

**LAPORAN TUGAS PRMROGRAMAN  
BERIORENTASI OBJEK**

T.A Semester Ganjil 2024/2025



DISUSUN OLEH :

Najwa Shafira Zahra (5230411293)

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

## **KODE PROGRAM**

```
import pymysql
from tabulate import tabulate
from decimal import Decimal

def read_choice(prompt, options):
    """Fungsi untuk membaca input dengan validasi opsi."""
    while True:
        print(prompt)
        choice = input("Pilih opsi: ")
        if choice in options:
            return choice
        print("\nOpsi tidak valid, coba lagi.")

# Konfigurasi koneksi database
db_config = {
    'host': 'localhost',
    'user': 'root', # Ganti dengan username MySQL Anda
    'password': '', # Ganti dengan password MySQL Anda
    'database': 'penjualan',
    'cursorclass': pymysql.cursors.DictCursor # Menentukan bahwa kursor yang
    digunakan adalah DictCursor
}

# Fungsi untuk membuat koneksi database
def get_db_connection():
    try:
        connection = pymysql.connect(**db_config)
        return connection
    except pymysql.MySQLError as e:
        print(f"Kesalahan koneksi ke database: {e}")
        return None

# Fungsi untuk menampilkan daftar pegawai
def show_pegawai():
    connection = get_db_connection()
    if connection is None:
        return

    cursor = connection.cursor()
    try:
        cursor.execute("SELECT * FROM Pegawai")
        pegawai = cursor.fetchall()
```

```

    if pegawai:
        print("\nDaftar Pegawai:")
        print(tabulate(pegawai, headers="keys", tablefmt="pretty"))
    else:
        print("\nBelum ada data pegawai.")
except pymysql.MySQLError as e:
    print(f"Kesalahan saat mengambil data pegawai: {e}")
finally:
    connection.close()

# Fungsi untuk menambahkan data pegawai
def add_pegawai():
    nik = input("Masukkan NIK Pegawai Baru: ")
    nama = input("Masukkan Nama Pegawai Baru: ")
    alamat = input("Masukkan Alamat Pegawai Baru: ")

    connection = get_db_connection()
    if connection is None:
        return

    cursor = connection.cursor()
    try:
        cursor.execute(
            "INSERT INTO Pegawai (NIK, Nama, Alamat) VALUES (%s, %s, %s)",
            (nik, nama, alamat)
        )
        connection.commit()
        print("\nPegawai baru berhasil ditambahkan!")
    except pymysql.IntegrityError:
        print(f"\nPegawai dengan NIK {nik} sudah ada di database!")
    except pymysql.MySQLError as e:
        print(f"\nGagal menambahkan pegawai: {e}")
    finally:
        connection.close()

# Fungsi untuk menampilkan daftar produk berdasarkan kategori
def show_produk(kategori=None):
    connection = get_db_connection()
    if connection is None:
        return

    cursor = connection.cursor()
    try:
        if kategori:
            cursor.execute("SELECT * FROM Produk WHERE JenisProduk = %s",
                (kategori,))

```

```

else:
    cursor.execute("SELECT * FROM Produk")
    produk = cursor.fetchall()
    if produk:
        print(f"\nDaftar Produk ({kategori if kategori else 'Semua Produk'}):")
        print(tabulate(produk, headers="keys", tablefmt="pretty"))
    else:
        print(f"\nBelum ada data produk di kategori {kategori if kategori else 'semua kategori'}.")
except pymysql.MySQLError as e:
    print(f"Kesalahan saat mengambil data produk: {e}")
finally:
    connection.close()

```

# Fungsi untuk menambahkan produk baru

```

def add_produk():
    kode_produk = input("Masukkan Kode Produk: ")
    nama_produk = input("Masukkan Nama Produk: ")
    harga = float(input("Masukkan Harga Produk: "))
    jenis_produk = input("Masukkan Jenis Produk (Snack/Makanan/Minuman): ")

    connection = get_db_connection()
    if connection is None:
        return

    cursor = connection.cursor()
    try:
        cursor.execute(
            "INSERT INTO Produk (KodeProduk, NamaProduk, Harga, JenisProduk) "
            "VALUES (%s, %s, %s, %s)",
            (kode_produk, nama_produk, harga, jenis_produk)
        )
        connection.commit()
        print("\nProduk baru berhasil ditambahkan!")
    except pymysql.MySQLError as e:
        print(f"\nGagal menambahkan produk: {e}")
    finally:
        connection.close()

```

# Fungsi untuk menampilkan data transaksi

```

def show_transaksi():
    connection = get_db_connection()
    if connection is None:
        return

    cursor = connection.cursor()

```

```

try:
    cursor.execute("SELECT * FROM Transaksi")
    transaksi = cursor.fetchall()
    if transaksi:
        print("\nDaftar Transaksi:")
        print(tabulate(transaksi, headers="keys", tablefmt="pretty"))
    else:
        print("\nBelum ada data transaksi.")
except pymysql.MySQLError as e:
    print(f"Kesalahan saat mengambil data transaksi: {e}")
finally:
    connection.close()

# Fungsi untuk menambahkan transaksi baru
def add_transaksi():
    connection = get_db_connection()
    if connection is None:
        return

    cursor = connection.cursor()
    try:
        # Generate nomor transaksi otomatis
        cursor.execute("SELECT IFNULL(MAX(NoTransaksi), 0) + 1 AS
NextNoTransaksi FROM Transaksi")
        next_no_transaksi = cursor.fetchone()['NextNoTransaksi'] # Mengakses nilai
        pertama dari dictionary

        # Pastikan next_no_transaksi adalah integer
        next_no_transaksi = int(next_no_transaksi)

        no_transaksi = f"T{next_no_transaksi:05d}"

        print(f"\nNomor Transaksi: {no_transaksi}")

        no_pegawai = input("Masukkan Nomor Pegawai: ")
        kode_produk = input("Masukkan Kode Produk: ")
        jumlah_produk = int(input("Masukkan Jumlah Produk: "))

        # Ambil harga produk
        cursor.execute("SELECT Harga FROM Produk WHERE KodeProduk = %s",
(kode_produk,))
        produk = cursor.fetchone()
        if produk:
            harga_produk = produk['Harga'] # Mengakses harga produk menggunakan
            nama kolom
            total_harga = harga_produk * jumlah_produk

```

```

print(f"\nTotal Harga: Rp{total_harga}")
uang_dibayar = input("Masukkan Uang yang Dibayarkan: ").replace('Rp',
").replace(',', '').strip()

try:
    uang_dibayar = Decimal(uang_dibayar) # Menggunakan Decimal di sini
except ValueError:
    print("\nInput tidak valid, pastikan hanya angka yang dimasukkan.")
    return # Menghentikan eksekusi jika terjadi kesalahan

# Pastikan total_harga juga menggunakan Decimal
total_harga = Decimal(total_harga) # Mengonversi total_harga menjadi
Decimal jika perlu

if uang_dibayar < total_harga:
    print("\nUang yang dibayarkan kurang, transaksi gagal!")
    return

kembalian = uang_dibayar - total_harga # Operasi ini sekarang aman
print(f"Kembalian: Rp{kembalian:.2f}")

# Simpan transaksi
cursor.execute(
    "INSERT INTO Transaksi (NoTransaksi, NoPegawai, KodeProduk,
JumlahProduk, TotalHarga, UangDibayar, Kembalian) "
    "VALUES (%s, %s, %s, %s, %s, %s, %s)",
    (no_transaksi, no_pegawai, kode_produk, jumlah_produk, total_harga,
uang_dibayar, kembalian)
)
connection.commit()

# Simpan struk
cursor.execute(
    "INSERT INTO Struk (NoTransaksi, NamaPegawai, NamaProduk,
JumlahProduk, TotalHarga) "
    "SELECT %s, p.Nama, pr.NamaProduk, %s, %s "
    "FROM Pegawai p, Produk pr WHERE p.NIK = %s AND pr.KodeProduk =
%s",
    (no_transaksi, jumlah_produk, total_harga, no_pegawai, kode_produk)
)
connection.commit()
print("\nTransaksi berhasil ditambahkan!")
else:
    print("\nProduk tidak ditemukan!")
except pymysql.MySQLError as e:

```

```

        print(f"\nGagal menambahkan transaksi: {e}")
    finally:
        connection.close()

# Menu Pegawai
def menu_pegawai():
    while True:
        choice = read_choice(
            "\n--- Menu Pegawai ---\n1. Lihat Daftar Pegawai\n2. Tambah Pegawai\n3.
Kembali ke Menu Utama",
            ['1', '2', '3']
        )

        if choice == '1':
            show_pegawai()
        elif choice == '2':
            add_pegawai()
        elif choice == '3':
            break

# Menu Produk
def menu_produk():
    while True:
        choice = read_choice(
            "\n--- Menu Produk ---\n1. Lihat Semua Produk\n2. Lihat Produk Snack\n3.
Lihat Produk Makanan\n4. Lihat Produk Minuman\n5. Tambah Produk\n6. Kembali
ke Menu Utama",
            ['1', '2', '3', '4', '5', '6']
        )

        if choice == '1':
            show_produk()
        elif choice == '2':
            show_produk(kategori="Snack")
        elif choice == '3':
            show_produk(kategori="Makanan")
        elif choice == '4':
            show_produk(kategori="Minuman")
        elif choice == '5':
            add_produk()
        elif choice == '6':
            break

# Menu Transaksi
def menu_transaksi():
    while True:

```

```

        choice = read_choice(
            "\n--- Menu Transaksi ---\n1. Lihat Daftar Transaksi\n2. Tambah
Transaksi\n3. Kembali ke Menu Utama",
            ['1', '2', '3']
        )

        if choice == '1':
            show_transaksi()
        elif choice == '2':
            add_transaksi()
        elif choice == '3':
            break

# Menu Utama
def main():
    while True:
        choice = read_choice(
            "\n--- Menu penjualan ---\n1. Menu Pegawai\n2. Menu Produk\n3. Menu
Transaksi\n4. Keluar",
            ['1', '2', '3', '4']
        )

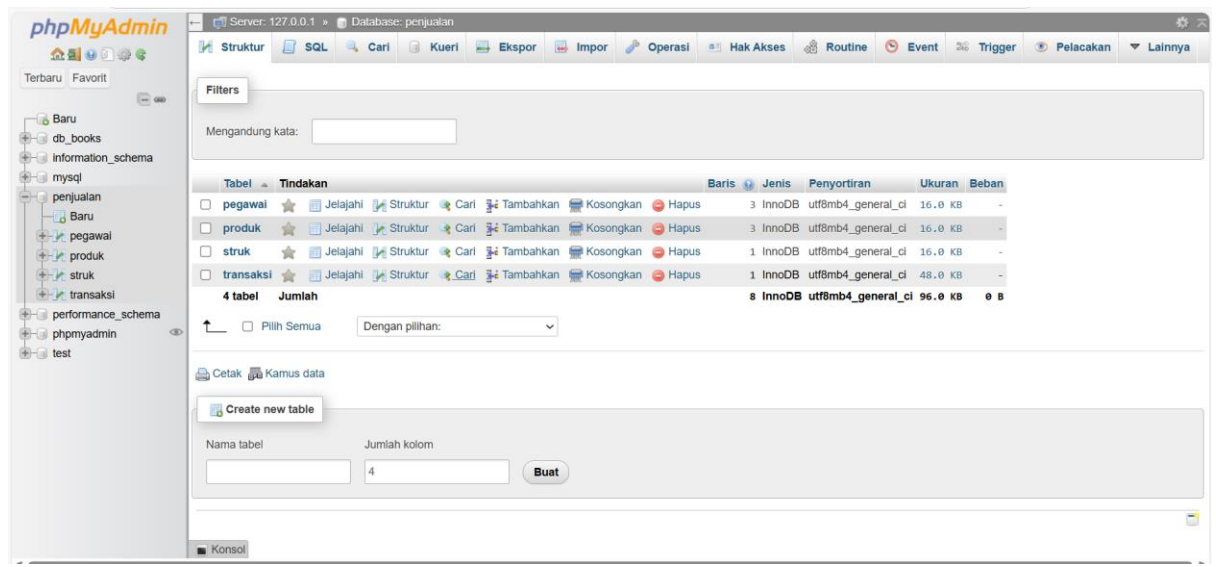
        if choice == '1':
            menu_pegawai()
        elif choice == '2':
            menu_produk()
        elif choice == '3':
            menu_transaksi()
        elif choice == '4':
            print("Terima kasih, program selesai.")
            break

if __name__ == "__main__":
    main()

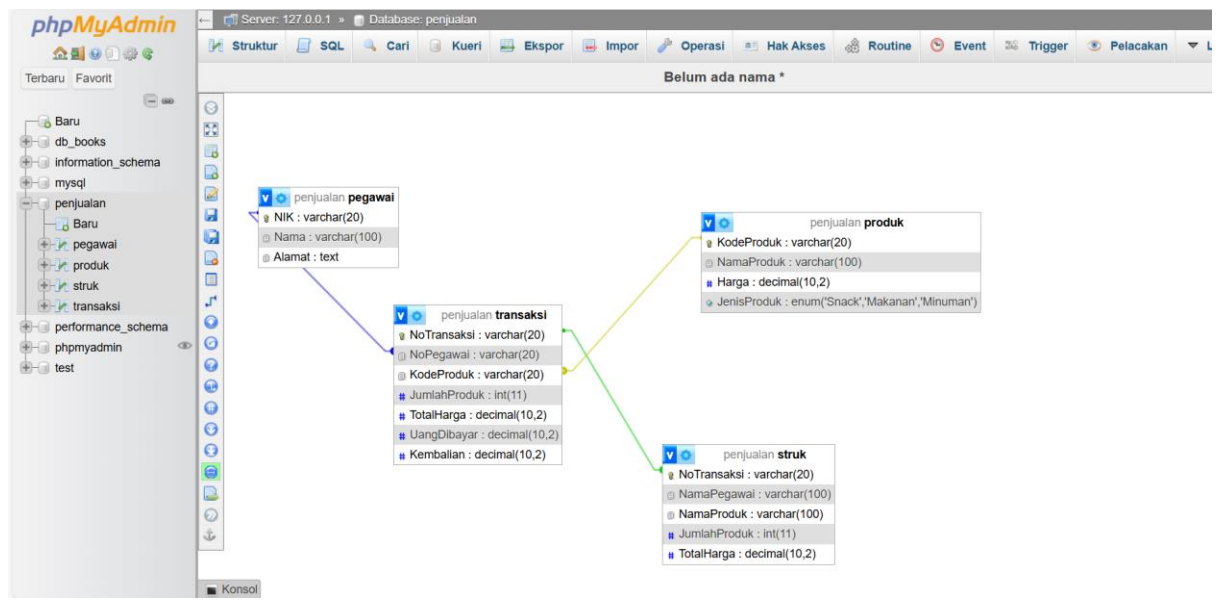
```



# DATABASE



# TABEL RELASI



## **HASIL KODE PROGRAM**

### **Menu utama**

```
--- Menu penjualan ---  
1. Menu Pegawai  
2. Menu Produk  
3. Menu Transaksi  
4. Keluar  
Pilih opsi: █
```

Berikut adalah hasil kode program di menu utama yang terdiri dari 4 opsi yaitu :

1. Menu pegawai
2. Menu Produk
3. Menu Transaksi
4. Keluar

### **Menu pegawai**

```
--- Menu penjualan ---  
1. Menu Pegawai  
2. Menu Produk  
3. Menu Transaksi  
4. Keluar  
Pilih opsi: 1  
  
--- Menu Pegawai ---  
1. Lihat Daftar Pegawai  
2. Tambah Pegawai  
3. Kembali ke Menu Utama  
Pilih opsi: █
```

Jika kita memilih opsi satu maka akan menampilkan sub menu dari menu pegawai yang terdiri dari tiga opsi yaitu lihat daftar pegawai, tambah pegawai, kembali ke menu utama.

## Menu pegawai ( Lihat Daftar Pegawai (1) )

PROBLEMS   OUTPUT   **TERMINAL**   DEBUG CONSOLE   PORTS

```
--- Menu Pegawai ---
1. Lihat Daftar Pegawai
2. Tambah Pegawai
3. Kembali ke Menu Utama
Pilih opsi: 1

Daftar Pegawai:
+-----+
| NIK | Nama | Alamat |
+-----+
| 2345 | Marsha | Semarang |
| 4567 | najwa | lapangan |
| 987 | kiki | bawang |
+-----+
```

NIK

Nama

Alamat

☐

2345

Marsha

Semarang

☐

4567

najwa

lapangan

☐

987

kiki

bawang

Pada menu tampilan ini akan menghasilkan output yaitu melihat daftar pegawai yang sudah di masukan sebelumnya, otomatis data data pegawai tersebut akan tersimpan di dalam database yang sudah kita buat sebelumnya.

## Menu pegawai( Tambah pegawai (2) )

```
--- Menu Pegawai ---
1. Lihat Daftar Pegawai
2. Tambah Pegawai
3. Kembali ke Menu Utama
Pilih opsi: 2
Masukkan NIK Pegawai Baru: 2323
Masukkan Nama Pegawai Baru: Tisti
Masukkan Alamat Pegawai Baru: Bali

Pegawai baru berhasil ditambahkan!
```

Berikut adalah tampilan dari menu tambah pegawai, jika kita memilih opsi dua maka user diminta untuk memasukkan pegawai baru, nama pegawai baru, alamat pegawai baru, setelah itu data pegawai akan berhasil ditambahkan dan otomatis akan tersimpan di dalam database.

Menampilkan baris 0 - 3 (total 4, Pencarian dilakukan dalam 0,000s)

```
SELECT * FROM `pegawai`
```

☐ Profil

[ Edit kotak ]

[ Ubah ]

[ Jelaskan SQL ]

[ Buat kode PHP ]

[ S ]

☐ Tampilkan semua

Jumlah baris: 25

Saring baris

Extra options

NIK

Nama

Alamat

☐

2323

Tisti

Bali

☐

2345

Marsha

Semarang

☐

4567

najwa

lapangan

☐

987

kiki

bawang

### **Menu pegawai (keluar (3) )**

```
--- Menu Pegawai ---
1. Lihat Daftar Pegawai
2. Tambah Pegawai
3. Kembali ke Menu Utama
Pilih opsi: 3

--- Menu penjualan ---
1. Menu Pegawai
2. Menu Produk
3. Menu Transaksi
4. Keluar
Pilih opsi: █
```

Di dalam menu ini jika kita memilih nomer tiga maka program akan kembali ke menu utama.

### **Menu Produk**

```
--- Menu penjualan ---
1. Menu Pegawai
2. Menu Produk
3. Menu Transaksi
4. Keluar
Pilih opsi: 2

--- Menu Produk ---
1. Lihat Semua Produk
2. Lihat Produk Snack
3. Lihat Produk Makanan
4. Lihat Produk Minuman
5. Tambah Produk
6. Kembali ke Menu Utama
```

Di dalam menu ini terdapat enam opsi sub menu lagi yaitu lihat semua produk, lihat produk snack, lihat produk makanan, lihat produk minuman, tambah produk, Kembali ke menu utama

## Menu Produk (Lihat semua produk (1) )

```
--- Menu Produk ---
1. Lihat Semua Produk
2. Lihat Produk Snack
3. Lihat Produk Makanan
4. Lihat Produk Minuman
5. Tambah Produk
6. Kembali ke Menu Utama
Pilih opsi: 1
```

Daftar Produk (Semua Produk):

KodeProduk	NamaProduk	Harga	JenisProduk
A001	Chitato	10000.00	Snack
A002	Bakso	5000.00	Makanan
A003	Ice Matca	7000.00	Minuman
A004	Coki coki	2000.00	Snack
A005	Mie ayam	12000.00	Makanan
A006	Pop ice	5000.00	Minuman
P001	Keripik	5000.00	Snack
P002	Nasi Goreng	15000.00	Makanan
P003	Jus Jeruk	10000.00	Minuman

Jika kita memilih nomer satu di menu produk maka akan menampilkan semua produk yang tersimpan di dalam database

## Menu Produk (Lihat produk snack (2))

```
--- Menu Produk ---
1. Lihat Semua Produk
2. Lihat Produk Snack
3. Lihat Produk Makanan
4. Lihat Produk Minuman
5. Tambah Produk
6. Kembali ke Menu Utama
Pilih opsi: 2
```

Daftar Produk (Snack):

KodeProduk	NamaProduk	Harga	JenisProduk
A001	Chitato	10000.00	Snack
A004	Coki coki	2000.00	Snack
P001	Keripik	5000.00	Snack

Didalam sub menu ini menampilkan hasil kode program yaitu daftar produk tetapi hanya kategori snack saja.

### Menu Produk ( Lihat produk makanan (3))

```
--- Menu Produk ---
1. Lihat Semua Produk
2. Lihat Produk Snack
3. Lihat Produk Makanan
4. Lihat Produk Minuman
5. Tambah Produk
6. Kembali ke Menu Utama
Pilih opsi: 3
```

Daftar Produk (Makanan):

KodeProduk	NamaProduk	Harga	JenisProduk
A002	Bakso	5000.00	Makanan
A005	Mie ayam	12000.00	Makanan
P002	Nasi Goreng	15000.00	Makanan

Jika kita memilih sub menu nomer tiga maka akan menampilkan output yaitu semua daftar produk makanan

### Menu produk (Lihat produk minuman (4))

```
--- Menu Produk ---
1. Lihat Semua Produk
2. Lihat Produk Snack
3. Lihat Produk Makanan
4. Lihat Produk Minuman
5. Tambah Produk
6. Kembali ke Menu Utama
Pilih opsi: 4
```

Daftar Produk (Minuman):

KodeProduk	NamaProduk	Harga	JenisProduk
A003	Ice Matca	7000.00	Minuman
A006	Pop ice	5000.00	Minuman
P003	Jus Jeruk	10000.00	Minuman

Jika kita memilih sub menu nomer empat maka akan menampilkan output yaitu semua daftar produk minuman

### **Menu produk (Tambah produk (5))**

```
--- Menu Produk ---
1. Lihat Semua Produk
2. Lihat Produk Snack
3. Lihat Produk Makanan
4. Lihat Produk Minuman
5. Tambah Produk
6. Kembali ke Menu Utama
Pilih opsi: 5
Masukkan Kode Produk: B001
Masukkan Nama Produk: Gerry
Masukkan Harga Produk: 2000
Masukkan Jenis Produk (Snack/Makanan/Minuman): snack

Produk baru berhasil ditambahkan!
```

Di dalam sub menu ini akan menampilkan output tambah menu, jika user memilih opsi ini maka user diminta untuk memasukkan kode produk, nama produk, harga produk, jenis produk tersebut, jika produk berhasil ditambahkan maka otomatis akan tersimpan di dalam database.

### **Menu transaksi (Lihat daftar))**

```
--- Menu penjualan ---
1. Menu Pegawai
2. Menu Produk
3. Menu Transaksi
4. Keluar
Pilih opsi: 3

--- Menu Transaksi ---
1. Lihat Daftar Transaksi
2. Tambah Transaksi
3. Kembali ke Menu Utama
Pilih opsi: █
```

Berikut adalah menu transaksi yang memiliki beberapa sub menu yaitu lihat daftar transaksi, tambah transaksi, Kembali ke menu utama.

## Menu transaksi (Lihat daftar transaksi))

```
--- Menu Transaksi ---
1. Lihat Daftar Transaksi
2. Tambah Transaksi
3. Kembali ke Menu Utama
Pilih opsi: 1

Daftar Transaksi:
+-----+-----+-----+-----+-----+-----+
| NoTransaksi | NoPegawai | KodeProduk | JumlahProduk | TotalHarga | UangDibayar | Kembalian |
+-----+-----+-----+-----+-----+-----+
| T00001      | 987       | P003       | 2             | 20000.00   | 10000.00   | 80000.00   |
+-----+-----+-----+-----+-----+-----+
```

Berikut adalah hasil output dari sub menu nomer satu, melihat semua daftar transaksi yang sudah dimasukan sebelumnya.

## Menu transaksi (Tambah transaksi))

```
1. Lihat Daftar Transaksi
2. Tambah Transaksi
3. Kembali ke Menu Utama
Pilih opsi: 2

Nomor Transaksi: T00001
Masukkan Nomor Pegawai: 2345
Masukkan Kode Produk: A001
Masukkan Jumlah Produk: 2

Total Harga: Rp20000.00
Masukkan Uang yang Dibayarkan: 10000

Uang yang dibayarkan kurang, transaksi gagal!

--- Menu Transaksi ---
```

Jika kita memilih opsi nomer dua maka untuk menambah transaksi, didalam menu tersebut user diminta untuk memasukan nomor transaksi, nomor pegawai, kode produk, jumlah produk, dan uang yang akan di bayarkan, contoh di atas menunjukan bahwa transaksi gagal di karenakan uang yang di bayarkan kurang, maka transaksi gagal , jika berhasil maka otomatis akan masuk ke dalam database.



