

Laporan Penggunaan JOIN dan VIEW di MySQL

I. Pendahuluan

Di laporan ini, saya akan menjelaskan LEFT JOIN dan RIGHT JOIN dalam MySQL, serta pembuatan VIEW dari hasil tersebut

II. Implementasi JOIN

A. LEFT JOIN

Kita ingin mengambil data buku dan genre yang terkait.

Query:

```
MySQL [dbtoko]> SELECT books.title, genres.name AS genre_name
-> FROM books
-> LEFT JOIN genres ON books.genre_id = genres.id;
```

title	genre_name
Harry Potter and the Sorcerer's Stone	Fiction
The Lord of the Rings	Non-Fiction
Foundation	Science Fiction
Dune	Science Fiction
1984	Non-Fiction
The Great Gatsby	Fiction

6 rows in set (0.001 sec)

- Dengan query ini, kita mendapatkan semua judul buku. Jika ada buku tanpa genre, kolom genre akan menunjukkan NULL.

B. LEFT JOIN

Kita ingin melihat nama pengguna dan jumlah transaksi yang mereka lakukan

Query:

```
MySQL [dbtoko]> SELECT users.name, COUNT(transactions.id) AS total_transactions
-> FROM users
-> LEFT JOIN transactions ON users.id = transactions.costumer_id
-> GROUP BY users.id;
```

name	total_transactions
Ramadhan Al Ghifari	2
Alucard	3
Nana	0
Balmond	0
Alice Johnson	0
Bob Smith	0
Charlie Brown	0
Diana Prince	0
Ethan Hunt	0

9 rows in set (0.005 sec)

- Query ini menampilkan nama pengguna dan jumlah total transaksi mereka. Jika pengguna tidak memiliki transaksi totalnya akan menjadi 0.

C. RIGHT JOIN

Kita ingin mengambil semua genre dan judul buku.

Query:

```
MySQL [dbtoko]> SELECT genres.name AS genre_name, books.title
-> FROM genres
-> RIGHT JOIN books ON genres.id = books.genre_id;
```

genre_name	title
Fiction	Harry Potter and the Sorcerer's Stone
Non-Fiction	The Lord of the Rings
Science Fiction	Foundation
Science Fiction	Dune
Non-Fiction	1984
Fiction	The Great Gatsby

6 rows in set (0.001 sec)

- Dengan query ini, kita mendapatkan semua gener dan judul buku. Jika adab uku yang tidak memili genre, nama genre akan ditampilkan sebagai NULL.

D. RIGHT JOIN

Kita ingin melihat semua transaksi dan pengguna yang terkait.

Query:

```
MySQL [dbtoko]> SELECT transactions.id, users.name
-> FROM transactions
-> RIGHT JOIN users ON transactions.costumer_id = users.id;
```

id	name
1	Ramadhan Al Ghifari
2	Ramadhan Al Ghifari
3	Alucard
4	Alucard
5	Alucard
NULL	Nana
NULL	Balmond
NULL	Alice Johnson
NULL	Bob Smith
NULL	Charlie Brown
NULL	Diana Prince
NULL	Ethan Hunt

12 rows in set (0.001 sec)

- Query ini menampilkan semua transaksi dan nama pengguna. Jika suatu transaksi tidak memiliki pengguna, nama pengguna akan di tampilkan sebagai NULL.

III. Implementasi VIEW

A. VIEW untuk LEFT JOIN Buku dan Genre

Query:

```
MySQL [dbtoko]> SELECT books.title, genres.name AS genre_name
-> FROM books
-> LEFT JOIN genres ON books.genre_id = genres.id;
```

title	genre_name
Harry Potter and the Sorcerer's Stone	Fiction
The Lord of the Rings	Non-Fiction
Foundation	Science Fiction
Dune	Science Fiction
1984	Non-Fiction
The Great Gatsby	Fiction

6 rows in set (0.001 sec)

B. VIEW untuk LEFT JOIN Pengguna dan Transaksi

Query:

```
MySQL [dbtoko]> SELECT users.name AS user_name, COUNT(transactions.id) AS total_transactions
-> FROM users
-> LEFT JOIN transactions ON users.id = transactions.costumer_id
-> GROUP BY users.id;
```

user_name	total_transactions
Ramadhan Al Ghifari	2
Alucard	3
Nana	0
Balmond	0
Alice Johnson	0
Bob Smith	0
Charlie Brown	0
Diana Prince	0
Ethan Hunt	0

```
9 rows in set (0.001 sec)
```

IV. Menggunakan VIEW

A. Mengambil data dari book_genre_view

Query:

```
MySQL [dbtoko]> SELECT * FROM book_genre_view;
```

title	genre_name
Harry Potter and the Sorcerer's Stone	Fiction
The Lord of the Rings	Non-Fiction
Foundation	Science Fiction
Dune	Science Fiction
1984	Non-Fiction
The Great Gatsby	Fiction

```
6 rows in set (0.001 sec)
```

B. Mengambil data dari user_transactions_summary_view

Query:

```
MySQL [dbtoko]> SELECT * FROM user_transaction_summary_view;
```

name	total_transactions
Ramadhan Al Ghifari	2
Alucard	3
Nana	0
Balmond	0
Alice Johnson	0
Bob Smith	0
Charlie Brown	0
Diana Prince	0
Ethan Hunt	0

```
9 rows in set (0.001 sec)
```