

# Query to populate the tables

```
CREATE database IF NOT EXISTS enote_01;
```

```
USE enote_01;
```

```
CREATE TABLE IF NOT EXISTS BI_assignment_person (  
    id_person INT NOT NULL AUTO_INCREMENT PRIMARY KEY,  
    name VARCHAR(255),  
    surname VARCHAR(255),  
    zip INT,  
    city VARCHAR(255),  
    country VARCHAR(255),  
    email VARCHAR(255),  
    phone_number VARCHAR(255),  
    birth_date DATE  
);
```

```
LOAD DATA LOCAL INFILE '/home/augustine/Desktop/enote1/person.csv'  
INTO TABLE BI_assignment_person  
CHARACTER SET 'utf8mb4'  
FIELDS TERMINATED BY ','  
OPTIONALLY ENCLOSED BY ''  
LINES TERMINATED BY '\n'  
IGNORE 1 ROWS  
SET phone_number = NULLIF(phone_number,"");
```

```
CREATE TABLE IF NOT EXISTS BI_assignment_account(  
    id_account INT NOT NULL AUTO_INCREMENT,  
    id_person int,  
    PRIMARY KEY (id_account),  
    FOREIGN KEY (id_person) REFERENCES BI_assignment_person(id_person),  
    account_type VARCHAR(50)  
);
```

```
LOAD DATA LOCAL INFILE '/home/augustine/Desktop/enote1/accounts.csv'  
INTO TABLE BI_assignment_account  
CHARACTER SET 'utf8mb4'  
FIELDS TERMINATED BY ','  
OPTIONALLY ENCLOSED BY ''  
LINES TERMINATED BY '\n'  
IGNORE 1 ROWS;  
# SET new_birth_date = STR_TO_DATE(birth_date,'%m/%d/%Y')
```

```
CREATE TABLE IF NOT EXISTS BI_assignment_transaction(  
    id_transaction INT NOT NULL AUTO_INCREMENT,  
    id_account int,  
    PRIMARY KEY(id_transaction),  
    FOREIGN KEY (id_account) REFERENCES BI_assignment_account(id_account),
```

```

transaction_type VARCHAR(10),
transaction_date VARCHAR(20),
transaction_amount FLOAT
);

```

```

LOAD DATA LOCAL INFILE '/home/augustine/Desktop/enote1/transactions.csv'
INTO TABLE BI_assignment_transaction
CHARACTER SET 'utf8mb4'
FIELDS TERMINATED BY ';'
OPTIONALLY ENCLOSED BY ''
LINES TERMINATED BY '\n'
IGNORE 1 ROWS
SET transaction_date = STR_TO_DATE(transaction_date, '%m/%d/%y');

```

## **Query to create a view and get monthly transactions.**

```

use enote_01;
CREATE VIEW yam AS
SELECT
    BI_assignment_account.id_account,
    BI_assignment_person.id_person,
    BI_assignment_transaction.transaction_type,
    BI_assignment_transaction.transaction_amount,
    BI_assignment_transaction.transaction_date,
    DATE_FORMAT(BI_assignment_transaction.transaction_date,
        '%m-%Y') AS month_year
FROM
    ((BI_assignment_account
        INNER JOIN BI_assignment_person ON BI_assignment_account.id_account =
BI_assignment_person.id_person)
        INNER JOIN BI_assignment_transaction ON BI_assignment_account.id_account =
BI_assignment_transaction.id_account)
WHERE
    BI_assignment_person.id_person = 345
    OR BI_assignment_account.id_account = 1234
    OR BI_assignment_person.id_person = 1234
    OR BI_assignment_transaction.id_account = 1234
    AND BI_assignment_transaction.transaction_date BETWEEN 2020 - 02 - 15 AND 2020 -
06 - 06
    order by month_year;

SELECT
    yam.id_person,
    month_year,
    SUM(transaction_amount) AS sum_of_transactions
FROM
    yam
GROUP BY month_year, id_person;

```