 20 |
| Name-last | | Varchar(20) | 20 |
| Email | | Varchar(40) | 40 |
| Phone Number | | Varchar(20) | 20 |
| CC-number(CARDS) | | Varchar(16) | 16 |
| CarID (CARS) | | Varchar(11) | 11 |
| SearchID | | Varchar(20) | 20 |
| ResultID | | Int | 50 |
| Total Record Size (Bytes) | | | 300 |
| **EXTRAS** | Reservation ID (RESERVATIONS) | | Int | 20 |
| Extra | | Varchar(30) | 30 |
| Price | | Decimal(10,2) | 9 |
| Qty | | Int | 2 |
| Total Record Size (Bytes) | | | 61 |

# 

**הנחות והסברים נוספים:**

* שדות התאריכים בטבלה SEARCHES נבחרו כתאריך ולא כתאריך/זמן, כיוון שהאתר אינו מאפשר חיפוש על פי שעות ספציפיות.
* טלפון מסוג נתונים של VARCHAR(20) מכיוון שבסיס הנתונים הוא עבור חברת השכרת רכבים בינלאומית ולכל מדינה קידומות שונה הכולל תווים שאינם מספרים. אנו לא נדרשים לבצע פעולות חישוב על שדה זה, באופן דומה גם עבור מספר כרטיסי האשראי ולוחיות רישוי של רכב.

## מטלה 3 (30%) - עיצוב פיסי (בסיס נתונים פעיל על שרת RDBMS)

**אילוצי מרחב ערכים**

**אילוצי בדיקה  (CHECK):**

* **אילוץ על EMAIL**: הוספנו אילוץ על השדה Email לטבלת RESERVATIONS בה הוא  מופיע. זהו שדה חשוב מכיוון שהלקוח צריך לקבל את כל פרטי ההזמנה לכתובת המייל שהזין.
* **אילוץ על Priceforallperiod**: הוספנו אילוץ על השדה Priceforallperiod לטבלת RESULTS בה הוא מופיע.  הוספנו אילוץ זה מכיוון ששדה זה הוא מחיר לתקופה מסוימת ומחיר בהכרח גדול מ-0.
* **אילוץ על Price:** הוספנו אילוץ על Price לטבלת EXTRAS בה הוא מופיע. הוספנו אילוץ זה מכיוון השדה מגדיר את מחיר התוספות להזמנה ומחיר בהכרח צריך להיות גדול מ-0.
* **אילוץ על :StartDate FinishDate** הוספנו אילוץ על StartDate , FinishDate לטבלת SEARCHES בה שדות אלו מופיעים. הוספנו אילוץ זה מכיוון שהתאריך סיום בהכרח צריך להיות אחרי התאריך התחלה.

**טבלאות חיפוש (Lookup Tables):**

**להוסיף בכתב מה כל טבלה עושה**

* **טבלת חיפוש** **ל-Paymentmethod:** הוספנו טבלת חיפוש עבור אופן התשלום מתוך הנחה כי לקוח יכול לשלם רק בכרטיסי אשראי באינטרנט או בשיחה טלפונית.
* **טבלת חיפוש ל-Gears types**: הוספנו טבלת חיפוש לסוגי ידיות ההילוכים מתוך הנחה שיש תיבת הילוכים אוטומטית-automatic או ידנית- manual.
* **טבלת חיפוש ל-Cars types:** זיהינו משמעות עסקית בנתון על אודות סוגי הרכבים המוחזקים באוויס.
* **טבלת חיפוש ל-Coin types**: מתוך הנחה שאפשר לשלם על הזמנה במטבע אותו הלקוח בוחר.

## מטלה 3 (30%) - עיצוב פיסי (בסיס נתונים פעיל על שרת RDBMS)

DROP TABLE IF EXISTS EXTRAS

DROP TABLE IF EXISTS RESERVATIONS

DROP TABLE IF EXISTS RESULTS

DROP TABLE IF EXISTS SEARCHES

DROP TABLE IF EXISTS CARS

DROP TABLE IF EXISTS BRANCHES

DROP TABLE IF EXISTS Paymentmethods

DROP TABLE IF EXISTS COINTYPES

DROP TABLE IF EXISTS GEARTYPES

DROP TABLE IF EXISTS CARTYPES

CREATE TABLE BRANCHES (

BranchID INT NOT NULL PRIMARY KEY,

BLocation VARCHAR(40) NULL,

HoursOfOperation VARCHAR(11) NULL

)

CREATE TABLE CARS (

CarID VARCHAR(11) NOT NULL PRIMARY KEY,

CarType VARCHAR(30) NULL,

Gear VARCHAR(10) NOT NULL,

Color VARCHAR(15) NOT NULL,

Door INT NOT NULL,

Seat INT NOT NULL,

Suitcase INT NOT NULL,

Bag INT NOT NULL,

BranchID INT NOT NULL,

)

ALTER TABLE CARS

ADD CONSTRAINT FK\_BranchID FOREIGN KEY (BranchID)

REFERENCES BRANCHES(BranchID)

DROP TABLE IF EXISTS SEARCHES

CREATE TABLE SEARCHES (

SearchIP VARCHAR(20) NOT NULL PRIMARY KEY,

Station VARCHAR(100) NULL,

StartDate DATE NOT NULL,

FinishDate DATE NOT NULL,

Club INT NULL,

BranchT INT NOT NULL,

BranchR INT NOT NULL,

)

ALTER TABLE SEARCHES

ADD CONSTRAINT FK\_BranchT FOREIGN KEY (BranchT)

REFERENCES BRANCHES(BranchID)

ALTER TABLE SEARCHES

ADD CONSTRAINT FK\_BranchR FOREIGN KEY (BranchR)

REFERENCES BRANCHES(BranchID)

CREATE TABLE RESULTS (

SearchID VARCHAR(20) NOT NULL,

ResultID INT NOT NULL,

PriceForAllPeriod DECIMAL(10, 2) NOT NULL,

CarID VARCHAR(11) NULL,

PRIMARY KEY (SearchID, ResultID)

)

ALTER TABLE RESULTS

ADD CONSTRAINT FK\_SearchID FOREIGN KEY (SearchID)

REFERENCES SEARCHES(SearchIP)

ALTER TABLE RESULTS

ADD CONSTRAINT FK\_CarID FOREIGN KEY (CarID)

REFERENCES CARS(CarID)

CREATE TABLE CARDS (

CC\_Number VARCHAR(19) NOT NULL PRIMARY KEY,

CC\_Date DATE NOT NULL,

CC\_Cvc CHAR(3) NOT NULL

)

CREATE TABLE RESERVATIONS (

ReservationID INT NOT NULL PRIMARY KEY,

ReservationDate DATE NOT NULL,

FlightNumber VARCHAR(50) NOT NULL,

Coin VARCHAR(10) NOT NULL,

PaymentMethod VARCHAR(15) NOT NULL,

NameFirst VARCHAR(20) NOT NULL,

NameLast VARCHAR(20) NOT NULL,

Email VARCHAR(40) NOT NULL,

PhoneNumber VARCHAR(20) NOT NULL,

CC\_Number VARCHAR(19) NULL,

CarID VARCHAR(11) NULL,

SearchID VARCHAR(20) NOT NULL,

ResultID INT NOT NULL

)

ALTER TABLE RESERVATIONS

ADD CONSTRAINT FK\_CarID\_Reservation FOREIGN KEY (CarID)

REFERENCES CARS(CarID)

ALTER TABLE RESERVATIONS

ADD CONSTRAINT FK\_CARD FOREIGN KEY (CC\_Number)

REFERENCES CARDS(CC\_Number)

ALTER TABLE RESERVATIONS

ADD CONSTRAINT FK\_SearchResult FOREIGN KEY (SearchID, ResultID)

REFERENCES RESULTS(SearchID, ResultID)

CREATE TABLE EXTRAS (

ReservationID INT NOT NULL,

Extra VARCHAR(30) NOT NULL,

Price DECIMAL(10, 2) NOT NULL,

Qty INT NULL,

PRIMARY KEY (ReservationID, Extra)

)

ALTER TABLE EXTRAS

ADD CONSTRAINT FK\_ReservationID FOREIGN KEY (ReservationID)

REFERENCES RESERVATIONS(ReservationID)

ALTER TABLE RESERVATIONS

ADD CONSTRAINT Ck\_Email

CHECK (Email like '%@%.%')

ALTER TABLE RESULTS

ADD CONSTRAINT Ck\_PriceForAllPeriod

CHECK (PriceForAllPeriod >0)

ALTER TABLE EXTRAS

ADD CONSTRAINT Ck\_Price

CHECK (Price >0)

ALTER TABLE SEARCHES

ADD CONSTRAINT CK\_StartDate\_before\_FinishDate

CHECK (StartDate<FinishDate)

INSERT INTO BRANCHES (BranchID, BLocation, HoursOfOperation) VALUES

(1, 'Tel Aviv', '08:00-20:00'),

(2, 'Jerusalem', '09:00-18:00'),

(3, 'Paris', '08:30-19:30'),

(4, 'Thessaloniki', '07:00-23:00'),

(5, 'Beer Sheva', '09:00-17:00')

INSERT INTO CARS (CarID, CarType, Gear, Color, Door, Seat, Suitcase, Bag, BranchID) VALUES

('A123', 'Sedan','Automatic', 'Red', 4, 5, 2, 3, 1),

('B456', 'SUV','Automatic', 'Blue', 4, 7, 3, 4, 2),

('C789', 'Hatchback','Manual', 'White', 4, 4, 1, 2, 3),

('D101', 'Convertible','Manual', 'Black', 2, 2, 1, 1, 4),

('E202', 'Minivan','Manual', 'Silver', 4, 8, 4, 5, 5)

INSERT INTO SEARCHES (SearchIP, Station, StartDate, FinishDate, Club, BranchT, BranchR) VALUES

('192.168.1.1', 'Tel Aviv', '2024-01-01', '2024-01-10', 1, 1, 2),

('192.168.1.2', 'Jerusalem', '2024-02-01', '2024-02-05', 0, 2, 3),

('192.168.1.3', 'Haifa', '2024-03-01', '2024-03-15', 1, 3, 4),

('192.168.1.4', 'Eilat', '2024-04-01', '2024-04-07', 0, 4, 5),

('192.168.1.5', 'Beer Sheva', '2024-05-01', '2024-05-10', 1, 5, 1)

INSERT INTO RESULTS (SearchID, ResultID, PriceForAllPeriod, CarID) VALUES

('192.168.1.1', 1, 500.00, 'A123'),

('192.168.1.2', 2, 750.00, 'B456'),

('192.168.1.3', 3, 600.00, 'C789'),

('192.168.1.4', 4, 850.00, 'D101'),

('192.168.1.5', 5, 1000.00, 'E202')

INSERT INTO CARDS (CC\_Number, CC\_Date, CC\_Cvc) VALUES

(1234567890123456, '2024-12-31', 123),

(2345678901234567, '2025-11-30', 234),

(3456789012345678, '2026-10-31', 345),

(4567890123456789, '2027-09-30', 456),

(5678901234567890, '2028-08-31', 567)

INSERT INTO RESERVATIONS (ReservationID, ReservationDate, FlightNumber, Coin, PaymentMethod, NameFirst, NameLast, Email, PhoneNumber, CC\_Number, CarID, SearchID, ResultID) VALUES

(1, '2024-01-01', 'LY101', 'USD', 'CC-Online', 'Yossi', 'Cohen', 'yossi.cohen@example.com', 1234567890, 1234567890123456, 'A123', '192.168.1.1', 1),

(2, '2024-02-01', 'LY102', 'EUR', 'CC-Phone Call', 'Miri', 'Levi', 'miri.levi@example.com', 2345678901, 2345678901234567, 'B456', '192.168.1.2', 2),

(3, '2024-03-01', 'LY103', 'ILS', 'CC-Online', 'Ronen', 'Avraham', 'ronen.avraham@example.com', 3456789012, 3456789012345678, 'C789', '192.168.1.3', 3),

(4, '2024-04-01', 'LY104', 'GBP', 'CC-Phone Call', 'Dana', 'Mizrahi', 'dana.mizrahi@example.com', 4567890123, 4567890123456789, 'D101', '192.168.1.4', 4),

(5, '2024-05-01', 'LY105', 'USD', 'CC-Online', 'Avi', 'Shalom', 'avi.shalom@example.com', 5678901234, 5678901234567890, 'E202', '192.168.1.5', 5)

INSERT INTO EXTRAS (ReservationID, Extra, Price, Qty) VALUES

(1, 'GPS', 10.00, 1),

(2, 'Baby Seat', 15.00, 2),

(3, 'WiFi', 8.00, 1),

(4, 'Additional Driver', 20.00, 1),

(5, 'Insurance', 30.00, 1)

CREATE TABLE Paymentmethods (

PaymentMethod VARCHAR(15) NOT NULL PRIMARY KEY

)

INSERT INTO Paymentmethods (PaymentMethod)

select distinct PaymentMethod

from RESERVATIONS

ALTER TABLE RESERVATIONS

ADD CONSTRAINT FK\_PaymentMethod

FOREIGN KEY (PaymentMethod)

REFERENCES Paymentmethods(PaymentMethod)

CREATE TABLE COINTYPES (

coin VARCHAR(10) NOT NULL PRIMARY KEY

)

INSERT INTO COINTYPES (Coin)

select distinct Coin

from RESERVATIONS

ALTER TABLE COINTYPES

ADD CONSTRAINT FK\_Coin

FOREIGN KEY (coin)

REFERENCES COINTYPES(Coin)

CREATE TABLE GEARTYPES (

Gear VARCHAR(10) NOT NULL PRIMARY KEY

)

INSERT INTO GEARTYPES(Gear)

select distinct Gear

from CARS

ALTER TABLE GEARTYPES

ADD CONSTRAINT FK\_Gear

FOREIGN KEY (Gear)

REFERENCES GEARTYPES(Gear)

CREATE TABLE CARTYPES (

CarType VARCHAR(30) NOT NULL PRIMARY KEY

)

INSERT INTO CARTYPES(CarType)

select distinct CarType

from CARS

ALTER TABLE CARS

ADD CONSTRAINT FK\_CarType

FOREIGN KEY (CarType)

REFERENCES CARTYPES(CarType)

ALTER TABLE EXTRAS DROP CONSTRAINT FK\_ReservationID

ALTER TABLE EXTRAS DROP CONSTRAINT Ck\_Price

DROP TABLE IF EXISTS EXTRAS

ALTER TABLE RESERVATIONS DROP CONSTRAINT FK\_CarID\_Reservation

ALTER TABLE RESERVATIONS DROP CONSTRAINT FK\_CARD

ALTER TABLE RESERVATIONS DROP CONSTRAINT FK\_SearchResult

ALTER TABLE RESERVATIONS DROP CONSTRAINT Ck\_Email

DROP TABLE IF EXISTS RESERVATIONS

ALTER TABLE RESULTS DROP CONSTRAINT FK\_SearchID

ALTER TABLE RESULTS DROP CONSTRAINT FK\_CarID

ALTER TABLE RESULTS DROP CONSTRAINT Ck\_PriceForAllPeriod

DROP TABLE IF EXISTS RESULTS

ALTER TABLE SEARCHES DROP CONSTRAINT FK\_BranchT

ALTER TABLE SEARCHES DROP CONSTRAINT FK\_BranchR

ALTER TABLE SEARCHES DROP CONSTRAINT CK\_StartDate\_before\_FinishDate

DROP TABLE IF EXISTS SEARCHES

DROP TABLE IF EXISTS CARDS

ALTER TABLE CARS DROP CONSTRAINT FK\_BranchID

ALTER TABLE CARS DROP CONSTRAINT FK\_CarType

DROP TABLE IF EXISTS CARS

DROP TABLE IF EXISTS CARTYPES

DROP TABLE IF EXISTS GEARTYPES

DROP TABLE IF EXISTS COINTYPES

DROP TABLE IF EXISTS Paymentmethods

DROP TABLE IF EXISTS BRANCHES