

Tabel awal

id_transaksi	id_user	id_barang	harga_sewa	durasi_sewa	total_harga	tgl_transaksi	tgl_selesai
AA001	IDS001	BRG001	650000	1 hari	650000	12/12/2024	13/12/2024
AA002	IDS002	BRG002,BRG007	600000, 700000	1hari, 2hari	600000, 1400000	12/12/2024	13-12-2024, 14-12-2024

Tabel 1NF

1nf

id_transaksi	id_user	id_barang	harga_sewa	durasi_sewa	total_harga	tgl_transaksi	tgl_selesai
AA001	IDS001	BRG001	650000	1 hari	650000	12/12/2024	13/12/2024
AA002	IDS002	BRG002	600000	1hari	600000	12/12/2024	13/12/2024
AA002	1DS002	BRG007	700000	2 hari	1400000	12/12/2024	14/12/2024

Tabel 2NF

2nf

table transaksi

id_transaksi	id_user	id_barang	total_harga	tgl_transaksi	tgl_selesai
AA001	IDS001	BRG001	650000	12/12/2024	13/12/2024
AA002	IDS002	BRG002	600000	12/12/2024	13/12/2024
AA002	1DS002	BRG007	1400000	12/12/2024	14/12/2024

table barang

id_barang	harga_sewa	durasi_sewa
BRG001	650000	1 hari
BRG002	600000	1hari
BRG007	700000	2 hari

Tabel 3NF

3nf

Definisi: Data harus dalam 2NF dan tidak ada ketergantungan transitif (atribut non-primer tidak boleh bergantung pada atribut non-primer lainnya).

table transaksi

id_transaksi	id_user	id_barang	total_harga	tgl_transaksi	tgl_selesai
AA001	IDS001	BRG001	650000	12/12/2024	13/12/2024
AA002	IDS002	BRG002	600000	12/12/2024	13/12/2024
AA002	1DS002	BRG007	1400000	12/12/2024	14/12/2024

table barang

id_barang	harga_sewa	durasi_sewa
BRG001	650000	1 hari
BRG002	600000	1hari
BRG007	700000	2 hari

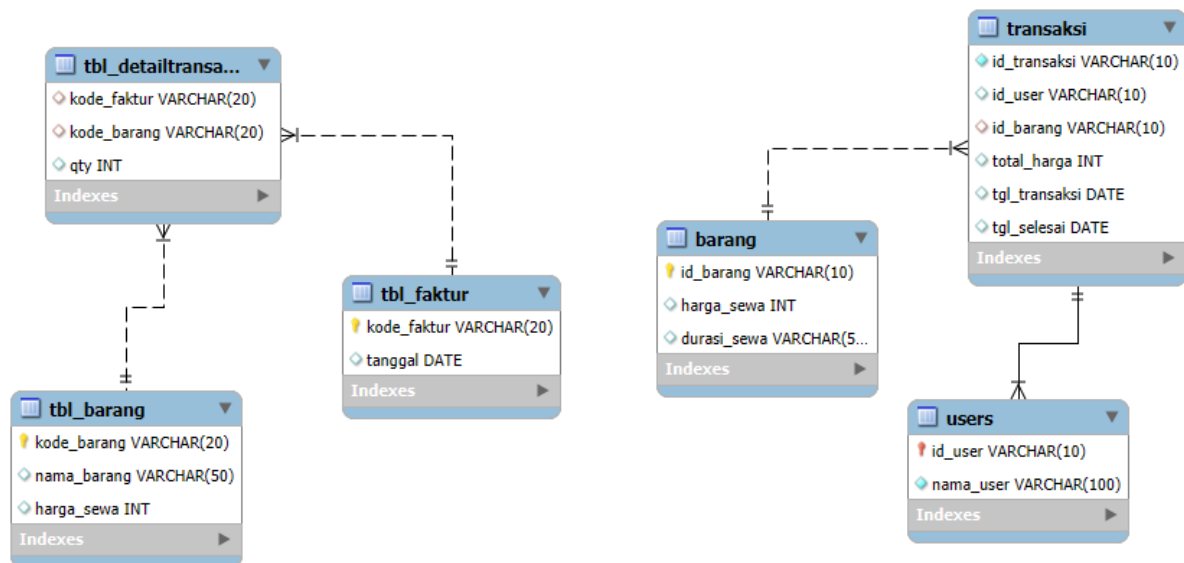
3NF (Third Normal Form)

Definisi: Data harus dalam 2NF dan tidak ada ketergantungan transitif (atribut non-primer tidak boleh bergantung pada atribut non-primer lainnya).

Masalah di tabel:

- Tidak ada ketergantungan transitif, sehingga data sudah memenuhi 3NF.

2. ERD



3. Query SQL untuk Kasus Bisnis

3.1 Kasus JOIN

```
1  SELECT
2      b.id_barang,
3      b.harga_sewa,
4      COUNT(t.id_transaksi) AS jumlah_sewa
5  FROM
6      Transaksi t
7  JOIN
8      Barang b ON t.id_barang = b.id_barang
9  GROUP BY
10     b.id_barang, b.harga_sewa
11  HAVING
12     COUNT(t.id_transaksi) > 1
13  ORDER BY
14     jumlah_sewa DESC;
15
```

3.2 subquery

```
1  SELECT
2      b.id_barang,
3      b.harga_sewa,
4      b.durasi_sewa
5  FROM
6      Barang b
7  WHERE
8      NOT EXISTS (
9          SELECT 1
10         FROM Transaksi t
11         WHERE b.id_barang = t.id_barang
12     );
13 |
```

4. Implementasi Database Objects

4.1 view

View untuk Menampilkan Transaksi Lengkap:

```
1  CREATE VIEW view_transaksi_lengkap AS
2  SELECT
3      t.id_transaksi,
4      u.nama_user,
5      b.id_barang,
6      b.harga_sewa,
7      t.total_harga,
8      t.tgl_transaksi,
9      t.tgl_selesai
10 FROM
11     transaksi t
12 JOIN
13     users u ON t.id_user = u.id_user
14 JOIN
15     barang b ON t.id_barang = b.id_barang;
16
```

4.2 Stored Procedure

Procedure untuk Menambahkan Transaksi Baru:

```
1 DELIMITER $$
2
3 CREATE PROCEDURE add_transaksi (
4     IN p_id_transaksi VARCHAR(10),
5     IN p_id_users VARCHAR(10),
6     IN p_id_barang VARCHAR(10),
7     IN p_total_harga INT,
8     IN p_tgl_transaksi DATE,
9     IN p_tgl_selesai DATE
10 )
11 BEGIN
12     INSERT INTO transaksi (id_transaksi, id_user, id_barang, total_harga, tgl_transaksi, tgl_selesai)
13     VALUES (p_id_transaksi, p_id_user, p_id_barang, p_total_harga, p_tgl_transaksi, p_tgl_selesai);
14 END$$
15
16 DELIMITER ;
17
```

4.3 Function untuk Menghitung Total Pendapatan:

```
1 DELIMITER $$
2
3 CREATE FUNCTION total_pendapatan()
4 RETURNS INT
5 DETERMINISTIC
6 BEGIN
7     DECLARE total INT;
8     SELECT SUM(total_harga) INTO total FROM transaksi;
9     RETURN total;
10 END$$
11
12 DELIMITER ;
13
```

4. Kesimpulan

- Database objects seperti **View, Procedure, Function** membantu menyederhanakan pengelolaan data dan meningkatkan otomatisasi.
- Query SQL dirancang untuk mendukung berbagai kebutuhan bisnis, seperti menampilkan data transaksi lengkap, menghitung total pengeluaran, dan mengotomatisasi pembersihan data lama.

Jika Anda memiliki kebutuhan tambahan, database dapat diperluas atau disesuaikan lebih lanjut.