

שם: ארי שישפורטיש
ת.ז. 319306684
קורס: מיני פרוייקט לבסיסי נתונים

דו"ח על בסיס נתונים- הלוואות בנק

פרוייקט העוסק בתחום הלוואות של הבנק. עוסק בנתונים והפרטים של הלווים ותאריכי ההלוואה.

DSD

- **Loan Table**
 - Loan_ID (PK)
 - LoanStartDate
 - Borrower_ID (FK)
- **PaymentSchedule Table**
 - Schedule_ID (PK)
 - DueDate
 - Loan_ID (FK)
- **Borrower Table**
 - Borrower_ID (PK)
 - DateOfBirth

Relationships:

- **Loan** (Loan_ID) → **Borrower** (Borrower_ID)
- **Loan** (Loan_ID) → **PaymentSchedule** (Loan_ID)

SQL

```
CREATE TABLE Borrower
```

```
(  
  Borrower_ID INT NOT NULL,  
  DateOfBirth DATE NOT NULL,  
  PRIMARY KEY (Borrower_ID)  
);
```

```
CREATE TABLE Loan
```

```
(  
  Loan_ID INT NOT NULL,  
  LoanStartDate DATE NOT NULL,  
  Borrower_ID INT NOT NULL,  
  PRIMARY KEY (Loan_ID),  
  FOREIGN KEY (Borrower_ID) REFERENCES Borrower(Borrower_ID)  
);
```

```
CREATE TABLE PaymentSchedule
```

```
(  
  Schedule_ID INT NOT NULL,  
  DueDate DATE NOT NULL,  
  Loan_ID INT NOT NULL,  
  PRIMARY KEY (Schedule_ID),  
  FOREIGN KEY (Loan_ID) REFERENCES Loan(Loan_ID)  
);
```

קובץ createTables.sql

-- Create the Loan table

```
CREATE TABLE Loan (
    LoanID INT PRIMARY KEY,
    LoanStartDate DATE NOT NULL,
    BorrowerID INT,
    FOREIGN KEY (BorrowerID) REFERENCES Borrower(BorrowerID)
);
```

-- Create the PaymentSchedule table

```
CREATE TABLE PaymentSchedule (
    ScheduleID INT PRIMARY KEY,
    DueDate DATE NOT NULL,
    LoanID INT,
    FOREIGN KEY (LoanID) REFERENCES Loan(LoanID)
);
```

-- Create the Borrower table

```
CREATE TABLE Borrower (
    BorrowerID INT PRIMARY KEY,
    BorrowerName VARCHAR(100) NOT NULL,
    BorrowerAddress VARCHAR(255) NOT NULL
);
```

קובץ dropTables.sql

-- Drop the PaymentSchedule table first

DROP TABLE PaymentSchedule;

-- Drop the Loan table second

DROP TABLE Loan;

-- Drop the Borrower table last

DROP TABLE Borrower;

קובץ insertTables.sql

-- Insert data into Borrower table

INSERT INTO Borrower (BorrowerID, BorrowerName, BorrowerAddress) VALUES (1, 'John Doe', '123 Elm Street');

-- Repeat with at least 200 records

-- Insert data into Loan table

INSERT INTO Loan (LoanID, LoanStartDate, BorrowerID) VALUES (1, '2024-01-01', 1);

-- Repeat with at least 200 records

-- Insert data into PaymentSchedule table

INSERT INTO PaymentSchedule (ScheduleID, DueDate, LoanID) VALUES (1, '2024-02-01', 1);

-- Repeat with at least 200 records

selectAll.sql קובץ

-- Select all data from Borrower table

SELECT * FROM Borrower;

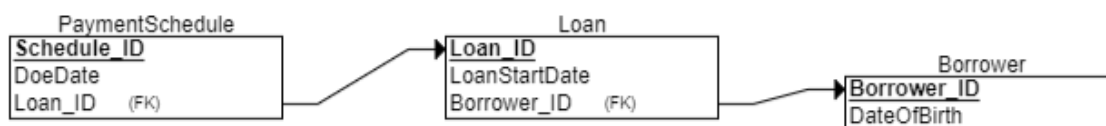
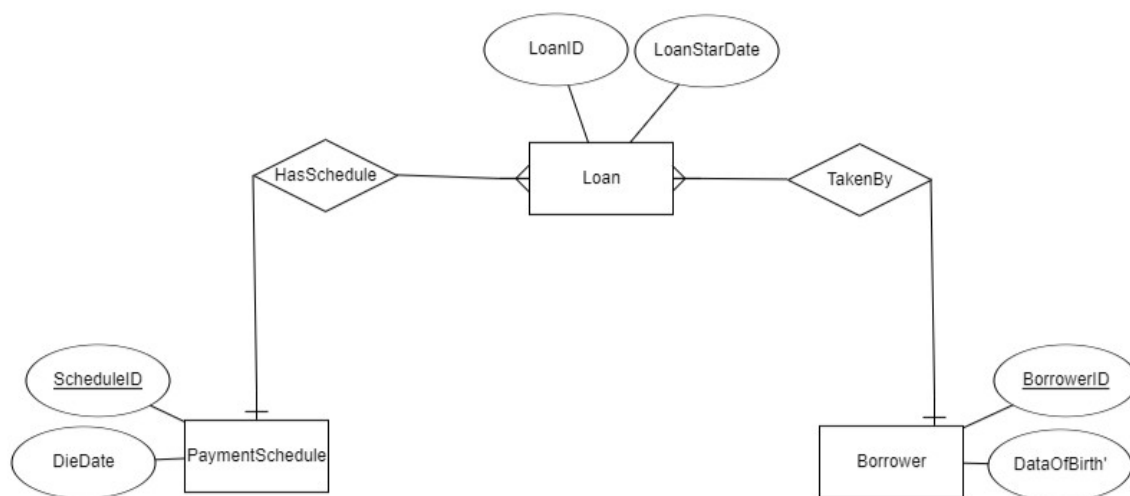
-- Select all data from Loan table

SELECT * FROM Loan;

-- Select all data from PaymentSchedule table

SELECT * FROM PaymentSchedule;

טבלאות ותרשימים



שלב 2

עכשיו נכתוב כאן סוגים שונים של שאילתות לוודא שמסד הנתונים שלנו עובד- וגם
השאילתות מביא לנו את מה שרצוי

SELECT

```
SELECT Loan.Loan_ID, Loan.LoanStartDate,  
Borrower.BorrowerName, Borrower.BorrowerAddress  
FROM Loan  
JOIN Borrower ON Loan.Borrower_ID = Borrower.Borrower_ID  
WHERE YEAR(Loan.LoanStartDate) = 2024;
```

```
SELECT PaymentSchedule.Schedule_ID,  
PaymentSchedule.DueDate, Loan.Loan_ID,  
Borrower.BorrowerName  
FROM PaymentSchedule  
JOIN Loan ON PaymentSchedule.Loan_ID = Loan.Loan_ID  
JOIN Borrower ON Loan.Borrower_ID = Borrower.Borrower_ID  
WHERE MONTH(PaymentSchedule.DueDate) = 8;
```

The screenshot shows the phpMyAdmin interface with a SQL query executed against the 'paymentschedule' table. The query is a JOIN between 'paymentschedule' and 'loan' tables, filtering by 'MONTH(PaymentSchedule.DueDate) = 8'. The results table shows columns: BorrowerName, Loan_ID, DueDate, and Schedule_ID. The results are sorted by 'DueDate' in descending order.

BorrowerName	Loan_ID	DueDate	Schedule_ID
Cherri O'Leahy	01-8182913	2033-08-15	18a10ab69571de729d7b5448f6cb9997dcd3b1e2
Anallise Tomkies	52-5809598	2031-08-30	1be6bd855deff6a6fb1a1984ae44cf08ac88c41ee
Wyn Artrick	33-0639173	2029-08-11	41ba9205b2ba54e3562c1f3203ca2f356be211b
Reinwald Phidgin	30-5075027	2054-08-04	4b1ebdaaf26930b9e8b16fcd4eb11f313ef8611
Gallagher Heggison	43-5370578	2050-08-02	4e70faa71c77e016f6055d9dd560b81a16b1f06
Stanly Orbine	66-4053517	2027-08-31	4f97899ebf8345f0c4d630e777bf48bd48f05f
Niels Gley	64-2000825	2043-08-28	5bd784ddfae1478ac14b469f5ae830d592a99f1
My Crawshay	84-2046842	2031-08-10	704419e45d3c7448d320b6526ac85abd1d5729f
Kelsi Seymer	44-9345493	2053-08-22	73c5622ea4fbb6b721e20a349a913d4a1e83ce99
Claudelle Halfhead	55-3739731	2027-08-03	7e10e0ee1705dedcb1b4fe77580981309ab49bc3
Denni Thurbbeck	47-4033495	2028-08-06	95e3d310bd1fb83f6be286437dfe6f64de0e81
Moina Anten	37-9792386	2025-08-30	977d7c895e2c8769f669c543cc344ed4d3e7977
Weider Kaikofen	58-8706230	2036-08-11	ab630c8075364665990135cd058ca5344131a07d
Crichton Scimoni	03-0081270	2033-08-26	b85a1ea2461f9fdctad89c520e03eb9a550393bc
Rollo Beare	43-3681574	2043-08-02	bc40c90f62ccaedf0774c6717ba073432a875d8
Caresa Masden	14-7187552	2047-08-08	ee78c67586ccb62a9a6890efabf75b0338c48d1
Conzain Entellae	88-0577612	2046-08-14	f083d32fbae0f0d4d4f7d73d457c72f5a77-f0f038f44d83

תמונה שהשאלות עובדות

DELETE

DELETE FROM Loan
WHERE YEAR(LoanStartDate) < 2020;

DELETE FROM PaymentSchedule
WHERE DueDate < CURDATE();

UPDATE

UPDATE Borrower
SET BorrowerAddress = 'New Address'
WHERE Borrower_ID IN (

```
SELECT Borrower_ID FROM Loan WHERE
YEAR(LoanStartDate) = 2023

);
```

```
UPDATE PaymentSchedule
SET DueDate = DATE_ADD(DueDate, INTERVAL 30 DAY)
WHERE DueDate < CURDATE();
```

