Nama : Rendi Alexander Hutagalung
NIM : 122450057
Kelas : RC

Tugas Mandiri

1. Membuat tabel
   Tabel Pelanggan

```sql
CREATE TABLE Pelanggan (
    pelanggan_id SERIAL PRIMARY KEY,
    pelanggan_name VARCHAR(255) NOT NULL,
    pelanggan_email VARCHAR(100) UNIQUE,
    pelanggan_address TEXT,
    phone_number VARCHAR(15) UNIQUE
);
```

   Tabel Pesanan

```sql
CREATE TABLE Pesanan (
    no_order SERIAL PRIMARY KEY,
    pelanggan_id INT REFERENCES Pelanggan(pelanggan_id), y
    date_orders DATE NOT NULL,
    product VARCHAR(100),
    quantity INT,
    price DECIMAL(15, 2),
    total_price DECIMAL(15, 2)
);
```

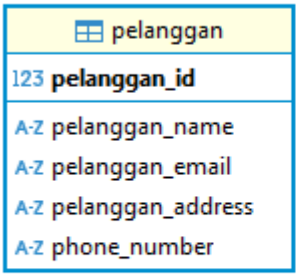
2. Insert Data
   Memasukan data pelanggan

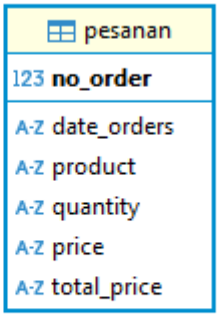
```sql
INSERT INTO customers (name, email, address, phone_number) VALUES
('Ahmad Rizky', 'ahmad.rizky@example.com', 'Jl. Melati No.1, Jakarta', '081234567890'),
('Siti Nuraini', 'siti.nuraini@example.com', 'Jl. Cempaka No.2, Surabaya', '082345678901'),
('Dedi Prasetyo', 'dedi.prasetyo@example.com', 'Jl. Mawar No.3, Bandung', '083456789012'),
('Tia Sulastri', 'tia.sulastri@example.com', 'Jl. Anggrek No.4, Yogyakarta', '084567890123'),
('Bayu Nugroho', 'bayu.nugroho@example.com', 'Jl. Kenanga No.5, Semarang', '085678901234'),
('Rina Pertiwi', 'rina.pertiwi@example.com', 'Jl. Melur No.6, Medan', '086789012345'),
('Fajar Kusuma', 'fajar.kusuma@example.com', 'Jl. Seroja No.7, Makassar', '087890123456'),
('Liana Indah', 'liana.indah@example.com', 'Jl. Angsana No.8, Bali', '088901234567'),
('Dwi Wibowo', 'dwi.wibowo@example.com', 'Jl. Taman No.9, Palembang', '089012345678'),
('Rudi Setiawan', 'rudi.setiawan@example.com', 'Jl. Pandan No.10, Malang', '090123456789');
```

   Memasukan data pesanan

```sql
INSERT INTO Orders (no_order, date_orders, product, quantity, price, total_price) VALUES
(1, '2025-03-06', 'Laptop', 2, 5000000.00, 10000000.00),
(2, '2025-03-06', 'Smartphone', 3, 3000000.00, 9000000.00),
(3, '2025-03-06', 'Tablet', 5, 2000000.00, 10000000.00),
(4, '2025-03-06', 'Headphones', 10, 500000.00, 5000000.00),
(5, '2025-03-06', 'Monitor', 4, 1500000.00, 6000000.00),
(6, '2025-03-06', 'Keyboard', 7, 350000.00, 2450000.00),
(7, '2025-03-06', 'Mouse', 15, 150000.00, 2250000.00),
(8, '2025-03-06', 'Printer', 3, 1200000.00, 3600000.00),
(9, '2025-03-06', 'Webcam', 6, 800000.00, 4800000.00),
(10, '2025-03-06', 'External Hard Drive', 4, 1000000.00, 4000000.00);
```

   ER-Diagram





Data pelanggan

	123 pelanggan_id	A-Z pelanggan_name	A-Z pelanggan_email	A-Z pelanggan_address	A-Z phone_number
1	1	Ahmad Rizky	ahmad.rizky@example.com	Jl. Melati No.1, Jakarta	081234567890
2	2	Siti Nuraini	siti.nuraini@example.com	Jl. Cempaka No.2, Surabaya	082345678901
3	3	Dedi Prasetyo	dedi.prasetyo@example.com	Jl. Mawar No.3, Bandung	083456789012
4	4	Tia Sulastri	tia.sulastri@example.com	Jl. Anggrek No.4, Yogyakarta	084567890123
5	5	Bayu Nugroho	bayu.nugroho@example.com	Jl. Kenanga No.5, Semarang	085678901234
6	6	Rina Pertiwi	rina.pertiwi@example.com	Jl. Melur No.6, Medan	086789012345
7	7	Fajar Kusuma	fajar.kusuma@example.com	Jl. Seroja No.7, Makassar	087890123456
8	8	Liana Indah	liana.indah@example.com	Jl. Angsana No.8, Bali	088901234567
9	9	Dwi Wibowo	dwi.wibowo@example.com	Jl. Taman No.9, Palembang	089012345678
10	10	Rudi Setiawan	rudi.setiawan@example.com	Jl. Pandan No.10, Malang	090123456789

Data pesanan

pesanan | Enter a SQL expression to filter results (use Ctrl+Space)

	123 no_order	A-Z date_orders	A-Z product	A-Z quantity	A-Z price	A-Z total_price
1	1	2025-03-06	Laptop	2	5000000.00	10000000.00
2	2	2025-03-06	Smartphone	3	3000000.00	9000000.00
3	3	2025-03-06	Tablet	5	2000000.00	10000000.00
4	4	2025-03-06	Headphones	10	500000.00	5000000.00
5	5	2025-03-06	Monitor	4	1500000.00	6000000.00
6	6	2025-03-06	Keyboard	7	350000.00	2450000.00
7	7	2025-03-06	Mouse	15	150000.00	2250000.00
8	8	2025-03-06	Printer	3	1200000.00	3600000.00
9	9	2025-03-06	Webcam	6	800000.00	4800000.00
10	10	2025-03-06	External Hard Drive	4	1000000.00	4000000.00

Melakukan Tuning

1. Spliting Data
   Vertikal

```sql
--Melakukan Tuning
-- Splitting Tabel
-- Tabel Pesanan_Header (Denormalisasi dengan Menyimpan Informasi Pelanggan)
-- Tabel Pelanggan (Bagian 1) - Kolom yang sering digunakan
--Vertikal
CREATE TABLE Pelanggan_Core (
    pelanggan_id SERIAL PRIMARY KEY,
    pelanggan_name VARCHAR(255) NOT NULL,
    pelanggan_email VARCHAR(100) UNIQUE NOT NULL,
    phone_number VARCHAR(15) UNIQUE NOT NULL
);

-- Tabel Pelanggan (Bagian 2) - Kolom yang jarang digunakan
CREATE TABLE Pelanggan_Alamat (
    pelanggan_id INT PRIMARY KEY,
    pelanggan_address TEXT,
    FOREIGN KEY (pelanggan_id) REFERENCES Pelanggan_Core(pelanggan_id)
);
```

Pelanggan_Core hanya menyimpan informasi dasar pelanggan yang sering diakses seperti nama, email, dan nomor telepon.
Pelanggan_Alamat menyimpan alamat pelanggan, yang hanya diakses jika diperlukan, misalnya saat mengirimkan barang atau dalam pencarian berbasis alamat.

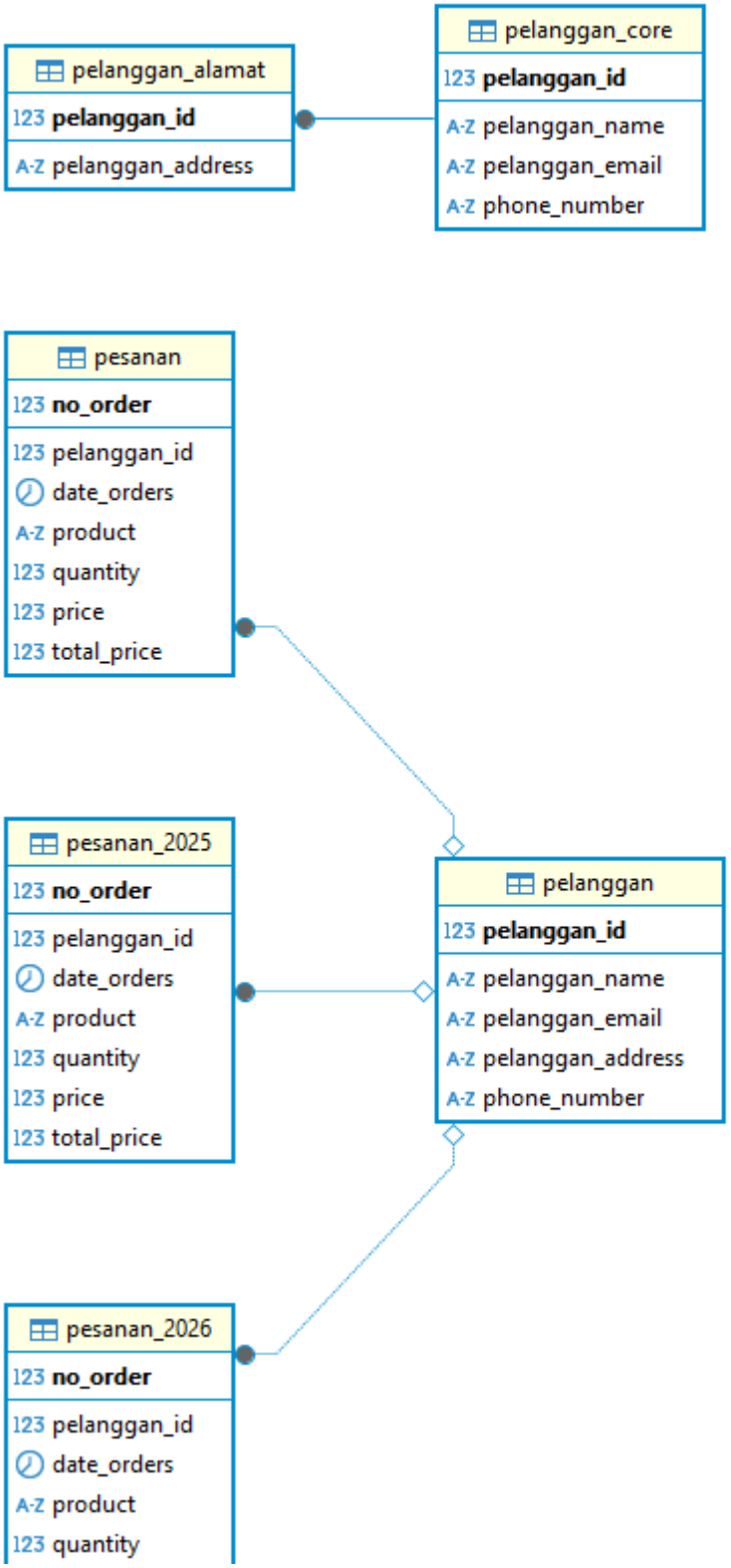
Horizontal

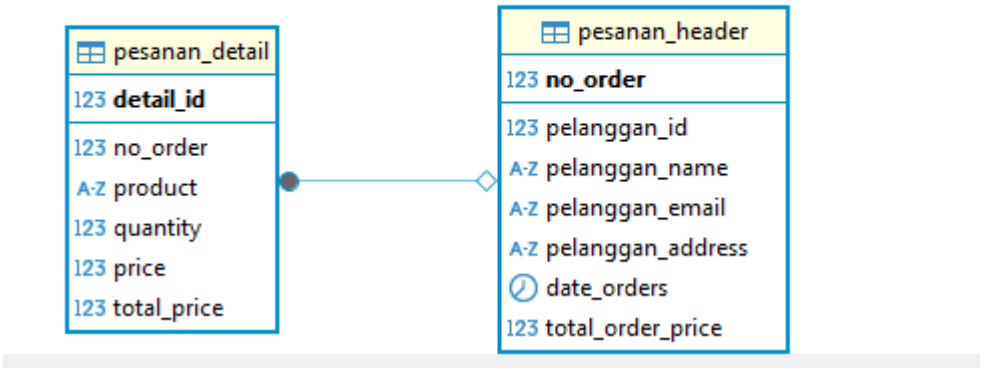
```sql
--Horizontal

-- Tabel Pesanan untuk tahun 2025
CREATE TABLE Pesanan_2025 (
    no_order SERIAL PRIMARY KEY,
    pelanggan_id INT REFERENCES Pelanggan(pelanggan_id),
    date_orders DATE NOT NULL,
    product VARCHAR(100),
    quantity INT,
    price DECIMAL(15, 2),
    total_price DECIMAL(15, 2)
);

-- Tabel Pesanan untuk tahun 2026
CREATE TABLE Pesanan_2026 (
    no_order SERIAL PRIMARY KEY,
    pelanggan_id INT REFERENCES Pelanggan(pelanggan_id),
    date_orders DATE NOT NULL,
    product VARCHAR(100),
    quantity INT,
    price DECIMAL(15, 2),
    total_price DECIMAL(15, 2)
);
```

Denormalisasi

```sql
--Denormalisai
CREATE TABLE Pesanan_Header (
    no_order SERIAL PRIMARY KEY,
    pelanggan_id INT NOT NULL,
    pelanggan_name VARCHAR(255) NOT NULL,
    pelanggan_email VARCHAR(100) NOT NULL,
    pelanggan_address TEXT NOT NULL,
    date_orders DATE NOT NULL,
    total_order_price DECIMAL(15, 2)
);

-- Tabel Pesanan_Detail (Menyimpan item per pesanan)
CREATE TABLE Pesanan_Detail (
    detail_id SERIAL PRIMARY KEY,
    no_order INT REFERENCES Pesanan_Header(no_order),
    product VARCHAR(100) NOT NULL,
    quantity INT NOT NULL,
    price DECIMAL(15, 2) NOT NULL,
    total_price DECIMAL(15, 2) NOT NULL
);

-- Menambahkan Pesanan_header
INSERT INTO Pesanan_Header (pelanggan_id, pelanggan_name, pelanggan_email, pelanggan_address, date_orders, total_order_price) VALUES
(1, 'Ahmad Rizky', 'ahmad.rizky@example.com', 'Jl. Melati No.1, Jakarta', '2025-03-06', 10000000.00),
(2, 'Siti Nuraini', 'siti.nuraini@example.com', 'Jl. Cempaka No.2, Surabaya', '2025-03-06', 9000000.00),
(3, 'Dedi Prasetyo', 'dedi.prasetyo@example.com', 'Jl. Mawar No.3, Bandung', '2025-03-06', 10000000.00);

-- Menambahkan data pesanan detail
INSERT INTO Pesanan_Detail (no_order, product, quantity, price, total_price) VALUES
(1, 'Laptop', 2, 5000000.00, 10000000.00),
(2, 'Smartphone', 3, 3000000.00, 9000000.00),
(3, 'Tablet', 5, 2000000.00, 10000000.00);
```

Hasil


pelanggan_alamat
123 pelanggan_id
A-Z pelanggan_address
pelanggan_core
123 pelanggan_id
A-Z pelanggan_name
A-Z pelanggan_email
A-Z phone_number
pesanan
123 no_order
123 pelanggan_id
date_orders
A-Z product
123 quantity
123 price
123 total_price
pesanan_2025
123 no_order
123 pelanggan_id
date_orders
A-Z product
123 quantity
123 price
123 total_price
pelanggan
123 pelanggan_id
A-Z pelanggan_name
A-Z pelanggan_email
A-Z pelanggan_address
A-Z phone_number
pesanan_2026
123 no_order
123 pelanggan_id
date_orders
A-Z product
123 quantity



Dalam Hasil ER Diagram Terlihat hubungan dan perlakuan tuning yaiitu dengan membagi tabel sama dengan hasil kode diatas dimana pesanan dengan pelanggan, pesanan detail, dan pelanggan core.

Hasil insert data

pelanggan | Enter a SQL expression to filter results (use Ctrl+Space)

	pelanggan_id	pelanggan_name	pelanggan_email	pelanggan_address	phone_number
1	1	Ahmad Rizky	ahmad.rizky@example.com	Jl. Melati No.1, Jakarta	081234567890
2	2	Siti Nuraini	siti.nuraini@example.com	Jl. Cempaka No.2, Surabaya	082345678901
3	3	Dedi Prasetyo	dedi.prasetyo@example.com	Jl. Mawar No.3, Bandung	083456789012
4	4	Tia Sulastri	tia.sulastri@example.com	Jl. Anggrek No.4, Yogyakarta	084567890123
5	5	Bayu Nugroho	bayu.nugroho@example.com	Jl. Kenanga No.5, Semarang	085678901234
6	6	Rina Pertiwi	rina.pertiwi@example.com	Jl. Melur No.6, Medan	086789012345
7	7	Fajar Kusuma	fajar.kusuma@example.com	Jl. Seroja No.7, Makassar	087890123456
8	8	Liana Indah	liana.indah@example.com	Jl. Angsana No.8, Bali	088901234567
9	9	Dwi Wibowo	dwi.wibowo@example.com	Jl. Taman No.9, Palembang	089012345678
10	10	Rudi Setiawan	rudi.setiawan@example.com	Jl. Pandan No.10, Malang	090123456789

	no_order	pelanggan_id	date_orders	product	quantity	price	total_price
1	1	[NULL]	2025-03-06	Laptop	2	5,000,000	10,000,000
2	2	[NULL]	2025-03-06	Smartphone	3	3,000,000	9,000,000
3	3	[NULL]	2025-03-06	Tablet	5	2,000,000	10,000,000
4	4	[NULL]	2025-03-06	Headphones	10	500,000	5,000,000
5	5	[NULL]	2025-03-06	Monitor	4	1,500,000	6,000,000
6	6	[NULL]	2025-03-06	Keyboard	7	350,000	2,450,000
7	7	[NULL]	2025-03-06	Mouse	15	150,000	2,250,000
8	8	[NULL]	2025-03-06	Printer	3	1,200,000	3,600,000
9	9	[NULL]	2025-03-06	Webcam	6	800,000	4,800,000
10	10	[NULL]	2025-03-06	External Hard Drive	4	1,000,000	4,000,000

Modul 2 Teknologi Basis Data Melakukan Tuning

1. Membuat Tabel

```sql
CREATE TABLE Publishers (
    publisher_id SERIAL PRIMARY KEY,
    publisher_name VARCHAR(255) NOT NULL,
    contact_email VARCHAR(100),
    contact_phone VARCHAR(15)
);

CREATE TABLE Books (
    book_id SERIAL PRIMARY KEY,
    title VARCHAR(255) NOT NULL,
    author VARCHAR(255),
    publisher_id INT,
    publication_year INT,
    genre VARCHAR(100),
    stock INT DEFAULT 0,
    FOREIGN KEY (publisher_id) REFERENCES Publishers(publisher_id)
);

CREATE TABLE Members (
    member_id SERIAL PRIMARY KEY,
    first_name VARCHAR(100),
    last_name VARCHAR(100),
    email VARCHAR(100) UNIQUE NOT NULL,
    phone_number VARCHAR(15),
    join_date DATE DEFAULT CURRENT_DATE
);

CREATE TABLE Borrowings (
    borrowing_id SERIAL PRIMARY KEY,
    member_id INT,
    book_id INT,
    borrow_date DATE DEFAULT CURRENT_DATE,
    return_date DATE,
    FOREIGN KEY (member_id) REFERENCES Members(member_id),
    FOREIGN KEY (book_id) REFERENCES Books(book_id)
);
```

Terdapat tabel publisher, books, members, dan borrowings dengan ketentuan masing masing tabel yang dibutuhkan.

Memasukan data Publisher

```sql
INSERT INTO Publishers (publisher_name, contact_email, contact_phone)
VALUES
('Gramedia', 'info@gramedia.com', '0211234567'),
('Erlangga', 'contact@erlangga.co.id', '0212345678'),
('Mizan', 'mizan@mizan.com', '0223456789'),
('HarperCollins', 'info@harpercollins.com', '0214567890'),
('Springer', 'springer@springer.com', '0225678901'),
('Penguin Random House', 'contact@penguinrandomhouse.com', '0216789012'),
('Oxford University Press', 'contact@oup.com', '0217890123'),
('Cambridge University Press', 'info@cambridge.org', '0218901234'),
('Wiley', 'support@wiley.com', '0219012345'),
('McGraw-Hill', 'support@mheducation.com', '0219123456'),
('Pearson', 'contact@pearson.com', '0219234567'),
('Routledge', 'info@routledge.com', '0219345678'),
('Elsevier', 'info@elsevier.com', '0219456789'),
('SAGE Publications', 'contact@sagepub.com', '0219567890'),
('Hachette Livre', 'support@hachette.com', '0219678901'),
('John Wiley & Sons', 'wiley@wiley.com', '0219789012'),
('Blackwell Publishing', 'support@blackwell.com', '0219890123'),
('Harvard University Press', 'contact@harvardpress.com', '0219901234'),
('MIT Press', 'info@mitpress.com', '0219912345'),
('Princeton University Press', 'support@press.princeton.edu', '0219923456'),
('Stanford University Press', 'contact@sup.org', '0219934567'),
('University of Chicago Press', 'info@press.uchicago.edu', '0219945678'),
('Palgrave Macmillan', 'support@palgrave.com', '0219956789'),
('Bloomsbury Publishing', 'contact@bloomsbury.com', '0219967890'),
('Kogan Page', 'info@koganpage.com', '0219978901'),
('Taylor & Francis', 'support@taylorandfrancis.com', '0219989012'),
('Springer Nature', 'info@springernature.com', '0219990123'),
('University Presses of California', 'support@calpress.edu', '0220001234'),
('Indigo Press', 'info@indigopress.com', '0220012345');
```

Memasukan data books

```sql
INSERT INTO Books (title, author, publisher_id, publication_year, genre,
stock)
VALUES
('To Kill a Mockingbird', 'Harper Lee', 1, 1960, 'Fiction', 10),
('1984', 'George Orwell', 2, 1949, 'Dystopian', 15),
('Pride and Prejudice', 'Jane Austen', 3, 1813, 'Romance', 5),
('The Great Gatsby', 'F. Scott Fitzgerald', 4, 1925, 'Fiction', 8),
('Moby Dick', 'Herman Melville', 5, 1851, 'Adventure', 12),
('War and Peace', 'Leo Tolstoy', 6, 1869, 'Historical Fiction', 7),
('Crime and Punishment', 'Fyodor Dostoevsky', 7, 1866, 'Psychological
Fiction', 9),
('The Odyssey', 'Homer', 8, -800, 'Epic', 6),
('The Catcher in the Rye', 'J.D. Salinger', 9, 1951, 'Fiction', 14),
('The Hobbit', 'J.R.R. Tolkien', 10, 1937, 'Fantasy', 20),
('Brave New World', 'Aldous Huxley', 1, 1932, 'Dystopian', 13),
('Frankenstein', 'Mary Shelley', 2, 1818, 'Gothic Fiction', 11),
('Dracula', 'Bram Stoker', 3, 1897, 'Horror', 10),
('Les Misérables', 'Victor Hugo', 4, 1862, 'Historical Fiction', 8),
('The Divine Comedy', 'Dante Alighieri', 5, 1320, 'Epic', 9),
('The Brothers Karamazov', 'Fyodor Dostoevsky', 6, 1880, 'Philosophical
Fiction', 5),
('The Iliad', 'Homer', 7, -750, 'Epic', 6),
('Wuthering Heights', 'Emily Brontë', 8, 1847, 'Gothic Fiction', 12),
('Jane Eyre', 'Charlotte Brontë', 9, 1847, 'Romance', 8),
('The Picture of Dorian Gray', 'Oscar Wilde', 10, 1890, 'Philosophical
Fiction', 7),
('A Tale of Two Cities', 'Charles Dickens', 1, 1859, 'Historical Fiction',
10),
('The Lord of the Rings', 'J.R.R. Tolkien', 2, 1954, 'Fantasy', 25),
('The Chronicles of Narnia', 'C.S. Lewis', 3, 1956, 'Fantasy', 18),
('The Da Vinci Code', 'Dan Brown', 4, 2003, 'Thriller', 30),
('The Shining', 'Stephen King', 5, 1977, 'Horror', 13),
('The Alchemist', 'Paulo Coelho', 6, 1988, 'Adventure', 22),
('Dune', 'Frank Herbert', 7, 1965, 'Science Fiction', 17),
('The Catcher in the Rye', 'J.D. Salinger', 8, 1951, 'Fiction', 19),
('The Hunger Games', 'Suzanne Collins', 9, 2008, 'Dystopian', 24),
('Gone with the Wind', 'Margaret Mitchell', 10, 1936, 'Historical Fiction',
```

Memasukan data member

```sql
INSERT INTO Members (first_name, last_name, email, phone_number)
VALUES
('Alice', 'Johnson', 'alice.johnson@example.com', '081234567890'),
('Bob', 'Smith', 'bob.smith@example.com', '081234567891'),
('Charlie', 'Brown', 'charlie.brown@example.com', '081234567892'),
('David', 'Williams', 'david.williams@example.com', '081234567893'),
('Emma', 'Davis', 'emma.davis@example.com', '081234567894'),
('Frank', 'Miller', 'frank.miller@example.com', '081234567895'),
('Grace', 'Wilson', 'grace.wilson@example.com', '081234567896'),
('Hannah', 'Moore', 'hannah.moore@example.com', '081234567897'),
('Ivy', 'Taylor', 'ivy.taylor@example.com', '081234567898'),
('Jack', 'Anderson', 'jack.anderson@example.com', '081234567899'),
('Kathy', 'Thomas', 'kathy.thomas@example.com', '081234567900'),
('Leo', 'Martinez', 'leo.martinez@example.com', '081234567901'),
('Mona', 'Hernandez', 'mona.hernandez@example.com', '081234567902'),
('Nina', 'Roberts', 'nina.roberts@example.com', '081234567903'),
('Oscar', 'King', 'oscar.king@example.com', '081234567904'),
('Paula', 'Scott', 'paula.scott@example.com', '081234567905'),
('Quinn', 'Adams', 'quinn.adams@example.com', '081234567906'),
('Ryan', 'Baker', 'ryan.baker@example.com', '081234567907'),
('Sara', 'Carter', 'sara.carter@example.com', '081234567908'),
('Tina', 'Gomez', 'tina.gomez@example.com', '081234567909'),
('Uma', 'Evans', 'uma.evans@example.com', '081234567910'),
('Vera', 'Clark', 'vera.clark@example.com', '081234567911'),
('Wendy', 'Lewis', 'wendy.lewis@example.com', '081234567912'),
('Xander', 'Young', 'xander.young@example.com', '081234567913'),
('Yara', 'Walker', 'yara.walker@example.com', '081234567914'),
('Zara', 'Nelson', 'zara.nelson@example.com', '081234567915');
```

Memasukan data borrowings

```sql
INSERT INTO Borrowings (member_id, book_id, borrow_date, return_date)
VALUES
(1, 1, '2025-03-01', '2025-03-15'),
(2, 2, '2025-03-02', '2025-03-16'),
(3, 3, '2025-03-03', '2025-03-17'),
(4, 4, '2025-03-04', '2025-03-18'),
(5, 5, '2025-03-05', '2025-03-19'),
(6, 6, '2025-03-06', '2025-03-20'),
(7, 7, '2025-03-07', '2025-03-21'),
(8, 8, '2025-03-08', '2025-03-22'),
(9, 9, '2025-03-09', '2025-03-23'),
(10, 10, '2025-03-10', '2025-03-24'),
(11, 11, '2025-03-11', '2025-03-25'),
(12, 12, '2025-03-12', '2025-03-26'),
(13, 13, '2025-03-13', '2025-03-27'),
(14, 14, '2025-03-14', '2025-03-28'),
(15, 15, '2025-03-15', '2025-03-29'),
(16, 16, '2025-03-16', '2025-03-30'),
(17, 17, '2025-03-17', '2025-03-31'),
(18, 18, '2025-03-18', '2025-04-01'),
(19, 19, '2025-03-19', '2025-04-02'),
(20, 20, '2025-03-20', '2025-04-03'),
(21, 21, '2025-03-21', '2025-04-04'),
(22, 22, '2025-03-22', '2025-04-05'),
(23, 23, '2025-03-23', '2025-04-06'),
(24, 24, '2025-03-24', '2025-04-07'),
(25, 25, '2025-03-25', '2025-04-08'),
(26, 26, '2025-03-26', '2025-04-09')
;
```

Melakukan Tuning

```sql
CREATE TABLE Borrowings_2025 (
borrowing_id SERIAL PRIMARY KEY,
member_id INT NOT NULL,
book_id INT NOT NULL,
borrow_date DATE DEFAULT CURRENT_DATE,
return_date DATE,
FOREIGN KEY (member_id) REFERENCES Members(member_id) ON DELETE
CASCADE,
FOREIGN KEY (book_id) REFERENCES Books(book_id) ON DELETE CASCADE
);
```

```sql
INSERT INTO Borrowings_2025 (member_id, book_id, borrow_date, return_date)
SELECT member_id, book_id, borrow_date, return_date
FROM Borrowings
WHERE EXTRACT(YEAR FROM borrow_date) = 2025;
```

```sql
CREATE TABLE Borrowing_Info (
borrowing_id SERIAL PRIMARY KEY,
member_id INT NOT NULL,
book_id INT NOT NULL,
FOREIGN KEY (member_id) REFERENCES Members(member_id) ON DELETE
CASCADE,
FOREIGN KEY (book_id) REFERENCES Books(book_id) ON DELETE CASCADE
);
```

```sql
CREATE TABLE Borrowing_Dates (
borrowing_id INT PRIMARY KEY,
borrow_date DATE DEFAULT CURRENT_DATE,
return_date DATE,
FOREIGN KEY (borrowing_id) REFERENCES Borrowing_Info(borrowing_id) ON
DELETE CASCADE
);
```

```sql
INSERT INTO Borrowing_Info (borrowing_id, member_id, book_id)
SELECT borrowing_id, member_id, book_id
FROM Borrowings;

INSERT INTO Borrowing_Dates (borrowing_id, borrow_date, return_date)
SELECT borrowing_id, borrow_date, return_date
FROM Borrowings;
```