

KATA PENGANTAR

Puji syukur selalu kami panjatkan kehadiran Allah Swt yang telah memberikan semua nikmatnya sehingga penulis berhasil menyelesaikan final project yang berjudul “Implementasi dan Manajemen Basis Data Menggunakan MariaDB Pada Studi Kasus Bisnis Laundry Flazz” ini dengan tepat waktu tanpa adanya kendala yang berarti. Tujuan dari penyusunan laporan ini adalah untuk memudahkan para mahasiswa sistem informasi dalam memahami basis data.

Keberhasilan penyusunan laporan ini tentunya bukan atas usaha penulis saja namun ada banyak pihak yang turut membantu dan memberikan dukungan untuk suksesnya penulisan laporan ini. Untuk itu, penulis mengucapkan terima kasih yang sebesar besarnya kepada semua pihak yang telah memberikan dukungan baik secara moril ataupun material sehingga laporan ini berhasil disusun.

Laporan yang ada di hadapan pembaca ini tentu tidak luput dari kekurangan. Selalu ada celah untuk perbaikan. Sehingga, kritik, saran serta masukan dari pembaca sangat kami harapkan dan kami sangat terbuka untuk itu supaya laporan ini semakin sempurna dan lengkap.

Surabaya, 9 Desember 2024

Penyusun

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI.....	ii
DAFTAR TABEL.....	ii
DAFTAR GAMBAR	ii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Tujuan dan Manfaat Proyek	1
1.2.1 Tujuan	1
1.2.2 Manfaat	2
1.3 Ruang Lingkup Proyek.....	2
BAB II PERANCANGAN BASIS DATA	4
2.1 Deskripsi Kebutuhan Sistem	4
2.1.1 Kebutuhan Fungsional	4
2.2 Entity-Relationship Diagram (ERD).....	5
2.2.1 Entitas, Atribut, dan Hubungan Antar-Entitas	5
2.2.2 Kardinalitas dan Domain Atribut.....	7
2.3 Normalisasi.....	10
BAB III IMPLEMENTASI BASIS DATA	14
3.1 Membuat Basis Data	14
3.1.1 Perintah Create Database	14
3.2 Pembuatan Tabel.....	14
3.2.1 Definisi Tabel, Tipe Data, Primary Key, dan Foreign Key	14
3.2.2 Contoh implementasi SQL.....	18
3.3 Manipulasi Data	22
3.3.1 Menambahkan Data (INSERT).....	22
3.3.2 Memodifikasi Data (UPDATE)	24
3.3.3 Menghapus Data (DELETE).....	25
3.4 Query data	26
3.4.1 Penggunaan SELECT dan JOIN.....	26
BAB IV VIEW.....	36
4.1 View	38

4.1.1 Membuat view	38
4.1.2 Select view transaksi.....	38
4.1.3 Mengubah view dengan perintah ALTER	39
4.1.4 Menghapus view dengan perintah DROP.....	39
BAB V KEAMANAN BASIS DATA.....	40
5.1 Pembuatan pengguna.....	40
5.1.1 Perintah CREATE USER	40
5.1.2 Pemberian Hak akses/ Priviledges	40
BAB VI UNDO MANAGEMENT.....	42
BAB VII BACKUP DAN RESTORE	46
7.1 Penjelasan Pentingnya Backup dan Restore.....	46
7.2 Proses backup basis data:	47
7.3 Proses restore basis data.....	50
BAB VIII KESIMPULAN DAN SARAN	54
8.2 Evaluasi implementasi.....	54
8.3 Rekomendasi untuk pengembangan lebih lanjut.....	55
8.4 Saran dan Pesan.....	55
DAFTAR PUSTAKA	56
LAMPIRAN.....	57

DAFTAR TABEL

Tabel 2. 1 Domain Atribut Database	10
Tabel 2. 2 Tabel 1NF	11
Tabel 2. 3 Tabel 2NF	12
Tabel 2. 4 Tabel 3 NF	13

DAFTAR GAMBAR

Gambar 2. 1 Tabel Universal 1	7
Gambar 3. 1 Tampilan Create Database di PhpMyAdmin.....	14
Gambar 3. 2 Create Database dengan SQL	19
Gambar 3. 3 SQL Create Setiap Tabel	21
Gambar 3. 4 SQL Insert Setiap Tabel	24
Gambar 3. 5 SQL Update Isi Tabel	25
Gambar 3. 6 SQL Delete Isi Tabel	25
Gambar 3. 7 SQL dan Bukti Select Setiap Tabel	33
Gambar 3. 8 SQL dan Bukti Join tabel.....	35
Gambar 3. 9 SQL dan Bukti Double Join	36
Gambar 3. 10 SQL dan Bukti Triple Join	36
Gambar 4. 1 SQL dan Bukti View.....	38
Gambar 5. 1 Query Create User	40
Gambar 5. 2 Query Privileges.....	40
Gambar 5. 3 Query Role	41
Gambar 6. 1 SQL dan Bukti Start Transaction	39
Gambar 6. 2 SQL dan Bukti Commit.....	45
Gambar 6. 3 SQL dan Bukti Savepoint	45
Gambar 7. 1 Query Logical Backup	47
Gambar 7. 2 Bukti backup logical database	47
Gambar 7. 3 Query dan Bukti Logical Backup Tabel pilihan	48
Gambar 7. 4 Query dan Bukti Logical Backup Tabel pilihan	48
Gambar 7. 5 Query Physical Backup	49
Gambar 7. 6 Query Restore Database	52

BAB I

PENDAHULUAN

1.1 Latar Belakang

Administrasi basis data merupakan elemen penting dalam pengelolaan informasi di era digital. Namun, masih banyak pelaku usaha kecil dan menengah yang menghadapi kendala dalam mengelola data mereka. Salah satu masalah utama adalah pencatatan data yang dilakukan secara manual, seperti menggunakan buku catatan atau spreadsheet, yang rentan terhadap kesalahan pencatatan, kehilangan data, dan ketidakakuratan dalam analisis. Hal ini dapat menyebabkan hambatan dalam operasional bisnis, seperti keterlambatan layanan, kesalahan dalam penghitungan transaksi, hingga sulitnya melacak data pelanggan.

Laundry Flazz, sebagai salah satu bisnis jasa laundry, juga menghadapi tantangan serupa. Ketergantungan pada metode pencatatan manual mengakibatkan sulitnya mengelola data transaksi yang terus meningkat seiring bertambahnya jumlah pelanggan. Kondisi ini tidak hanya menyulitkan operasional harian, tetapi juga membatasi kemampuan bisnis untuk berkembang dan bersaing di tengah persaingan industri laundry yang semakin ketat.

MariaDB hadir sebagai solusi modern untuk mengatasi permasalahan tersebut. Sebagai salah satu sistem manajemen basis data relasional open-source yang andal, MariaDB menawarkan berbagai fitur, seperti kemudahan dalam pengelolaan data, kemampuan transaksi yang terjamin, dan dukungan terhadap keamanan data [1]. Dengan implementasi MariaDB, Laundry Flazz diharapkan dapat mengelola data secara lebih efisien, terstruktur, dan aman, sehingga mampu meningkatkan kualitas layanan serta mendukung pengambilan keputusan berbasis data.

Melalui proyek ini, implementasi dan manajemen basis data menggunakan MariaDB pada Laundry Flazz diharapkan dapat menjadi contoh penerapan administrasi basis data yang efektif, sekaligus mengatasi berbagai kendala yang selama ini dihadapi oleh bisnis jasa laundry.

1.2 Tujuan dan Manfaat Proyek

1.2.1 Tujuan

Berdasarkan rumusan masalah yang telah dikemukakan diatas, maka tujuan dari penelitian ini adalah :

- 1) Mengimplementasikan basis data menggunakan MariaDB yang dirancang khusus untuk mendukung kebutuhan bisnis Laundry Flazz.
- 2) Mengelola data pelanggan, transaksi, dan inventaris secara sistematis dan terintegrasi.
- 3) Memberikan pemahaman praktis mengenai administrasi basis data menggunakan MariaDB, termasuk keamanan dan pemulihan data.
- 4) Membuktikan potensi MariaDB sebagai solusi manajemen basis data yang handal, efisien, dan terjangkau bagi usaha kecil dan menengah.

1.2.2 Manfaat

Adapun manfaat yang diharapkan dari pelaksanaan penelitian ini adalah sebagai berikut:

1. Memberikan solusi praktis bagi Laundry Flazz untuk meningkatkan efisiensi operasional dan manajemen data.
2. Meningkatkan akurasi pencatatan dan pengelolaan data melalui basis data yang terstruktur.
3. Menyediakan laporan yang mendukung pengambilan keputusan yang lebih tepat dan berbasis data.
4. Memberikan wawasan kepada pembaca atau pengguna akhir tentang praktik terbaik dalam implementasi basis data.

1.3 Ruang Lingkup Proyek

1. Perancangan Basis Data:

- Analisis kebutuhan sistem untuk bisnis laundry Flazz
- Pembuatan Entity-Relationship Diagram (ERD) yang menggambarkan entitas, atribut, dan relasi pada basis data
- Normalisasi basis data hingga mencapai bentuk Third Normal Form (3NF)

2. Implementasi Basis Data:

- Pembuatan database MariaDB untuk mengelola data bisnis Flazz
- Pembuatan tabel-tabel sesuai rancangan ERD dengan definisi tipe data, primary key, dan foreign key yang tepat
- Implementasi operasi CRUD (Create, Read, Update, Delete) untuk memanipulasi data pada tabel-tabel

3. Manipulasi Data:

- Pengisian data awal (INSERT) pada tabel-tabel basis data
- Modifikasi data (UPDATE) sesuai kebutuhan bisnis
- Penghapusan data (DELETE) jika diperlukan
- Penggunaan SELECT dan JOIN untuk melakukan query data

4. Penggunaan View:

- Pembuatan view untuk mempermudah akses data tertentu
- Manipulasi data melalui view

5. Keamanan Basis Data:

- Pembuatan pengguna (user) basis data
- Pemberian hak akses (privileges) dan peran (roles) sesuai kebutuhan

6. Undo Management:

- Implementasi manajemen transaksi menggunakan START TRANSACTION, ROLLBACK, dan COMMIT
- Penggunaan SAVEPOINT untuk memungkinkan pembatalan sebagian perubahan

7. Backup dan Restore:

- Proses backup basis data secara berkala
- Implementasi restore basis data jika diperlukan

8. Evaluasi dan Rekomendasi:

- Ringkasan hasil implementasi dan manajemen basis data MariaDB pada bisnis laundry Flazz
- Evaluasi kinerja dan efektivitas solusi yang diterapkan
- Rekomendasi untuk pengembangan lebih lanjut

BAB II

PERANCANGAN BASIS DATA

2.1 Deskripsi Kebutuhan Sistem

Studi kasus pada bisnis Laundry Flazz membutuhkan sistem basis data untuk mendukung proses operasional yang lebih efektif dan efisien. Berikut adalah deskripsi kebutuhan sistem yang mencakup aspek fungsional dan non-fungsional:

2.1.1 Kebutuhan Fungsional

- 1) Manajemen Pelanggan
 - a. Sistem harus mampu mencatat data pelanggan, seperti nama, alamat, nomor telepon, dan email.
 - b. Data pelanggan harus dapat diubah, dihapus, dan dicari dengan cepat.
- 2) Manajemen Transaksi
 - a. Sistem harus mencatat setiap transaksi, termasuk detail seperti tanggal transaksi, jenis layanan (cuci kering, setrika, atau paket lengkap), jumlah item, dan biaya total.
 - b. Memungkinkan pencatatan status transaksi (misalnya: diterima, dalam proses, selesai, diambil pelanggan).
- 3) Manajemen Inventaris
 - a. Sistem harus mencatat inventaris bahan pendukung laundry seperti deterjen, pelembut, plastik, dan lainnya.\
 - b. Memantau ketersediaan stok inventaris dan memberikan peringatan saat stok menipis.
- 4) Manajemen Laporan
 - a. Sistem harus mampu menghasilkan laporan seperti:
 - Laporan harian, mingguan, atau bulanan tentang transaksi.
 - Laporan pendapatan berdasarkan layanan.
 - Laporan pelanggan berdasarkan frekuensi transaksi.
- 5) Keamanan Data
 - o Sistem harus memastikan data pelanggan dan transaksi hanya dapat diakses oleh pihak yang berwenang melalui pemberian hak akses.
- 6) Backup dan Restore
 - o Sistem harus memiliki fitur untuk mencadangkan data dan mengembalikannya jika terjadi kegagalan sistem.

2.1.2 Kebutuhan Non-Fungsional

1. Kinerja Sistem
 - i. Sistem harus dapat menangani setidaknya 1.000 transaksi per bulan tanpa mengalami penurunan performa.
2. Kemudahan Penggunaan
 - i. Sistem harus memiliki struktur data yang sederhana dan mendukung query yang mudah dimengerti oleh pengguna.
3. Skalabilitas
 - i. Sistem harus dapat dikembangkan untuk mendukung pertumbuhan data pelanggan dan transaksi dalam jangka panjang.
4. Keandalan
 - i. Sistem harus memastikan data tersimpan dengan aman, mencegah duplikasi, dan memberikan kemampuan rollback pada transaksi yang belum selesai.
5. Kompatibilitas
 - i. Basis data harus kompatibel dengan aplikasi front-end yang mungkin akan dikembangkan di masa depan untuk mendukung operasional Laundry Flazz.
6. Waktu Respons
 - i. Query terhadap data harus memberikan waktu respons kurang dari 2 detik untuk pencarian umum.

2.2 Entity-Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh Sistem analis dalam tahap analisis persyaratan proyek pengembangan sistem. Selain itu, Entity Relationship Diagram (ERD) adalah salah satu metode pemodelan basis data yang digunakan untuk menghasilkan skema konseptual untuk jenis atau model data semantik sistem [2].

2.2.1 Entitas, Atribut, dan Hubungan Antar-Entitas

Pada ERD dibawah ini terdapat 8 entity antara lain :

- Karyawan

Dalam entity karyawan terdapat atribut nama_karyawan, jabatan, alamat, email, gaji, no_telp, tanggal_bergabung serta terdapat primary key atribut yaitu id_karyawan.

- Transaksi

Dalam entity Transaksi terdapat atribut id_pelanggan, id_karyawan, status, email, tanggal_masuk, tanggal_selesai, total_harga, metode_pembayaran serta terdapat primary key atribut yaitu id_transaksi.

- Detail Transaksi

Dalam entity Detail_Transaksi terdapat atribut id_transaksi, id_layanan, subtotal, jumlah serta terdapat primary key atribut yaitu id_detail.

- Admin

Dalam entity Admin terdapat atribut username, password, nama_lengkap, email, tanggal_dibuat, no_telp serta terdapat primary key attribut yaitu id_admin.

- Barang

Dalam entity Barang terdapat atribut nama_barang, kategori, stok, satuan, harga_beli, tanggal_update serta terdapat primary key attribut yaitu id_update.

- Pelanggan

Dalam entity Pelanggan terdapat atribut nama_pelanggan, email, alamat, no_telp, tanggal_bergabung, poin serta terdapat primary key attribut yaitu id_pelanggan.

- Layanan

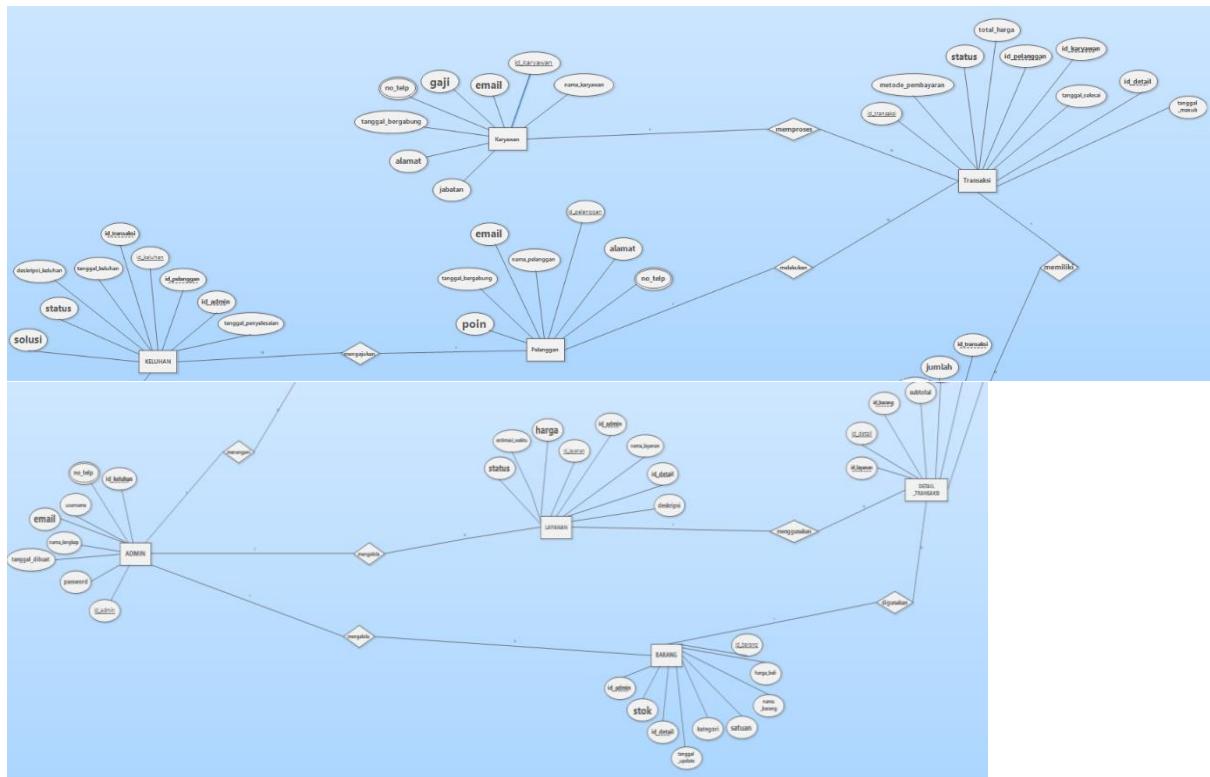
Dalam entity Layanan terdapat atribut nama_layanan, deskripsi, status, harga, estimasi_waktu, serta terdapat primary key attribut yaitu id_layanan.

- Keluhan

Dalam entity Keluhan terdapat atribut id_pelanggan, id_transaksi, status, tanggal_keluhan, deskripsi_keluhan, solusi, tanggal_penyyelesaian serta terdapat primary key attribut yaitu id_keluhan.

➤ Relasi :

- Entity karyawan memiliki relasi One to Many pada entity transaksi.
- Entity keluhan memiliki relasi Many to One pada entity pelanggan.
- Entity pelanggan memiliki relasi One to Many pada entity transaksi.
- Entity detail_transaksi memiliki relasi Many to One pada entity transaksi.
- Entity detail_transaksi memiliki relasi Many to One pada entity barang.
- Entity detail_transaksi memiliki relasi Many to One pada entity layanan.
- Entity barang memiliki relasi Many to One pada entity admin.
- Entity layanan memiliki relasi Many to One pada entity Admin.



Gambar 2. 1 Tabel Universal 1

2.2.2 Kardinalitas dan Domain Atribut

1. Kardinalitas Antar Entitas

Kardinalitas Relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain [2]. Dan berikut dibawah ini merupakan hasil kardinalitas dari penelitian.

a. Relasi Admin:

- Admin mengelola Layanan: 1..n (Satu admin dapat mengelola banyak layanan, setiap layanan hanya dikelola oleh satu admin).
- Admin menangani Keluhan: 1..n (Satu admin dapat menangani banyak keluhan, setiap keluhan hanya ditangani oleh satu admin).

b. Relasi Pelanggan:

- Pelanggan membuat Transaksi: 1..n (Satu pelanggan dapat memiliki banyak transaksi, tetapi satu transaksi hanya dimiliki oleh satu pelanggan).
- Pelanggan dapat memiliki Poin: 1..1 (Setiap pelanggan memiliki poin yang terakumulasi secara tunggal).

c. Relasi Transaksi:

- Transaksi memiliki Detail Transaksi: 1..n (Satu transaksi memiliki banyak detail transaksi, tetapi satu detail transaksi hanya untuk satu transaksi).

- Transaksi menggunakan Barang: n..n (Satu transaksi dapat menggunakan banyak barang, dan satu barang dapat digunakan di banyak transaksi).
- d. Relasi Layanan:
- Layanan digunakan dalam Detail Transaksi: 1..n (Satu layanan dapat digunakan dalam banyak detail transaksi, tetapi satu detail transaksi hanya berhubungan dengan satu layanan).

2. Domain Atribut

Atribut merupakan sifat-sifat dari sebuah entitas atau tipe relasi. Himpunan nilai yang diperbolehkan untuk satu atau lebih atribut disebut sebagai domain atribut. Ini merupakan aturan atau batasan yang menentukan tipe, format, dan karakteristik data yang dapat disimpan. [3]

Berikut adalah domain atribut untuk setiap entitas dalam ERD penelitian ini:

➤ Entitas Admin

Atribut	Domain
ID_Admin	Integer
Username	String, maks. 20 karakter
Password	String, dienkripsi, maks. 50 karakter
Nama_Lengkap	String, maks. 50 karakter
Email	String, format email
No_Telp	String, angka, maks. 15 digit
Tanggal_Dibuat	Format tanggal (DD/MM/YYYY)
Id_Keluhan	Integer

➤ Entitas Karyawan

Id_karyawan	Integer
nama_karyawan	Varchar(100)
jabatan	Varchar(50)
alamat	text
email	Varchar(100)
gaji	Decimal(10,2)
no_telp	Varchar(15)
tanggal_bergabung	Format tanggal (DD/MM/YYYY)

➤ Entitas Pelanggan

Id_pelanggan	Integer
Nama_pelanggan	Varchar (100)
email	Varchar (100)
alamat	Text
No_telp	Varchar (15)
Tanggal_bergabung	Date
poin	Integer (10)

➤ Entitas Transaksi

id_transaksi	Integer
id_pelanggan	Integer
id_karyawan	Integer
status	Enum
tanggal_masuk	Datetime
tanggal_selesai	Datetime
total_harga	Decimal (10,2)
metode_pembayaran	Enum

➤ Entitas Detail Transaksi

Id_detail	Integer
Id_transaksi	Integer
Id_layanan	Integer
subtotal	Decimal (10,2)
jumlah	Integer

➤ Entitas Layanan

id_layanan	Integer
nama_layanan	Varchar (100)
deskripsi	Text
status	Enum
harga	Decimal (10,2)
estimasi_waktu	Integer

➤ Entitas Barang

id_barang	Integer
nama_barang	Varchar (100)
kategori	Enum
stok	Integer
satuan	Varchar (20)
harga_beli	Decimal (10,2)
tanggal_update	Timestamp

➤ Entitas Keluhan

id_keluhan	Integer
id_pelanggan	Integer
id_transaksi	Integer
status	Enum
tanggal_keluhan	Datetime
deskripsi_keluhan	Text
solusi	Text
tanggal_keluhan	Datetime

Tabel 2. 1 Domain Atribut Database

2.3 Normalisasi

Normalisasi adalah proses yang berkaitan dengan model data relational untuk mengorganisasi himpunan data dengan ketergantungan dan keterkaitan yang tinggi atau erat. Hasil dari proses normalisasi adalah himpunan-himpunan data (table-table) dalam bentuk normal (normal form) [4].

2.3.1 Proses Normalisasi Hingga 3NF (Third Normal Form)

- Tabel Universal

Menggabungkan semua atribut dari entitas tabel universal.

id_karyawan	nama_karyawa	jabatan_karyawa	alamat_karyawa	email_karyawa	gaj	no_lp	tanggal_bergbung_karyawan
-------------	--------------	-----------------	----------------	---------------	-----	-------	---------------------------

id_transaksi	status_transaksi	tanggal_masuk	tanggal_selesai	total_harga	metode_pembayaran
--------------	------------------	---------------	-----------------	-------------	-------------------

subtotal	jumlah
----------	--------

id_admin	username_admin	password_admin	nama_lengkap_admin	email_admin	tanggal_dibuat_admin	no_telp_admin
----------	----------------	----------------	--------------------	-------------	----------------------	---------------

id_barang	nama_barang	kategori	stock	satuan	harga_beli	tanggal_update
-----------	-------------	----------	-------	--------	------------	----------------

id_pelanggan	nama_pelanggan	email_pelanggan	alamat_pelanggan	no_telp_pelanggan	tanggal_bergbung_pelanggan	poin_pelanggan
--------------	----------------	-----------------	------------------	-------------------	----------------------------	----------------

id_layanan	nama_layanan	deskripsi_layanan	status_layanan	harga_layanan	estimasi_waktu
------------	--------------	-------------------	----------------	---------------	----------------

id_keluhan	status_keluhan	tanggal_keluhan	deskripsi_keluhan	solusi_keluhan	tanggal_penyehesian
------------	----------------	-----------------	-------------------	----------------	---------------------

Tabel 2.3.1.1 Tabel Universal

- Normalisasi 1NF

Syarat 1NF adalah setiap kolom hanya menyimpan satu nilai (tidak ada atribut multivalued) dan setiap baris dalam tabel bersifat unik.

Semua atribut sudah memenuhi 1NF karena data dalam tabel universal tidak mengandung nilai yang berulang dalam satu kolom. Tabel tetap sama dengan tabel universal.

id_karyawan	nama_karyawa	jabatan_karyawa	alamat_karyawa	email_karyawa	gaj	no_te	tanggal_bergabung_karyawan
-------------	--------------	-----------------	----------------	---------------	-----	-------	----------------------------

id_transaksi	status_transaksi	tanggal_masuk	tanggal_selesai	total_harga	metode_pembayaran
--------------	------------------	---------------	-----------------	-------------	-------------------

subtotal	jumlah
----------	--------

id_admin	username_admin	password_admin	nama_lengkap_admin	email_admin	tanggal_dibuat_admin	no_telp_admin
----------	----------------	----------------	--------------------	-------------	----------------------	---------------

id_barang	nama_barang	kategori	stock	satuan	harga_beli	tanggal_update
-----------	-------------	----------	-------	--------	------------	----------------

id_pelanggan	nama_pelanggan	email_pelanggan	alamat_pelanggan	no_telp_pelanggan	tanggal_bergabung_pelanggan	poin_pelanggan
--------------	----------------	-----------------	------------------	-------------------	-----------------------------	----------------

id_layanan	nama_layanan	deskripsi_layanan	status_layanan	harga_layanan	estimasi_waktu
------------	--------------	-------------------	----------------	---------------	----------------

id_keluhan	status_keluhan	tanggal_keluhan	deskripsi_keluhan	solusi_keluhan	tanggal_penyehesian
------------	----------------	-----------------	-------------------	----------------	---------------------

Tabel 2. 2 Tabel 1NF

- Normalisasi 2NF

Syarat 2NF adalah tabel sudah memenuhi 1NF dan menghapus ketergantungan parsial, yaitu memindahkan atribut yang bergantung hanya pada sebagian kunci utama ke tabel terpisah.

Tabel karyawan

id_karyawan	nama_karyawan	jabatan_karyawan	alamat_karyawan	email_karyawan	gaji	no_telp	tanggal_bergabung_karyawan
-------------	---------------	------------------	-----------------	----------------	------	---------	----------------------------

Tabel transaksi

id_transaksi	id_pelanggan	id_karyawan	status_transaksi	tanggal_masuk	tanggal_selesai	total_harga	metode_pembayaran
--------------	--------------	-------------	------------------	---------------	-----------------	-------------	-------------------

Tabel detail_transaksi

id_detail	id_transaksi	id_layanan	subtotal	jumlah
-----------	--------------	------------	----------	--------

Tabel admin

id_admin	username_admin	password_admin	nama_length_kap_admin	email_admin	tanggal_dibuat_admin	no_telp_admin
----------	----------------	----------------	-----------------------	-------------	----------------------	---------------

Tabel barang

id_barang	nama_barang	kategori	stok	satuan	harga_beli	tanggal_update
-----------	-------------	----------	------	--------	------------	----------------

Tabel pelanggan

id_pelanggan	nama_pelanggan	email_pelanggan	alamat_pelanggan	no_telp_pelanggan	tanggal_bergabung_pelanggan	poin_pelanggan
--------------	----------------	-----------------	------------------	-------------------	-----------------------------	----------------

Tabel layanan

id_layanan	nama_layanan	deskripsi_layanan	status_layanan	harga_layanan	estimasi_waktu
------------	--------------	-------------------	----------------	---------------	----------------

Tabel keluhan

id_keluhan	id_pelanggan	id_transaksi	status_keluhan	tanggal_keluhan	deskripsi_keluhan	solusi_keluhan	tanggal_penyelesaian
------------	--------------	--------------	----------------	-----------------	-------------------	----------------	----------------------

Tabel 2. 3 Tabel 2NF

- **Tabel 3NF**

Syarat 3NF adalah tabel sudah memenuhi 2NF dan menghapus ketergantungan transitif, yaitu atribut non-pk yang bergantung pada atribut non-pk lainnya.

Tidak ada atribut yang memiliki ketergantungan transitif pada setiap tabel di 2NF. Semua tabel telah memenuhi 3NF.

♥ Tabel karyawan

id_karyawan	nama_karyawa n	jabatan_karyawa n	alamat_karyawa n	email_karyawa n	gaj	no_tlp	tanggal_bergabung_karyawan
-------------	----------------	-------------------	------------------	-----------------	-----	--------	----------------------------

♥ Tabel transaksi

id_transaksi	id_pelanggan	id_karyawan	status_transaksi	tanggal_masuk	tanggal_selesai	total_harga	metode_pembayaran
--------------	--------------	-------------	------------------	---------------	-----------------	-------------	-------------------

♥ Tabel detail_transaksi

id_detail	id_transaksi	id_layanan	subtotal	jumlah
-----------	--------------	------------	----------	--------

♥ Tabel admin

id_admin	username_admin	password_admin	nama_lengkap_admin	email_admin	tanggal_dibuat_admin	no_telp_admin
----------	----------------	----------------	--------------------	-------------	----------------------	---------------

♥ Tabel barang

id_barang	nama_barang	kategori	stock	satuan	harga_beli	tanggal_update
-----------	-------------	----------	-------	--------	------------	----------------

♥ Tabel pelanggan

id_pelanggan	nama_pelanggan	email_pelanggan	alamat_pelanggan	no_telp_pelanggan	tanggal_bergabung_pelanggan	poin_pelanggan
--------------	----------------	-----------------	------------------	-------------------	-----------------------------	----------------

♥ Tabel layanan

id_layanan	nama_layanan	deskripsi_layanan	status_layanan	harga_layanan	estimasi_waktu
------------	--------------	-------------------	----------------	---------------	----------------

♥ Tabel keluhan

id_keluhan	id_pelanggan	id_transaksi	status_keluhan	tanggal_keluhan	deskripsi_keluhan	solusi_keluhan	tanggal_penyyelesaian
------------	--------------	--------------	----------------	-----------------	-------------------	----------------	-----------------------

Tabel 2. 4 Tabel 3 NF

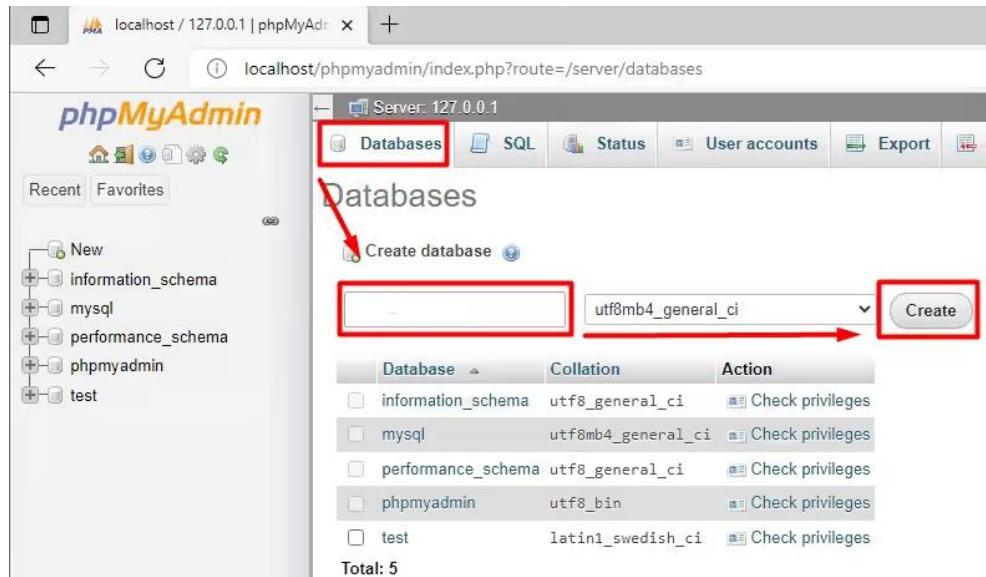
BAB III

IMPLEMENTASI BASIS DATA

3.1 Membuat Basis Data

DDL merupakan sekumpulan perintah SQL yang menggambarkan desain dari database tersebut, selain itu DDL juga digunakan untuk membuat (CREATE), merubah (ALTER) ataupun menghapus (DROP). [5]

3.3.1 Perintah Create Database



Gambar 3. 1 Tampilan Create Database di PhpMyAdmin

Pada gambar, terlihat tampilan antarmuka dari PhpMyAdmin. Disini dapat melihat daftar database yang ada dan melakukan berbagai operasi pada database tersebut. Di bagian "Databases", terdapat opsi "Create database" yang dapat digunakan untuk membuat database baru. Di bagian nama "Database" kami beri nama "bisniskaundry".

Selain itu, juga dapat membuat database baru menggunakan perintah SQL, seperti "CREATE DATABASE nama_database;". Perintah ini dapat dijalankan melalui panel SQL di phpMyAdmin.

3.2 Pembuatan Tabel

3.2.1 Definisi Tabel, Tipe Data, Primary Key, dan Foreign Key

Tabel Admin

Tabel admin di atas mencakup semua kolom yang dibutuhkan untuk menyimpan informasi detail admin pada sistem bisnis laundry. Primary key adalah id_admin yang berfungsi sebagai identitas unik untuk setiap admin. Tidak terdapat foreign key karena tabel admin berdiri sendiri tanpa relasi dengan tabel lain.

- Primary Key
 - id_admin

- Tipe data :
 - id_admin : integer (11), NOT NULL
 - Username : varchar(50), NOT NULL
 - Password : varchar(255), NOT NULL
 - nama_lengkap : varchar(100), NOT NULL
 - Email : varchar(100), NOT NULL
 - no_telp : varchar(15), DEFAULT NULL
 - tanggal_dibuat : timestamp, NOT NULL DEFAULT CURRENT_TIMESTAMP

Tabel Barang

Tabel barang digunakan untuk menyimpan informasi mengenai barang atau bahan-bahan yang digunakan dalam proses layanan laundry. Data barang disimpan berdasarkan kategori, seperti deterjen, pewangi, peralatan, dan lain-lain. Informasi stok, harga beli, dan satuan barang diperlukan untuk mengelola persediaan dan pembelian barang. Tanggal update digunakan untuk melacak kapan data barang terakhir diperbarui.

- Primary Key
 - id_barang

- Tipe Data
 - id_barang : int(11) NOT NULL,
 - nama_barang : varchar(100) NOT NULL,
 - Kategori : enum('Deterjen','Pewangi','Peralatan','Lainnya') NOT NULL,
 - stok : int(11) NOT NULL,
 - harga_beli : decimal(10,2) NOT NULL,
 - Satuan : varchar(20) NOT NULL,
 - tanggal_update : timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE current_timestamp()

Tabel Detail Transaksi

Berfungsi untuk mencatat rincian layanan yang dipesan oleh pelanggan dalam setiap transaksi. Setiap transaksi dapat terdiri dari beberapa jenis layanan, maka tabel ini akan mencatat jumlah dan subtotal untuk masing-masing layanan. Tabel ini memiliki relasi dengan tabel transaksi dan layanan melalui foreign key, sehingga dapat menghubungkan data transaksi dengan jenis layanan yang dipilih.

- Primary key
 - id_detail

- Foreign Key
 - id_transaksi (referensi ke tabel transaksi)
 - id_layanan (referensi ke tabel layanan)

- Tipe Data
 - id_detail : int(11) NOT NULL,
 - id_transaksi : int(11) DEFAULT NULL,
 - id_layanan : int(11) DEFAULT NULL,
 - jumlah : int(11) NOT NULL,
 - subtotal : decimal(10,2) NOT NULL

Tabel Karyawan

Digunakan untuk menyimpan data karyawan yang bekerja di bisnis laundry. Informasi karyawan seperti nama, jabatan, kontak, dan gaji perlu dicatat untuk keperluan pengelolaan sumber daya manusia. Email karyawan disimpan secara unik untuk memastikan tidak ada duplikasi data.

- Primary Key
 - id_karyawan

- Tipe Data
 - id_karyawan : int(11) NOT NULL,
 - nama_karyawan : varchar(100) NOT NULL,
 - jabatan : varchar(50) NOT NULL,
 - alamat : text DEFAULT NULL,
 - no_telp : varchar(15) NOT NULL,
 - email : varchar(100) DEFAULT NULL,
 - tanggal_bergabung : date NOT NULL,
 - gaji : decimal(10,2) NOT NULL

Tabel Keluhan

Berfungsi untuk mencatat setiap keluhan yang disampaikan oleh pelanggan terkait layanan laundry. Setiap keluhan akan disimpan beserta informasi seperti tanggal keluhan, deskripsi keluhan, status penanganan, tanggal penyelesaian, dan solusi yang diberikan. Tabel ini memiliki relasi dengan tabel transaksi dan pelanggan sehingga dapat mengidentifikasi transaksi dan pelanggan yang terkait dengan keluhan.

- Primary Key
 - id_keluhan

- Foreign Key
 - id_transaksi (referensi ke tabel transaksi)
 - id_pelanggan (referensi ke tabel pelanggan)

- Tipe Data
 - id_keluhan : int(11) NOT NULL,
 - id_transaksi : int(11) DEFAULT NULL,
 - id_pelanggan : int(11) DEFAULT NULL,
 - tanggal_keluhan : datetime NOT NULL,
 - deskripsi_keluhan : text NOT NULL,
 - status : enum('Belum Ditangani','Sedang Ditangani','Selesai') DEFAULT 'Belum Ditangani',
 - tanggal_penyehesaian : datetime DEFAULT NULL,
 - solusi : text DEFAULT NULL

Tabel Layanan

Digunakan untuk menyimpan informasi mengenai jenis-jenis layanan yang ditawarkan oleh bisnis laundry. Data layanan mencakup nama, deskripsi, harga, estimasi waktu penggerjaan, dan status ketersediaan.

- Primary Key
 - id_layanan

- Tipe Data
 - id_layanan : int(11) NOT NULL,
 - nama_layanan : varchar(100) NOT NULL,
 - deskripsi : text DEFAULT NULL,
 - harga : decimal(10,2) NOT NULL,
 - estimasi_waktu : int(11) NOT NULL COMMENT ',
 - status : enum('Aktif','Tidak Aktif') DEFAULT 'Aktif'

Tabel Pelanggan

Berfungsi untuk menyimpan data pelanggan yang menggunakan layanan laundry. Informasi pelanggan seperti nama, alamat, kontak, email, tanggal bergabung, dan poin yang dimiliki perlu dicatat. Data pelanggan ini digunakan untuk mengelola profil dan riwayat transaksi setiap pelanggan.

- Primary Key
 - id_pelanggan

- Tipe Data
 - id_pelanggan : int(11) NOT NULL,
 - nama_pelanggan : varchar(100) NOT NULL,
 - alamat : text DEFAULT NULL,
 - no_telp : varchar(15) NOT NULL,
 - email : varchar(100) DEFAULT NULL,
 - tanggal_bergabung : date DEFAULT curdate(),
 - poin : int(11) DEFAULT 0

Tabel Transaksi

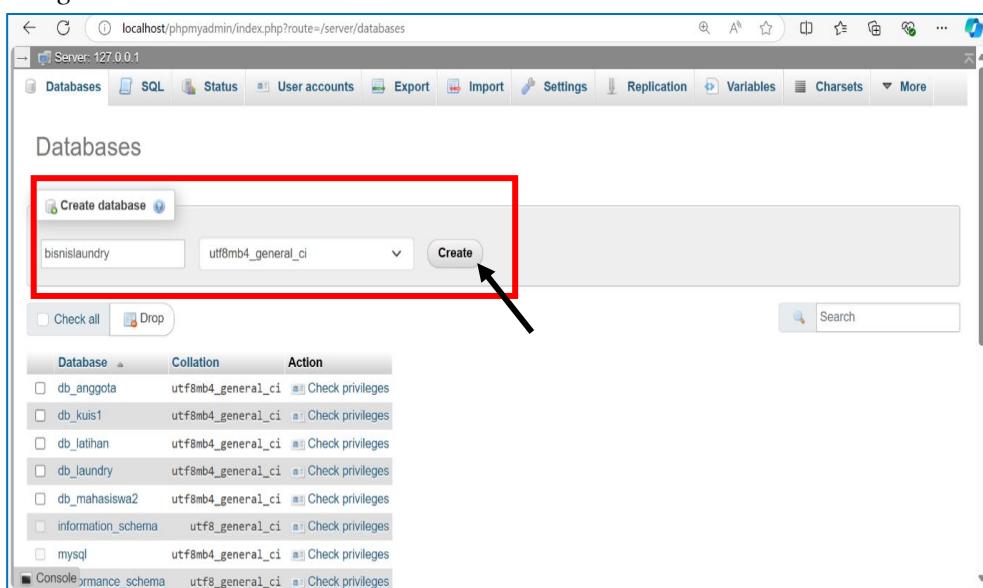
Digunakan untuk mencatat setiap transaksi yang dilakukan oleh pelanggan. Setiap transaksi akan disimpan dengan informasi seperti tanggal masuk, tanggal selesai, status, total harga, dan metode pembayaran. Tabel ini memiliki relasi dengan tabel pelanggan dan karyawan sehingga dapat mengidentifikasi pelanggan dan karyawan yang terlibat dalam suatu transaksi.

- Primary Key
 - id_transaksi
- Foreign Key
 - id_pelanggan (referensi ke tabel pelanggan)
 - id_karyawan (referensi ke tabel karyawan)
- Tipe Data
 - id_transaksi : int(11) NOT NULL,
 - id_pelanggan : int(11) DEFAULT NULL,
 - id_karyawan : int(11) DEFAULT NULL,
 - tanggal_masuk : datetime NOT NULL,
 - tanggal_selesai : datetime DEFAULT NULL,
 - status : enum('Proses','Selesai','Diambil') DEFAULT 'Proses',
 - total_harga : decimal(10,2) NOT NULL,
 - metode_pembayaran : enum('Tunai','Kartu Kredit','Transfer') NOT NULL

3.2.2 Contoh implementasi SQL

a) Perintah CREATE DATABASE

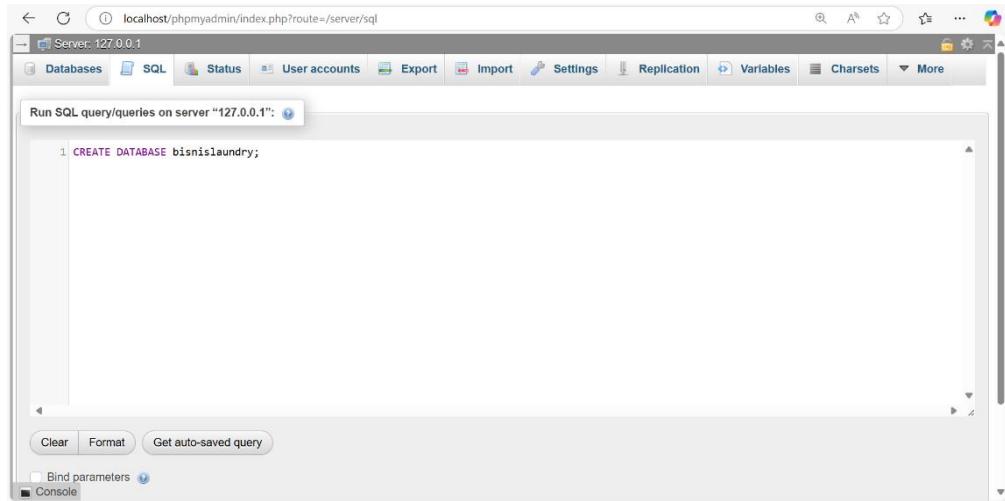
- *Dengan GUI*



The screenshot shows the phpMyAdmin interface for MySQL management. The top navigation bar includes tabs for Databases, SQL, Status, User accounts, Import, Export, Settings, Replication, Variables,Charsets, and More. The main area is titled 'Databases' and displays a list of existing databases: db_anggota, db_kuis1, db_latihan, db_laundry, db_mahasiswa2, information_schema, mysql, and Console. A new database, 'bisnislaundry', is being created in the 'Create database' input field. The 'Collation' dropdown is set to 'utf8mb4_general_ci'. A red box highlights the 'Create' button, which is being clicked by a mouse cursor. Below the list of databases, there is a table with columns Database, Collation, and Action, showing various privileges for each database.

Database	Collation	Action
db_anggota	utf8mb4_general_ci	Check privileges
db_kuis1	utf8mb4_general_ci	Check privileges
db_latihan	utf8mb4_general_ci	Check privileges
db_laundry	utf8mb4_general_ci	Check privileges
db_mahasiswa2	utf8mb4_general_ci	Check privileges
information_schema	utf8_general_ci	Check privileges
mysql	utf8mb4_general_ci	Check privileges
Console	utf8_general_ci	Check privileges

- *Dengan SQL*



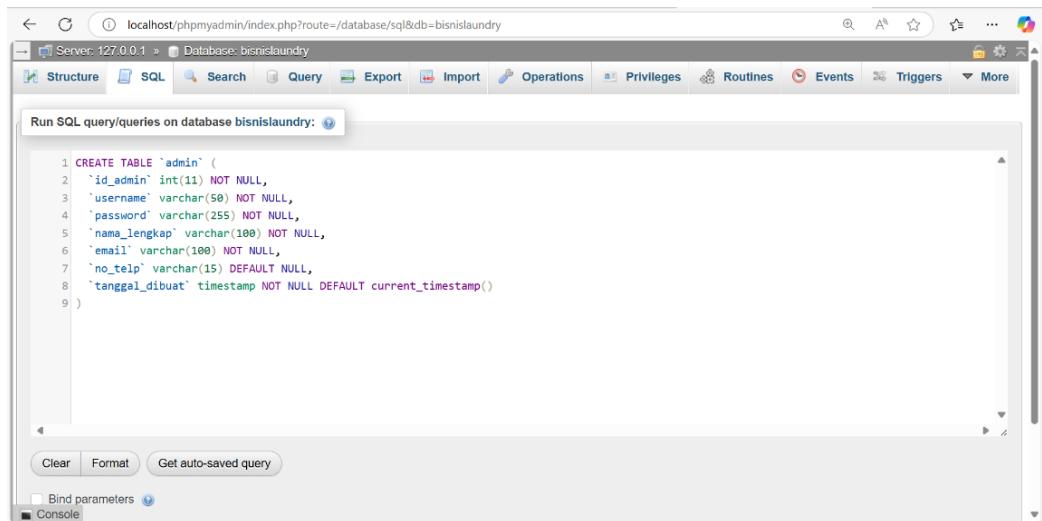
The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1
- Databases:** bisnislaunder
- SQL Tab:** Active
- Query Editor:** Contains the SQL command: `CREATE DATABASE bisnislaunder;`
- Buttons:** Clear, Format, Get auto-saved query
- Checkboxes:** Bind parameters, Console

Gambar 3. 2 Create Database dengan SQL

b) Perintah CREATE TABEL

- *CREATE Tabel Admin*



The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1 > Database: bisnislaunder
- Structure Tab:** Active
- SQL Tab:** Active
- Query Editor:** Contains the SQL command for creating the 'admin' table:

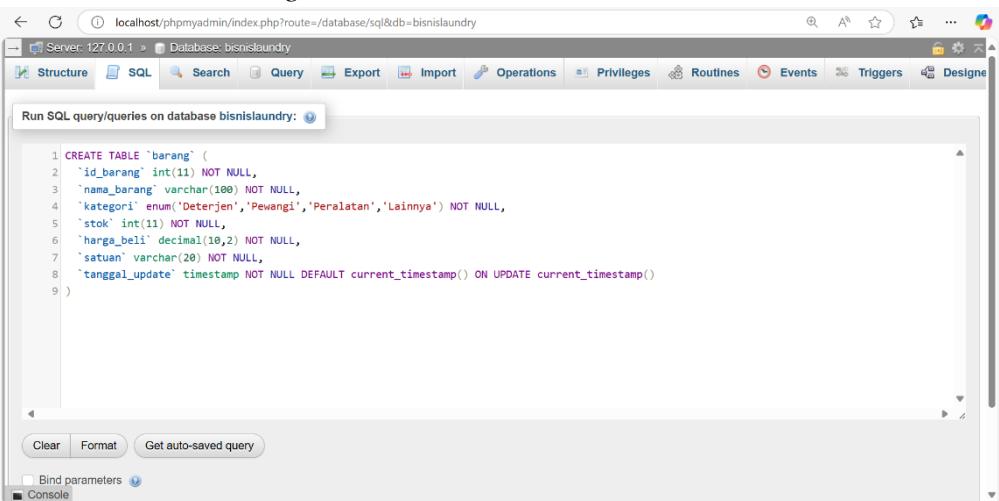
```

1 CREATE TABLE `admin` (
2   `id_admin` int(11) NOT NULL,
3   `username` varchar(50) NOT NULL,
4   `password` varchar(255) NOT NULL,
5   `nama_lengkap` varchar(100) NOT NULL,
6   `email` varchar(100) NOT NULL,
7   `no_telp` varchar(15) DEFAULT NULL,
8   `tanggal_dibuat` timestamp NOT NULL DEFAULT current_timestamp()
9 )

```

- Buttons:** Clear, Format, Get auto-saved query
- Checkboxes:** Bind parameters, Console

- *CREATE Tabel Barang*



The screenshot shows the phpMyAdmin interface with the following details:

- Server:** 127.0.0.1 > Database: bisnislaunder
- Structure Tab:** Active
- SQL Tab:** Active
- Query Editor:** Contains the SQL command for creating the 'barang' table:

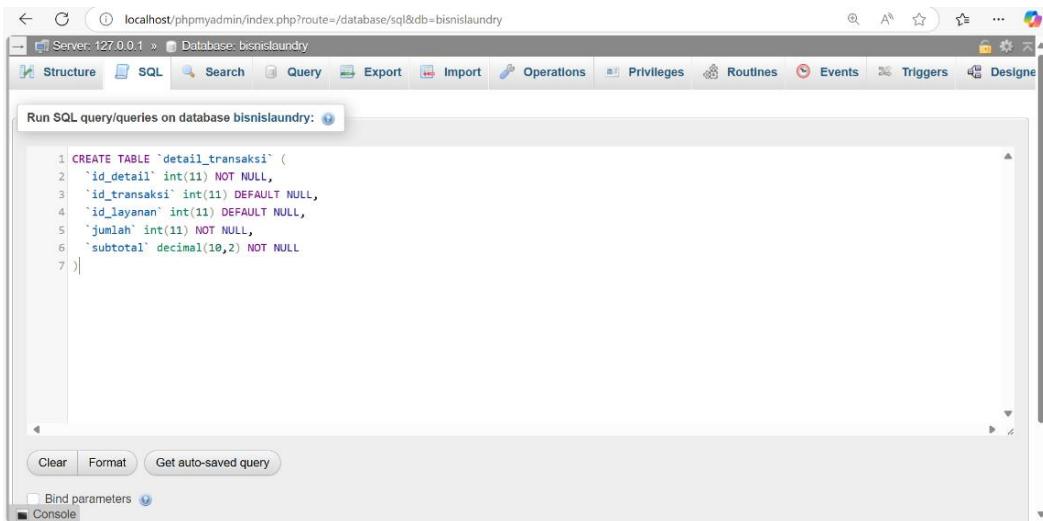
```

1 CREATE TABLE `barang` (
2   `id_barang` int(11) NOT NULL,
3   `name_barang` varchar(100) NOT NULL,
4   `kategori` enum('Deterjen','Pewangi','Peralatan','Lainnya') NOT NULL,
5   `stok` int(11) NOT NULL,
6   `harga_beli` decimal(10,2) NOT NULL,
7   `satuan` varchar(20) NOT NULL,
8   `tanggal_update` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE current_timestamp()
9 )

```

- Buttons:** Clear, Format, Get auto-saved query
- Checkboxes:** Bind parameters, Console

- *CREATE Tabel Detail Transaksi*



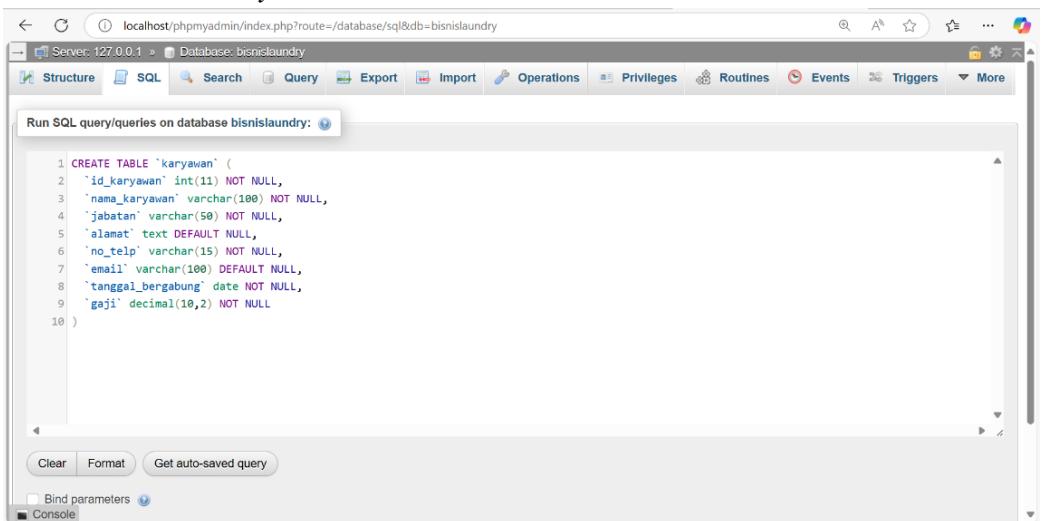
```
1 CREATE TABLE `detail_transaksi` (
2   `id_detail` int(11) NOT NULL,
3   `id_transaksi` int(11) DEFAULT NULL,
4   `id_layanan` int(11) DEFAULT NULL,
5   `jumlah` int(11) NOT NULL,
6   `subtotal` decimal(10,2) NOT NULL
7 )|
```

The screenshot shows the phpMyAdmin interface with the 'SQL' tab selected. A SQL query is entered in the main area:

```
1 CREATE TABLE `detail_transaksi` (
2   `id_detail` int(11) NOT NULL,
3   `id_transaksi` int(11) DEFAULT NULL,
4   `id_layanan` int(11) DEFAULT NULL,
5   `jumlah` int(11) NOT NULL,
6   `subtotal` decimal(10,2) NOT NULL
7 )|
```

Below the query, there are buttons for 'Clear', 'Format', and 'Get auto-saved query'. There is also a checkbox for 'Bind parameters' and a dropdown menu for 'Console'.

- *CREATE Tabel Karyawan*



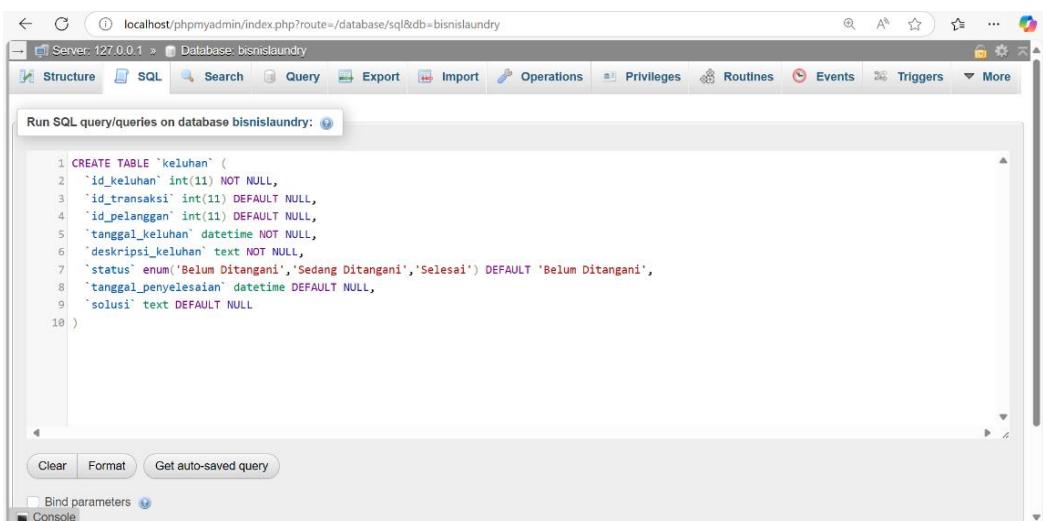
```
1 CREATE TABLE `karyawan` (
2   `id_karyawan` int(11) NOT NULL,
3   `nama_karyawan` varchar(100) NOT NULL,
4   `jabatan` varchar(50) NOT NULL,
5   `alamat` text DEFAULT NULL,
6   `no_telp` varchar(15) NOT NULL,
7   `email` varchar(100) DEFAULT NULL,
8   `tanggal Bergabung` date NOT NULL,
9   `gaji` decimal(10,2) NOT NULL
10 )|
```

The screenshot shows the phpMyAdmin interface with the 'SQL' tab selected. A SQL query is entered in the main area:

```
1 CREATE TABLE `karyawan` (
2   `id_karyawan` int(11) NOT NULL,
3   `nama_karyawan` varchar(100) NOT NULL,
4   `jabatan` varchar(50) NOT NULL,
5   `alamat` text DEFAULT NULL,
6   `no_telp` varchar(15) NOT NULL,
7   `email` varchar(100) DEFAULT NULL,
8   `tanggal Bergabung` date NOT NULL,
9   `gaji` decimal(10,2) NOT NULL
10 )|
```

Below the query, there are buttons for 'Clear', 'Format', and 'Get auto-saved query'. There is also a checkbox for 'Bind parameters' and a dropdown menu for 'Console'.

- *CREATE Tabel Keluhan*



```
1 CREATE TABLE `keluhan` (
2   `id_keluhan` int(11) NOT NULL,
3   `id_transaksi` int(11) DEFAULT NULL,
4   `id_pelanggan` int(11) DEFAULT NULL,
5   `tanggal_keuhan` datetime NOT NULL,
6   `deskripsi_keuhan` text NOT NULL,
7   `status` enum('Belum Ditangani','Sedang Ditangani','Selesai') DEFAULT 'Belum Ditangani',
8   `tanggal_penyelasaian` datetime DEFAULT NULL,
9   `solusi` text DEFAULT NULL
10 )|
```

The screenshot shows the phpMyAdmin interface with the 'SQL' tab selected. A SQL query is entered in the main area:

```
1 CREATE TABLE `keluhan` (
2   `id_keluhan` int(11) NOT NULL,
3   `id_transaksi` int(11) DEFAULT NULL,
4   `id_pelanggan` int(11) DEFAULT NULL,
5   `tanggal_keuhan` datetime NOT NULL,
6   `deskripsi_keuhan` text NOT NULL,
7   `status` enum('Belum Ditangani','Sedang Ditangani','Selesai') DEFAULT 'Belum Ditangani',
8   `tanggal_penyelasaian` datetime DEFAULT NULL,
9   `solusi` text DEFAULT NULL
10 )|
```

Below the query, there are buttons for 'Clear', 'Format', and 'Get auto-saved query'. There is also a checkbox for 'Bind parameters' and a dropdown menu for 'Console'.

- *CREATE Tabel Layanan*



```
1 CREATE TABLE `layanan` (
2   `id_layanan` int(11) NOT NULL,
3   `nama_layanan` varchar(100) NOT NULL,
4   `deskripsi` text DEFAULT NULL,
5   `harga` decimal(10,2) NOT NULL,
6   `estimasi_waktu` int(11) NOT NULL COMMENT 'Dalam jam',
7   `status` enum('Aktif','Tidak Aktif') DEFAULT 'Aktif'
8 )
```

The screenshot shows the 'Query' tab in phpMyAdmin with the SQL code for creating the 'layanan' table. The table has columns: id_layanan (int(11) NOT NULL), nama_layanan (varchar(100) NOT NULL), deskripsi (text DEFAULT NULL), harga (decimal(10,2) NOT NULL), estimasi_waktu (int(11) NOT NULL COMMENT 'Dalam jam'), and status (enum('Aktif','Tidak Aktif') DEFAULT 'Aktif'). The code is enclosed in a single CREATE TABLE statement.

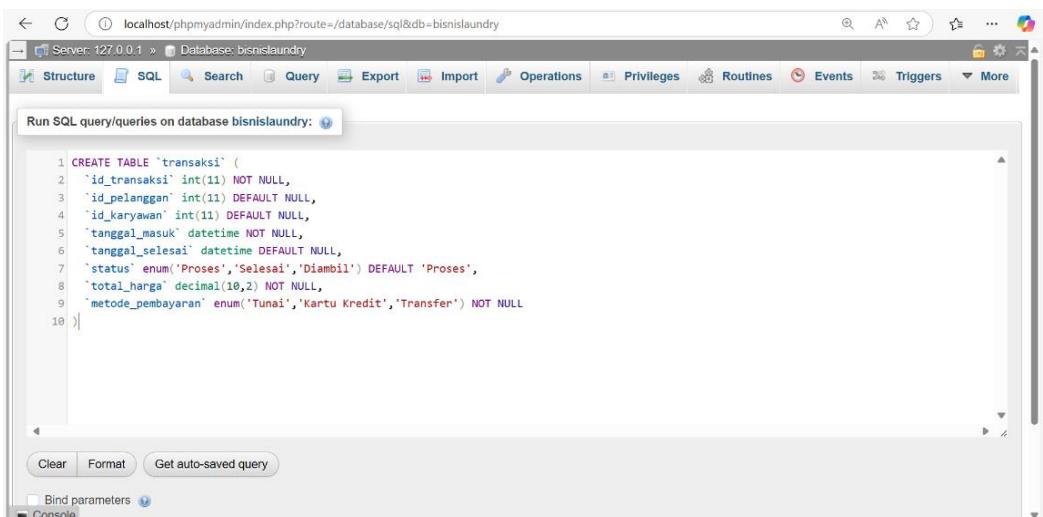
- *CREATE Tabel Pelanggan*



```
1 CREATE TABLE `pelanggan` (
2   `id_pelanggan` int(11) NOT NULL,
3   `nama_pelanggan` varchar(100) NOT NULL,
4   `alamat` text DEFAULT NULL,
5   `no_telp` varchar(15) NOT NULL,
6   `email` varchar(100) DEFAULT NULL,
7   `tanggal Bergabung` date DEFAULT curdate(),
8   `poin` int(11) DEFAULT 0
9 )
```

The screenshot shows the 'Query' tab in phpMyAdmin with the SQL code for creating the 'pelanggan' table. The table has columns: id_pelanggan (int(11) NOT NULL), nama_pelanggan (varchar(100) NOT NULL), alamat (text DEFAULT NULL), no_telp (varchar(15) NOT NULL), email (varchar(100) DEFAULT NULL), tanggal Bergabung (date DEFAULT curdate()), and poin (int(11) DEFAULT 0). The column names contain a space, which is likely a typo in the screenshot.

- *CREATE Tabel Transaksi*



```
1 CREATE TABLE `transaksi` (
2   `id_transaksi` int(11) NOT NULL,
3   `id_pelanggan` int(11) DEFAULT NULL,
4   `id_karyawan` int(11) DEFAULT NULL,
5   `tanggal_masuk` datetime NOT NULL,
6   `tanggal_selesai` datetime DEFAULT NULL,
7   `status` enum('Proses','Selesai','Dambil') DEFAULT 'Proses',
8   `total_harga` decimal(10,2) NOT NULL,
9   `metode_pembayaran` enum('Tunai','Kartu Kredit','Transfer') NOT NULL
10 )
```

The screenshot shows the 'Query' tab in phpMyAdmin with the SQL code for creating the 'transaksi' table. The table has columns: id_transaksi (int(11) NOT NULL), id_pelanggan (int(11) DEFAULT NULL), id_karyawan (int(11) DEFAULT NULL), tanggal_masuk (datetime NOT NULL), tanggal_selesai (datetime DEFAULT NULL), status (enum('Proses','Selesai','Dambil') DEFAULT 'Proses'), total_harga (decimal(10,2) NOT NULL), and metode_pembayaran (enum('Tunai','Kartu Kredit','Transfer') NOT NULL).

Gambar 3. 3 SQL Create Setiap Tabel

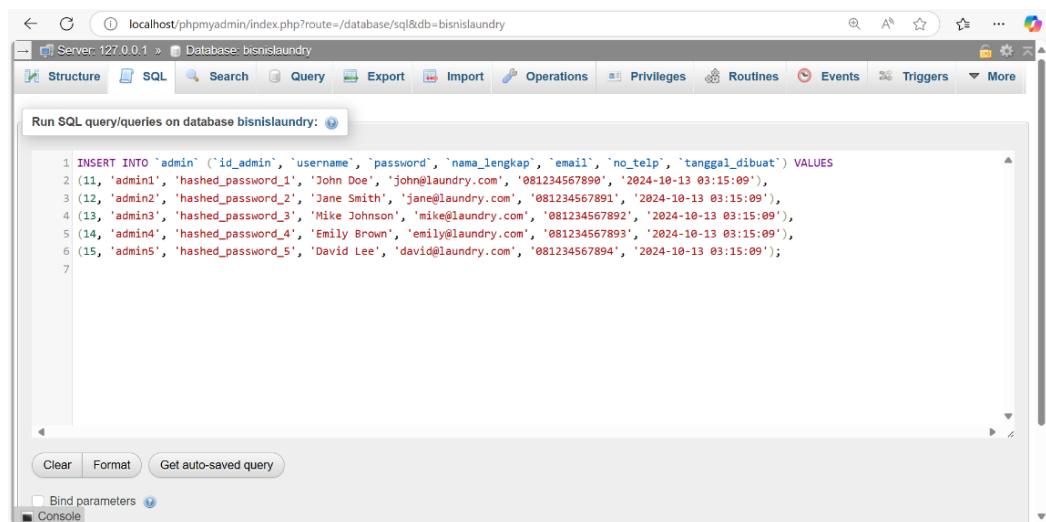
3.3 Manipulasi Data

Data Manipulation Language (DML) merupakan suatu metode perintah SQL yang digunakan untuk proses pengolahan isi data di dalam tabel seperti menampilkan, memasukan, mengubah, menghapus isi data dan tidak terkait dengan perubahan struktur serta definisi tipe data dari objek database. Perintah SQL yang termasuk dalam DML antara lain select, update, dan delete. [6]

3.3.1 Menambahkan Data (INSERT)

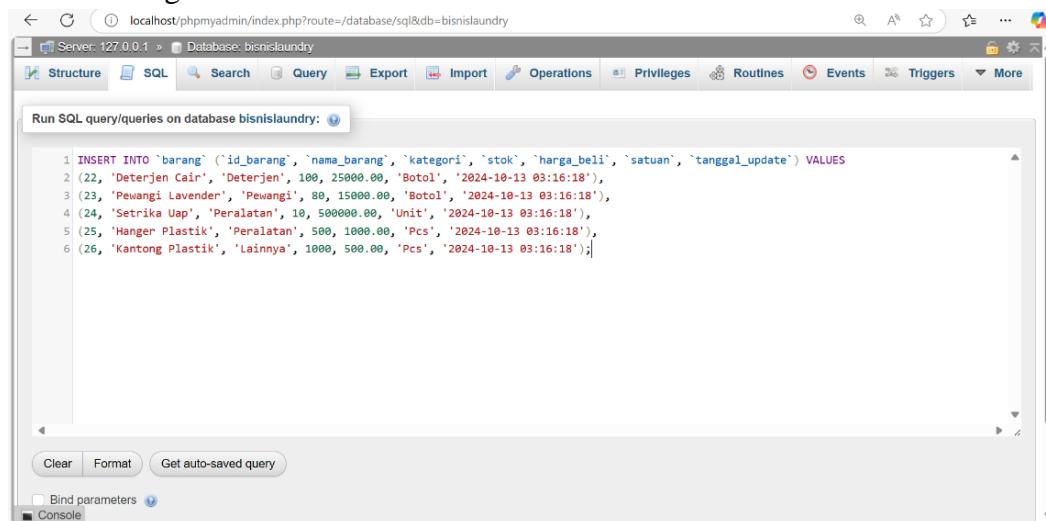
Dari gambar yang ditunjukkan dibawah, merupakan perintah SQL untuk 'Menambahkan data (INSERT)' pada masing-masing tabel. Perintah 'INSERT' biasanya digunakan dalam SQL (Structured Query Language) untuk menambahkan baris data baru ke dalam tabel database. Secara umum, proses INSERT akan melibatkan penyusunan pernyataan SQL INSERT dengan nama-nama kolom yang sesuai dan nilai-nilai yang akan dimasukkan ke dalam tabel.

- Tabel Admin



```
Run SQL query/queries on database bisniskaundry: ⓘ  
1 INSERT INTO `admin` (`id_admin`, `username`, `password`, `nama_lengkap`, `email`, `no_telp`, `tanggal_dibuat`) VALUES  
2 (11, 'admin1', 'hashed_password_1', 'John Doe', 'john@laundry.com', '081234567890', '2024-10-13 03:15:09'),  
3 (12, 'admin2', 'hashed_password_2', 'Jane Smith', 'jane@laundry.com', '081234567891', '2024-10-13 03:15:09'),  
4 (13, 'admin3', 'hashed_password_3', 'Mike Johnson', 'mike@laundry.com', '081234567892', '2024-10-13 03:15:09'),  
5 (14, 'admin4', 'hashed_password_4', 'Emily Brown', 'emily@laundry.com', '081234567893', '2024-10-13 03:15:09'),  
6 (15, 'admin5', 'hashed_password_5', 'David Lee', 'david@laundry.com', '081234567894', '2024-10-13 03:15:09');  
7
```

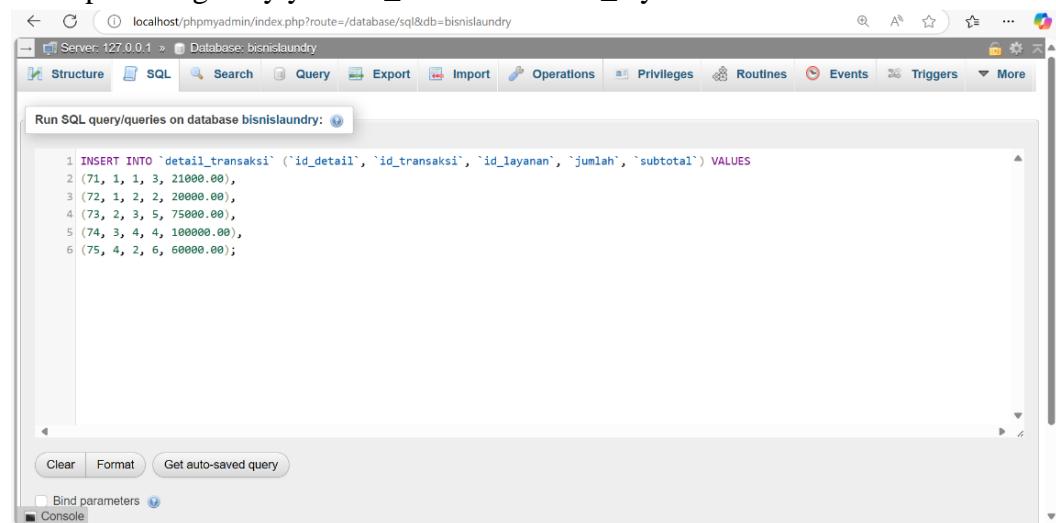
- Tabel Barang



```
Run SQL query/queries on database bisniskaundry: ⓘ  
1 INSERT INTO `barang` (`id_barang`, `nama_barang`, `kategori`, `stok`, `harga_beli`, `satuan`, `tanggal_update`) VALUES  
2 (22, 'Deterjen Cair', 'Deterjen', 100, 25000.00, 'Botol', '2024-10-13 03:16:18'),  
3 (23, 'Pewangi Lavender', 'Pewangi', 80, 15000.00, 'Botol', '2024-10-13 03:16:18'),  
4 (24, 'Setrika Uap', 'Peralatan', 10, 500000.00, 'Unit', '2024-10-13 03:16:18'),  
5 (25, 'Hanger Plastik', 'Peralatan', 500, 1000.00, 'Pcs', '2024-10-13 03:16:18'),  
6 (26, 'Kantong Plastik', 'Lainnya', 1000, 500.00, 'Pcs', '2024-10-13 03:16:18');
```

- Tabel Detail Transaksi

Terdapat foreign key yaitu id_transaksi dan id_layanan.

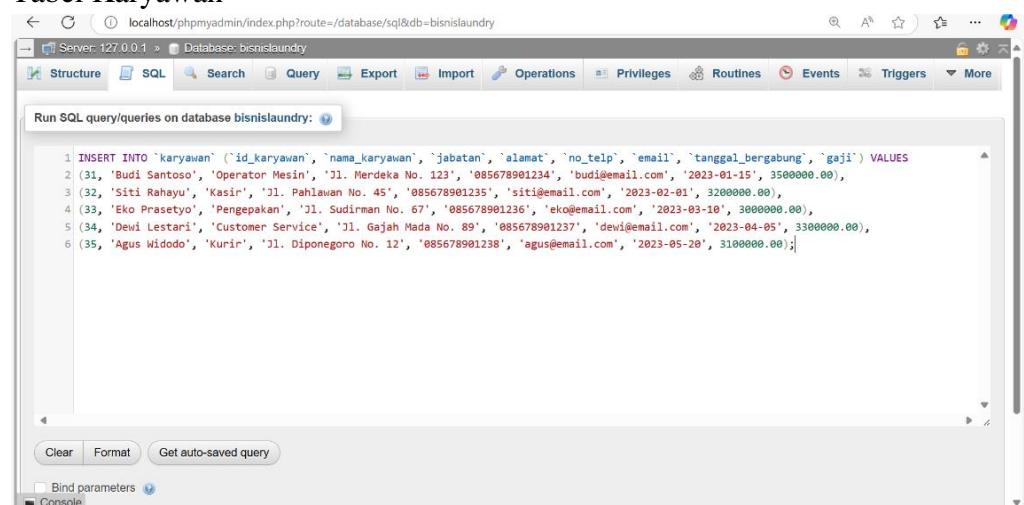


```
Run SQL query/queries on database bisniskauntry: ⓘ

1 INSERT INTO `detail_transaksi`(`id_detail`, `id_transaksi`, `id_layanan`, `jumlah`, `subtotal`) VALUES
2 (71, 1, 1, 3, 21000.00),
3 (72, 1, 2, 2, 20000.00),
4 (73, 2, 3, 5, 75000.00),
5 (74, 3, 4, 4, 100000.00),
6 (75, 4, 2, 6, 60000.00);
```

Clear Format Get auto-saved query
Bind parameters Console

- Tabel Karyawan



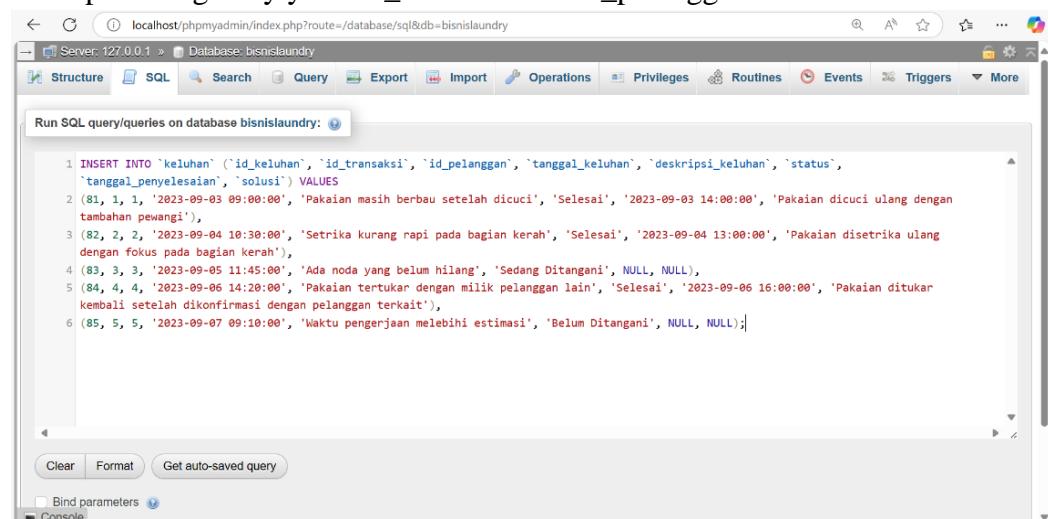
```
Run SQL query/queries on database bisniskauntry: ⓘ

1 INSERT INTO `karyawan`(`id_karyawan`, `nama_karyawan`, `jabatan`, `alamat`, `no_telp`, `email`, `tanggal Bergabung`, `gaji`) VALUES
2 (31, 'Budi Santoso', 'Operator Mesin', 'Jl. Merdeka No. 123', '085678901234', 'budi@email.com', '2023-01-15', 3500000.00),
3 (32, 'Siti Rahayu', 'Kasir', 'Jl. Pahlawan No. 45', '085678901235', 'siti@email.com', '2023-02-01', 3200000.00),
4 (33, 'Eko Prasetyo', 'Pengepakan', 'Jl. Sudirman No. 67', '085678901236', 'eko@email.com', '2023-03-10', 3000000.00),
5 (34, 'Dewi Lestari', 'Customer Service', 'Jl. Gajah Mada No. 89', '085678901237', 'devi@email.com', '2023-04-05', 3300000.00),
6 (35, 'Agus Widodo', 'Kurir', 'Jl. Diponegoro No. 12', '085678901238', 'agus@email.com', '2023-05-20', 3100000.00);
```

Clear Format Get auto-saved query
Bind parameters Console

- Tabel Keluhan

Terdapat foreign key yaitu id_transaksi dan id_pelanggan.

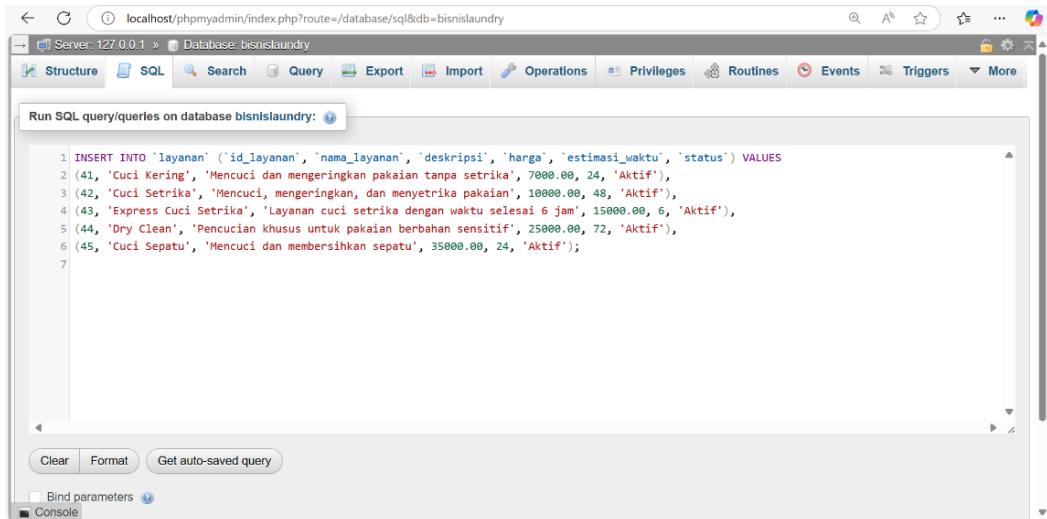


```
Run SQL query/queries on database bisniskauntry: ⓘ

1 INSERT INTO `keluhan`(`id_keluhan`, `id_transaksi`, `id_pelanggan`, `tanggal_keluhan`, `deskripsi_keluhan`, `status`, `tanggal_penyelusian`, `solusi`) VALUES
2 (81, 1, 1, '2023-09-03 09:00:00', 'Pakaian masih berbau setelah dicuci', 'Selesai', '2023-09-03 14:00:00', 'Pakaian dicuci ulang dengan tambahan pewangi'),
3 (82, 2, 2, '2023-09-04 10:30:00', 'Setrika kurang rapi pada bagian kerah', 'Selesai', '2023-09-04 13:00:00', 'Pakaian disetrika ulang dengan fokus pada bagian kerah'),
4 (83, 3, 3, '2023-09-05 11:45:00', 'Ada nodai yang belum hilang', 'Sedang Ditanganani', NULL, NULL),
5 (84, 4, 4, '2023-09-06 14:20:00', 'Pakaian tertukar dengan milik pelanggan lain', 'Selesai', '2023-09-06 16:00:00', 'Pakaian ditukar kembali setelah dikonfirmasi dengan pelanggan terkait'),
6 (85, 5, 5, '2023-09-07 09:10:00', 'Waktu pengiriman melebihi estimasi', 'Belum Ditanganani', NULL, NULL);
```

Clear Format Get auto-saved query
Bind parameters Console

- Tabel Layanan

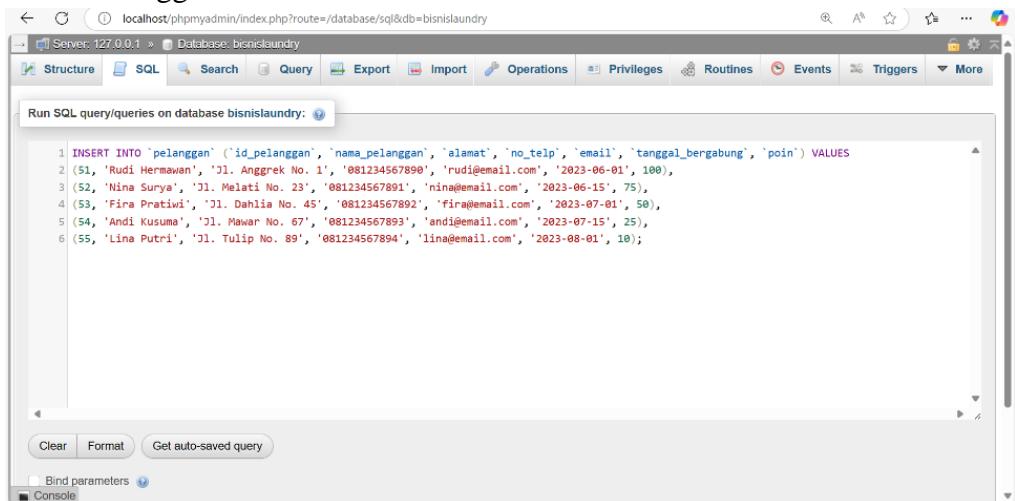


```

1 INSERT INTO `layanan` ('id_layanan', 'nama_layanan', 'deskripsi', 'harga', 'estimasi_waktu', 'status') VALUES
2 (41, 'Cuci Kering', 'Mencuci dan mengeringkan pakaian tanpa setrika', 7000.00, 24, 'Aktif'),
3 (42, 'Cuci Setrika', 'Mencuci, mengeringkan, dan menyetrika pakaian', 10000.00, 48, 'Aktif'),
4 (43, 'Express Cuci Setrika', 'Layanan cuci setrika dengan waktu selesai 6 jam', 15000.00, 6, 'Aktif'),
5 (44, 'Dry Clean', 'Pencucian khusus untuk pakaian berbahan sensitif', 25000.00, 72, 'Aktif'),
6 (45, 'Cuci Sepatu', 'Mencuci dan membersihkan sepatu', 35000.00, 24, 'Aktif');
7

```

- Tabel Pelanggan



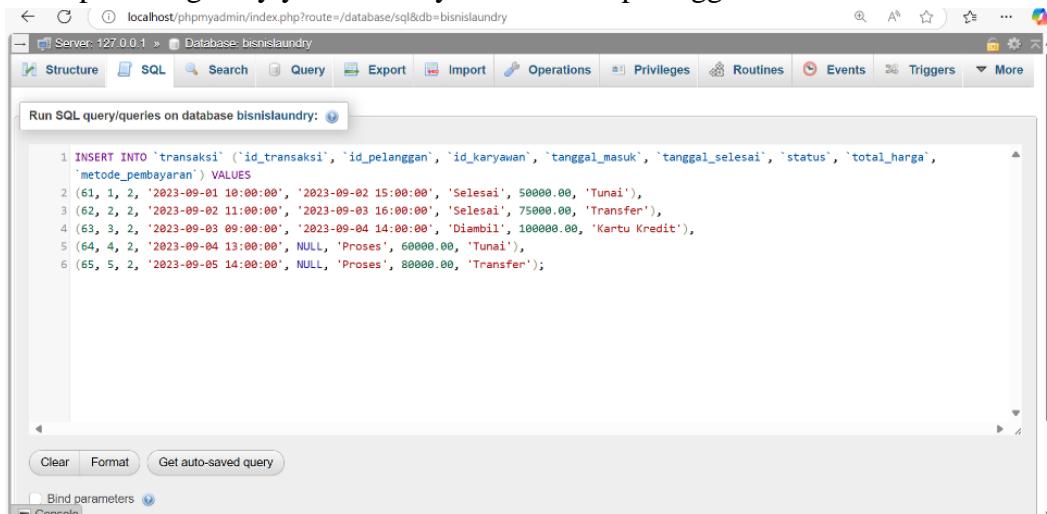
```

1 INSERT INTO `pelanggan` ('id_pelanggan', 'nama_pelanggan', 'alamat', 'no_telp', 'email', 'tanggal Bergabung', 'poin') VALUES
2 (51, 'Rudi Hermawan', 'Jl. Anggrek No. 1', '081234567890', 'rudi@email.com', '2023-06-01', 100),
3 (52, 'Nina Surya', 'Jl. Melati No. 23', '081234567891', 'nina@email.com', '2023-06-15', 75),
4 (53, 'Fira Pratiwi', 'Jl. Dahlia No. 45', '081234567892', 'fira@email.com', '2023-07-01', 50),
5 (54, 'Andi Kusuma', 'Jl. Mawar No. 67', '081234567893', 'andi@email.com', '2023-07-15', 25),
6 (55, 'Lina Putri', 'Jl. Tulip No. 89', '081234567894', 'lina@email.com', '2023-08-01', 10);

```

- Tabel Transaksi

Terdapat foreign key yaitu id_karyawan dan id_pelanggan.



```

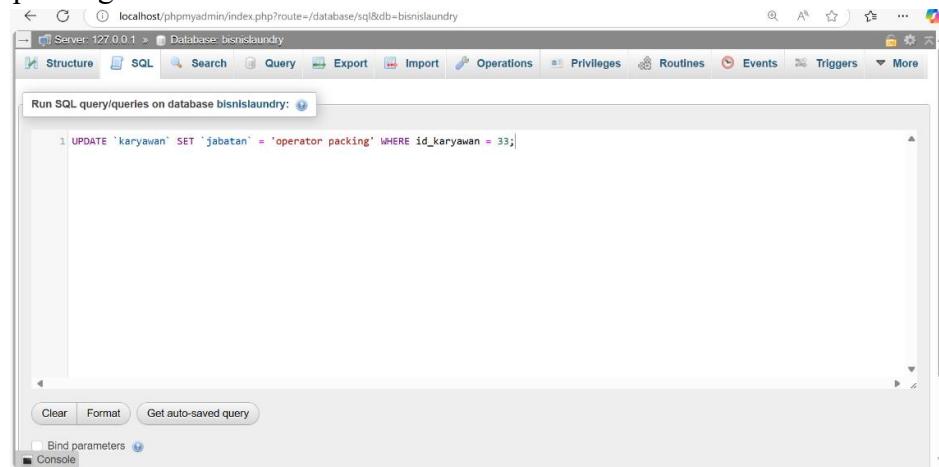
1 INSERT INTO `transaksi` ('id_transaksi', 'id_pelanggan', 'id_karyawan', 'tanggal_masuk', 'tanggal_selesai', 'status', 'total_harga', 'metode_pembayaran') VALUES
2 (61, 1, 2, '2023-09-01 10:00:00', '2023-09-02 15:00:00', 'Selesai', 50000.00, 'Tunai'),
3 (62, 2, 2, '2023-09-02 11:00:00', '2023-09-03 16:00:00', 'Selesai', 75000.00, 'Transfer'),
4 (63, 3, 2, '2023-09-03 09:00:00', '2023-09-04 14:00:00', 'Diambil', 100000.00, 'Kartu Kredit'),
5 (64, 4, 2, '2023-09-04 13:00:00', NULL, 'Proses', 60000.00, 'Tunai'),
6 (65, 5, 2, '2023-09-05 14:00:00', NULL, 'Proses', 80000.00, 'Transfer');

```

Gambar 3. 4 SQL Insert Setiap Tabel

3.3.2 Memodifikasi Data (UPDATE)

- Karyawan dengan id_karyawan = 33 di Update menjadi jabatan ‘operator packing’



The screenshot shows the phpMyAdmin interface for the 'bisniskauntry' database. The 'Query' tab is selected. In the query editor, the following SQL command is entered:

```
1 UPDATE `karyawan` SET `jabatan` = 'operator packing' WHERE id_karyawan = 33;
```

- Layanan dengan id_layanan = 41 di Update menjadi harga ‘9000’



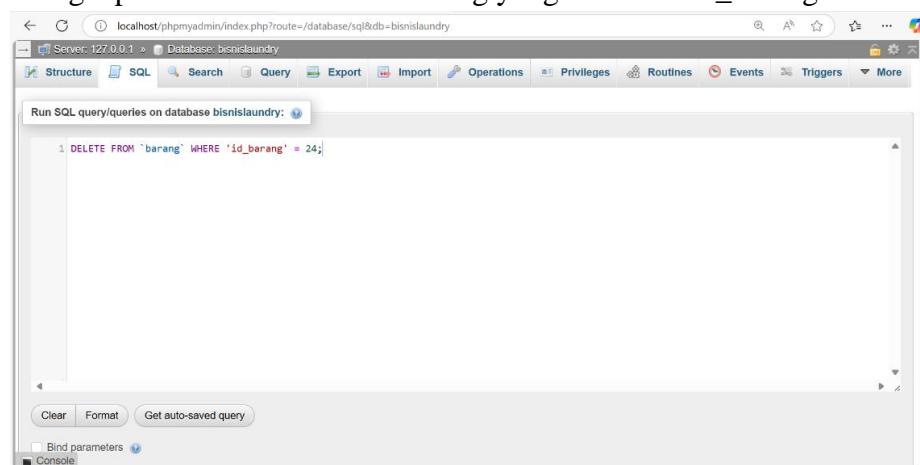
The screenshot shows the phpMyAdmin interface for the 'bisniskauntry' database. The 'Query' tab is selected. In the query editor, the following SQL command is entered:

```
1 UPDATE `layanan` SET `harga` = '9000' WHERE id_layanan = 41;
```

Gambar 3. 5 SQL Update Isi Tabel

3.3.3 Menghapus Data (DELETE)

- Menghapus baris data di tabel barang yang memiliki id_barang = 24.



The screenshot shows the phpMyAdmin interface for the 'bisniskauntry' database. The 'Query' tab is selected. In the query editor, the following SQL command is entered:

```
1 DELETE FROM `barang` WHERE `id_barang` = 24;
```

Gambar 3. 6 SQL Delete Isi Tabel

3.4 Query data

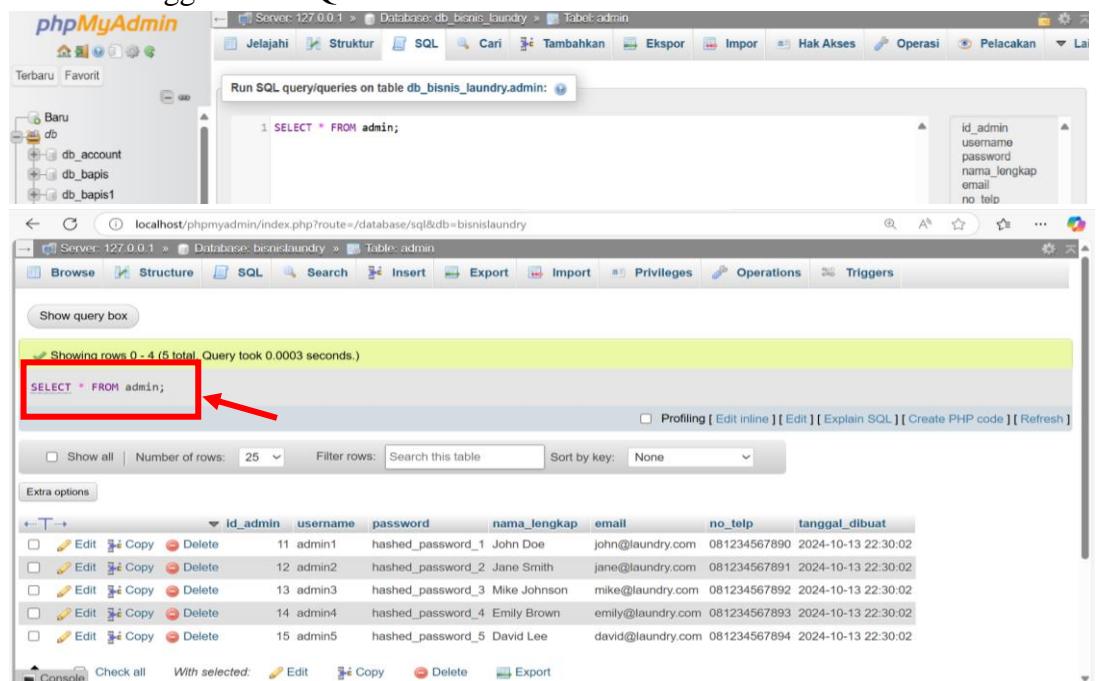
Query Adalah kemampuan untuk menampilkan dan atau meminta data yang digunakan untuk mengakses sebuah data pada sistem database dengan menggunakan perintah tertentu. Dimana sebuah data yang di peroleh dari satu atau beberapa tabel pada database sesuai dengan apa yang user perlukan. Data tersebut juga dapat berinteraksi dengan data atau tabel-tabel lainnya dengan tujuan mempermudah user dalam penggunaan nya. [7]

3.4.1 Penggunaan SELECT dan JOIN

Bagian ini menjelaskan tentang penggunaan perintah SELECT dan JOIN dalam query data. Dalam bagian ini, dijelaskan bahwa query akan menggunakan perintah SELECT untuk mengambil data, dan juga akan menggunakan JOIN untuk menggabungkan beberapa tabel.

- **SELECT Tabel Admin**

→ Menggunakan SQL



The screenshot shows the phpMyAdmin interface. In the top navigation bar, 'Server: 127.0.0.1' and 'Database: db_bisnis_laundry' are selected. The 'Tabel: admin' tab is active. The main area contains a SQL query box with the code 'SELECT * FROM admin;'. Below the query box is a results table showing 5 rows of data from the 'admin' table. The columns are: id_admin, username, password, nama_lengkap, email, no_telp, and tanggal_dibuat. The data is as follows:

	id_admin	username	password	nama_lengkap	email	no_telp	tanggal_dibuat
1	11	admin1	hashed_password_1	John Doe	john@laundry.com	081234567890	2024-10-13 22:30:02
2	12	admin2	hashed_password_2	Jane Smith	jane@laundry.com	081234567891	2024-10-13 22:30:02
3	13	admin3	hashed_password_3	Mike Johnson	mike@laundry.com	081234567892	2024-10-13 22:30:02
4	14	admin4	hashed_password_4	Emily Brown	emily@laundry.com	081234567893	2024-10-13 22:30:02
5	15	admin5	hashed_password_5	David Lee	david@laundry.com	081234567894	2024-10-13 22:30:02

→ Menggunakan Shell atau CMD

```
MariaDB [laundry_tugasnew]> select * from admin;
+-----+-----+-----+-----+-----+-----+-----+
| id_admin | username | password | nama_lengkap | email | no_telp | tanggal_dibuat |
+-----+-----+-----+-----+-----+-----+-----+
| 11 | admin1 | hashed_password_1 | John Doe | john@Laundry.com | 081234567890 | 2024-10-13 22:30:02 |
| 12 | admin2 | hashed_password_2 | Jane Smith | jane@Laundry.com | 081234567891 | 2024-10-13 22:30:02 |
| 13 | admin3 | hashed_password_3 | Mike Johnson | mike@Laundry.com | 081234567892 | 2024-10-13 22:30:02 |
| 14 | admin4 | hashed_password_4 | Emily Brown | emily@Laundry.com | 081234567893 | 2024-10-13 22:30:02 |
| 15 | admin5 | hashed_password_5 | David Lee | david@Laundry.com | 081234567894 | 2024-10-13 22:30:02 |
+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.006 sec)
```

- **SELECT Tabel Barang**
→ Menggunakan SQL

The screenshot shows the phpMyAdmin interface. On the left, there's a tree view of databases and tables. The main area has a query editor with the SQL command: `1 SELECT * FROM barang;`. To the right, the results of the query are displayed in a grid format with columns: id_barang, nama_barang, kategori, stok, harga_beli, satuan, and tanggal_update. There are five rows of data.

	id_barang	nama_barang	kategori	stok	harga_beli	satuan	tanggal_update
1	22	deterjenn	Deterjen	100	25000.00	Botol	2024-10-18 11:32:08
2	23	Pewangi Lavender	Pewangi	80	15000.00	Botol	2024-10-13 22:31:17
3	24	Setrika Uap	Peralatan	10	500000.00	Unit	2024-10-13 22:31:17
4	25	Hanger Plastik	Peralatan	500	1000.00	Pcs	2024-10-13 22:31:17
5	26	Kantong Plastik	Lainnya	1000	500.00	Pcs	2024-10-13 22:31:17

The screenshot shows a browser window with the URL `localhost/phpmyadmin/index.php?route=/database/sql&db=bisnislaundry`. It displays the same table structure and data as the previous screenshot. A red arrow points to the query input field where the SQL command is entered.

	id_barang	nama_barang	kategori	stok	harga_beli	satuan	tanggal_update
1	22	deterjenn	Deterjen	100	25000.00	Botol	2024-10-18 11:32:08
2	23	Pewangi Lavender	Pewangi	80	15000.00	Botol	2024-10-13 22:31:17
3	24	Setrika Uap	Peralatan	10	500000.00	Unit	2024-10-13 22:31:17
4	25	Hanger Plastik	Peralatan	500	1000.00	Pcs	2024-10-13 22:31:17
5	26	Kantong Plastik	Lainnya	1000	500.00	Pcs	2024-10-13 22:31:17

→ Menggunakan Shell atau CMD

```
MariaDB [laundry_tugasnew]> select * from barang;
+-----+-----+-----+-----+-----+-----+-----+
| id_barang | nama_barang | kategori | stok | harga_beli | satuan | tanggal_update |
+-----+-----+-----+-----+-----+-----+-----+
|      22 | deterjenn | Deterjen | 100 | 25000.00 | Botol | 2024-10-18 11:32:08 |
|      23 | Pewangi Lavender | Pewangi | 80 | 15000.00 | Botol | 2024-10-13 22:31:17 |
|      24 | Setrika Uap | Peralatan | 10 | 500000.00 | Unit | 2024-10-13 22:31:17 |
|      25 | Hanger Plastik | Peralatan | 500 | 1000.00 | Pcs | 2024-10-13 22:31:17 |
|      26 | Kantong Plastik | Lainnya | 1000 | 500.00 | Pcs | 2024-10-13 22:31:17 |
+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)
```

- **SELECT Tabel Detail Transaksi**

→ Menggunakan SQL

Run SQL query/queries on table db_bisnis_laundry.detail_transaksi:

```
1 SELECT * FROM detail_transaksi;
```

	id_detail_transaksi	id_transaksi	id_layanan	jumlah	subtotal
1	71	61	41	3	21000.00
2	72	61	42	2	20000.00
3	73	62	43	5	75000.00
4	74	63	44	4	100000.00
5	75	64	42	6	60000.00

Show query box

Showing rows 0 - 4 (5 total, Query took 0.0003 seconds.)

```
SELECT * FROM detail_transaksi;
```

	id_detail_transaksi	id_transaksi	id_layanan	jumlah	subtotal
1	71	61	41	3	21000.00
2	72	61	42	2	20000.00
3	73	62	43	5	75000.00
4	74	63	44	4	100000.00
5	75	64	42	6	60000.00

→ Menggunakan Shell atau CMD

```
MariaDB [laundry_tugasnew]> select * from detail_transaksi;
+-----+-----+-----+-----+
| id_detail_transaksi | id_transaksi | id_layanan | jumlah | subtotal |
+-----+-----+-----+-----+
| 71 | 61 | 41 | 3 | 21000.00 |
| 72 | 61 | 42 | 2 | 20000.00 |
| 73 | 62 | 43 | 5 | 75000.00 |
| 74 | 63 | 44 | 4 | 100000.00 |
| 75 | 64 | 42 | 6 | 60000.00 |
+-----+-----+-----+-----+
5 rows in set (0.001 sec)
```

- **SELECT Tabel Karyawan**
→ Menggunakan SQL

Run SQL query/queries on table db_bisnis_laundry.karyawan:

```
1 SELECT * FROM karyawan;
```

id_karyawan
nama_karyawan
jabatan
alamat
no_telp
email
tanggal_bergabung
gaji

Show query box

Showing rows 0 - 4 (total, Query took 0.0003 seconds.)

```
SELECT * FROM karyawan;
```

	id_karyawan	nama_karyawan	jabatan	alamat	no_telp	email	tanggal_bergabung	gaji
1	31	Budi Santoso	Operator Mesin	Jl. Merdeka No. 123	085678901234	budi@email.com	2023-01-15	3500000.00
2	32	Siti Rahayu	Kasir	Jl. Pahlawan No. 45	085678901235	siti@email.com	2023-02-01	3200000.00
3	33	Eko Prasetyo	operator packing	Jl. Sudirman No. 67	085678901236	eko@email.com	2023-03-10	3000000.00
4	34	Dewi Lestari	Customer Service	Jl. Gajah Mada No. 89	085678901237	dewi@email.com	2023-04-05	3300000.00
5	35	Agus Widodo	Kurir	Jl. Diponegoro No. 12	085678901238	agus@email.com	2023-05-20	3100000.00

→ Menggunakan Shell atau CMD

```
MariaDB [laundry_tugasnew]> select * from karyawan;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_karyawan | nama_karyawan | jabatan | alamat | no_telp | email | tanggal_bergabung | gaji |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 31 | Budi Santoso | Operator Mesin | Jl. Merdeka No. 123 | 085678901234 | budi@email.com | 2023-01-15 | 3500000 |
| 32 | Siti Rahayu | Kasir | Jl. Pahlawan No. 45 | 085678901235 | siti@email.com | 2023-02-01 | 3200000 |
| 33 | Eko Prasetyo | Pengepakan | Jl. Sudirman No. 67 | 085678901236 | eko@email.com | 2023-03-10 | 3000000 |
| 34 | Dewi Lestari | Customer Service | Jl. Gajah Mada No. 89 | 085678901237 | dewi@email.com | 2023-04-05 | 3300000 |
| 35 | Agus Widodo | Kurir | Jl. Diponegoro No. 12 | 085678901238 | agus@email.com | 2023-05-20 | 3100000 |
+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.060 sec)
```

- **SELECT Tabel Keluhan**

→ Menggunakan SQL

The screenshot shows the phpMyAdmin interface. On the left, there's a tree view of databases and tables. The main area has a SQL query window with the command 'SELECT * FROM keluhan;'. To the right, the results of the query are displayed in a table format.

	<code>id_keluhan</code>	<code>id_transaksi</code>	<code>id_pelanggan</code>	<code>tanggal_keluhan</code>	<code>deskripsi_keluhan</code>	<code>status</code>	<code>tanggal_penyeleseian</code>	<code>solusi</code>
1	81	61	51	2023-09-03 09:00:00	Pakaian masih berbau setelah dicuci	Selesai	2023-09-03 14:00:00	Pakaian dicuci ulang dengan tambahan pewangi
2	82	61	52	2023-09-04 10:30:00	Setrika kurang rapi pada bagian kerah	Selesai	2023-09-04 13:00:00	Pakaian disetrika ulang dengan fokus pada bagian k...
3	83	62	53	2023-09-05 11:45:00	Ada noda yang belum hilang	Sedang ditangani	NULL	NULL
4	84	63	54	2023-09-06 14:20:00	Pakaian tertukar dengan milik pelanggan lain	Selesai	2023-09-06 16:00:00	Pakaian ditukar kembali setelah dikonfirmasi dengan...
5	85	64	55	2023-09-07 09:10:00	Waktu pengerjaan melebihi estimasi	Belum ditangani	NULL	NULL

The screenshot shows a web-based MySQL interface. A red box highlights the SQL query 'SELECT * FROM keluhan;'. Below it, the results are displayed in a table.

	<code>id_keluhan</code>	<code>id_transaksi</code>	<code>id_pelanggan</code>	<code>tanggal_keluhan</code>	<code>deskripsi_keluhan</code>	<code>status</code>	<code>tanggal_penyeleseian</code>	<code>solusi</code>
1	81	61	51	2023-09-03 09:00:00	Pakaian masih berbau setelah dicuci	Selesai	2023-09-03 14:00:00	Pakaian dicuci ulang dengan tambahan pewangi
2	82	61	52	2023-09-04 10:30:00	Setrika kurang rapi pada bagian kerah	Selesai	2023-09-04 13:00:00	Pakaian disetrika ulang dengan fokus pada bagian k...
3	83	62	53	2023-09-05 11:45:00	Ada noda yang belum hilang	Sedang ditangani	NULL	NULL
4	84	63	54	2023-09-06 14:20:00	Pakaian tertukar dengan milik pelanggan lain	Selesai	2023-09-06 16:00:00	Pakaian ditukar kembali setelah dikonfirmasi dengan...
5	85	64	55	2023-09-07 09:10:00	Waktu pengerjaan melebihi estimasi	Belum ditangani	NULL	NULL

→ Menggunakan Shell atau CMD

```
MariaDB [laundry_tugasnew]> select * from keluhan;
+-----+-----+-----+-----+-----+-----+-----+
| id_keluhan | id_transaksi | id_pelanggan | tanggal_keluhan | deskripsi_keluhan | status | tanggal_penyeleseian |
+-----+-----+-----+-----+-----+-----+-----+
| 81 | 61 | 51 | 2023-09-03 09:00:00 | Pakaian masih berbau setelah dicuci | Selesai |
| 82 | 61 | 52 | 2023-09-04 10:30:00 | Setrika kurang rapi pada bagian kerah | Selesai |
| 83 | 62 | 53 | 2023-09-05 11:45:00 | Ada noda yang belum hilang | Sedang ditangani |
| NULL |
| 84 | 63 | 54 | 2023-09-06 14:20:00 | Pakaian tertukar dengan milik pelanggan lain | Selesai |
| 2023-09-06 16:00:00 | Pakaian ditukar kembali setelah dikonfirmasi dengan pelanggan terkait | Selesai |
| 85 | 64 | 55 | 2023-09-07 09:10:00 | Waktu pengerjaan melebihi estimasi | Belum ditangani |
| NULL |
+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.000 sec)
```

- **SELECT Tabel Layanan**
→ Menggunakan SQL

Screenshot of phpMyAdmin interface showing the database structure and a query execution window.

The left sidebar shows the database structure:

```

        Baru
        db
        db_account
        db_bapis
        db_bapis1
        db_bisnis_laundry
        Baru
        admin
        barang
        detail_transaksi
        karyawan
        kuhulan
        layanan
        pelanggan
        transaksi
        db_lathar07
        db_lathar1
        db_lathar3
        db_lathar4
        db_laundrfazz
        db_laundry
    
```

The main area shows a query window with the following content:

```

    Run SQL query/queries on table db_bisnis_laundry.layanan:
    1 SELECT * FROM layanan;
    
```

Below the query window are buttons for SELECT*, SELECT, INSERT, UPDATE, DELETE, Bersihkan, Format, Get auto-saved query, Bind parameters, and a search field for marking the query.

Screenshot of a web browser displaying the results of a SQL query in phpMyAdmin.

The URL is localhost/phpmyadmin/index.php?route=/database/sql&db=bisnislaunder

The query executed is:

```

    SELECT * FROM layanan;
    
```

The results table shows the following data:

	id_layanan	nama_layanan	deskripsi	harga	estimasi_waktu	status
41	Cuci Kering	Mencuci dan mengeringkan pakaian tanpa setrika	9000.00	24	Aktif	
42	Cuci Setrika	Mencuci, mengeringkan, dan menyetrika pakaian	10000.00	48	Aktif	
43	Express Cuci Setrika	Layanan cuci setrika dengan waktu selesai 6 jam	15000.00	6	Aktif	
44	Dry Clean	Pencucian khusus untuk pakaian berbahan sensitif	25000.00	72	Aktif	
45	Cuci Sepatu	Mencuci dan membersihkan sepatu	35000.00	24	Aktif	

→ Menggunakan Shell atau CMD

```

MariaDB [laundry_tugasnew]> select * from layanan;
+-----+-----+-----+-----+-----+-----+
| id_layanan | nama_layanan | deskripsi | harga | estimasi_waktu | status |
+-----+-----+-----+-----+-----+-----+
| 41 | Cuci Kering | Mencuci dan mengeringkan pakaian tanpa setrika | 7000.00 | 24 | Aktif |
| 42 | Cuci Setrika | Mencuci, mengeringkan, dan menyetrika pakaian | 10000.00 | 48 | Aktif |
| 43 | Express Cuci Setrika | Layanan cuci setrika dengan waktu selesai 6 jam | 15000.00 | 6 | Aktif |
| 44 | Dry Clean | Pencucian khusus untuk pakaian berbahan sensitif | 25000.00 | 72 | Aktif |
| 45 | Cuci Sepatu | Mencuci dan membersihkan sepatu | 35000.00 | 24 | Aktif |
+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)
    
```

- **SELECT Tabel Pelanggan**

→ Menggunakan SQL

```
1 SELECT * FROM pelanggan;
```

	<code>id_pelanggan</code>	<code>nama_pelanggan</code>	<code>alamat</code>	<code>no_telp</code>	<code>email</code>	<code>tanggal_bergabung</code>	<code>poin</code>
1	51	Rudi Hermawan	Jl. Angrek No. 1	081234567890	rudi@email.com	2023-06-01	100
2	52	Nina Surya	Jl. Melati No. 23	081234567891	nina@email.com	2023-06-15	75
3	53	Fira Pratiwi	Jl. Dahlia No. 45	081234567892	fira@email.com	2023-07-01	50
4	54	Andi Kusuma	Jl. Mawar No. 67	081234567893	andi@email.com	2023-07-15	25
5	55	Lina Putri	Jl. Tulip No. 89	081234567894	lina@email.com	2023-08-01	10

```
SELECT * FROM pelanggan;
```

	<code>id_pelanggan</code>	<code>nama_pelanggan</code>	<code>alamat</code>	<code>no_telp</code>	<code>email</code>	<code>tanggal_bergabung</code>	<code>poin</code>
1	51	Rudi Hermawan	Jl. Angrek No. 1	081234567890	rudi@email.com	2023-06-01	100
2	52	Nina Surya	Jl. Melati No. 23	081234567891	nina@email.com	2023-06-15	75
3	53	Fira Pratiwi	Jl. Dahlia No. 45	081234567892	fira@email.com	2023-07-01	50
4	54	Andi Kusuma	Jl. Mawar No. 67	081234567893	andi@email.com	2023-07-15	25
5	55	Lina Putri	Jl. Tulip No. 89	081234567894	lina@email.com	2023-08-01	10

→ Menggunakan Shell atau CMD

```
MariaDB [laundry_tugasnew]> select * from pelanggan;
+-----+-----+-----+-----+-----+-----+-----+
| id_pelanggan | nama_pelanggan | alamat | no_telp | email | tanggal_bergabung | poin |
+-----+-----+-----+-----+-----+-----+-----+
| 51 | Rudi Hermawan | Jl. Angrek No. 1 | 081234567890 | rudi@email.com | 2023-06-01 | 100 |
| 52 | Nina Surya | Jl. Melati No. 23 | 081234567891 | nina@email.com | 2023-06-15 | 75 |
| 53 | Fira Pratiwi | Jl. Dahlia No. 45 | 081234567892 | fira@email.com | 2023-07-01 | 50 |
| 54 | Andi Kusuma | Jl. Mawar No. 67 | 081234567893 | andi@email.com | 2023-07-15 | 25 |
| 55 | Lina Putri | Jl. Tulip No. 89 | 081234567894 | lina@email.com | 2023-08-01 | 10 |
+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)
```

- **SELECT Tabel Transaksi**

→ Menggunakan Shell

The screenshot shows the phpMyAdmin interface. On the left, there's a tree view of databases and tables. The main area has a SQL query editor with the query 'SELECT * FROM transaksi;'. To the right, a results pane displays the schema of the 'transaksi' table:

```

id_transaksi
id_pelanggan
id_karyawan
tanggal_masuk
tanggal_selesai
status
total_harga
metode_pembayaran

```

The screenshot shows the MySQL Workbench interface. It's similar to the phpMyAdmin one, with a tree view on the left and a results grid on the right. The results grid shows the same data as the previous screenshot. A red arrow points to the 'Edit inline' button in the toolbar.

→ Menggunakan Shell atau CMD

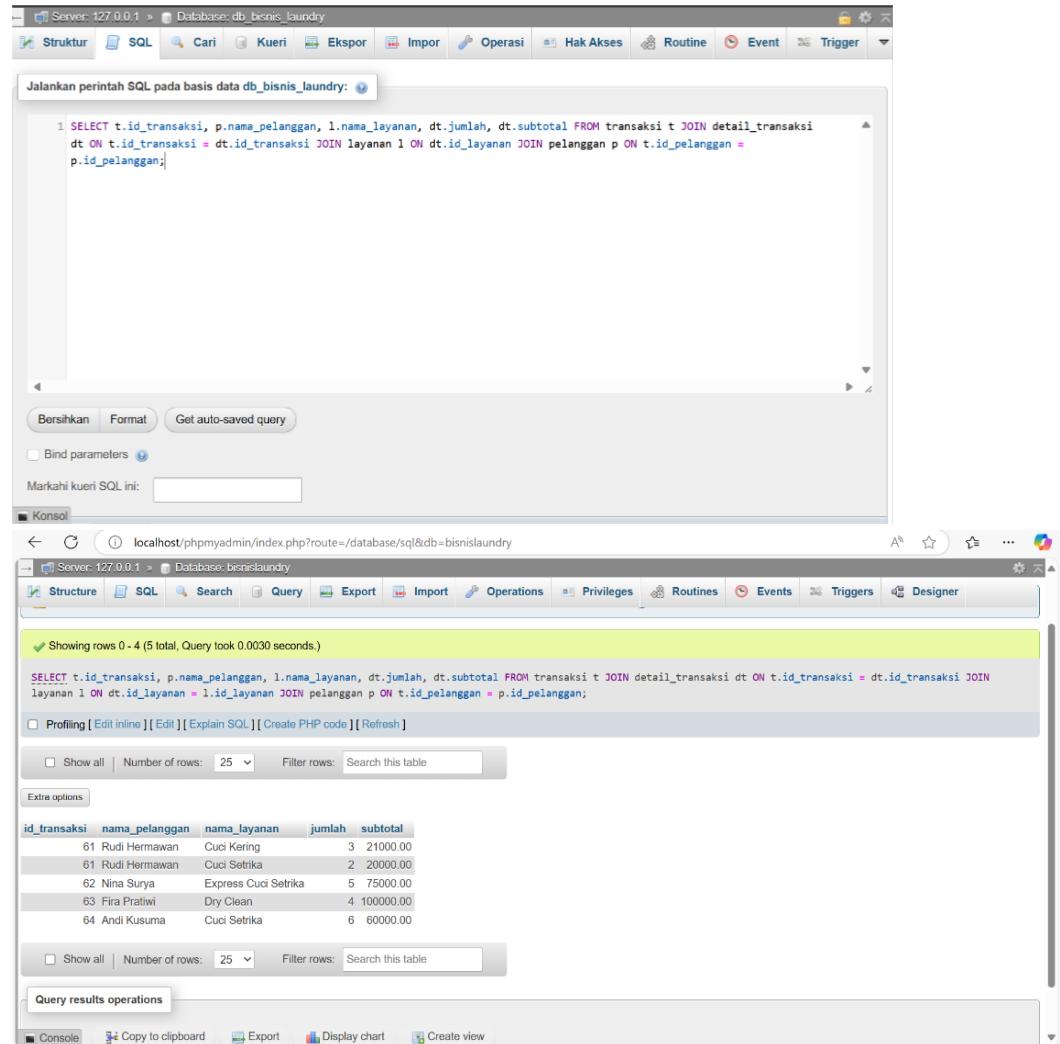
```

MariaDB [laundry_tugasnew]> select * from transaksi;
+-----+-----+-----+-----+-----+-----+-----+
| id_transaksi | id_pelanggan | id_karyawan | tanggal_masuk | tanggal_selesai | status | total_harga | metode_pembayaran |
+-----+-----+-----+-----+-----+-----+-----+
|       61 |         51 |         34 | 2023-09-01 10:00:00 | 2023-09-02 15:00:00 | Selesai |   50000.00 | Tunai
|       62 |         52 |         34 | 2023-09-02 11:00:00 | 2023-09-03 16:00:00 | Selesai |   72000.00 | Transfer
|       63 |         53 |         34 | 2023-09-03 09:00:00 | 2023-09-04 14:00:00 | Diambil |  99000.00 | Kartu Kredit
|       64 |         54 |         34 | 2023-09-04 13:00:00 | NULL             | Proses  |   60000.00 | Tunai
|       65 |         55 |         34 | 2023-09-05 14:00:00 | NULL             | Proses  |   80000.00 | Transfer
+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.006 sec)

```

Gambar 3. 7 SQL dan Bukti Select Setiap Tabel

- JOIN Tabel Detail Transaksi, Layanan, dan Pelanggan
Menampilkan daftar transaksi beserta rincian layanan yang digunakan



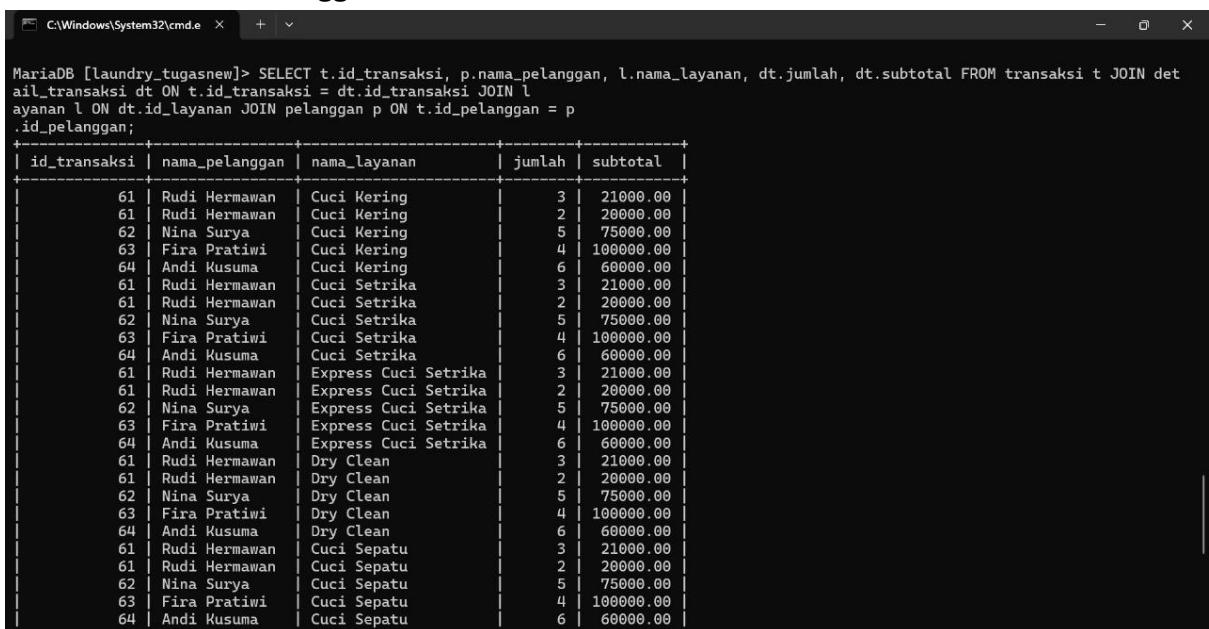
The screenshot shows the phpMyAdmin interface with the following details:

- Top Bar:** Server: 127.0.0.1 > Database: db_bisnis_laundry. The tabs include Struktur, SQL, Cari, Kueri, Eksport, Impor, Operasi, Hak Akses, Routine, Event, Trigger.
- SQL Tab:** Contains the following SQL query:


```
1 SELECT t.id_transaksi, p.nama_pelanggan, l.nama_layanan, dt.jumlah, dt.subtotal FROM transaksi t JOIN detail_transaksi dt ON t.id_transaksi = dt.id_transaksi JOIN layanan l ON dt.id_layanan = l.id_layanan JOIN pelanggan p ON t.id_pelanggan = p.id_pelanggan;
```
- Buttons:** Bersihkan, Format, Get auto-saved query, Bind parameters.
- Message:** Markahi kueri SQL ini: []
- Konsol Tab:** Shows the results of the executed query. The output message is: Showing rows 0 - 4 (5 total). The query took 0.0030 seconds.


```
SELECT t.id_transaksi, p.nama_pelanggan, l.nama_layanan, dt.jumlah, dt.subtotal FROM transaksi t JOIN detail_transaksi dt ON t.id_transaksi = dt.id_transaksi JOIN layanan l ON dt.id_layanan = l.id_layanan JOIN pelanggan p ON t.id_pelanggan = p.id_pelanggan;
```
- Table Results:** Displays the joined data from three tables. The columns are id_transaksi, nama_pelanggan, nama_layanan, jumlah, and subtotal. The data includes transactions for Rudi Hermawan, Nina Surya, Fira Pratiwi, and Andi Kusuma across various laundry services like Cuci Kering, Setrika, Express, and Dry Clean.

→ Menggunakan CMD



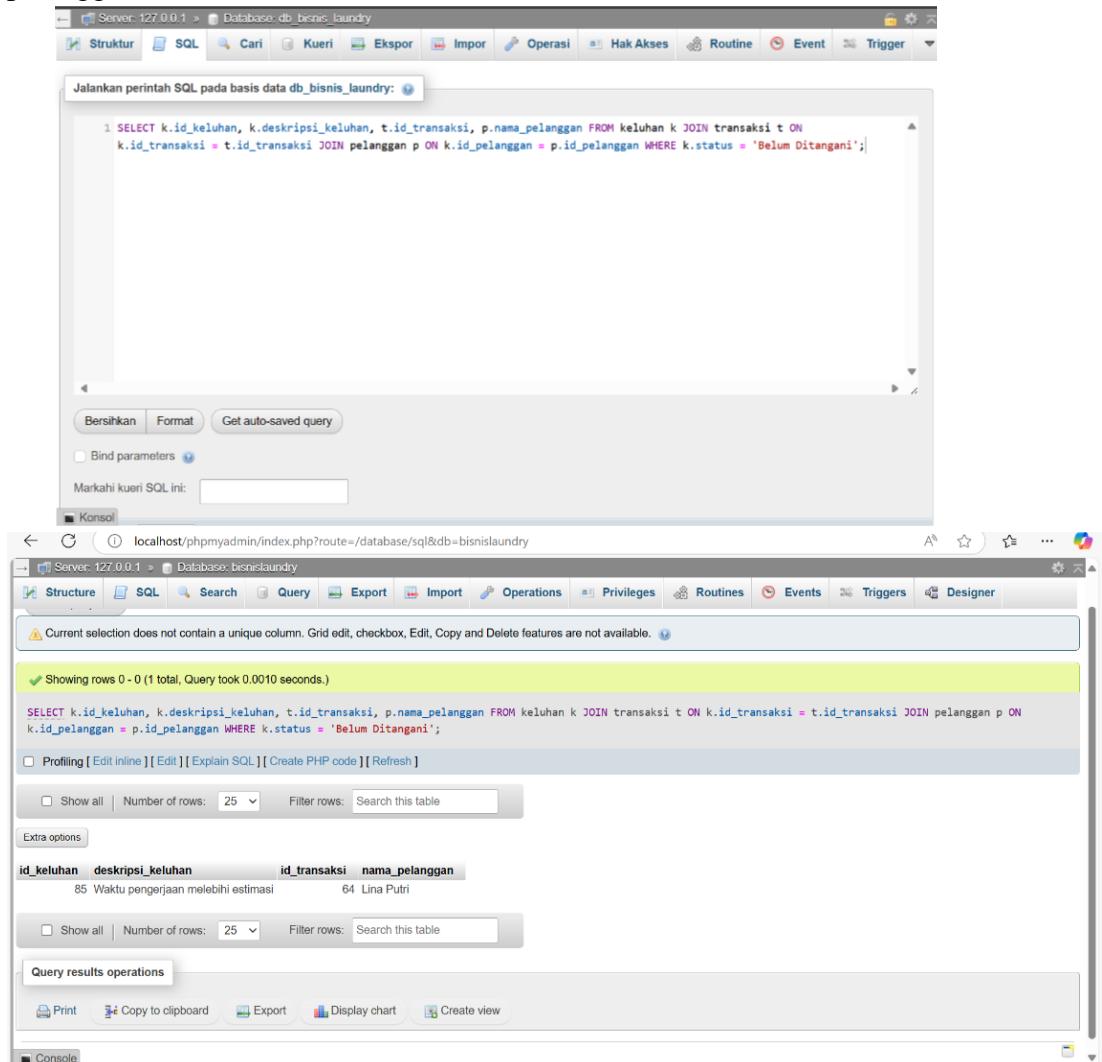
The screenshot shows a Windows Command Prompt window titled 'C:\Windows\System32\cmd.e' running the MariaDB command-line client. The query and its results are identical to those shown in the phpMyAdmin screenshot above.

```
MariaDB [laundry_tugasnew]> SELECT t.id_transaksi, p.nama_pelanggan, l.nama_layanan, dt.jumlah, dt.subtotal FROM transaksi t JOIN detail_transaksi dt ON t.id_transaksi = dt.id_transaksi JOIN layanan l ON dt.id_layanan = l.id_layanan JOIN pelanggan p ON t.id_pelanggan = p.id_pelanggan;
+-----+-----+-----+-----+-----+
| id_transaksi | nama_pelanggan | nama_layanan | jumlah | subtotal |
+-----+-----+-----+-----+-----+
|       61 | Rudi Hermawan | Cuci Kering |     3 | 21000.00 |
|       61 | Rudi Hermawan | Cuci Kering |     2 | 20000.00 |
|       62 | Nina Surya    | Cuci Kering |     5 | 75000.00 |
|       63 | Fira Pratiwi  | Cuci Kering |     4 | 100000.00 |
|       64 | Andi Kusuma   | Cuci Kering |     6 | 60000.00 |
|       61 | Rudi Hermawan | Cuci Setrika |    3 | 21000.00 |
|       61 | Rudi Hermawan | Cuci Setrika |    2 | 20000.00 |
|       62 | Nina Surya    | Cuci Setrika |    5 | 75000.00 |
|       63 | Fira Pratiwi  | Cuci Setrika |    4 | 100000.00 |
|       64 | Andi Kusuma   | Cuci Setrika |    6 | 60000.00 |
|       61 | Rudi Hermawan | Express Cuci Setrika | 3 | 21000.00 |
|       61 | Rudi Hermawan | Express Cuci Setrika | 2 | 20000.00 |
|       62 | Nina Surya    | Express Cuci Setrika | 5 | 75000.00 |
|       63 | Fira Pratiwi  | Express Cuci Setrika | 4 | 100000.00 |
|       64 | Andi Kusuma   | Express Cuci Setrika | 6 | 60000.00 |
|       61 | Rudi Hermawan | Dry Clean | 3 | 21000.00 |
|       61 | Rudi Hermawan | Dry Clean | 2 | 20000.00 |
|       62 | Nina Surya    | Dry Clean | 5 | 75000.00 |
|       63 | Fira Pratiwi  | Dry Clean | 4 | 100000.00 |
|       64 | Andi Kusuma   | Dry Clean | 6 | 60000.00 |
|       61 | Rudi Hermawan | Cuci Sepatu | 3 | 21000.00 |
|       61 | Rudi Hermawan | Cuci Sepatu | 2 | 20000.00 |
|       62 | Nina Surya    | Cuci Sepatu | 5 | 75000.00 |
|       63 | Fira Pratiwi  | Cuci Sepatu | 4 | 100000.00 |
|       64 | Andi Kusuma   | Cuci Sepatu | 6 | 60000.00 |
```

- **DOUBLE JOIN**

Join Tabel Keluhan, Transaksi, dan Pelanggan

Menampilkan keluhan yang belum ditangani beserta data transaksi dan pelanggan terkait.



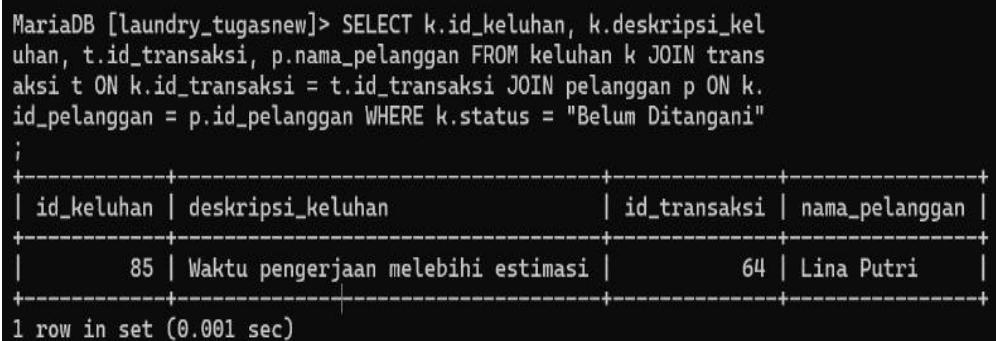
The screenshot shows two windows of MySQL Workbench. The top window is titled 'Jalankan perintah SQL pada basis data db_bisnis_laundry:' and contains the following SQL query:

```
1 SELECT k.id_keluhan, k.deskripsi_keluhan, t.id_transaksi, p.nama_pelanggan FROM keluhan k JOIN transaksi t ON k.id_transaksi = t.id_transaksi JOIN pelanggan p ON k.id_pelanggan = p.id_pelanggan WHERE k.status = 'Belum Ditangani';
```

The bottom window is titled 'localhost/phpmyadmin/index.php?route=/database/sql&db=bisnislaundry' and shows the results of the query execution:

id_keluhan	deskripsi_keluhan	id_transaksi	nama_pelanggan
85	Waktu pengerjaan melebihi estimasi	64	Lina Putri

→ Menggunakan Shell atau CMD



```
MariaDB [laundry_tugasnew]> SELECT k.id_keluhan, k.deskripsi_keluhan, t.id_transaksi, p.nama_pelanggan FROM keluhan k JOIN transaksi t ON k.id_transaksi = t.id_transaksi JOIN pelanggan p ON k.id_pelanggan = p.id_pelanggan WHERE k.status = "Belum Ditangani"
;
+-----+-----+-----+
| id_keluhan | deskripsi_keluhan | id_transaksi | nama_pelanggan |
+-----+-----+-----+
| 85 | Waktu pengerjaan melebihi estimasi | 64 | Lina Putri |
+-----+-----+-----+
1 row in set (0.001 sec)
```

Gambar 3. 8 SQL dan Bukti Double Join tabel

- TRIPLE JOIN

Pada table pelanggan, table transaksi dan table karyawan lakukan select terlebih dahulu untuk menampilkan ketiga table tersebut kemudian lakukan FROM untuk mengambil data dari table ‘pelanggan’ dengan (p), JOIN untuk melakukan join antara table pelanggan dengan table transaksi berdasarkan id pelanggan lalu JOIN untuk table karyawan dan transaksi berdasarkan id karyawan dimana dengan nama pelanggan misalnya : Nina Surya dan Andi Kusuma.

❖ Triple Join
→ Menggunakan SQL

The screenshot shows two instances of the phpMyAdmin interface. Both instances have the same database structure on the left sidebar, which includes tables such as db, db_account, db_bapis, db_bapis1, db_bisnis_laundry, Baru, admin, barang, detail_transaksi, karyawan, keluhan, layanan, pelanggan, and transaksi. The top instance shows a SQL query window with the following code:

```

1 SELECT
2     p.id_pelanggan,
3     p.nama_pelanggan,
4     t.id_transaksi,
5     k.id_karyawan,
6     k.nama_karyawan
7 FROM
8     pelanggan p
9 JOIN
10    transaksi t ON p.id_pelanggan = t.id_pelanggan
11 JOIN
12    karyawan k ON t.id_karyawan = k.id_karyawan WHERE
13 p.nama_pelanggan = 'Nina Surya';

```

The bottom instance shows the results of the query in the results window:

```

Menampilkan baris 0 - 0 (total 1, Pencarian dilakukan dalam 0,0004 detik.)

SELECT p.id_pelanggan, p.nama_pelanggan, t.id_transaksi, k.id_karyawan, k.nama_karyawan FROM pelanggan p JOIN transaksi t ON p.id_pelanggan = t.id_pelanggan JOIN karyawan k ON t.id_karyawan = k.id_karyawan WHERE p.nama_pelanggan = 'Nina Surya';

id_pelanggan  nama_pelanggan  id_transaksi  id_karyawan  nama_karyawan
52  Nina Surya          62            34  Dewi Lestari

```

Gambar 3. 9 SQL dan Bukti Triple Join

❖ Left Join

```
MariaDB [laundry_tugasnew]> select * from pelanggan LEFT JOIN keluhan ON pelanggan.id_pelanggan = keluhan.id_keluhan;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id_pelanggan | nama_pelanggan | alamat | no_telp | email | tanggal_bergabung | poin | id_keluhan | id_transaksi |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 51 | Rudi Hermawan | Jl. Anggrek No. 1 | 081234567890 | rudi@email.com | 2023-06-01 | 100 | NULL |
| 52 | Nina Surya | Jl. Melati No. 23 | 081234567891 | nina@email.com | 2023-06-15 | 75 | NULL |
| 53 | Fira Pratiwi | Jl. Dahlia No. 45 | 081234567892 | fira@email.com | 2023-07-01 | 50 | NULL |
| 54 | Andi Kusuma | Jl. Mawar No. 67 | 081234567893 | andi@email.com | 2023-07-15 | 25 | NULL |
| 55 | Lina Putri | Jl. Tulip No. 89 | 081234567894 | lina@email.com | 2023-08-01 | 10 | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.001 sec)

MariaDB [laundry_tugasnew]>
```

❖ Right Join

```
MariaDB [laundry_tugasnew]> select * from karyawan RIGHT JOIN layanan ON karyawan.id_karyawan = layanan.id_layanan;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id_karyawan | nama_karyawan | jabatan | alamat | no_telp | email | tanggal_bergabung | gaji | id_layanan | nama_layanan | d |
| eskripsi |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| NULL | NULL | NULL | NULL | NULL | NULL | 24 | Aktif | 41 | Cuci Kering | M |
encuci dan mengeringkan pakaian tanpa setrika | 7000.00 |
| NULL | NULL | NULL | NULL | NULL | NULL | 48 | Aktif | 42 | Cuci Setrika | M |
encuci, mengeringkan, dan menyetrika pakaian | 10000.00 |
| NULL | NULL | NULL | NULL | NULL | NULL | 6 | Aktif | 43 | Express Cuci Setrika | L |
ayanan cuci setrika dengan waktu selesai 6 jam | 15000.00 |
| NULL | NULL | NULL | NULL | NULL | NULL | 72 | Aktif | 44 | Dry Clean | P |
encucian khusus untuk pakaian berbahan sensitif | 25000.00 |
| NULL | NULL | NULL | NULL | NULL | NULL | 24 | Aktif | 45 | Cuci Sepatu | M |
encuci dan membersihkan sepatu | 35000.00 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.012 sec)
```

❖ Inner Join

```
C:\Windows\System32\cmd.e > + | 
MariaDB [db_bisnis_laundry]> select * from transaksi INNER JOIN detail_transaksi ON transaksi.id_transaksi = detail_transaksi.id_detail_transaksi;
Empty set (0.001 sec)

MariaDB [db_bisnis_laundry]> select * from detail_transaksi INNER JOIN transaksi ON detail_transaksi.id_detail_transaksi = transaksi.id_transaksi;
Empty set (0.001 sec)

MariaDB [db_bisnis_laundry]> SELECT detail_transaksi.* ,transaksi.* FROM detail_transaksi INNER JOIN transaksi ON detail_transaksi.id_transaksi = transaksi.id_transaksi;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id_detail_transaksi | id_transaksi | id_layanan | jumlah | subtotal | id_transaksi | id_pelanggan | id_karyawan | tanggal_masuk |
| tanggal_selesai | status | total_harga | metode_pembayaran |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 71 | 2023-09-02 15:00:00 | Selesai | 50000.00 | Tunai | 61 | 51 | 34 | 2023-09-01 10:00 |
| 72 | 2023-09-02 15:00:00 | Selesai | 50000.00 | Tunai | 61 | 51 | 34 | 2023-09-01 10:00 |
| 73 | 2023-09-03 16:00:00 | Selesai | 75000.00 | Transfer | 62 | 52 | 34 | 2023-09-02 11:00 |
| 74 | 2023-09-04 14:00:00 | Dibambil | 100000.00 | Kartu Kredit | 63 | 53 | 34 | 2023-09-03 09:00 |
| 75 | 2023-09-04 14:00:00 | Proses | 60000.00 | Tunai | 64 | 54 | 34 | 2023-09-04 13:00 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.002 sec)
```

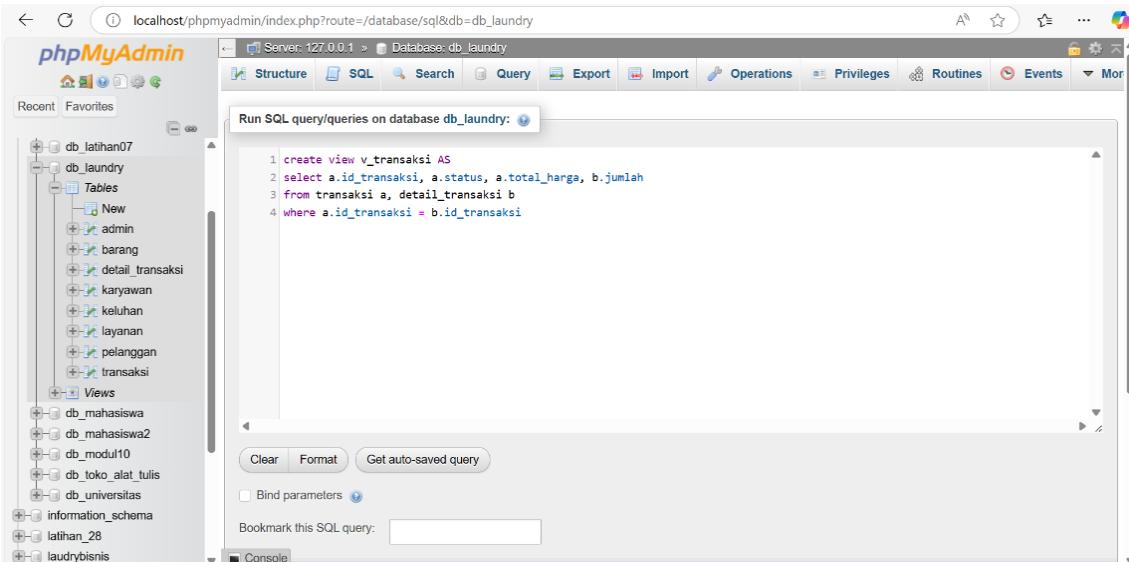
BAB IV

VIEW

4.1 View

Pada MySQL dapat dikatakan sebagai virtual tabel atau *stored query* yang dibuat dengan *query* SQL SELECT. Data yang terdapat pada *view* tidak disimpan dalam database sebagai objek yang nyata, hanya perintah SELECTnya yang disimpan dalam database. Hasil dari SELECT tersebut yang membentuk tabel virtual yang ditampilkan pada *view* yang telah dibuat.

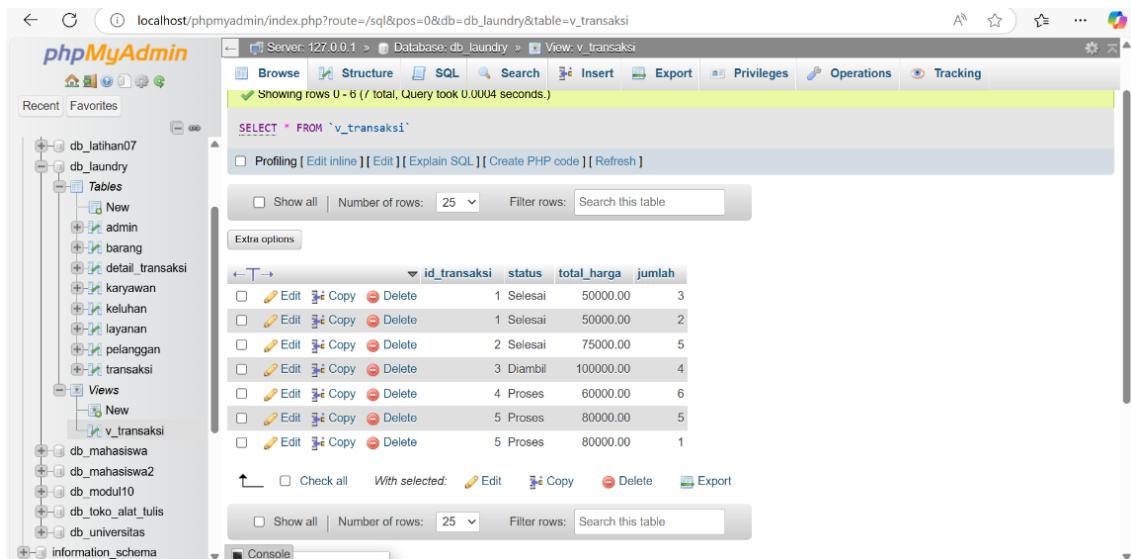
4.1.1 Membuat view



The screenshot shows the phpMyAdmin interface for the 'db_laundry' database. In the left sidebar, under 'Views', there is a new entry for 'v_transaksi'. The main area displays the following SQL code:

```
1 create view v_transaksi AS
2 select a.id_transaksi, a.status, a.total_harga, b.jumlah
3 from transaksi a, detail_transaksi b
4 where a.id_transaksi = b.id_transaksi
```

4.1.2 Select view transaksi

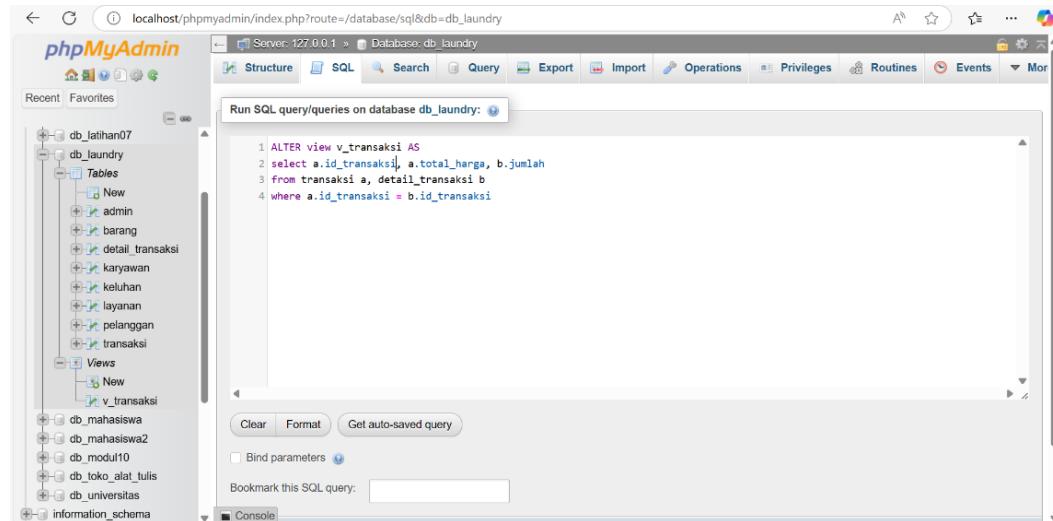


The screenshot shows the phpMyAdmin interface displaying the contents of the 'v_transaksi' view. The table structure is as follows:

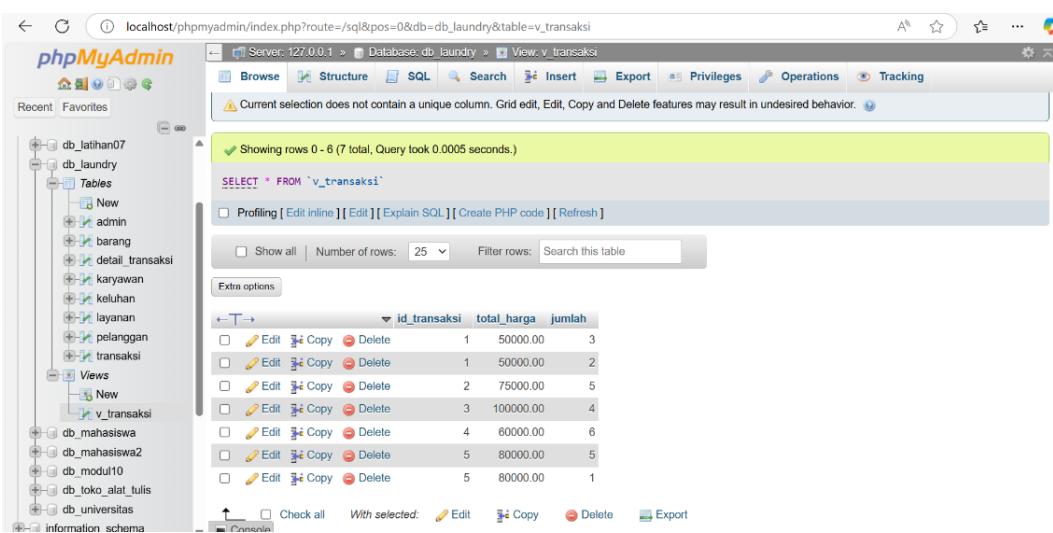
	id_transaksi	status	total_harga	jumlah
<input type="checkbox"/>	1	Selesai	50000.00	3
<input type="checkbox"/>	1	Selesai	50000.00	2
<input type="checkbox"/>	2	Selesai	75000.00	5
<input type="checkbox"/>	3	Dambil	100000.00	4
<input type="checkbox"/>	4	Proses	60000.00	6
<input type="checkbox"/>	5	Proses	80000.00	5
<input type="checkbox"/>	5	Proses	80000.00	1

Gambar 4. 1 SQL dan Bukti View

4.1.3 Mengubah view dengan perintah ALTER

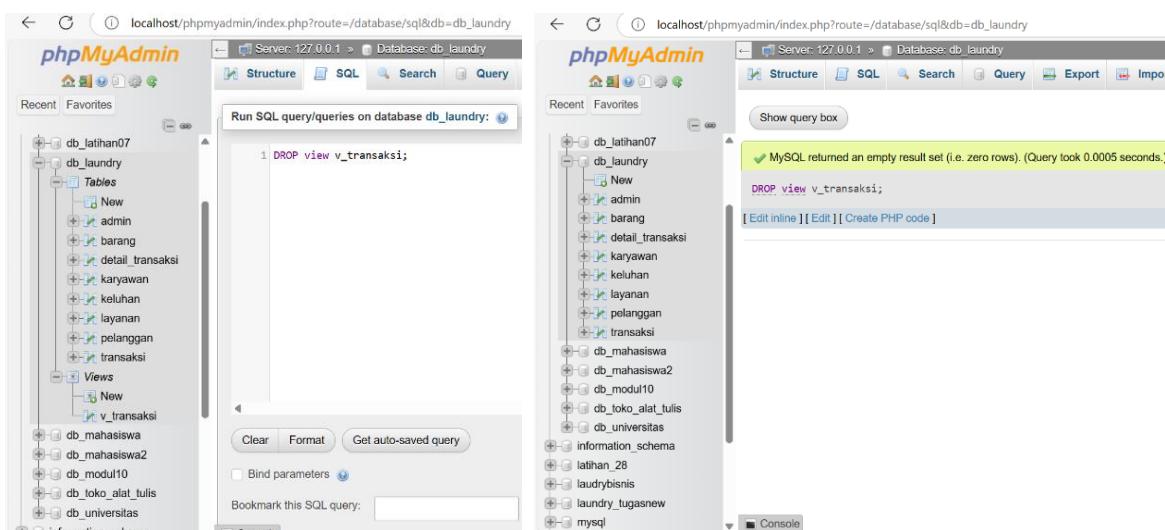


```
ALTER view v_transaksi AS
select a.id_transaksi, a.total_harga, b.jumlah
from transaksi a, detail_transaksi b
where a.id_transaksi = b.id_transaksi
```



	id_transaksi	total_harga	jumlah
<input type="checkbox"/>	1	50000.00	3
<input type="checkbox"/>	1	50000.00	2
<input type="checkbox"/>	2	75000.00	5
<input type="checkbox"/>	3	100000.00	4
<input type="checkbox"/>	4	60000.00	6
<input type="checkbox"/>	5	80000.00	5
<input type="checkbox"/>	5	80000.00	1

4.1.4 Menghapus view dengan perintah DROP



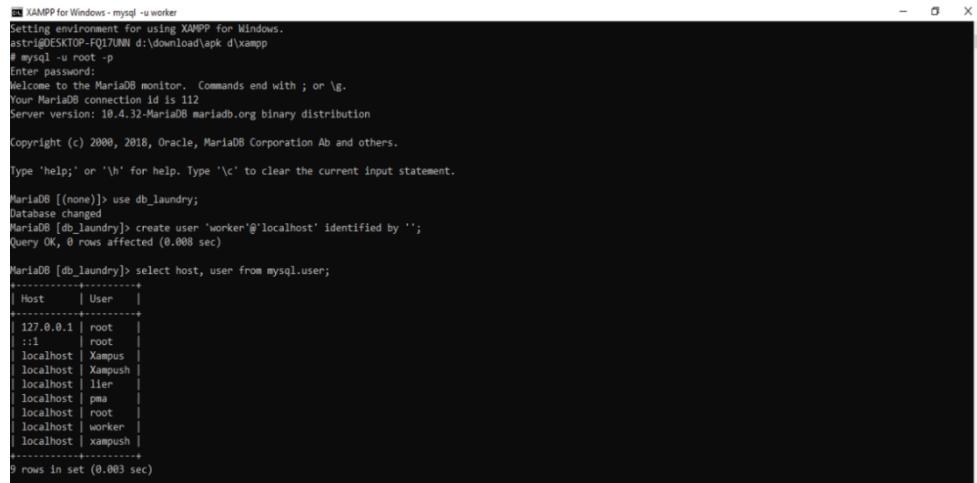
```
DROP view v_transaksi;
```

BAB V

KEAMANAN BASIS DATA

5.1 Pembuatan pengguna

5.1.1 Perintah CREATE USER



```
Setting environment for using XAMPP for Windows.
asti@DESKTOP-FQ17UNN d:/download/apk/dxampp
# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 112
Server version: 10.4.32-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> use db_laundry;
Database changed
MariaDB [db_laundry]> create user 'worker'@'localhost' identified by '';
Query OK, 0 rows affected (0.008 sec)

MariaDB [db_laundry]> select host, user from mysql.user;
+-----+-----+
| Host | User |
+-----+-----+
| 127.0.0.1 | root |
| ::1 | root |
| localhost | Xampus |
| localhost | Xampush |
| localhost | llieR |
| localhost | pma |
| localhost | root |
| localhost | worker |
| localhost | xampush |
+-----+-----+
9 rows in set (0.003 sec)
```

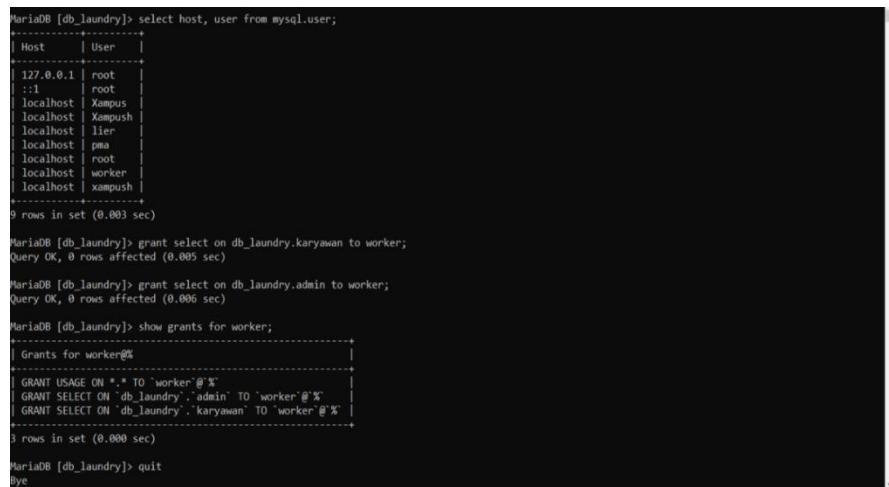
Gambar 5. 1 Query Create User

5.1.2 Pemberian Hak akses/ Priviledges

Data Control Language (DCL) adalah perintah yang digunakan untuk keperluan keamanan (security) database dengan membuat hak akses tertentu bagi setiap user. Statement pada DCL antara lain grant, revoke, set dan lock table. [8]

❖ Privileges

Pemberian hak akses (privileges) adalah proses mengatur izin pengguna untuk berinteraksi dengan objek-objek database. Ini merupakan mekanisme keamanan penting untuk melindungi data dan membatasi akses sesuai kebutuhan. [9]



```
MariaDB [db_laundry]> select host, user from mysql.user;
+-----+-----+
| Host | User |
+-----+-----+
| 127.0.0.1 | root |
| ::1 | root |
| localhost | Xampus |
| localhost | Xampush |
| localhost | llieR |
| localhost | pma |
| localhost | root |
| localhost | worker |
| localhost | xampush |
+-----+-----+
9 rows in set (0.003 sec)

MariaDB [db_laundry]> grant select on db_laundry.karyawan to worker;
Query OK, 0 rows affected (0.005 sec)

MariaDB [db_laundry]> grant select on db_laundry.admin to worker;
Query OK, 0 rows affected (0.006 sec)

MariaDB [db_laundry]> show grants for worker;
+-----+
| Grants for worker@% |
+-----+
| GRANT USAGE ON *.* TO 'worker'@'%'
| GRANT SELECT ON `db_laundry`.`admin` TO 'worker'@'%'
| GRANT SELECT ON `db_laundry`.`karyawan` TO 'worker'@'%'
+-----+
3 rows in set (0.000 sec)

MariaDB [db_laundry]> quit
Bye
```

Gambar 5. 2 Query Privileges

❖ Role

Role adalah mekanisme pengaturan hak akses yang memungkinkan pengelompokan izin-izin tertentu untuk memudahkan manajemen keamanan database. [9]

```
MariaDB [(none)]> create role 'owner';
Query OK, 0 rows affected (0.008 sec)

MariaDB [(none)]> select host, user from mysql.user;
+-----+-----+
| Host | User |
+-----+-----+
| %    | owner  |
| 127.0.0.1 | worker |
| :1   | root   |
| localhost | Xampus |
| localhost | Xampush |
| localhost | ller   |
| localhost | pma   |
| localhost | root   |
| localhost | worker |
| localhost | xampush |
+-----+-----+
11 rows in set (0.001 sec)

MariaDB [(none)]> grant insert, update, delete on db_laundry.* to 'owner';
Query OK, 0 rows affected (0.008 sec)

MariaDB [(none)]> grant 'owner' to 'worker';
Query OK, 0 rows affected (0.005 sec)

MariaDB [(none)]> show grants for worker;
+-----+
| Grants for worker@% |
+-----+
| GRANT 'owner' TO 'worker'@'%'
| GRANT USAGE ON *.* TO 'worker'@'%'
| GRANT SELECT ON `db_laundry`.`admin` TO 'worker'@'%'
| GRANT SELECT ON `db_laundry`.`karyawan` TO 'worker'@'%'
+-----+
4 rows in set (0.000 sec)

MariaDB [(none)]> quit
Bye

astrid@DESKTOP-FQ17UNN d:\download\apk d\xampp
# mysql -u worker
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 115
Server version: 10.4.32-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> select current_role();
+-----+
| current_role() |
+-----+
| NULL          |
+-----+
1 row in set (0.000 sec)

MariaDB [(none)]> set role owner;
Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> select current_role();
+-----+
| current_role() |
+-----+
| owner          |
+-----+
1 row in set (0.000 sec)
```

Gambar 5. 3 Query Role

BAB VI

UNDO MANAGEMENT

★ START TRANSACTION menggunakan xampp untuk memulai transaksi.

The figure consists of four vertically stacked screenshots of the phpMyAdmin interface, showing the process of starting a transaction and updating data in a MySQL database.

Screenshot 1: Shows the 'SQL' tab with the following SQL code:

```

1 START TRANSACTION;
2
3 UPDATE transaksi SET status = 'Diambil' WHERE id_transaksi = 61;
4 UPDATE transaksi SET total_harga = 80000 WHERE id_transaksi = 62;
5 UPDATE transaksi SET metode_pembayaran = 'Tunai' WHERE id_transaksi = 63;
6
7 SELECT * FROM transaksi WHERE id_transaksi IN (61, 62, 63);
    
```

Screenshot 2: Shows the results of the SQL query from Screenshot 1. The table 'transaksi' now contains the following data:

	id_transaksi	id_pelanggan	id_karyawan	tanggal_masuk	tanggal_selesai	status	total_harga	metode_pem
1	61	51	34	2023-09-01 10:00:00	2023-09-02 15:00:00	Selesai	50000.00	Tunai
2	62	52	34	2023-09-02 11:00:00	2023-09-03 16:00:00	Selesai	75000.00	Transfer
3	63	53	34	2023-09-03 09:00:00	2023-09-04 14:00:00	Diambil	100000.00	Kartu Kredit
4	64	54	34	2023-09-04 13:00:00	NULL	Proses	60000.00	Tunai
5	65	55	34	2023-09-05 14:00:00	NULL	Proses	80000.00	Transfer

Screenshot 3: Shows the 'SQL' tab with the following SQL code:

```

START TRANSACTION;

```

Screenshot 4: Shows the results of the SQL query from Screenshot 3. The table 'transaksi' now contains the following data:

	id_transaksi	id_pelanggan	id_karyawan	tanggal_masuk	tanggal_selesai	status	total_harga	metode_pem
1	61	51	34	2023-09-01 10:00:00	2023-09-02 15:00:00	Diambil	50000.00	Tunai
2	62	52	34	2023-09-02 11:00:00	2023-09-03 16:00:00	Selesai	80000.00	Transfer
3	63	53	34	2023-09-03 09:00:00	2023-09-04 14:00:00	Diambil	100000.00	Tunai

Gambar 6. 1 SQL dan Bukti Start Transaction

★ START TRANSACTION menggunakan shell untuk memulai transaksi.

```
MariaDB [bisnislaunder] > UPDATE `transaksi` SET `total_harga` = '50000' WHERE `id_transaksi` = 61;
Query OK, 0 rows affected (0.001 sec)
Rows matched: 1  Changed: 0  Warnings: 0

MariaDB [bisnislaunder] > commit;
Query OK, 0 rows affected (0.006 sec)

MariaDB [bisnislaunder] > |
```

★ Perintah SQL seperti INSERT, UPDATE, atau DELETE untuk mengubah data.
➤ INSERT

The screenshot shows the phpMyAdmin interface for the 'bisnislaunder' database. In the top navigation bar, 'Structure' and 'SQL' tabs are visible. The main area contains an SQL query window with the following code:

```
1 INSERT INTO transaksi (id_transaksi, id_pelanggan, id_karyawan, tanggal_masuk, tanggal_selesai, status, total_harga, metode_pembayaran)
VALUES
2 (61, 51, 34, '2023-09-01 10:00:00', '2023-09-02 15:00:00', 'Selesai', 50000.00, 'Tunai'),
3 (62, 52, 34, '2023-09-02 11:00:00', '2023-09-03 16:00:00', 'Selesai', 75000.00, 'Transfer'),
4 (63, 53, 34, '2023-09-03 09:00:00', '2023-09-04 14:00:00', 'Diambil', 100000.00, 'Kartu Kredit'),
5 (64, 54, 34, '2023-09-04 13:00:00', NULL, 'Proses', 60000.00, 'Tunai'),
6 (65, 55, 34, '2023-09-05 14:00:00', NULL, 'Proses', 80000.00, 'Transfer');
```

➤ UPDATE

The screenshot shows the phpMyAdmin interface for the 'bisnislaunder' database, displaying two separate update operations.

Top Window (karyawan Table):

Query: `UPDATE `karyawan` SET `jabatan`='operator packing' WHERE `id_karyawan` = 33;`

Result: 1 baris terpengaruh. (Pencarian dilakukan dalam 0,0036 detik.)

#	id_karyawan	nama_karyawan	jabatan	alamat	no_telp	email	tanggal Bergabung	gaji
31	Budi Santoso	operator mesin	Jl. Merdeka No. 123	085678901234	budi@email.com	2023-01-15	3500000.00	
32	Siti Rahayu	Kasir	Jl. Pahlawan No. 45	085678901235	siti@email.com	2023-02-01	3200000.00	
33	Eko Prasetyo	operator packing	Jl. Sudirman No. 67	085678901236	eko@email.com	2023-03-10	3000000.00	
34	Dewi Lestari	Customer Service	Jl. Gajah Mada No. 89	085678901237	dewi@email.com	2023-04-05	3300000.00	
35	Agus Widodo	Kurir	Jl. Diponegoro No. 12	085678901238	agus@email.com	2023-05-20	3100000.00	

Bottom Window (layanan Table):

Query: `UPDATE `layanan` SET `harga`='9000' WHERE `id_layanan` = 41;`

Result: 1 baris terpengaruh. (Pencarian dilakukan dalam 0,0034 detik.)

#	id_layanan	nama_layanan	deskripsi	harga	estimasi_waktu	status
41	Cuci Kering	Mencuci dan mengeringkan pakaian tanpa setrika	9000.00	Dalam jem	24	Aktif
42	Cuci Setrika	Mencuci, mengeringkan, dan menyetrika pakaian	10000.00	Dalam jem	48	Aktif
43	Express Cuci Setrika	Layanan cuci setrika dengan waktu selesai 6 jam	15000.00	Dalam jem	6	Aktif
44	Dry Clean	Pencucian khusus untuk pakaian berbahan sensitif	25000.00	Dalam jem	72	Aktif
45	Cuci Sepatu	Mencuci dan membersihkan sepatu	35000.00	Dalam jem	24	Aktif

➤ DELETE

The screenshot shows the phpMyAdmin interface for the db_bisnis_laundry database. In the 'Tabel: barang' (Table: barang) section, a message indicates '1 baris telah dihapus. (Pencarian dilakukan dalam 0,0030 detik.)' (1 row deleted. (Search took 0.0030 seconds)). Below this, a SQL query is shown: `DELETE FROM barang WHERE id_barang = 24;`. The results grid shows 25 rows of data, with the 24th row being the one that was deleted.

★ ROLLBACK untuk membatalkan perubahan.

The image contains three stacked screenshots of phpMyAdmin illustrating the use of the ROLLBACK command:

- Top Screenshot:** Shows the SQL tab with the command `ROLLBACK;` entered. The results show a success message: 'MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0002 detik.)' (MySQL returns empty results (or zero rows). (Search took 0.0002 seconds.))
- Middle Screenshot:** Shows the results of a previous SELECT query on the transaksi table, displaying 5 rows of transaction data.
- Bottom Screenshot:** Shows the SQL tab with the command `SELECT * FROM `transaksi`;` entered. The results show the same 5 rows of transaction data, indicating that the ROLLBACK command did not affect the data in the table.

- ★ COMMIT untuk menyimpan perubahan yang berhasil.

The image contains three separate screenshots of the phpMyAdmin interface, each showing a different database and table setup:

- Screenshot 1:** Shows the 'transaksi' table in the 'db_bisnis_laundry' database. A SQL query window contains the command `COMMIT;`. A tooltip message says: "Anda memiliki kueri yang tersimpan sebelumnya. Klik \"Dapatkan kueri terimpan otomatis\" untuk memuat kueri." The results pane shows the table structure with columns: id_transaksi, id_pelanggan, id_karyawan, tanggal_masuk, tanggal_selesai, status, total_harga, metode_pembayaran.
- Screenshot 2:** Shows the same 'transaksi' table in the 'db_bisnis_laundry' database. A SQL query window contains the command `COMMIT;`. A message box says: "MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0003 detik.)". Below it, another `COMMIT;` command is shown.
- Screenshot 3:** Shows the 'transaksi' table in the 'db_administrator' database. A SQL query window contains the command `SELECT * FROM `transaksi`;`. The results pane displays several rows of transaction data, including columns: id_transaksi, id_pelanggan, id_karyawan, tanggal_masuk, tanggal_selesai, status, total_harga, metode_pembayaran.

Gambar 6. 2 SQL dan Bukti Commit

- ★ Penggunaan SAVEPOINT untuk memungkinkan pembatalan sebagian perubahan.

The image contains two screenshots of the phpMyAdmin interface, both showing the 'transaksi' table in the 'db_administrator' database:

- Screenshot 1:** A SQL query window contains the command `1. SAVEPOINT sp1;`. The results pane shows the table structure with columns: id_transaksi, id_pelanggan, id_karyawan, tanggal_masuk, tanggal_selesai, status, total_harga, metode_pembayaran.
- Screenshot 2:** A SQL query window contains the command `SAVEPOINT sp1;`. A message box says: "MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0004 detik.)". Below it, another `SAVEPOINT sp1;` command is shown.

Gambar 6.3 SQL dan Bukti Savepoint

BAB VII

BACKUP DAN RESTORE

7.1 Penjelasan Pentingnya Backup dan Restore

Backup dan restore adalah aspek penting dalam pengelolaan data yang berfungsi untuk melindungi informasi penting dari kehilangan, kerusakan, atau ancaman lainnya. Berikut adalah penjelasan tentang pentingnya kedua hal ini:

1. Pentingnya Backup

Backup adalah proses membuat salinan data yang digunakan untuk memulihkan data jika terjadi kerusakan atau kehilangan. [10]

- Perlindungan terhadap kehilangan data

Kehilangan data dapat terjadi karena kerusakan perangkat keras, serangan malware, kesalahan pengguna, atau bencana alam. Backup memastikan data tetap tersedia meskipun hal-hal tersebut terjadi.

- Menjamin kelangsungan operasional

Dalam bisnis, kehilangan data dapat mengganggu operasional dan menyebabkan kerugian finansial. Backup membantu organisasi memulihkan data dengan cepat dan melanjutkan operasi tanpa gangguan besar.

- Keamanan data

Dengan backup reguler, pengguna dapat memastikan bahwa informasi sensitif atau penting tidak hilang secara permanen.

- Pematuhan regulasi

Beberapa industri memiliki aturan ketat tentang perlindungan dan pengelolaan data. Backup dapat membantu organisasi mematuhi standar tersebut.

2. Pentingnya Restore

Restore adalah proses mengembalikan data dari backup ke kondisi semula setelah data asli rusak, hilang, atau terhapus.[11]

- Pemulihan cepat dari kegagalan

Restore memungkinkan data yang hilang untuk dipulihkan dengan cepat, mengurangi dampak buruk terhadap individu atau organisasi.

- Meminimalkan waktu downtime

Dalam bisnis, waktu adalah uang. Restore yang efektif membantu mengembalikan sistem atau data dalam waktu singkat.

- Pengembalian kondisi semula

Restore memungkinkan sistem atau data dikembalikan ke kondisi sebelum terjadi kerusakan, termasuk struktur folder, konfigurasi, dan file individual.

- Uji coba keamanan

Restore juga penting untuk memverifikasi bahwa backup berfungsi dengan baik. Ini membantu memastikan bahwa backup tidak rusak atau usang.

- 

Tips untuk Backup dan Restore yang Efektif

 - Gunakan strategi backup 3-2-1: 3 salinan data, 2 media penyimpanan berbeda, 1 di lokasi yang berbeda.
 - Lakukan backup secara rutin.
 - Simpan data dalam format yang mudah dipulihkan.
 - Uji proses restore secara berkala untuk memastikan keandalan backup.
 - Gunakan solusi backup otomatis untuk mengurangi risiko kesalahan manusia.

7.2 Proses backup basis data:

- Logical Backup

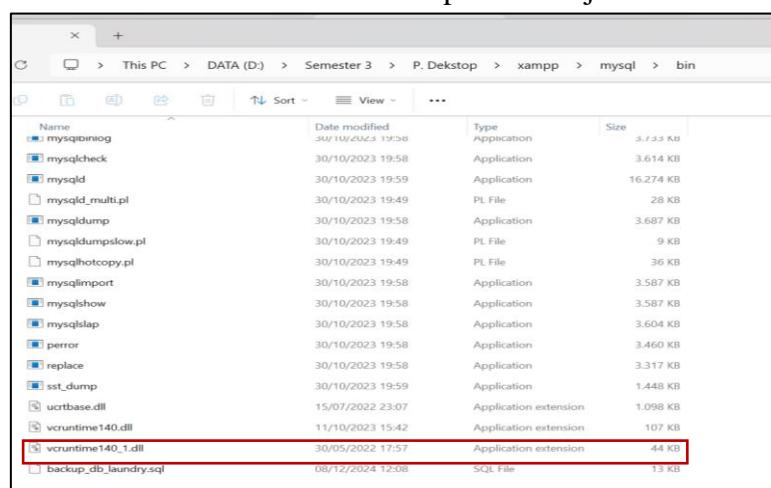
Untuk nge-backup seluruh database, maka dapat menggunakan query dibawah ini. Pastikan run cmd pada file xampp.

```
C:\Windows\System32\cmd.e + ▾ Microsoft Windows [Version 10.0.22631.4460]  
(c) Microsoft Corporation. All rights reserved.  
D:\Semester 3\P. Dekstop\xampp\mysql\bin>mysqldump -u root -p db_laundry > backup_db_laundry.sql  
Enter password:  
D:\Semester 3\P. Dekstop\xampp\mysql\bin>
```

Gambar 7. 1 Query Logical Backup

- ✓ Penjelasan :
 - mysqldump : perintah untuk mengekspor database.
 - -u root : pengguna menggunakan user root.
 - -p : password pengguna (jika tidak menggunakan password bisa dikosongin)
 - db_laundry adalah nama_database yang ingin dibackup.
 - backup_db_laundry.sql adalah nama setelah dibackup dan menggunakan jenis file sql.

Gambar dibawah ini adalah hasil backup database jika berhasil.

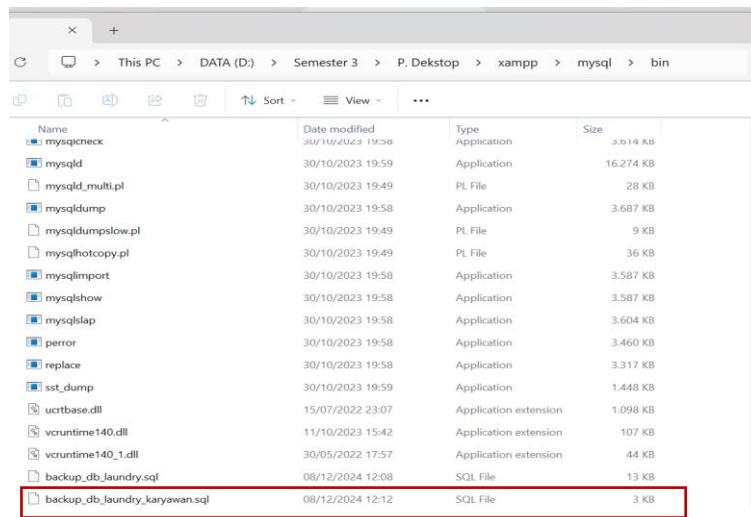


Gambar 7. 2 Bukti backup logical database

➤ Logical Backup Tabel Pilihan

Jika pengguna hanya ingin backup data pada tabel karyawan, maka dapat menggunakan query dibawah ini:

```
D:\Semester 3\P. Dekstop\xampp\mysql\bin>mysqldump -u root -p db_laundry karyawan > backup_db_laundry_karyawan.sql  
Enter password:  
D:\Semester 3\P. Dekstop\xampp\mysql\bin>
```

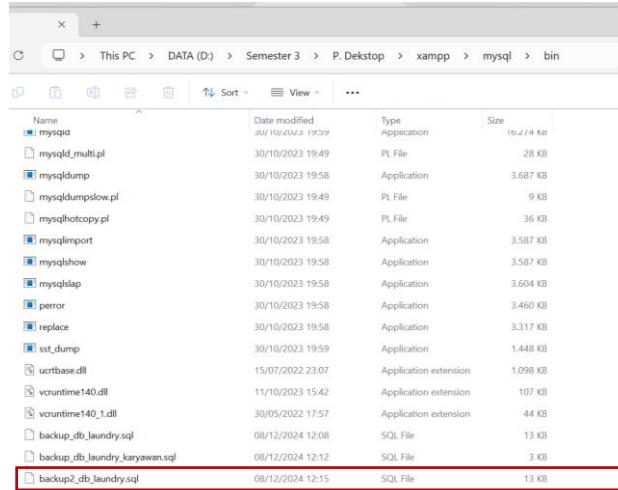


Gambar 7. 3 Query dan Bukti Logical Backup Tabel pilihan

➤ Logical Backup dengan Opsi Tambahan

Pengguna dapat menambahkan opsi seperti `--add-drop-table` untuk menambahkan perintah `DROP TABLE` pada file backup dengan tujuan menghapus tabel lama sebelum membuat tabel baru saat restore, berikut adalah :

```
D:\Semester 3\P. Dekstop\xampp\mysql\bin>mysqldump -u root -p --add-drop-table db_laundry > backup2_db_laundry.sql  
Enter password:  
D:\Semester 3\P. Dekstop\xampp\mysql\bin>
```



Gambar 7. 4 Query dan Bukti Logical Backup Tabel pilihan

- Physical Backup

```

C:\Windows\System32\cmd.exe + - 
Microsoft Windows [Version 10.0.26100.2310]
(c) Microsoft Corporation. All rights reserved.

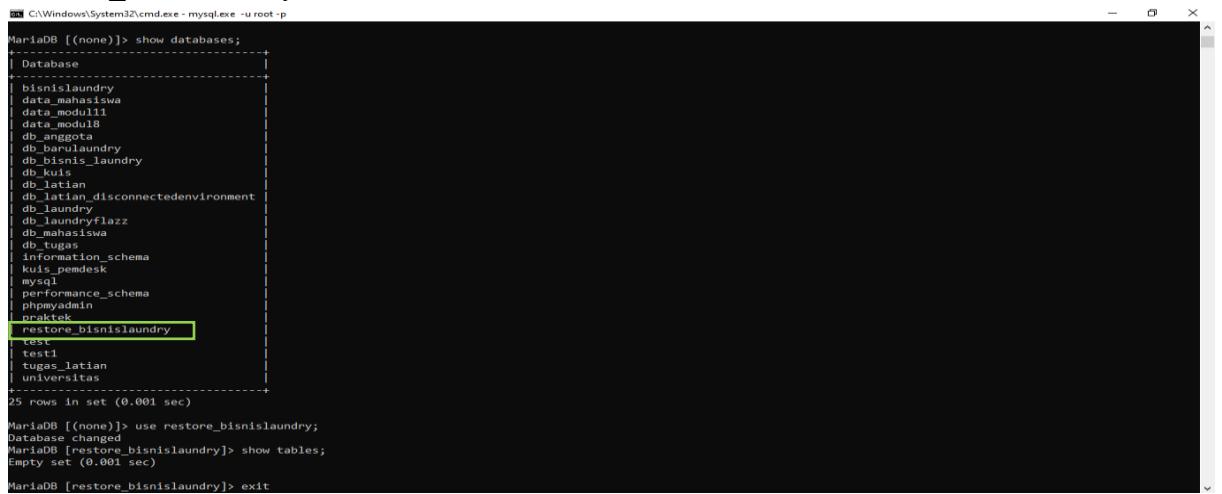
C:xampp\mysql\data>xcopy C:xampp\mysql\data D:\backup1 /E /I
C:xampp\mysql\data\mysql.pid
C:xampp\mysql\data\mysql.dmp
C:xampp\mysql\data\mysql_error.log
C:xampp\mysql\data\db_account\accounts.frm
C:xampp\mysql\data\db_account\accounts.ibd
C:xampp\mysql\data\db_account\db.opt
C:xampp\mysql\data\db_anggota\tbl_anggota.frm
C:xampp\mysql\data\db_anggota\tbl_anggota.ibd
C:xampp\mysql\data\db_barang\barang.frm
C:xampp\mysql\data\db_barang\barang.ibd
C:xampp\mysql\data\db_barang\tbl_barang.frm
C:xampp\mysql\data\db_bisnis\laundry\admin.frm
C:xampp\mysql\data\db_bisnis\laundry\admin.ibd
C:xampp\mysql\data\db_bisnis\laundry\barang.frm
C:xampp\mysql\data\db_bisnis\laundry\barang.ibd
C:xampp\mysql\data\db_bisnis\laundry\db.opt
C:xampp\mysql\data\db_bisnis\laundry\detail_transaksi.frm
C:xampp\mysql\data\db_bisnis\laundry\pelanggan.frm
C:xampp\mysql\data\db_bisnis\laundry\transaksi.frm
C:xampp\mysql\data\db_bisnis\laundry\transaksi.ibd
C:xampp\mysql\data\db_bisnis\laundry\karyawan.frm
C:xampp\mysql\data\db_bisnis\laundry\keluhan.frm
C:xampp\mysql\data\db_bisnis\laundry\layanan.frm
C:xampp\mysql\data\db_bisnis\laundry\pelanggan.frm
C:xampp\mysql\data\db_bisnis\laundry\pelangan.ibd
C:xampp\mysql\data\db_bisnis\laundry\transaksi.ibd
C:xampp\mysql\data\db_latihan\db.opt
C:xampp\mysql\data\db_latihan\tbl_mahasiswa.frm
C:xampp\mysql\data\db_latihan\tbl_mahasiswa.ibd
C:xampp\mysql\data\db_latihan\db.opt
C:xampp\mysql\data\db_latihan\tbl_barang.frm
C:xampp\mysql\data\db_latihan\tbl_barang.ibd
C:xampp\mysql\data\db_latihan\tbl_jenis_barang.frm
C:xampp\mysql\data\db_latihan\tbl_jenis_barang.ibd
C:xampp\mysql\data\db_latihan\tbl_satuhan.frm
C:xampp\mysql\data\db_latihan\tbl_satuhan.ibd
C:xampp\mysql\data\db_latihan\tbl_mahasiswa.frm
C:xampp\mysql\data\db_latihan\tbl_mahasiswa.ibd
C:xampp\mysql\data\db_latihan\tbl_pelanggan.frm
C:xampp\mysql\data\db_latihan\tbl_pelanggan.ibd
C:xampp\mysql\data\db_latihan\tbl_perjualan.frm
C:xampp\mysql\data\db_latihan\tbl_perjualan_ibd
C:xampp\mysql\data\db_latihan\tbl_perjualan_rinci.frm
C:xampp\mysql\data\db_latihan\tbl_perjualan_rinci.ibd
C:xampp\mysql\data\db_latihan\tbl_stock.frm
C:xampp\mysql\data\db_latihan\tbl_stock.ibd
C:xampp\mysql\data\db_latihan2\db.opt
C:xampp\mysql\data\db_latihan2\tbl_barang.frm
C:xampp\mysql\data\db_latihan2\tbl_barang.ibd
C:xampp\mysql\data\db_latihan2\tbl_pelanggan.frm
C:xampp\mysql\data\db_latihan2\tbl_pelanggan.ibd
C:xampp\mysql\data\db_latihan2\tbl_perjualan.frm
C:xampp\mysql\data\db_latihan2\tbl_perjualan.ibd
C:xampp\mysql\data\db_latihan2\tbl_satuhan.frm
C:xampp\mysql\data\db_latihan2\tbl_satuhan.ibd
C:xampp\mysql\data\db_latihan2\tbl_mahasiswa.frm
C:xampp\mysql\data\db_latihan2\tbl_mahasiswa.ibd
C:xampp\mysql\data\db_latihan2\tbl_pelanggan.frm
C:xampp\mysql\data\db_latihan2\tbl_pelanggan.ibd
C:xampp\mysql\data\db_latihan2\tbl_perjualan.frm
C:xampp\mysql\data\db_latihan2\tbl_perjualan.ibd
C:xampp\mysql\data\db_latihan2\tbl_perjualan_rinci.frm
C:xampp\mysql\data\db_latihan2\tbl_perjualan_rinci.ibd
C:xampp\mysql\data\db_laundry\tbl_penjualan_rinci.frm
C:xampp\mysql\data\db_laundry\tbl_penjualan_rinci.ibd
C:xampp\mysql\data\db_laundry\tbl_stock.frm
C:xampp\mysql\data\db_laundry\tbl_stock.ibd
C:xampp\mysql\data\db_laundry2\db.opt
C:xampp\mysql\data\db_laundry2\tbl_barang.frm
C:xampp\mysql\data\db_laundry2\tbl_barang.ibd
C:xampp\mysql\data\db_laundry2\tbl_pelanggan.frm
C:xampp\mysql\data\db_laundry2\tbl_pelanggan.ibd
C:xampp\mysql\data\db_laundry2\tbl_perjualan.frm
C:xampp\mysql\data\db_laundry2\tbl_perjualan.ibd
C:xampp\mysql\data\db_laundry2\tbl_karyawan.frm
C:xampp\mysql\data\db_laundry2\tbl_karyawan.ibd
C:xampp\mysql\data\db_laundry2\tbl_keluhan.frm
C:xampp\mysql\data\db_laundry2\tbl_keluhan.ibd
C:xampp\mysql\data\db_laundry2\tbl_layanan.frm
C:xampp\mysql\data\db_laundry2\tbl_layanan.ibd
C:xampp\mysql\data\db_laundry2\tbl_pelanggan.frm
C:xampp\mysql\data\db_laundry2\tbl_pelanggan.ibd
C:xampp\mysql\data\db_laundry2\tbl_perjualan.frm
C:xampp\mysql\data\db_laundry2\tbl_perjualan.ibd
C:xampp\mysql\data\db_laundry2\tbl_transaksi.frm
C:xampp\mysql\data\db_laundry2\tbl_transaksi.ibd
C:xampp\mysql\data\db_laundry2\tbl_laundry.frm
C:xampp\mysql\data\db_laundry2\tbl_laundry.ibd
C:xampp\mysql\data\db_laundry2\tbl_laundry_rinci.frm
C:xampp\mysql\data\db_laundry2\tbl_laundry_rinci.ibd
C:xampp\mysql\data\db_laundry2\tbl_laundry2.frm
C:xampp\mysql\data\db_laundry2\tbl_laundry2.ibd
C:xampp\mysql\data\db_laundry2\tbl_laundry2_rinci.frm
C:xampp\mysql\data\db_laundry2\tbl_laundry2_rinci.ibd
C:xampp\mysql\data\laundry_db\detail_transaksi.TRG
C:xampp\mysql\data\laundry_db\hitung_subtotal.TRN
C:xampp\mysql\data\laundry_db\layanan.frm
C:xampp\mysql\data\laundry_db\layanan.ibd
C:xampp\mysql\data\laundry_db\pelanggan.frm
C:xampp\mysql\data\laundry_db\pelanggan.ibd
C:xampp\mysql\data\laundry_db\pendapatan.frm
C:xampp\mysql\data\laundry_db\pendapatan.ibd
C:xampp\mysql\data\laundry_db\transaksi.frm
C:xampp\mysql\data\laundry_db\transaksi.ibd
C:xampp\mysql\data\laundry_db\transaksi.TRG
C:xampp\mysql\data\laundry_db\update_pendapatan.TRN
C:xampp\mysql\data\mysql\MAO-246925220331.BAK
C:xampp\mysql\data\tbl_mahasiswa\db.opt
C:xampp\mysql\data\tbl_mahasiswa\tbl_mahasiswa.frm
C:xampp\mysql\data\tbl_mahasiswa\tbl_mahasiswa.ibd
C:xampp\mysql\data\tugas@0020abd\cabang.frm
C:xampp\mysql\data\tugas@0020abd\cabang.ibd
C:xampp\mysql\data\tugas@0020abd\db.opt
C:xampp\mysql\data\tugas@0020abd\detail_transaksi.frm
C:xampp\mysql\data\tugas@0020abd\detail_transaksi.ibd
C:xampp\mysql\data\tugas@0020abd\layanan.frm
C:xampp\mysql\data\tugas@0020abd\layanan.ibd
C:xampp\mysql\data\tugas@0020abd\pelanggan.frm
C:xampp\mysql\data\tugas@0020abd\pelanggan.ibd
C:xampp\mysql\data\tugas@0020abd\pendapatan.frm
C:xampp\mysql\data\tugas@0020abd\pendapatan.ibd
C:xampp\mysql\data\tugas@0020abd\transaksi.frm
C:xampp\mysql\data\tugas@0020abd\transaksi.ibd
133 File(s) copied
C:xampp\mysql\data>

```

Gambar 7. 5 Query Physical Backup

7.3 Proses restore basis data

- 1) Langkah awal proses restore menyiapkan database yang kosong yaitu "restore_bisnislaundery"

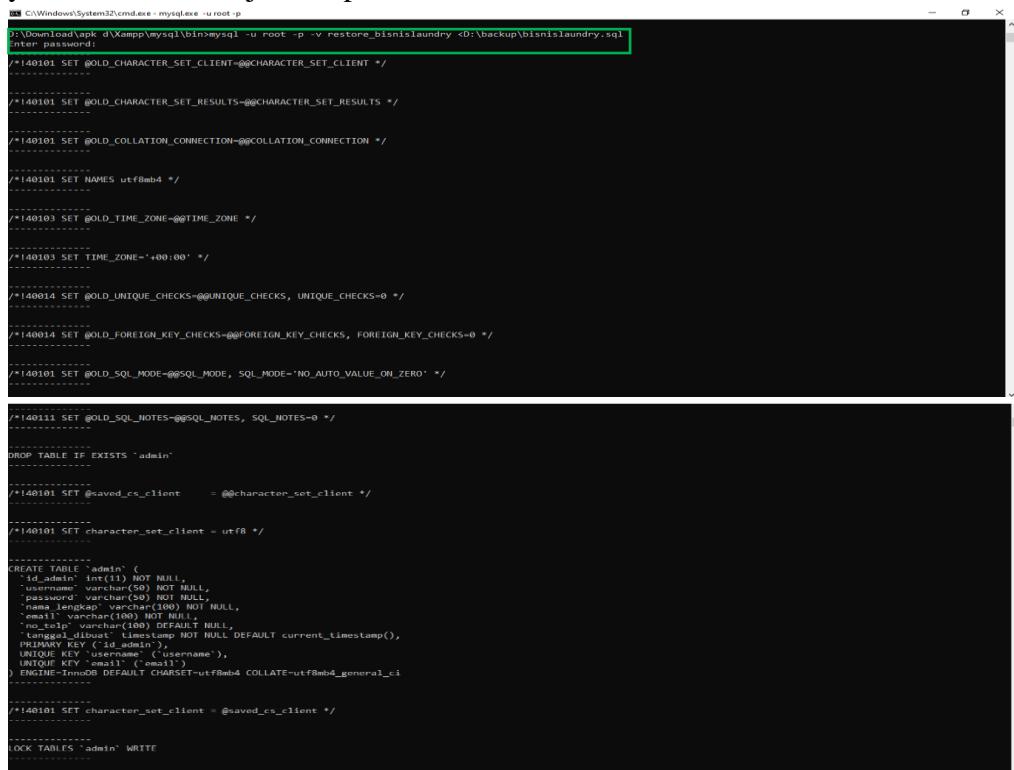


```
C:\Windows\System32\cmd.exe - mysql.exe -u root -p
MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| bisnislaundery |
| data_mahasiswa |
| data_modul11 |
| data_modul18 |
| db_anggota |
| db_bisnislaundry |
| db_bisnis_laundry |
| db_kuis |
| db_latian |
| db_latian_disconnectedenvironment |
| db_laundry |
| db_laundryflazz |
| db_mahasiswa |
| db_tugas |
| information_schema |
| kuis_pemdesk |
| mysql |
| performance_schema |
| phpmayadmin |
| projectek |
| restore_bisnislaundery |
| test |
| ttest1 |
| tugas_latian |
| universitas |
+-----+
25 rows in set (0.001 sec)

MariaDB [(none)]> use restore_bisnislaundery;
Database changed
MariaDB [restore_bisnislaundery]> show tables;
Empty set (0.001 sec)

MariaDB [restore_bisnislaundery]> exit
```

- 2) Selanjutnya, mengetikkan query yang terletak dikotak merah tersebut. Kegunaan –v yaitu untuk menunjukkan proses restore.



```
C:\Windows\System32\cmd.exe - mysql.exe -u root -p
D:\Download\sql\dxXamp\mysql\bin>mysql -u root -p -v < D:\backup\bisnislaundery.sql
Enter password:
/*I40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */

/*I40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */

/*I40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */

/*I40101 SET NAMES utf8mb4 */

/*I40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */

/*I40103 SET TIME_ZONE='+00:00' */

/*I40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */

/*I40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */

/*I40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */

/*I40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */

-- TABLE `admin` EXISTS `admin`

/*I40101 SET @saved_cs_client      = @@character_set_client */

/*I40101 SET character_set_client = utf8 */

CREATE TABLE `admin` (
  `id_admin` int(11) NOT NULL,
  `username` varchar(50) NOT NULL,
  `password` varchar(50) NOT NULL,
  `email_gmail` varchar(50) NOT NULL,
  `email` varchar(100) NOT NULL,
  `no_telp` varchar(100) DEFAULT NULL,
  `last_update` timestamp NOT NULL DEFAULT current_timestamp(),
  PRIMARY KEY (`id_admin`),
  UNIQUE KEY `username` (`username`),
  UNIQUE KEY `email` (`email`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci

/*I40101 SET character_set_client = @saved_cs_client */

-- TABLE `admin` WRITE
```

```

-----+*140000 ALTER TABLE `admin` DISABLE KEYS *+
-----+
-----+INSERT INTO `admin` VALUES (1,'admin','hashed_password_1','John Doe','john@laundry.com','081234567898','2024-10-13 15:30:02'),(12,'admin2','hashed_password_2','Jane S
mith','jane@laundry.com','081234567891','2024-10-13 15:30:02'),(13,'admin3','hashed_password_3','Mike Johnson','mike@laundry.com','081234567892','2024-10-13 15:30:02'),
(14,'admin4','hashed_password_4','Emily Brown','emily@laundry.com','081234567893','2024-10-13 15:30:02'),(15,'admin5','hashed_password_5','David Lee','david@laundry.com
-----+
-----+*140000 ALTER TABLE `admin` ENABLE KEYS *+
-----+
-----+UNLOCK TABLES
-----+
-----+DROP TABLE IF EXISTS `barang`
-----+
-----+/*140101 SET @saved_cs_client      = @@character_set_client */
-----+
-----+/*140101 SET character_set_client = utf8 */
-----+
-----+CREATE TABLE `barang` (
`id_barang` int(11) NOT NULL,
`nama_barang` varchar(100) NOT NULL,
`kategori` enum('Deterjen','Peuangi','Peralatan','Lainnya') NOT NULL,
`stok` int(11) NOT NULL,
`harga_beli` decimal(10,2) NOT NULL,
`satuan` varchar(20) NOT NULL,
`tanggal_update` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE current_timestamp(),
PRIMARY KEY (`id_barang`),
KEY `id_barang`(`id_barang`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
-----+
-----+*140000 ALTER TABLE `barang` DISABLE KEYS *
-----+
-----+INSERT INTO `barang` VALUES (22,'Deterjen Cair','Deterjen',100,25000.00,'Botol','2024-10-13 15:31:17'),(23,'Peuangi Lavender','Peuangi',80,15000.00,'Botol','2024-10-13 15:31:17'),(25,'Hanger Plastik','Peralatan',500,1000.00,'Pcs','2024-10-13 15:31:17'),(26,'Kantong Plastik','Lainnya',1000,500.00,'Pcs','2024-10-13 15:31:17')
-----+
-----+*140000 ALTER TABLE `barang` ENABLE KEYS *
-----+
-----+UNLOCK TABLES
-----+
-----+DROP TABLE IF EXISTS `detail_transaksi`
-----+
-----+/*140101 SET @saved_cs_client      = @@character_set_client */
-----+
-----+/*140101 SET character_set_client = utf8 */
-----+
-----+CREATE TABLE `detail_transaksi` (
`id_detail_transaksi` int(11) NOT NULL,
`id_transaksi` int(11) DEFAULT NULL,
`id_layanan` int(11) DEFAULT NULL,
`jumlah` int(11) NOT NULL,
`subtotal` decimal(10,2) NOT NULL,
PRIMARY KEY (`id_detail_transaksi`),
KEY `id_transaksi`(`id_transaksi`),
KEY `id_layanan`(`id_layanan`),
CONSTRAINT `detail_transaksi_ibfk_1` FOREIGN KEY (`id_transaksi`) REFERENCES `transaksi` (`id_transaksi`),
CONSTRAINT `detail_transaksi_ibfk_2` FOREIGN KEY (`id_layanan`) REFERENCES `layanan` (`id_layanan`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
-----+
-----+*140000 ALTER TABLE `detail_transaksi` DISABLE KEYS *
-----+
-----+INSERT INTO `detail_transaksi` VALUES (71,61,41,3,21000.00),(72,61,42,2,20000.00),(73,62,43,5,75000.00),(74,63,44,4,100000.00),(75,64,42,6,60000.00)
-----+
-----+*140000 ALTER TABLE `detail_transaksi` ENABLE KEYS *
-----+
-----+UNLOCK TABLES
-----+
-----+DROP TABLE IF EXISTS `karyawan`
-----+
-----+/*140101 SET @saved_cs_client      = @@character_set_client */
-----+
-----+/*140101 SET character_set_client = utf8 */
-----+
-----+CREATE TABLE `karyawan` (
`id_karyawan` int(11) NOT NULL,
`nama_karyawan` varchar(100) NOT NULL,
`jabatan` varchar(50) NOT NULL,
`alamat` text DEFAULT NULL,
`no_telp` varchar(15) DEFAULT NULL,
`email` varchar(100) DEFAULT NULL,
`tanggal Bergabung` date NOT NULL,
`gaji` decimal(10,2) NOT NULL,
`status_karyawan` enum('Aktif','Tidak Aktif') NOT NULL,
UNIQUE KEY `email` (`email`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
-----+

```

```

-----+
LOCK TABLES `karyawan` WRITE
-----+
/*1400000 ALTER TABLE `karyawan` DISABLE KEYS */
-----+
INSERT INTO `karyawan` VALUES (31,'Rudi Sentoso','operator mail','Jl. Merdeka No. 123','085678901234','rudi@mail.com','2023-01-15',350000.00),(32,'Siti Bahayu','Kasi PT. ABC','Jl. Jendral Soedirman No. 45 ','085678901235','siti@mail.com','2023-03-10',300000.00),(33,'Eduard Lestari','Customer Service','Jl. Sulirman No. 67 ','085678901236','eduard@mail.com','2023-07-05',300000.00),(34,'Gajah Mada No. 89 ','085678901237','gajah@mail.com','2023-04-05',300000.00),(35,'Agni Widodo','Kurir','Jl. Diponegoro No. 127','085678901238','agni@mail.com','2023-05-20',300000.00)
-----+
/*1400000 ALTER TABLE `karyawan` ENABLE KEYS */
-----+
UNLOCK TABLES
-----+
DROP TABLE IF EXISTS `layanan`
-----+
/*1400101 SET @saved_cs_client      = @@character_set_client */
-----+
/*1400101 SET character_set_client = utf8 */
-----+
CREATE TABLE `layanan` (
  `id_layanan` int(11) NOT NULL,
  `nama_layanan` varchar(100) NOT NULL,
  `deskripsi` text DEFAULT NULL,
  `harga` decimal(10,2) NOT NULL,
  `estimer_waktu` int(11) NOT NULL COMMENT 'Dalam jam',
  `status` enum('Aktif','Tidak Aktif','','') NOT NULL DEFAULT 'Aktif',
  PRIMARY KEY (`id_layanan`),
  ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
-----+
DROP TABLE IF EXISTS `layanan`
-----+
/*1400101 SET @saved_cs_client      = @@character_set_client */
-----+
/*1400101 SET character_set_client = utf8 */
-----+
CREATE TABLE `layanan` (
  `id_layanan` int(11) NOT NULL,
  `nama_layanan` varchar(100) NOT NULL,
  `deskripsi` text DEFAULT NULL,
  `harga` decimal(10,2) NOT NULL,
  `estimer_waktu` int(11) NOT NULL COMMENT 'Dalam jam',
  `status` enum('Aktif','Tidak Aktif','','') NOT NULL DEFAULT 'Aktif',
  PRIMARY KEY (`id_layanan`),
  ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
-----+
/*1400101 SET character_set_client = @saved_cs_client */
-----+
LOCK TABLES `layanan` WRITE
-----+
/*1400000 ALTER TABLE `layanan` DISABLE KEYS */
-----+
INSERT INTO `layanan` VALUES (41,'Cuci Kering','Mencuci dan mengeringkan pakaian tanpa setrika',90000.00,24,'Aktif'),(42,'Cuci Setrika','Mencuci, mengeringkan, dan menyeprti pakaian',10000.00,48,'Aktif'),(43,'Express Cuci Setrika','Layanan cuci setrika dengan waktu selesai 6 jam',15000.00,6,'Aktif'),(44,'Dry Clean','Pencucian khusus untuk pakaian berbahan sensitif',25000.00,12,'Aktif'),(45,'Cuci Sepatu','Mencuci dan membersihkan sepatu',35000.00,24,'Aktif')
-----+
/*1400000 ALTER TABLE `layanan` ENABLE KEYS */
-----+
DROP TABLE IF EXISTS `pelanggan`
-----+
/*1400101 SET @saved_cs_client      = @@character_set_client */
-----+
/*1400101 SET character_set_client = utf8 */
-----+
CREATE TABLE `pelanggan` (
  `id_pelanggan` int(11) NOT NULL,
  `nama_pelanggan` varchar(100) NOT NULL,
  `alamat` varchar(100) NOT NULL,
  `no_telp` varchar(15) NOT NULL,
  `email` varchar(100) DEFAULT curdate(),
  `pin` int(11) DEFAULT 0,
  PRIMARY KEY (`id_pelanggan`),
  UNIQUE KEY `uk_email` (`email`),
  ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
-----+
/*1400101 SET character_set_client = @saved_cs_client */
-----+
LOCK TABLES `pelanggan` WRITE
-----+
/*1400000 ALTER TABLE `pelanggan` DISABLE KEYS */
-----+
INSERT INTO `pelanggan` VALUES (51,'Rudi Hermawan','Jl. Anggrek No. 1','081234567890','rudi@mail.com','2023-06-01',100),(52,'Nina Surya','Jl. Melati No. 23','081234567891','nina@mail.com','2023-07-01',100),(53,'Andi Pratama','Jl. Pemuda No. 45 ','081234567892','andi@mail.com','2023-08-01',100),(54,'Andi Kusuma','Jl. Mawar No. 67 ','081234567893','andi@mail.com','2023-07-15',100),(55,'Lina Putri','Jl. Tugu Negara No. 89 ','081234567894','linamail@mail.com','2023-08-01',100)
-----+
/*1400000 ALTER TABLE `pelanggan` ENABLE KEYS */
-----+
DROP TABLE IF EXISTS `transaksi`
-----+
/*1400101 SET @saved_cs_client      = @@character_set_client */
-----+
/*1400101 SET character_set_client = utf8 */
-----+
CREATE TABLE `transaksi` (
  `id_transaksi` int(11) NOT NULL,
  `id_pelanggan` int(11) DEFAULT NULL,
  `id_karyawan` int(11) DEFAULT NULL,
  `tarif` decimal(10,2) NOT NULL,
  `tanggak_musik` datetime NOT NULL,
  `tanggak_selesai` datetime NOT NULL,
  `total_proses` decimal(10,2) DEFAULT '0.00',
  `total_kerja` decimal(10,2) DEFAULT '0.00',
  `total_kartu_kredit` decimal(10,2) DEFAULT '0.00',
  `total_transfer` decimal(10,2) DEFAULT '0.00',
  PRIMARY KEY (`id_transaksi`),
  KEY `id_karyawan` (`id_karyawan`),
  KEY `id_kar_transaksi` (`id_karyawan`),
  CONSTRAINT `transaksi_ibfk_1` FOREIGN KEY (`id_karyawan`) REFERENCES `karyawan` (`id_karyawan`),
  CONSTRAINT `transaksi_ibfk_2` FOREIGN KEY (`id_karyawan`) REFERENCES `karyawan` (`id_karyawan`),
  ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
-----+
/*1400101 SET character_set_client = @saved_cs_client */
-----+
LOCK TABLES `transaksi` WRITE
-----+
/*1400000 ALTER TABLE `transaksi` DISABLE KEYS */
-----+
INSERT INTO `transaksi` VALUES (61,51,34,'2023-09-01 10:00:00','2023-09-02 15:00:00','Selesai','50000.00','Tunai'),(62,52,34,'2023-09-02 11:00:00','2023-09-03 16:00:00','Selesai','75000.00','Transfer'),(63,53,34,'2023-09-03 09:00:00','2023-09-04 14:00:00','Kartu Kredit'),(64,54,34,'2023-09-04 13:00:00','null','Proses','60000.00','Tunai'),(65,55,34,'2023-09-05 14:00:00','null','Proses','60000.00','Transfer')
-----+
/*1400000 ALTER TABLE `transaksi` ENABLE KEYS */
-----+
UNLOCK TABLES
-----+

```

Gambar 7. 6 Query Restore Database

- 3) Jika berhasil, tahap akhir pada restore yaitu melihat database yang sebelumnya berisi kosong menjadi ada table bisniskaundry

```
C:\Windows\System32\cmd.exe
Your MariaDB connection id is 147
Server version: 10.4.32-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| bisniskaundry |
| data_mahasiswa |
| data_modull1 |
| data_modul8 |
| db_anggota |
| db_barulaundry |
| db_bisnis_laundry |
| db_kuis |
| db_latian |
| db_latian_disconnectedenvironment |
| db_laundry |
| db_laundryflazz |
| db_mahasiswa |
| db_tugas |
| information_schema |
| kuis_pemdesk |
| mysql |
| performance_schema |
| phpmyadmin |
| praktik |
| restore_bisniskaundry |
| test |
| test1 |
| tugas_latian |
| universitas |
+-----+
25 rows in set (0.001 sec)

MariaDB [(none)]> use restore_bisniskaundry;
Database changed

MariaDB [(none)]> use restore_bisniskaundry;
Database changed
MariaDB [restore_bisniskaundry]> show tables;
+-----+
| Tables_in_restore_bisniskaundry |
+-----+
| admin |
| barang |
| detail_transaksi |
| karyawan |
| layanan |
| pelanggan |
| transaksi |
+-----+
7 rows in set (0.001 sec)

MariaDB [restore_bisniskaundry]> select * from detail_transaksi;
+-----+-----+-----+-----+-----+
| id_detail_transaksi | id_transaksi | id_layanan | jumlah | subtotal |
+-----+-----+-----+-----+-----+
| 71 | 61 | 41 | 3 | 21000.00 |
| 72 | 61 | 42 | 2 | 20000.00 |
| 73 | 62 | 43 | 5 | 75000.00 |
| 74 | 63 | 44 | 4 | 100000.00 |
| 75 | 64 | 42 | 6 | 60000.00 |
+-----+-----+-----+-----+-----+
5 rows in set (0.000 sec)

MariaDB [restore_bisniskaundry]> exit
Bye
```

Gambar 7.7 Bukti Berhasil Restore Database

BAB VIII

KESIMPULAN DAN SARAN

8.1 Ringkasan Hasil Proyek

Proyek ini berhasil mengimplementasikan sistem basis data menggunakan MariaDB untuk mendukung operasional bisnis Laundry Flazz. Sistem basis data yang dirancang mampu memenuhi kebutuhan fungsional seperti pencatatan data pelanggan, transaksi, inventaris, keluhan, dan layanan, serta menghasilkan laporan yang mendukung pengambilan keputusan.

Melalui proses perancangan, implementasi, hingga pengujian, struktur basis data yang dibuat telah memastikan data tersimpan secara efisien, terstruktur, dan aman. Fitur seperti transaksi, rollback, backup, dan restore telah diterapkan untuk memastikan keandalan data. Selain itu, keamanan data ditingkatkan melalui pemberian hak akses terkontrol kepada pengguna yang berwenang.

Dengan sistem ini, Laundry Flazz dapat meningkatkan efisiensi operasional, mengurangi kesalahan pencatatan manual, dan menyediakan laporan yang lebih akurat untuk mendukung strategi bisnis.

8.2 Evaluasi implementasi

a. Keberhasilan:

- Sistem mampu menangani kebutuhan dasar pencatatan dan pengelolaan data pelanggan, transaksi, serta inventaris.
- Fitur keamanan dan pemulihan data berjalan dengan baik, memastikan perlindungan terhadap kehilangan data.
- Penggunaan view dan laporan yang dihasilkan mendukung pengambilan keputusan berbasis data.

b. Tantangan:

- Masih ada kebutuhan akan integrasi dengan aplikasi front-end untuk mempermudah interaksi pengguna non-teknis.
- Kurangnya pelatihan bagi staf Laundry Flazz mengenai penggunaan sistem ini sehingga perlu waktu adaptasi.

c. Potensi Peningkatan:

- Optimalisasi performa query pada skenario data yang lebih besar.
- Peningkatan skalabilitas untuk mengakomodasi pertumbuhan bisnis di masa depan.

8.3 Rekomendasi untuk pengembangan lebih lanjut

- Integrasi Front-End: Kembangkan aplikasi berbasis web atau mobile yang terhubung dengan basis data ini untuk memberikan kemudahan akses kepada pengguna akhir.
- Automasi Proses: Implementasikan fitur automasi, seperti pengingat stok menipis atau pengiriman notifikasi kepada pelanggan terkait status transaksi.
- Analitik Lanjutan: Tambahkan fitur analitik berbasis data seperti prediksi permintaan layanan berdasarkan riwayat transaksi atau analisis pelanggan loyal.
- Peningkatan Keamanan: Gunakan enkripsi data dan audit log untuk meningkatkan perlindungan terhadap data sensitif.
- Pelatihan Pengguna: Adakan pelatihan rutin bagi staf Laundry Flazz untuk memastikan penggunaan sistem secara optimal.

8.4 Saran dan Pesan

- ✓ Pesan untuk modul ABD mohon di perbaiki lagi, dikarenakan ada banyak kesalahan untuk dipraktekkan.
- ✓ Saran untuk mata kuliah ABD agar diperbanyak prakteknya.

DAFTAR PUSTAKA

- [1] Prasetya & Satria, “Analisis dan perancangan basis data sistem informasi laundry kiloan untuk peningkatan kualitas layanan dan administrasi keuangan,” no. September, pp. 1963–1970, 2019.
- [2] R. Sihotang, H. Saputro, and S. Novari, “Sistem Informasi Penggajian LKP English Academy Menggunakan Embarcadero XE2 Berbasis Client Server,” *JTIM: Jurnal Teknik Informatika Mahakarya*, vol. 04, no. 1, pp. 28–36, 2021.
- [3] Connolly, T., & Begg, C. (2010). Database Systems: A Practical Approach to Design, Implementation, and Management (5th ed.). United States: Pearson.
- [4] Z. Efendy, “Normalisasi dalam Desain Database,” *Jurnal CoreIT*, vol. 4, no. 1, pp. 34–43, 2018.
- [5] E. Sebastian and M. R. Ridho, “Perancangan Sistem Informasi Stok Kontainer Dengan Metode Scrum Pada Pt Putra Batam Jasa Mandiri Utama,” *Jurnal Comasie*, vol. 6, no. 2, pp. 40–51, 2021, [Online]. Available: [http://ejournal.upbatam.ac.id/index.php/comasiejurnal%0AJurnal Comasie ISSN \(Online\) 2715-6265%0APERANCANGAN](http://ejournal.upbatam.ac.id/index.php/comasiejurnal%0AJurnal Comasie ISSN (Online) 2715-6265%0APERANCANGAN)
- [6] D. Meidi and K. K. Purnamasari, “Penerjemah Teks Bahasa Indonesia Ke Dalam Dml (Data Manipulation Language) Dengan Sub-Query,” *Teknik Informatika Universitas Komputer Indonesia*, vol. 3, pp. 36–48, 2020.
- [7] H. Jurnal et al., “Jurnal Informatika Dan Teknologi Komputer Analisa Perbandingan Kinerja Response Time Query Mysql Dan Mongodb,” *Juli*, vol. 2, no. 2, pp. 158–166, 2022.
- [8] F. Wongso, “Speizer et al 2001.pdf,” *Jurnal Ilmiah EkoWongso, F. (2015). Speizer et al 2001.pdf. Jurnal Ilmiah Ekonomi Dan Bisnis, 12(1), 46–60.nomi dan Bisnis*, vol. 12, no. 1, pp. 46–60, 2015.
- [9] J. Triyono, “Penerapan Hak Akses pada Perancangan Database Akademik untuk Meningkatkan Keamanan Data,” *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 3, no. 1, pp. 50–59, 2023, doi: 10.57152/malcom.v3i1.747.
- [10] K. S. Mahedy, “Implementasi Sistem Backup Data Pada Sistem Informasi Perpustakaan Universitas Pendidikan Ganesha,” *Jurnal Pendidikan Teknologi dan Kejuruan*, vol. 18, no. 2, pp. 216–225, 2021.
- [11] I. Arnomo, “Simulasi Backup Dan Restore Database Repository,” *Jurnal Sistem Informasi, Teknologi Informasi dan Komputer*, vol. 9, no. 2, pp. 92–99, 2019.