

STRUCTURED QUERY LANGUAGE (SQL) BASIS DATA LANJUT

Pada laporan akhir basis data lanjut ini membahas mengenai penerapan SQL kedalam suatu rancangan basis data, yakni “*Travel Tour Agent*” yang akan dibahas lebih lanjut dengan sub-bab seperti dibawah ini

1.1 BASIS DATA

Basis data (*database*) adalah kumpulan data yang terstruktur dan diorganisir agar dapat diakses, dikelola, dan diperbarui dengan mudah. Basis data digunakan untuk menyimpan informasi yang dapat diakses oleh berbagai aplikasi, dan ini merupakan komponen integral dari sistem informasi di berbagai bidang.

Atau, dapat diartikan pula bahwa Basis data merupakan sistem terdiri dari komponen yang terhubung untuk memfasilitasi aliran informasi guna mencapai tujuan. Data adalah catatan fakta yang memerlukan pengolahan untuk menghasilkan kesimpulan dan keputusan. Basis data adalah kumpulan data yang disimpan secara sistematis dalam komputer untuk memudahkan pencarian informasi, membantu pengambilan keputusan, dan menjaga integritas data operasional organisasi.

1.2 STRUCTURED QUERY LANGUAGE (SQL)

Structured Query Language (SQL) merupakan sebuah bahasa yang dipergunakan untuk mengakses data dalam basis data relasional. Bahasa ini secara de facto merupakan bahasa standar yang digunakan dalam manajemen basis data relasional. Saat ini hampir semua server basis data yang ada mendukung bahasa ini untuk melakukan manajemen datanya.

Atau dapat diartikan pula bahwa *Structured Query Language* (SQL) merupakan sebuah pemrograman khusus yang untuk dipergunakan untuk mengelola dan mengakses basis data relasional. Basis data relasional adalah cara penyimpanan dan pengelolaan data yang menggunakan tabel untuk menyimpan informasi dan perlu diperhatikan bahwa tabel-tabel tersebut saling terkait antara satu dengan yang lainnya.

1.3 TRAVEL TOUR AGENT

Pada project basis data lanjut ini, kami menggunakan judul *project* dengan studi kasus “*Travel Tour Agent*”. “*Travel Tour Agent*” merupakan suatu basis data untuk agen perjalanan yang di rancang untuk menyimpan, mengelola, dan mengakses informasi terkait perjalanan tur yang telah diorgansir dengan baik. Hal ini bertujuan untuk mengelola informasi pelanggan, destinasi, akomodasi, dan juga transportasi. Dimana pada basis data ini memiliki kolom paket, pesan, user, admin, hotel, dan daerah hotel.

Tentunya untuk melakukan proses penerapan “*Travel Tour Agent*” ke dalam *SQL*, diperlukan beberapa tahapan yakni diperlukannya pembuatan ERD (Entity Relationship Diagram), Flowchart, CDM (Conceptual Data Model), PDM (Physical Data Model), dan dilakukan *import* ke dalam MySQL yang akan dijabarkan lebih lanjut seperti berikut.

1.4 ERD, FLOWCHART, CDM, PDM, DAN MYSQL, DATA DUMMY

Untuk hasil dari pembuatan ERD, Flowchart, CDM, PDM, MYSQL, dan Data Dummy akan dijabarkan lebih lanjut dan ditampilkan didalam link-link yang dituliskan, seperti berikut.

- **Link ERD** : http://bit.ly/ERD-Tour_Agent-10
- **Link Flowchart** : https://bit.ly/Flowchart-Tour_Agent-10
- **Link CDM** : https://bit.ly/CDM-Tour_Agent-10
- **Link PDM** : https://bit.ly/PDM-Tour_Agent-10
- **Link MYSQL** : https://bit.ly/SQL-Tour_Agent-10

1.5 PROSES PREPARING QUERY STRUCTURED QUERY LANGUAGE (SQL)

Perlu diketahui bahwa sebelum melakukan proses *query* di SQL perlu dilakukan proses *import* data dummy yang telah dibuat terlebih dahulu, diantaranya, yakni.

1.5.1 Di Lakukan Proses Impor Data Dummy “Pesan_Travel_Tour” Ke Dalam SQL

ID_PESAN	ID_HOTEL	ID_PAKET	ID_BUKTI	STATUS_PESAN	TANGGAL_PESAN	TANGGAL_TOUR_PESAN
E01	H01	C01	D01	Belum Luna	01/01/2019	03/01/2020
E02	H02	C02	D02	Lunas	05/01/2019	04/01/2020
E03	H03	C03	D03	Lunas	16/01/2019	08/01/2020
E04	H04	C04	D04	Lunas	24/01/2019	10/01/2020
E05	H05	C05	D05	Lunas	26/01/2019	15/01/2020
E06	H06	C06	D06	Lunas	29/01/2019	17/01/2020
E07	H07	C07	D07	Belum Luna	30/01/2019	26/01/2020
E08	H08	C08	D08	Lunas	02/02/2019	27/01/2020
E09	H09	C09	D09	Belum Luna	09/02/2019	30/01/2020
E10	H10	C10	D10	Lunas	11/02/2019	30/01/2020

1.5.2 Di Lakukan Proses Impor Data Dummy “Paket_Travel_Tour” Ke Dalam SQL

ID_PAKET	ID_PESAN	NAMA_PAKET	HARGA_PAKET	KETERANGAN_PAKET
C01	E01	Stone	Rp500.000,00	Tur Kota Dasar
C02	E02	Silver	Rp800.000,00	Destinasi Wisata Populer
C03	E03	Golden	Rp1.100.000,00	Pengalaman Premium dan Keindahan Alam
C04	E04	Platinum	Rp1.300.000,00	Destinasi Mewah dan Eksklusif
C05	E05	Titanium	Rp1.600.000,00	Perjalanan Mewah dan Pribadi
C06	E06	Silver	Rp800.000,00	Destinasi Wisata Populer
C07	E07	Golden	Rp1.100.000,00	Pengalaman Premium dan Keindahan Alam
C08	E08	Silver	Rp800.000,00	Destinasi Wisata Populer
C09	E09	Stone	Rp500.000,00	Tur Kota Dasar

1.5.3 Di Lakukan Proses Impor Data Dummy “Admin_Travel_Tour” Ke Dalam SQL

ID_ADMIN	PASS_ADMIN	NAMA_ADMIN	STATUS_ADMIN
A01	1[2suVsb	Catherine Ferguson	Offline
A02	@{qNH5^	Laura Webster	Online
A03	oB#Xlqz:	Stephanie Phelps	Offline
A04	_][[-B'F	Candace Thompson	Online
A05	n4]K >*	Jennifer Ward	Online
A06	A<V9uYv5	Robert Dorsey	Online
A07	i3c[psiA	Morgan Bennett	Online
A08	a"s0F))	Aaron Jackson	Online
A09	RDWPILw	Billy Jones	Offline
A10	s[bDi#<=	Riley Fisher	Online

1.5.4 Di Lakukan Proses Impor Data Dummy “User_Travel_Tour” Ke Dalam SQL

ID_USER	PASS_USER	NAMA_USER	NO_TELP_USER
A102	Rojijog	Gavin Turner	+62 81156116169
A108	Nuwjdwrn	Nora Carter	+62 85912161215
A112	Epqafcsq	Yuri Suzuki	+62 87297044856
A116	Jruciswn	Mackenzie Davis	+62 89390837925
A119	Zchgipd	Teguh Susilo	+62 87095969514
A140	Czdmeqnz	Victoria Davis	+62 89431266122
A149	Kccvackm	Qodir Nugroho	+62 81620710395
A171	Dr@g0nFlyI	Isaac Taylor	+62 89576412520
A174	Qlepwrqk	Rendra Wijaya	+62 84701128398
A207	Trwryqua	Wahyu Nugroho	+62 85788481832

1.5.5 Di Lakukan Proses Impor Data Dummy “Bukti_Travel_Tour” Ke Dalam SQL

ID_BUKTI	ID_ADMIN	ID_USER	FILE_BUKTI
D01	A01	E296	Digital
D02	A02	C915	Digital
D03	A03	D849	Tercetak
D04	A04	B680	Digital
D05	A05	C913	Tercetak
D06	A06	D273	Tercetak
D07	A07	C939	Digital
D08	A08	B951	Digital
D09	A09	C278	Digital
D10	A10	E755	Tercetak

1.5.6 Di Lakukan Proses Impor Data Dummy “Hotel_Travel_Tour” Ke Dalam SQL

ID_HOTEL	ID_DAERAH	NAMA_HOTEL	BINTANG_HOTEL	HARGA_HOTEL
H01	G01	Grand Orch	5	Rp1 100.000,00
H02	G02	Emerald Ho	4	Rp780 000,00
H03	G03	Paradise H	4	Rp675 000,00
H04	G04	Indah Pala	3	Rp400 000,00
H05	G05	Harmoni He	5	Rp1 220.000,00
H06	G06	Emerald Ho	4	Rp780 000,00
H07	G07	Paradise H	4	Rp675 000,00
H08	G08	Emerald Ho	4	Rp780 000,00
H09	G09	Grand Orch	5	Rp1 100.000,00
H10	G10	Indah Pala	3	Rp400 000,00

1.5.7 Di Lakukan Proses Impor Data Dummy “Daerah Hotel_Travel_Tour” Ke Dalam SQL



ID_DAERAH	NAMA_DAERAH
G01	Bukittinggi
G02	Padang
G03	Padangpanjang
G04	Makassar
G05	Jambi
G06	Sukabumi
G07	Bekasi
G08	Pangkalpinang
G09	Kotamobagu
G10	Makassar

1.6 PENERAPAN QUERY STRUCTURED QUERY LANGUAGE (SQL)

Setelah melakukan pembuatan ERD sebagai rancangan *entity*, Flowchart sebagai diagram alir CDM kemudian dikonversi menjadi PDM, dan dilanjutkan dengan pembuatan data *record* sebanyak masing-masing 500 data pada setiap kolom dan proses *preparing*. Lalu, dapat dilakukan beberapa penerapan *query language* (SQL) yang akan dijabarkan lebih lanjut, seperti di bawah ini.

1.6.1 Melakukan Penerapan CRUD (Create, Read Update, Delete)

Dilakukan proses penerapan CRUD (Create, Read, Update, Delete), yang akan dijabarkan lebih lanjut seperti dibawah ini.

A. Insert

Insert (penyisipan) merupakan suatu *query* yang dipergunakan untuk menyisipkan atau menambahkan suatu data baru kedalam tabel atau kolom.

Dilakukan penerapan.

• Query



• Output

ID_DAERAH	NAMA_DAERAH
G496	Solok
G497	Jayapura
G498	Solok
G499	Madiun
G50	Jayapura
G500	Bekasi
G501	Ubud
G51	Pekalongan
G52	Yogyakarta
G53	Depok
G54	Metro

- **Penjelasan Singkat**

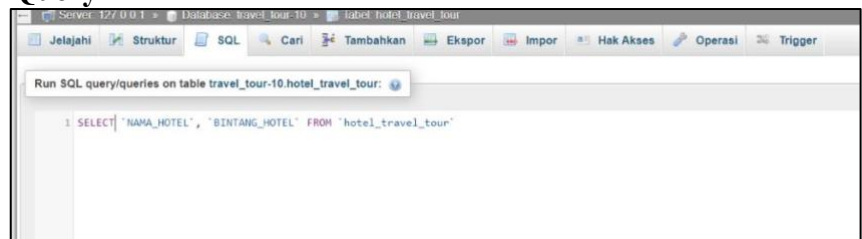
Pada penerapan ini dilakukan penambahan data pada tabel “daerah_hotel_travel_tour” dengan menggunakan **Insert**. Dengan insiasi atribut meliputi id_daerah sebesar G501 dan nama_daerah dengan “Ubud”. Maka, akan ditampilkan hasil seperti pada gambar *output* sehingga pada tabel “daerah_travel_tour” terdapat 501 *record* data.

B. Select

Select (pemilihan) merupakan *query* yang dipergunakan untuk mengambil atau menampilkan data dari suatu tabel dalam basis data.

B.1 Menampilkan Select Pada Syarat Kriteria Tertentu

- **Query**



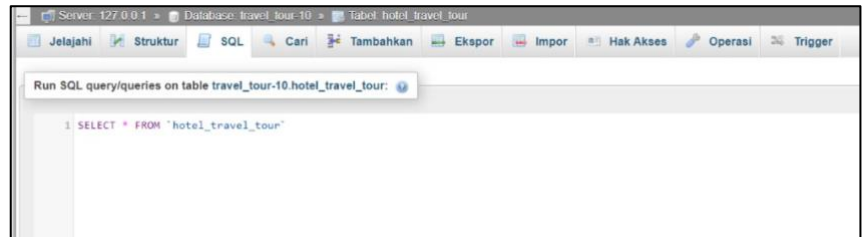
- **Output**

	NAMA_HOTEL	BINTANG_HOTEL
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	Grand Orch	5
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	Emerald Ho	4
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	Paradise H	4
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	Indah Pala	3
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	Harmoni He	5
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	Emerald Ho	4
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	Paradise H	4
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	Emerald Ho	4

• Penjelasan Penerapan

Pada penerapan ini dilakukan pemilihan dalam penampilan data pada tabel “hotel_travel_tour” dengan menggunakan **Select From**. Dengan inisiasi pemilihan atribut NAMA_HOTEL dan BINTANG_HOTEL. Maka, akan ditampilkan hasil seperti pada gambar *output* yakni data pada “hotel_travel_tour” dengan atribut nama NAMA_HOTEL dan BINTANG_HOTEL.

B.2 Menampilkan Select Pada Hotel Travel Tour • Query



• Output

	ID_HOTEL	ID_DAERAH	NAMA_HOTEL	BINTANG_HOTEL	HARGA_HOTEL
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H01	G01	Grand Orch	5	Rp1.100.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H02	G02	Emerald Ho	4	Rp780.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H03	G03	Paradise H	4	Rp675.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H04	G04	Indah Pala	3	Rp400.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H05	G05	Harmoni He	5	Rp1.220.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H06	G06	Emerald Ho	4	Rp780.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H07	G07	Paradise H	4	Rp675.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H08	G08	Emerald Ho	4	Rp780.000,00

• Penjelasan Singkat

Delete (penghapusan) merupakan *query* yang dipergunakan untuk menghapus satu data atau beberapa data, suatu baris atau beberapa baris data dari tabel didalam *SOL*.

- **Query**



- **Output**

The screenshot shows the table view of the 'daerah_hotel_travel_tour' table. The table has two columns: 'ID_DAERAH' and 'NAMA_DAERAH'. There are 8 rows of data, with the first row being 'G01' and 'Bukittinggi'.

ID_DAERAH	NAMA_DAERAH
G01	Bukittinggi
G02	Padang
G03	Padangpanjang
G04	Makassar
G05	Jambi
G06	Sukabumi
G07	Bekasi
G08	Pangkalpinang

- **Penjelasan Penerapan**

Pada penerapan ini, dilakukan proses penghapusan suatu baris data dalam tabel “daerah_hotel_travel_tour” dengan menggunakan **Delete**. Dengan inisiasi melakukan penghapusan pada data dengan id_daerah sebesar G501. Maka, akan ditampilkan hasil seperti pada gambar *output* yakni hilangnya data dengan id_daerah G501.

1.6.2 Eksplorasi Perintah SQL

Dilakukan proses penerapan eksplorasi perintah SQL, yang akan dijabarkan lebih lanjut seperti dibawah ini.

A. Where

Where (dimana) merupakan *query* yang dipergunakan untuk menetapkan kriteria pencarian tertentu dengan melakukan filter baris-baris yang akan diambil dari tabel berdasarkan kondisi tertentu.

A.1 Menampilkan Where Pada Daerah Hotel Di Surabaya

- **Query**



- **Output**

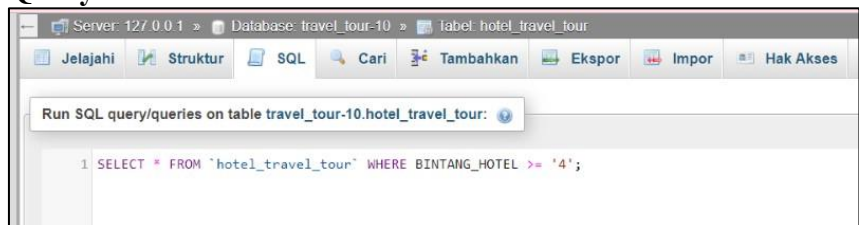
ID_DAERAH	NAMA_DAERAH
G166	Surabaya
G211	Surabaya
G336	Surabaya
G391	Surabaya
G429	Surabaya
G43	Surabaya
G437	Surabaya

- **Penjelasan Penerapan**

Pada penerapan ini dilakukan proses pemilihan penampilan dan pencarian pada tabel “daerah_hotel_travel_tour” dengan NAMA_DAERAH Surabaya dengan menggunakan **Select * From Where**. Maka, akan ditampilkan hasil seperti gambar output yakni id_daerah dengan nama daerah Surabaya.

A.2 Menampilkan Where Pada Bintang Hotel Lebih Dari Sama Dengan 4

- **Query**



- **Output**

Server: 127.0.0.1 - Database: travel_tour-10 - label: hotel_travel_tour					
Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi					
	ID_HOTEL	ID_DAERAH	NAMA_HOTEL	BINTANG_HOTEL	HARGA_HOTEL
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H01	G01	Grand Orch	5	Rp1.100.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H02	G02	Emerald Ho	4	Rp780.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H03	G03	Paradise H	4	Rp675.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H05	G05	Harmoni He	5	Rp1.220.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H06	G06	Emerald Ho	4	Rp780.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H07	G07	Paradise H	4	Rp675.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H08	G08	Emerald Ho	4	Rp780.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H09	G09	Grand Orch	5	Rp1.100.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H101	G101	Emerald Ho	4	Rp780.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H102	G102	Grand Orch	5	Rp1.100.000,00

- **Penjelasan Penerapan**

Pada penerapam ini dilakukan proses pemilihan, penampilan dan pencarian bintang hotel dengan kondisi lebih dari sama dengan 4 dalam tabel “hotel_travel_tour” dengan menggunakan **Select * From Where >=**. Maka, akan ditampilkan hasil seperti gambar *output* yakni nama-nama hotel yang memiliki bintang hotel lebih besar sama dengan 4, meliputi 4 dan 5.

A.3 Menampilkan Where Pada Bintang Hotel Tidak Sama Dengan 4 •

Query

Server: 127.0.0.1 - Database: travel_tour-10 - label: hotel_travel_tour					
Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi					
Run SQL query/queries on table travel_tour-10.hotel_travel_tour:					
1 SELECT * FROM 'hotel_travel_tour' WHERE BINTANG_HOTEL <> '4';					

- **Output**

Server: 127.0.0.1 - Database: travel_tour-10 - label: hotel_travel_tour					
Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi Trigger					
	ID_HOTEL	ID_DAERAH	NAMA_HOTEL	BINTANG_HOTEL	HARGA_HOTEL
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H01	G01	Grand Orch	5	Rp1.100.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H04	G04	Indah Pala	3	Rp400.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H05	G05	Harmoni He	5	Rp1.220.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H09	G09	Grand Orch	5	Rp1.100.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H10	G10	Indah Pala	3	Rp400.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H100	G100	Indah Pala	3	Rp400.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H102	G102	Grand Orch	5	Rp1.100.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H105	G105	Grand Orch	5	Rp1.100.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H106	G106	Indah Pala	3	Rp400.000,00

- **Penjelasan Penerapan**

Pada penerapan ini dilakukan proses pemilihan, penampilan dan pencarian bintang hotel dengan kondisi tidak sama dengan 4 dalam tabel “hotel_travel_tour” dengan menggunakan **Select * From <>**. Maka, akan ditampilkan hasil seperti gambar *output* yakni nama-nama hotel yang memiliki bintang hotel tidak sama dengan 4, meliputi 3 dan 5.

A.4 Menampilkan Where Pada Bintang Hotel Lebih Kecil Dari 4 •

Query



• Output

The screenshot shows the output of the SQL query. It displays a table with the following data:

ID_HOTEL	ID_DAERAH	NAMA_HOTEL	BINTANG_HOTEL	HARGA_HOTEL
H04	G04	Indah Pala	3	Rp400,000.00
H10	G10	Indah Pala	3	Rp400,000.00
H100	G100	Indah Pala	3	Rp400,000.00
H106	G106	Indah Pala	3	Rp400,000.00
H109	G109	Indah Pala	3	Rp400,000.00
H112	G112	Indah Pala	3	Rp400,000.00

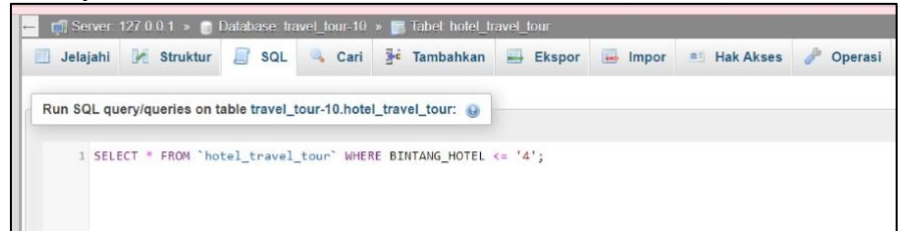
• Penjelasan Penerapan

Pada penerapan ini dilakukan proses pemilihan, penampilan dan pencarian bintang hotel dengan kondisi lebih kecil dari 4 dalam tabel “hotel_travel_tour” dengan menggunakan **Select * From <**. Maka, akan ditampilkan hasil seperti gambar *output* yakni namanama hotel yang memiliki bintang hotel lebih kecil dari 4, meliputi

3.

A.5 Menampilkan Where Pada Bintang Hotel Lebih Kecil Sama Dengan 4

- Query



- Output

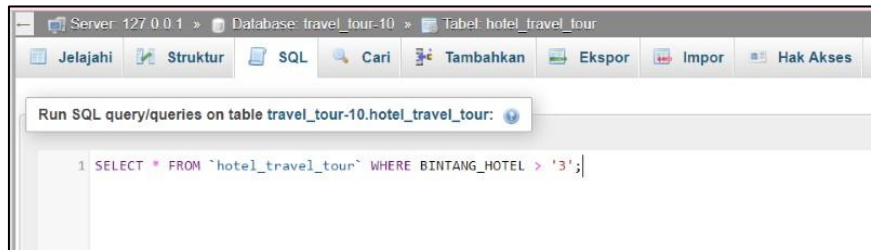
	ID_HOTEL	ID_DAERAH	NAMA_HOTEL	BINTANG_HOTEL	HARGA_HOTEL
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H02	G02	Emerald Ho	4	Rp780.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H03	G03	Paradise H	4	Rp675.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H04	G04	Indah Pala	3	Rp400.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H06	G06	Emerald Ho	4	Rp780.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H07	G07	Paradise H	4	Rp675.000,00

- Penjelasan Penerapan

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pencarian bintang hotel dengan kondisi lebih kecil sama dengan 4 dalam tabel “hotel_travel_tour” dengan menggunakan **Select * From <=**. Maka, akan ditampilkan hasil seperti gambar *output*, yakni nama-nama hotel yang memiliki bintang lebih kecil sama dengan 4, meliputi 3 dan 4.

A.6 Menampilkan Where Pada Bintang Hotel Lebih Besar Dari 3

- Query



- **Output**

ID_HOTEL	ID_DAERAH	NAMA_HOTEL	BINTANG_HOTEL	HARGA_HOTEL
H01	G01	Grand Orch	5	Rp1.100.000,00
H02	G02	Emerald Ho	4	Rp780.000,00
H03	G03	Paradise H	4	Rp675.000,00
H05	G05	Harmoni He	5	Rp1.220.000,00
H06	G06	Emerald Ho	4	Rp780.000,00
H07	G07	Paradise H	4	Rp675.000,00
H08	G08	Emerald Ho	4	Rp780.000,00

- **Penjelasan Penerapan**

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pencarian bintang hotel dengan kondisi lebih besar dari 3 dalam tabel “hotel_travel_tour” dengan menggunakan **Select * From >**. Maka, akan ditampilkan hasil seperti gambar *output*, yakni namanama hotel yang memiliki bintang lebih besar dari 3, meliputi 4,5.

B. Operator Logika And, Or, Dan Not

B.1 And

And (dan) merupakan *query* yang dipergunakan untuk menggabungkan dua atau lebih kondisi dan akan menampilkan hasil yang memenuhi semua kondisi yang dinyatakan.

- **Query**



- **Output**

Server: 127.0.0.1 - Database: travel_tour-10 - Tabel: paket_travel_tour

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi

Tampilkan kotak kueri

Menampilkan baris 0 - 24 (total 167, Pencarian dilakukan dalam 0,0004 detik.)

SELECT * FROM 'paket_travel_tour' WHERE NAMA_PAKET = 'Stone' AND KETERANGAN_PAKET = 'Tur Kota Dasar';

Profil [Edit kotak] [Ubah] [Jelaskan SQL] [Buat kode PHP] [Segarkan]

1 > >> ☐ Tampilkan semua | Jumlah baris: 25 Saring baris: Cari di tabel ini Sort by key: T

Extra options

	ID_PAKET	ID_PESAN	NAMA_PAKET	HARGA_PAKET	KETERANGAN_PAKET
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	C01	E01	Stone	Rp500.000,00	Tur Kota Dasar
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	C09	E09	Stone	Rp500.000,00	Tur Kota Dasar
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	C102	E102	Stone	Rp500.000,00	Tur Kota Dasar
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	C105	E105	Stone	Rp500.000,00	Tur Kota Dasar
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	C108	E108	Stone	Rp500.000,00	Tur Kota Dasar

• Penjelasan Penerapan

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan penarian pada tabel “paket_travel_tour” dengan kondisi NAMA_PAKET “Stone” dan KETERANGAN_PAKET “Tur Kota Dasar” dengan menggunakan **Select * From Where And**. Maka, akan ditampilkan hasil seperti gambar *output*, yakni nama-nama paket “Stone” dengan keterangan “Tur Kota Dasar” sesuai dengan kondisi yang telah dituliskan.

B.2 Or

Or (atau) merupakan *query* yang dipergunakan untuk menggabungkan dua atau lebih kondisi dengan syarat jika setidaknya satu kondisi harus terpenuhi, maka baris tersebut akan ditampilkan kedalam hasil.

• Query

Server: 127.0.0.1 - Database: travel_tour-10 - Tabel: paket_travel_tour

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi

Run SQL query/queries on table travel_tour-10.paket_travel_tour:

1 SELECT * FROM 'paket_travel_tour' WHERE NAMA_PAKET = 'Stone' OR NAMA_PAKET = 'Silver';

• Output

Server: 127.0.0.1 > Database: travel_tour-10 > Tabel: paket_travel_tour

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi

Tampilkan kotak kuen

Menampilkan baris 0 - 24 (total 317, Pencarian dilakukan dalam 0,0003 detik.)

SELECT * FROM `paket_travel_tour` WHERE NAMA_PAKET = 'Stone' OR NAMA_PAKET = 'Silver';

Profil [Edit dikotak] [Ubah] [Jelaskan SQL] [Buat kode PHP] [Segarkan]

1 > >> ☐ Tampilkan semua Jumlah baris: 25 Saring baris: Cari di tabel ini Sort by key:

Extra options

	ID_PAKET	ID_PESAN	NAMA_PAKET	HARGA_PAKET	KETERANGAN_PAKET
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	C01	E01	Stone	Rp500.000,00	Tur Kota Dasar
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	C02	E02	Silver	Rp800.000,00	Destinasi Wisata Populer
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	C06	E06	Silver	Rp800.000,00	Destinasi Wisata Populer
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	C08	E08	Silver	Rp800.000,00	Destinasi Wisata Populer
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	C09	E09	Stone	Rp500.000,00	Tur Kota Dasar

- **Penjelasan Penerapan**

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pencarian pada tabel “paket_travel_tour” dengan kondisi NAMA_PAKET “Stone” atau “Silver” dengan menggunakan **Select * From Where Or**. Maka, akan ditampilkan hasil seperti pada gambar *output*, yakni nama-nama paket “Stone” atau “Silver”

B.3 Not

- **Query**

Server: 127.0.0.1 > Database: travel_tour-10 > Tabel: paket_travel_tour

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi

Run SQL query/queries on table travel_tour-10.paket_travel_tour:

1 SELECT * FROM `paket_travel_tour` WHERE NOT NAMA_PAKET = 'Golden';

- **Output**

ID_PAKET	ID_PESAN	NAMA_PAKET	HARGA_PAKET	KETERANGAN_PAKET
C01	E01	Stone	Rp500.000,00	Tur Kota Dasar
C02	E02	Silver	Rp800.000,00	Destinasi Wisata Populer
C04	E04	Platinum	Rp1.300.000,00	Destinasi Mewah dan Eksklusif
C05	E05	Titanium	Rp1.600.000,00	Perjalanan Mewah dan Pribadi
C06	E06	Silver	Rp800.000,00	Destinasi Wisata Populer

- **Penjelasan Penerapan**

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pencarian tabel “paket_travel_tour” dengan kondisi NAMA_PAKET selain “Golden” dengan menggunakan **Select * From Where Not**. Maka, akan ditampilkan nama-nama paket selain “Golden” meliputi stone, silver, platinum, titanium.

C. Between Dan Nor Between

C.1 Between

Between (diantara) merupakan *query* yang dipergunakan untuk menentukan rentang nilai dalam suatu kondisi dalam pernyataan *query* atau dipergunakan untuk menyaring data berdasarkan suatu kolom yang berada direntang tertentu.

- **Query**

```

1 SELECT * FROM 'hotel_travel_tour' WHERE BINTANG_HOTEL BETWEEN 4 AND 5;

```

- **Output**

Menampilkan baris 0 - 24 (total 227. Pencarian dilakukan dalam 0.0004 detik.)

SELECT * FROM 'hotel_travel_tour' WHERE BINTANG_HOTEL NOT BETWEEN 3 AND 4;

Profil [Edit dikotak] [Ubah] [Jelaskan SQL] [Buat kode PHP] [Segarkan]

1 > >> ☐ Tampilkan semua Jumlah baris: 25 Saring baris: Cari di tabel ini Sort by key: Tidak ada

Extra options:

	ID_HOTEL	ID_DAERAH	NAMA_HOTEL	BINTANG_HOTEL	HARGA_HOTEL
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H01	G01	Grand Orch	5	Rp1.100.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H05	G05	Harmoni He	5	Rp1.220.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H09	G09	Grand Orch	5	Rp1.100.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H102	G102	Grand Orch	5	Rp1.100.000,00

- **Penjelasan Penerapan**

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pencarian pada “hotel_travel_tour” dengan kondisi bintang hotel direntang 4 dan 5 dengan menggunakan **Select * From Where Between**. Maka, akan ditampilkan semua data ditabel “hotel_travel_tour” dengan rentang 4 dan 5.

C.2 Not Between

Not Between (tidak diantara) merupakan *query* yang digunakan untuk menyaring data yang berada di luar rentang nilai tertentu dalam suatu kondisi dalam pernyataan *query* atau Ketika ingin mengecualikan data yang berada dalam rentang nilai yang didefinisikan.

- **Query**

Run SQL query/queries on table travel_tour-10.hotel_travel_tour:

1 SELECT * FROM 'hotel_travel_tour' WHERE BINTANG_HOTEL NOT BETWEEN 3 AND 4;

- **Output**

Menampilkan baris 0 - 24 (total 227, Pencarian dilakukan dalam 0,0004 detik)

SELECT * FROM "hotel_travel_tour" WHERE BINTANG_HOTEL NOT BETWEEN 3 AND 4;

Profil [Edit dikotak] [Ubah] [Jelaskan SQL] [Buat kode PHP] [Segarkan]

1 > >> ☐ Tampilkan semua Jumlah baris: 25 Saring baris: Cari di tabel ini Sort by key: Tidak ada

Extra options

	ID_HOTEL	ID_DAERAH	NAMA_HOTEL	BINTANG_HOTEL	HARGA_HOTEL
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H01	G01	Grand Orch	5	Rp1.100.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H05	G05	Harmoni He	5	Rp1.220.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H09	G09	Grand Orch	5	Rp1.100.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H102	G102	Grand Orch	5	Rp1.100.000,00
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	H105	G105	Grand Orch	5	Rp1.100.000,00

- **Penjelasan Penerapan**

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pencarian pada “hotel_travel_tour” dengan kondisi bintang hotel tidak berada di rentang 3 dan 4 dengan menggunakan **Select * From Where Not Between**. Maka, akan ditampilkan semua data ditabel “hotel_travel_tour” dengan rentang 5 saja, karena mengecualikan data yang berada dalam rentang.

D. Like Dan Not Like

D.1 Like

Like digunakan untuk mencari suatu teks dengan suatu pola berdasarkan kata depan (prefix), kata tengah (infix), dan kata akhir (suffix). Pada operator ini terdapat penggunaan tanda persen (%) untuk mewakili satu atau lebih kecocokan karakter.

- **Query**

Run SQL query/queries on table travel_tour-10.user_travel_tour:

1 SELECT * FROM USER_TRAVEL_TOUR WHERE NAMA_USER LIKE 'a%';

- **Output**

ID_USER	PASS_USER	NAMA_USER	NO_TELP_USER
Ubah Salin Hapus A226	Dcnhzyjw	Aaron Foster	+62 86166459039
Ubah Salin Hapus A392	Wlhasoqb	Alexis Bennett	+62 86632647076
Ubah Salin Hapus A677	Qlispndw	Alexis Bennett	+62 81563318494
Ubah Salin Hapus B183	Qtdclcpq	Amanda Kusuma	+62 87183864821
Ubah Salin Hapus B207	Nstaokfq	Aaron Foster	+62 82142088693
Ubah Salin Hapus B865	Kttjydee	Adrian Tan	+62 81725819240

Penjelasan Penerapan

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pencarian pada “user_travel_tour” dengan kondisi nama user berawalan dengan huruf ‘a’ menggunakan **Select * From Where**

LIKE ‘a%’. Maka, akan ditampilkan semua data ditabel “user_travel_tour” dengan nama user berhuruf awalan A saja.

D.2 Not Like

Not Like merupakan kebalikan dari *like* yang digunakan untuk mencari nilai kolom dengan suatu pola yang tidak sesuai dengan kriteria *like*. Pada operator ini terdapat penggunaan tanda persen (%) untuk mewakili satu atau lebih kecocokan karakter.

• Query

```
1 SELECT * FROM user_travel_tour WHERE NAMA_USER NOT LIKE 'a%';
```

• Output

Menampilkan baris 0 - 24 (total 453. Pencarian dilakukan dalam 0.0006 detik)

SELECT * FROM user_travel_tour WHERE NAMA_USER NOT LIKE 'a%';

Profil [Edit dikotak] [Ubah] [Jelaskan SQL] [Buat kode PHP] [Segarkan]

1 > >> ☐ Tampilkan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini | Sort by key: Tidak ada

Extra options

	ID_USER	PASS_USER	NAMA_USER	NO_TELP_USER
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	A102	Rojjog	Gavin Turner	+62 81156116169
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	A108	Nuwjdwrm	Nora Carter	+62 85912161215
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	A112	Epqafcsq	Yuni Suzuki	+62 87297044856
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	A116	Jruciswn	Mackenzie Davis	+62 89390837925
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	A119	Zchggygd	Teguh Susilo	+62 87095969514
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	A140	Czdmegnz	Victoria Davis	+62 89431266122
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	A149	Kccvackm	Qodir Nugroho	+62 81620710395

• Penjelasan Penerapan

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pencarian pada “user_travel_tour” dengan kondisi nama user yang mengecualikan nama berawalan dengan huruf ‘a’ menggunakan **Select * From Where NOT LIKE ‘a%’**. Maka, akan ditampilkan semua data ditabel “user_travel_tour” dengan nama user berhuruf awalan kecuali A.

E. In Dan Not In

E.1 In

In (di dalam) merupakan *query* yang digunakan untuk mencocokkan nilai dari suatu kolom dengan nilai-nilai yang terdapat dalam suatu daftar arau list. Dengan menggunakan *query* “*In*”, dapat membuat query yang lebih fleksibel dan spesifik.

• Query

Run SQL query/queries on table travel_tour-10.daerah_hotel_travel_tour:

```
1 SELECT * FROM daerah_hotel_travel_tour WHERE NAMA_DAERAH IN ('Surabaya', 'Bandung');
```

• Output

Menampilkan baris 0 - 15 (total 16). Pencarian dilakukan dalam 0,0005 detik.

```
SELECT * FROM "daerah_hotel_travel_tour" WHERE NAMA_DAERAH IN ('Surabaya', 'Bandung');
```

Profil [Edit dikotak] [Ubah] [Jelaskan SQL] [Buat kode PHP] [Segarkan]

Tampilkan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini | Sort by key: Tidak ada

	ID_DAERAH	NAMA_DAERAH
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	G135	Bandung
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	G160	Bandung
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	G166	Surabaya
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	G198	Bandung
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	G211	Surabaya
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	G248	Bandung

• Penjelasan Penerapan

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pencarian pada “daerah_hotel_travel_tour” dengan kondisi nama daerah yang berada di dalam rentang ‘Surabaya’ dan ‘Bandung’ menggunakan **Select * From Where IN** . Maka, akan ditampilkan semua data di tabel “daerah_hotel_travel_tour” dengan nama daerah yang berada di ‘Surabaya’ dan ‘bandung’ saja.

E.2 Not In

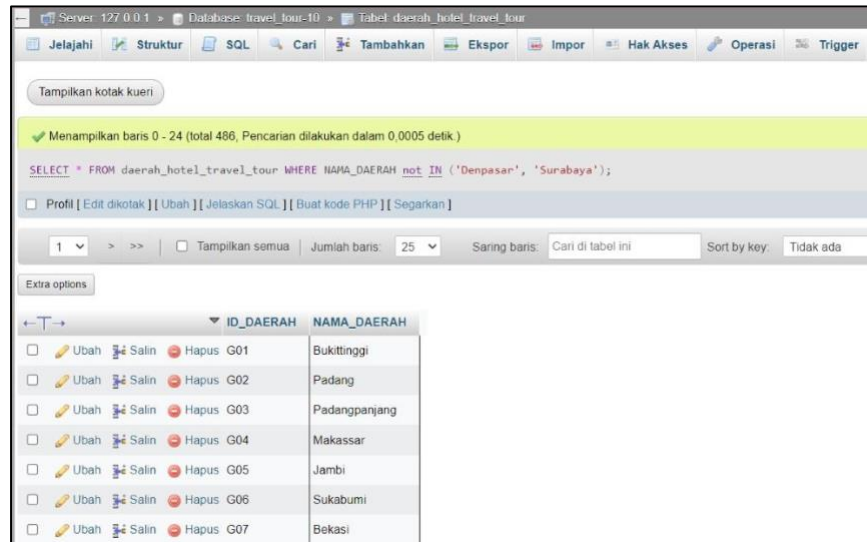
Not In (tidak di dalam) merupakan *query* yang keterbalikan dengan *In* digunakan untuk mencocokkan nilai dari suatu kolom dengan nilai-nilai yang tidak terdapat dalam suatu daftar arau list.

• Query

Run SQL query/queries on table travel_tour-10.daerah_hotel_travel_tour:

```
1 SELECT * FROM daerah_hotel_travel_tour WHERE NAMA_DAERAH not IN ('Denpasar', 'Surabaya');
```

• Output



- **Penjelasan Penerapan**

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pencarian pada “daerah_hotel_travel_tour” dengan kondisi nama daerah yang tidak berada di dalam rentang ‘Surabaya’ dan ‘Bandung’ menggunakan **Select * From Where NOT IN** . Maka, akan ditampilkan semua data di tabel “daerah_hotel_travel_tour” dengan nama daerah yang tidak berada di ‘Surabaya’ dan ‘Bandung’ saja atau selain dari daerah tersebut.

F. Order By

Order By merupakan suatu *query* yang digunakan untuk mengurutkan hasil *query* berdasarkan satu atau lebih kolom. *Query* ini dapat menyusun data dalam urutan tertentu, baik secara naik (ascending) ataupun turun (descending).

- **Query**



- **Output**

ID_DAERAH	NAMA_DAERAH
G178	Ambon
G374	Ambon
G396	Ambon
G355	Ambon
G249	Ambon
G19	Ambon
G57	Ambon
G305	Balikpapan

- **Penjelasan Penerapan**

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pengurutan pada “daerah_hotel_travel_tour” dengan kondisi nama daerah yang diurutkan secara *ascending* menggunakan **Select * From Where ORDER BY**. Maka, akan ditampilkan semua data di tabel “daerah_hotel_travel_tour” dengan nama daerah datanya berurutan dari huruf A hingga Z.

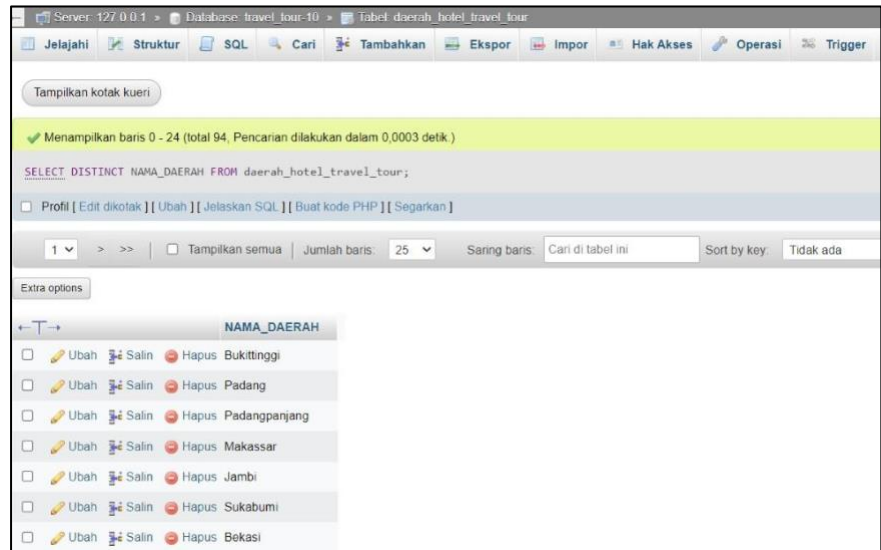
G. Select Distinct

Select Distinct merupakan query yang digunakan untuk menghilangkan nilai ganda pada suatu tampilan, dengan cara mengambil nilai unik dari suatu kolom. Jadi, hanya menampilkan nilai yang berbeda dari kolom yang ditentukan.

- **Query**

```
1 SELECT DISTINCT NAMA_DAERAH FROM daerah_hotel_travel_tour;
```

- **Output**



• Penjelasan Penerapan

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pengurutan pada “daerah_hotel_travel_tour” dengan kondisi nama daerah yang tidak ada duplikasinya menggunakan **Select DISTINCT From**. Maka, akan ditampilkan semua data di tabel “daerah_hotel_travel_tour” dengan nama daerah yang tidak terduplikasi.

H. Group By

Group By merupakan suatu *query* yang digunakan untuk mengelompokkan baris berdasarkan nilai kelompok tertentu.

• Query

Berdasarkan nama paket



• Output

Server: 127.0.0.1 » Database: travel_tour-10 » Tabel: paket_travel_tour

Tampilkan kotak kueri

Menampilkan baris 0 - 4 (total 5, Pencarian dilakukan dalam 0,0008 detik)

SELECT COUNT(ID_PAKET), NAMA_PAKET FROM paket_travel_tour GROUP BY NAMA_PAKET;

Profil | Edit kotak | Ubah | Jelaskan SQL | Buat kode PHP | Segarkan

☐ Tampilkan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini

Extra options

COUNT(ID_PAKET)	NAMA_PAKET
34	Golden
89	Platinum
150	Silver
167	Stone
60	Titanium

- **Penjelasan Penerapan**

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pencarian pada “paket_travel_tour” dengan kondisi nama paket yang telah dikelompokkan menggunakan **Select Count From Group By**. Maka, akan ditampilkan semua data di tabel “paket_travel_tour” dengan nama paket yang telah dikelompokkan.

- **Query**

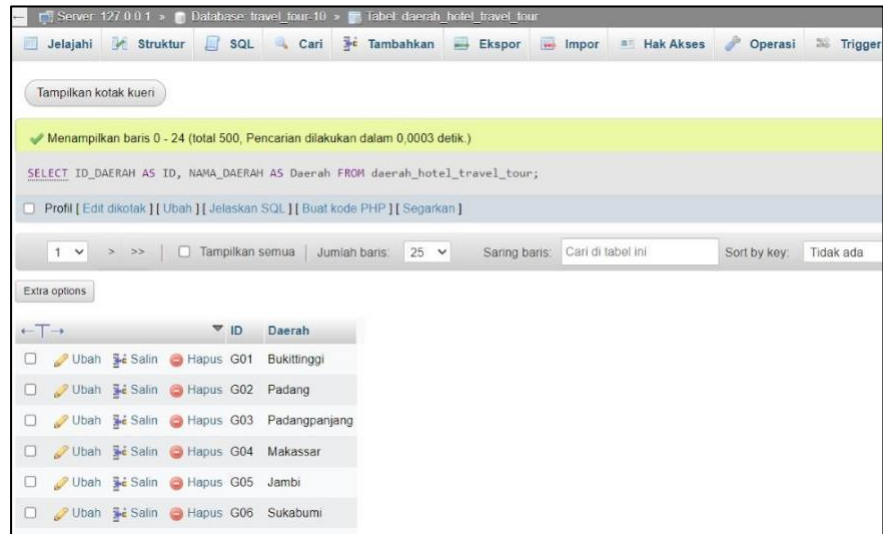
Mengubah ID_Daerah menjadi ID dan Nama_Daerah menjadi Daerah

Server: 127.0.0.1 » Database: travel_tour-10 » Tabel: daerah_hotel_travel_tour

Run SQL query/queries on table travel_tour-10.daerah_hotel_travel_tour:

```
1 SELECT ID_DAERAH AS ID, NAMA_DAERAH AS Daerah FROM daerah_hotel_travel_tour;
```

- **Output**



- **Penjelasan Penerapan**

Pada penerapan ini dilakukan proses pemilihan, penampilan, dan pencarian pada “daerah_hotel_travel_tour” dengan kondisi nama kolom ID_Daerah telah diubah menjadi ID dan nama kolom Nama_Daerah diubah menjadi Daerah menggunakan **Select As From**. Maka, akan ditampilkan semua data di tabel “daerah_hotel_travel_tour” dengan nama-nama kolom yang telah diubah.

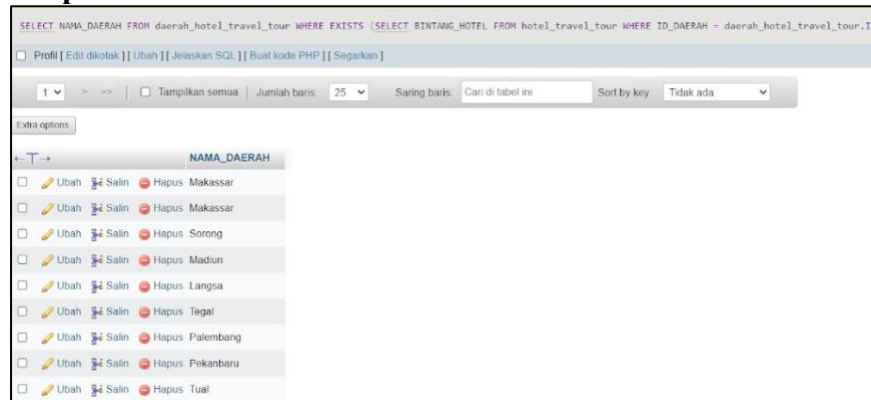
I. Exist dan Not Exist

Exist dan Not Exist merupakan suatu query yang digunakan untuk memeriksa keberadaan dari suatu obyek pada database apakah subquery menghasilkan setidaknya satu baris data atau tidak.

- **Query**



- **Output**



NAMA_DAERAH
Makassar
Makassar
Sorong
Madun
Langsa
Tegal
Palembang
Pekanbaru
Tual

- **Penjelasan Penerapan**

Pada penerapan ini dilakukan proses pemilihan kolom pada “daerah_hotel_travel_tour” dengan kondisi satu baris memenuhi kondisi yang ditentukan subquery, di mana subquery ini mengambil kolom ‘Bintang_Hotel’ dari tabel ‘Hotel_travel_tour’. Subquery hanya akan mengembalikan hasil jika terdapat satu baris dengan ‘ID_daerah’ yang sama dengan ‘Daerah_hotel_travel_tour.ID_Daerah’ dan memiliki nilai ‘Bintang_hotel’ kurang dari 4 menggunakan **Select From WHERE Exists (Select From WHERE Condition)**.

J. Having

HAVING dalam SQL dipakai ketika kata kunci WHERE tidak dapat digunakan dengan fungsi agregat. HAVING digunakan untuk mengatur kondisi bagi GROUP BY pada kelompok hasil agregat, maka kelompok yang memenuhi HAVING saja yang akan dihasilkan.

- **Query**



```

SELECT COUNT(BINTANG_HOTEL), NAMA_HOTEL FROM hotel_travel_tour
GROUP BY NAMA_HOTEL HAVING COUNT(BINTANG_HOTEL) > 4

```

- **Output**

Menampilkan baris 0 - 4 (total 5, Pencarian dilakukan dalam 0,0013 detik.)

`SELECT COUNT(BINTANG_HOTEL), NAMA_HOTEL FROM hotel_travel_tour GROUP BY NAMA_HOTEL HAVING COUNT(BINTANG_HOTEL) > 4`

☐ Profil ☐ Edit dikotak ☐ Ubah ☐ Jelaskan SQL ☐ Buat kode PHP ☐ Segarkan

☐ Tampilkan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini

Extra options

COUNT(BINTANG_HOTEL)	NAMA_HOTEL
150	Emerald Ho
167	Grand Orch
60	Harmoni He
89	Indah Pala
34	Paradise H

- **Penjelasan Penerapan**

Penerapan kode ini digunakan untuk menghitung bintang hotel yang memiliki Bintang > 4 dengan mengelompokkan berdasarkan nama_hotel. `SELECT COUNT(BINTANG_HOTEL), NAMA_HOTEL FROM hotel_travel_tour` digunakan untuk mengambil 2 kolom yakni bintang_hotel dan nama_hotel dalam tabel bernama hotel_travel_tour yang kemudian di kelompokkan berdasarkan nama_hotel dalam subquery `GRUUP BY NAMA_HOTEL` dengan klausul `HAVING COUNT(BINTANG_HOTEL) > 4` untuk membuat kelompok yang memenuhi kondisi bintang_hotel > 4

K. Union

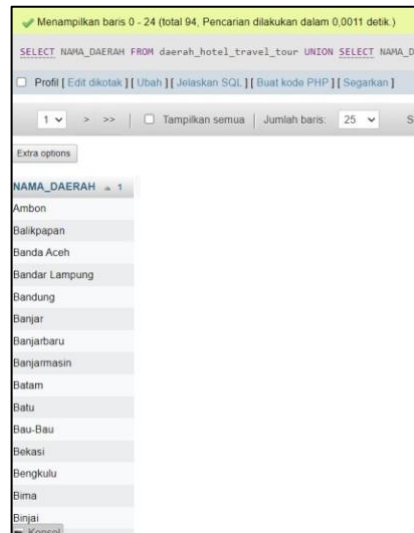
UNION dalam sql digunakan untuk menggabungkan hasil dari dua atau lebih pernyataan SELECT dengan syarat bahwa pernyataan SELECT dalam UNION harus memiliki kolom dengan jumlah, tipe data, dan urutan yang sama.

- **Query**

Run SQL query/queries on table travel_tour-10.daerah_hotel_travel_tour:

```
1. SELECT NAMA_DAERAH FROM daerah_hotel_travel_tour UNION SELECT NAMA_DAERAH FROM daerah_hotel_travel_tour ORDER BY NAMA_DAERAH;
```

• Output



• Penjelasan Penerapan

Dalam query ini UNION digunakan untuk menggabungkan hasil dari dua pernyataan SELECT yang serupa dengan pernyataan pertama untuk mengambil kolom NAMA_DAERAH dari tabel daerah_hotel_travel_tour dan pernyataan select kedua yang memiliki pernyataan yang sama dengan pernyataan sebelumnya, setelah itu hasil kedua pernyataan SELECT digabungkan dengan menggunakan ORDER BY untuk Menyusun hasil alfabetis berdasarkan kolom NAMA_DAERAH.

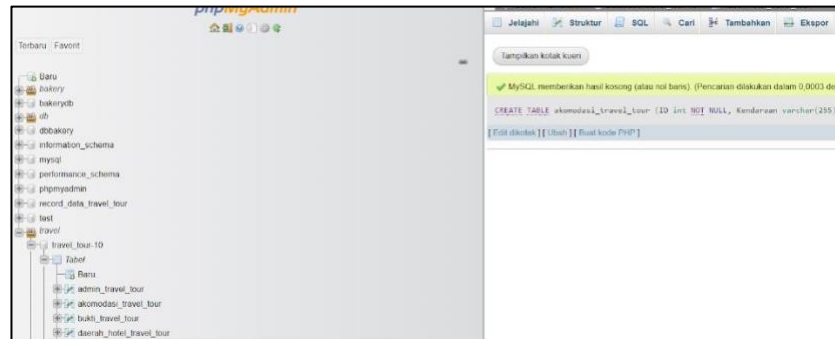
L. View

View merupakan hasil sebuah query yang disimpan sebagai objek virtual atau tabel virtual dalam basis data yang terdiri dari baris dan kolom seperti tabel sebenarnya.

• Query



- **Output**



- **Penjelasan Penerapan**

Query diatas digunakan untuk membuat tabel virtual dengan nama akomodasi_travel_tour dengan query CREATE_TABELE akomodasi_travel_tour yang didalamnya terdapat kolom-kolom berupa ID dengan tipe data integer yang tidak diperbolehkan kosong, Kendaraan dengan tipe data vaerchar(255) yang dapat menyimpan 255 karakter dan tidak boleh kosong, serta Waktu dengan tipe data integer namun diperbolehkan kosong dengan menggunakan query (ID int NOT NULL, Kendaraan varchar(255) NOT NUL, Waktu(int);.

1.6.3 Operasi Join

Operasi Join dalam sql digunakan untuk menghasilkan Kumpulan record dengan menggabungkan dua baris dari dua atau lebih tabel berdasarkan suatu kondisi yang sesuai.

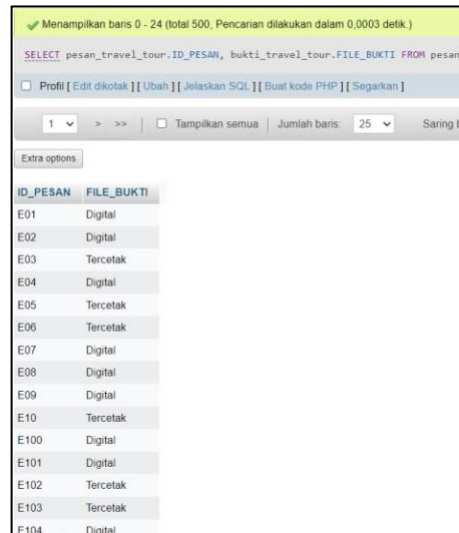
A. Inner Join

Inner Join merupakan operasi penggabungan yang menggabungkan baris dari dua tabel berdasarkan kondisi yang telah ditentukan.

- **Query**



- **Output**



The screenshot shows a database query result in a web interface. At the top, a green status bar indicates 'Menampilkan baris 0 - 24 (total 500, Pencarian dilakukan dalam 0.0003 detik.)'. Below it, the SQL query is displayed: `SELECT pesan_travel_tour.ID_PESAN, bukti_travel_tour.FILE_BUKTI FROM pesan_travel_tour INNER JOIN bukti_travel_tour ON pesan_travel_tour.ID_BUKTI = bukti_travel_tour.ID_BUKTI`. The interface includes buttons for 'Profil', 'Edit dikotak', 'Ubah', 'Jelaskan SQL', 'Buat kode PHP', and 'Segarkan'. Below the query, there are controls for page navigation (1, >, >>), a checkbox for 'Tampilkan semua', 'Jumlah baris' set to 25, and a 'Saring' button. The main area displays a table with two columns: 'ID_PESAN' and 'FILE_BUKTI'. The table contains 14 rows of data, alternating between 'Digital' and 'Tercetak' file types.

ID_PESAN	FILE_BUKTI
E01	Digital
E02	Digital
E03	Tercetak
E04	Digital
E05	Tercetak
E06	Tercetak
E07	Digital
E08	Digital
E09	Digital
E10	Tercetak
E100	Digital
E101	Digital
E102	Tercetak
E103	Tercetak
E104	Digital

- **Penjelasan Penerapan**

Query diatas digunakan untuk menggabungkan baris dari tabel pesan dengan baris dari tabel bukti berdasarkan kolom ID_BUKTI yang sama, query `SELECT pesan_travel_tour, ID_PESAN, bukti_travel_tour.FILE_BUKTI` digunakan untuk mengambil kolom ID_PESAN dalam tabel pesan_travel_tour dan kolom FILE_BUKTI dari tabel bukti_travel_tour dengan data dari tabel pesan yang ada di query `FROM pesan_travel_tour` yang kemudian digabungkan menggunakan baris tersebut menggunakan query `INNER JOIN bukti_travel_tour ON pesan_travel_tour.ID_BUKTI = bukti_travel_tour.ID_BUKTI`;

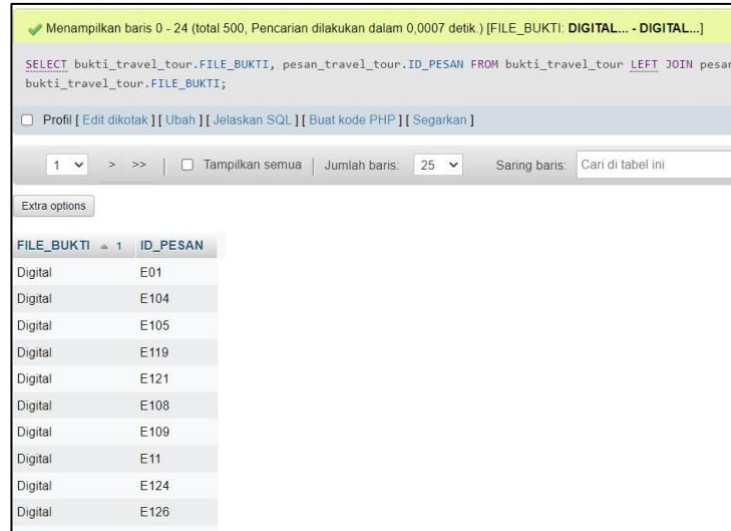
B. Left Outer Join

Left Outer Join merupakan operasi penggabungan dalam sql yang menggabungkan semua baris dari tabel kiri dan baris yang cocok dari tabel sebelah kanan.

- **Query**



- **Output**



- **Penjelasan Penerapan**

Query tersebut digunakan untuk mencakup seluruh FILE_BUKTI dari tabel bukti_travel_tour jika ada ID_PESAN yang sesuai dari tabel pesan_travel_tour query `SELECT bukti_travel_tour.FILE_BUKTI pesan_travel_tour .ID_PESAN`, digunakan untuk mengambil kolom FILE_BUKTI dari tabel bukti_travel_tour dan kolom ID_PESAN dalam tabel pesan_travel_tour dengan mengambil data dari bukti_travel_tour kemudian menggabungkan baris berdasarkan kesamaan kolom ID_PESAN dengan subquery `LEFT JOIN pesan_travel_tour ON bukti_travel_tour.ID_BUKTI = pesan_travel_tour. ID_BUKTI` yang kemudian diurutkan berdasarkan `ORDER BY bukti_travel_tour.FILE_BUKTI;`

C. Right Outer Join

Left Outer Join merupakan operasi penggabungan dalam sql yang menggabungkan semua baris dari tabel kanan dan baris yang cocok dari tabel sebelah kiri.

- **Query**



- **Output**

The screenshot shows the output of the SQL query in a database application. The table has 3 columns: ID_PESAN, NAMA_PAKET, and KETERANGAN_PAKET. The data is as follows:

ID_PESAN	NAMA_PAKET	KETERANGAN_PAKET
E01	Stone	Tur Kota Dasar
E02	Silver	Destinasi Wisata Populer
E03	Golden	Pengalaman Premium dan Keindahan Alam
E04	Platinum	Destinasi Mewah dan Eksklusif
E05	Titanium	Pengalaman Mewah dan Pribadi
E06	Silver	Destinasi Wisata Populer
E07	Golden	Pengalaman Premium dan Keindahan Alam
E08	Silver	Destinasi Wisata Populer
E09	Stone	Tur Kota Dasar
E10	Platinum	Destinasi Mewah dan Eksklusif
E100	Platinum	Destinasi Mewah dan Eksklusif
E101	Silver	Destinasi Wisata Populer
E102	Stone	Tur Kota Dasar
E103	Golden	Pengalaman Premium dan Keindahan Alam

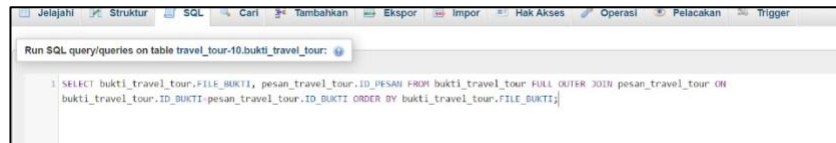
- **Penjelasan Penerapan**

Query diatas digunakan untuk menggabungkan baris dari tabel pesan dengan baris dari tabel paket berdasarkan kolom ID_PAKET yang sama, query SELECT pesan_travel_tour.ID_PESAN, paket_travel_tour.NAMA_PAKET digunakan untuk mengambil kolom ID_PESAN dalam tabel pesan_travel_tour dan kolom NAMA_PAKET dari tabel paket_travel_tour dengan data dari tabel pesan yang ada di subquery FROM pesan_travel_tour yang kemudian digabungkan menggunakan baris tersebut menggunakan subquery RIGHT JOIN paket_travel_tour ON pesan_travel_tour.ID_PAKET = paket_travel_tour.ID_PAKET yang kemudian mengurutkan hasil query tersebut berdasarkan ID_PESAN dengan subquery ORDER BY pesan_travel_tour.ID_PESAN;

D. Full Outer Join

FULL OUTER JOIN merupakan operasi penggabungan yang menggabungkan semua baris dari kedua tabel sebelah kiri dan sebelah kanan. •

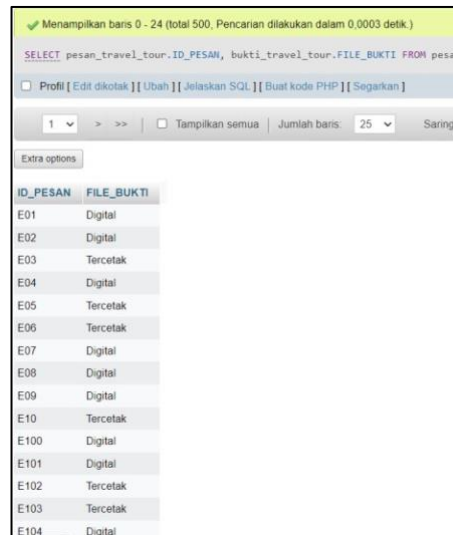
Query



```
Run SQL query/queries on table travel_tour-10.bukti_travel_tour:

1 SELECT bukti_travel_tour.FILE_BUKTI, pesan_travel_tour.ID_PESAN FROM bukti_travel_tour FULL OUTER JOIN pesan_travel_tour ON
bukti_travel_tour.ID_BUKTI=pesan_travel_tour.ID_BUKTI ORDER BY bukti_travel_tour.FILE_BUKTI;
```

• Output



Menampilkan baris 0 - 24 (total 500, Pencarian dilakukan dalam 0.0003 detik.)

```
SELECT pesan_travel_tour.ID_PESAN, bukti_travel_tour.FILE_BUKTI FROM pesan
```

☐ Profil ☐ Edit kotak ☐ Ubah ☐ Jelaskan SQL ☐ Buat kode PHP ☐ Segarkan

1 > >> ☐ Tampilkan semua Jumlah baris: 25 Saring t

Extra options

ID_PESAN	FILE_BUKTI
E01	Digital
E02	Digital
E03	Tercetak
E04	Digital
E05	Tercetak
E06	Tercetak
E07	Digital
E08	Digital
E09	Digital
E10	Tercetak
E100	Digital
E101	Digital
E102	Tercetak
E103	Tercetak
E104	Digital

• Penjelasan Penerapan

Query tersebut digunakan untuk mengambil semua baris dari ID_PESAN dan FILE_BUKTI SELECT bukti_travel_tour.FILE_BUKTI pesan_travel_tour .ID_PESAN, digunakan untuk mengambil kolom FILE_BUKTI dari tabel bukti_travel_tour dan kolom ID_PESAN dalam tabel pesan_travel_tour dengan mengambil data dari tabel bukti_travel_tour kemudian dengan subquery LEFT JOIN pesan_travel_tour ON bukti_travel_tour.ID_BUKTI = pesan_travel_tour.ID_BUKTI untuk menggabungkan semua baris dari pesan_travel_tou dan bukti_travel_tour berdasarkan kesamaan ID_PESAN yang kemudian diurutkan berdasarkan ORDER BY bukti_travel_tour.FILE_BUKTI;