

LAPORAN
TUGAS PEMROGRAMAN BERORIENTASI OBJEK



IBRA ERLANGGA PUTRA
5230411321

PROGRAM STUDI INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS TEKNOLOGI YOGYAKARTA
TAHUN 2024

```
def connect_to_db():  
    try:  
        return mysql.connector.connect(host="localhost",  
                                         user="root", password="",  
                                         database="penjualanrafli"  
                                         )  
    except mysql.connector.Error as err:  
        print(f"Error:  
        {err}")return None
```

Membuat database dan tabel

```
def setup_database():  
    try:  
        db = mysql.connector.connect(  
            host="localhost",  
            user="root",  
            password=""  
        )  
        cursor = db.cursor()  
        cursor.execute("CREATE DATABASE IF NOT EXISTS  
        penjualanrafli")cursor.execute("USE penjualanrafli")
```

Menghapus tabel jika sudah ada (opsional, untuk memastikan struktur tabel benar)

```
cursor.execute("DROP TABLE IF EXISTS Struk")  
cursor.execute("DROP TABLE IF EXISTS Produk")
```

```
cursor.execute("DROP TABLE IF EXISTS  
Transaksi")cursor.execute("DROP TABLE IF  
EXISTS Pegawai")
```

Tabel Pegawai

```
cursor.execute("""  
    CREATE TABLE IF NOT EXISTS  
        Pegawai (NIK INT PRIMARY  
        KEY,  
        Nama VARCHAR(100) NOT  
        NULL,Alamat TEXT NOT  
        NULL  
    )  
""")
```

Tabel Transaksi

```
cursor.execute("""  
    CREATE TABLE IF NOT EXISTS Transaksi (  
        NoTransaksi INT PRIMARY  
        KEY,NIK INT NOT NULL,  
        DetailTransaksi TEXT,  
        FOREIGN KEY (NIK) REFERENCES Pegawai(NIK) ON DELETE CASCADE  
    )  
""")
```

Tabel Produk

```
cursor.execute("""  
    CREATE TABLE IF NOT EXISTS Produk (  
        KodeProduk INT PRIMARY KEY,  
        NamaProduk VARCHAR(100) NOT NULL,  
        JenisProduk ENUM('Snack', 'Makanan', 'Minuman') NOT  
        NULL,HargaProduk DECIMAL(10, 2) NOT NULL
```

```
)  
""")
```

```
# Tabel Struk
```

```
cursor.execute("""
```

```
CREATE TABLE IF NOT EXISTS Struk (
```

```
    IdStruk INT AUTO_INCREMENT PRIMARY KEY,
```

```
    NoTransaksi INT NOT
```

```
    NULL,KodeProduk INT
```

```
    NOT NULL,
```

```
    JumlahProduk INT NOT NULL,
```

```
    TotalHarga DECIMAL(10, 2) NOT
```

```
    NULL,
```

```
    FOREIGN KEY (NoTransaksi) REFERENCES Transaksi(NoTransaksi) ON DELETE
```

```
    CASCADE,FOREIGN KEY (KodeProduk) REFERENCES Produk(KodeProduk) ON
```

```
    DELETE CASCADE
```

```
)
```

```
""")
```

```
print("Database dan tabel berhasil dibuat.")
```

```
db.close()
```

```
except mysql.connector.Error as err:
```

```
    print(f"Error: {err}")
```

```
# Fungsi untuk menambahkan data awal
```

```
def tambah_data_awal():
```

```
    try:
```

```
        # Data awal Pegawai
```

```
        data_pegawai = [
```

```
            (101, "Rafli", "Jl. Merdeka No. 1"),
```

```
            (102, "Andi", "Jl. Sudirman No. 2"),
```

```
            (103, "Siti", "Jl. Thamrin No. 3")
```

```

]

cursor.execute("SELECT COUNT(*) FROM
Pegawai")if cursor.fetchone()[0] == 0:

    cursor.executemany("INSERT INTO Pegawai (NIK, Nama, Alamat) VALUES (%s, %s, %s)",
data_pegawai)

# Data awal Produk
data_produk = [

    (1, "Sabun Mandi", "Snack", 10000.00),

    (2, "Air Mineral", "Minuman", 5000.00),

    (3, "Snack Ringan", "Makanan", 15000.00)

]

cursor.execute("SELECT COUNT(*) FROM
Produk")if cursor.fetchone()[0] == 0:

    cursor.executemany("INSERT INTO Produk (KodeProduk, NamaProduk, JenisProduk,
HargaProduk)VALUES (%s, %s, %s, %s)", data_produk)

# Data awal
Transaksi
data_transaksi = [

    (1, 101, "Pembelian sabun mandi 2 pcs"),

    (2, 102, "Pembelian air mineral 5 pcs"),

    (3, 103, "Pembelian snack ringan 10 pcs")

]

cursor.execute("SELECT COUNT(*) FROM
Transaksi")if cursor.fetchone()[0] == 0:

    cursor.executemany("INSERT INTO Transaksi (NoTransaksi, NIK, DetailTransaksi) VALUES (%s,
%s,
%s)", data_transaksi)

# Data awal Struk
data_struk = [

```

```

        (1, 1, 2, 20000.00),
        (2, 2, 5, 25000.00),
        (3, 3, 10, 50000.00)
    ]

    cursor.execute("SELECT COUNT(*) FROM
    Struk")if cursor.fetchone()[0] == 0:

        cursor.executemany("INSERT INTO Struk (NoTransaksi, KodeProduk, JumlahProduk, TotalHarga)
VALUES (%s, %s, %s, %s)", data_struk)

    connection.commit()

    print("Data awal berhasil dimasukkan.")

except mysql.connector.Error as err:

    print(f"Error: {err}")

# Fungsi Utilitas

def input_integer(prompt):

    while True:

        try:

            return int(input(prompt))

        except ValueError:

            print("Input harus berupa angka.")

def input_float(prompt):

    while True:

        try:

            return float(input(prompt))

        except ValueError:

            print("Input harus berupa angka desimal.")

```

```
def tampilkan_data(table):
```

```
    cursor.execute(f"SELECT * FROM
```

```
    {table}")rows = cursor.fetchall()
```

```
    if rows:
```

```
        for row in rows:
```

```
            print(row)
```

```
    else:
```

```
        print(f"Tidak ada data di tabel {table}.")
```

```
def hapus_data(table, key_column):
```

```
    key_value = input(f"Masukkan {key_column} untuk data yang ingin dihapus: ")
```

```
    cursor.execute(f"DELETE FROM {table} WHERE {key_column} = %s", (key_value,))
```

```
    connection.commit()
```

```
    print("Data berhasil dihapus jika ditemukan.")
```

```
def input_data_pegawai():
```

```
    nik = input_integer("Masukkan NIK:
```

```
    ")nama = input("Masukkan Nama: ")
```

```
    alamat = input("Masukkan Alamat: ")
```

```
    cursor.execute("INSERT INTO Pegawai (NIK, Nama, Alamat) VALUES (%s, %s, %s)", (nik, nama,
```

```
    alamat))connection.commit()
```

```
    print("Pegawai berhasil ditambahkan.")
```

```
def input_data_produk():
```

```
    kode_produk = input_integer("Masukkan Kode Produk: ")
```

```
    nama_produk = input("Masukkan Nama Produk: ")
```

```
    jenis_produk = input("Masukkan Jenis Produk (Snack/Makanan/Minuman): ")harga_produk
```

```
    = input_float("Masukkan Harga Produk: ")
```

```
cursor.execute("INSERT INTO Produk (KodeProduk, NamaProduk, JenisProduk, HargaProduk)
VALUES(%s, %s, %s, %s)",
```

```
(kode_produk, nama_produk, jenis_produk, harga_produk))
```

```
connection.commit()
```

```
print("Produk berhasil ditambahkan.")
```

```
def input_data_transaksi():
```

```
no_transaksi = input_integer("Masukkan No Transaksi: ")
```

```
nik = input_integer("Masukkan NIK Pegawai: ")
```

```
detail_transaksi = input("Masukkan Detil Transaksi: ")
```

```
cursor.execute("INSERT INTO Transaksi (NoTransaksi, NIK, DetailTransaksi) VALUES (%s, %s,
%s)",(no_transaksi, nik, detail_transaksi))
```

```
connection.commit()
```

```
print("Transaksi berhasil ditambahkan.")
```

```
def input_data_struk():
```

```
no_transaksi = input_integer("Masukkan No Transaksi: ")
```

```
kode_produk = input_integer("Masukkan Kode Produk: ")
```

```
jumlah_produk = input_integer("Masukkan Jumlah Produk: ")
```

```
total_harga = input_float("Masukkan Total Harga: ")
```

```
cursor.execute("INSERT INTO Struk (NoTransaksi, KodeProduk, JumlahProduk, TotalHarga) VALUES
(%s,
%s, %s, %s)",
```

```
(no_transaksi, kode_produk, jumlah_produk, total_harga))
```

```
connection.commit()
```

```
print("Struk berhasil ditambahkan.")
```

```
# Menjalankan setup database
```

```
setup_database()
```



```
# Koneksi database
```

```
connection = connect_to_db()
```

```
cursor = connection.cursor()
```

```
# Menambahkan data awal
```

```
tambah_data_awal()
```

```
# Sistem Menu
```

```
while True:
```

```
    print("\n=== MENU UTAMA ===")
```

```
    print("1. Tampilkan
```

```
Data")print("2. Tambah
```

```
Data") print("3. Hapus
```

```
Data") print("4. Keluar")
```

```
    pilihan = input_integer("Pilih menu: ")
```

```
    if pilihan == 1:
```

```
        # Sub-menu Tampilkan
```

```
Datawhile True:
```

```
    print("\n=== TAMPILKAN DATA ===")
```

```
    print("1. Pegawai")
```

```
    print("2. Produk")
```

```
    print("3.
```

```
Transaksi")print("4.
```

```
Struk")
```

```
    print("5. Kembali ke Menu Utama")
```

```
    sub_pilihan = input_integer("Pilih menu: ")
```

```
if sub_pilihan == 1:
    print("\nData Pegawai:")
    tampilkan_data("Pegawai")
elif sub_pilihan == 2:
    print("\nData Produk:")
    tampilkan_data("Produk")
elif sub_pilihan == 3:
    print("\nData Transaksi:")
    tampilkan_data("Transaksi")
elif sub_pilihan == 4:
    print("\nData Struk:")
    tampilkan_data("Struk")
elif sub_pilihan == 5:
    break
else:
    print("Pilihan tidak valid. Silakan coba lagi.")
```

```
elif pilihan == 2:
    # Sub-menu Tambah Data
    while True:
        print("\n=== TAMBAH DATA ===")
        print("1. Pegawai")
        print("2. Produk")
        print("3. Transaksi")
        print("4. Struk")
        print("5. Kembali ke Menu Utama")

        sub_pilihan = input_integer("Pilih menu: ")
```

```

if sub_pilihan == 1:
    input_data_pegawai()
elif sub_pilihan == 2:
    input_data_produk()
elif sub_pilihan == 3:
    input_data_transaksi()
elif sub_pilihan == 4:
    input_data_struk()
elif sub_pilihan ==
5:
    break
else:
    print("Pilihan tidak valid. Silakan coba lagi.")

elif pilihan == 3:
    # Sub-menu Hapus Data
    while True:
        print("\n=== HAPUS DATA ===")
        print("1. Pegawai")
        print("2. Produk")
        print("3.
Transaksi")print("4.
Struk")
        print("5. Kembali ke Menu Utama")

    sub_pilihan = input_integer("Pilih menu: ")

    if sub_pilihan == 1:
        hapus_data("Pegawai",
        "NIK")
    elif sub_pilihan == 2:
        hapus_data("Produk", "KodeProduk")

```

```

elif sub_pilihan == 3:
    hapus_data("Transaksi",
"NoTransaksi")elif sub_pilihan ==
4:
    hapus_data("Struk",
"IdStruk")elif sub_pilihan
== 5:
    b
re
a
k
el
se
:
    print("Pilihan tidak valid. Silakan coba lagi.")

```

```

elif pilihan == 4:
    print("Keluar dari program. Terima
kasih!")break

```

```

else:
    print("Pilihan tidak valid. Silakan coba lagi.")

```

```

=== MENU UTAMA ===
1. Tampilkan Data
2. Tambah Data
3. Hapus Data
4. Keluar
Pilih menu: 1

=== TAMPILKAN DATA ===
1. Pegawai
2. Produk
3. Transaksi
4. Struk
5. Kembali ke Menu Utama
Pilih menu: █

```

Pilih menu: 1

Data Pegawai:

(101, 'Rafla', 'Jl. Merdeka No. 1')
(102, 'Andi', 'Jl. Sudirman No. 2')
(103, 'Siti', 'Jl. Thamrin No. 3')

=== TAMPILKAN DATA ===

1. Pegawai
2. Produk
3. Transaksi
4. Struk
5. Kembali ke Menu Utama

Pilih menu: 2

Data Produk:

(1, 'Sabun Mandi', 'Snack', Decimal('10000.00'))
(2, 'Air Mineral', 'Minuman', Decimal('5000.00'))
(3, 'Snack Ringan', 'Makanan', Decimal('15000.00'))

=== TAMPILKAN DATA ===

1. Pegawai
2. Produk
3. Transaksi
4. Struk
5. Kembali ke Menu Utama

Pilih menu: 3

Data Transaksi:

(1, 101, 'Pembelian sabun mandi 2 pcs')
(2, 102, 'Pembelian air mineral 5 pcs')
(3, 103, 'Pembelian snack ringan 10 pcs')

=== TAMPILKAN DATA ===

1. Pegawai
2. Produk
3. Transaksi
4. Struk
5. Kembali ke Menu Utama

Pilih menu: 4

Data Struk:

(1, 1, 1, 2, Decimal('20000.00'))
(2, 2, 2, 5, Decimal('25000.00'))
(3, 3, 3, 10, Decimal('50000.00'))

=== TAMPILKAN DATA ===

1. Pegawai
2. Produk
3. Transaksi
4. Struk
5. Kembali ke Menu Utama

Pilih menu: 5

Filters

Containing the word:

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> pegawai	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/> produk	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/> struk	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_general_ci	48.0 KiB	-
<input type="checkbox"/> transaksi	Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_general_ci	32.0 KiB	-
4 tables	Sum	12	InnoDB	utf8mb4_general_ci	112.0 KiB	0 B

☐ Check all

With selected: