UAS Pemrograman 3

Nama: Maulana Azhri
Npm: 202343501710
Kelas: R3V

1. **Proses Perhitungan**

1. Output dari No 1 adalah
Digit awal NPM anda : 2
Digit akhir NPM anda : 0
Hasil digit terakhir NPM 0 : 730

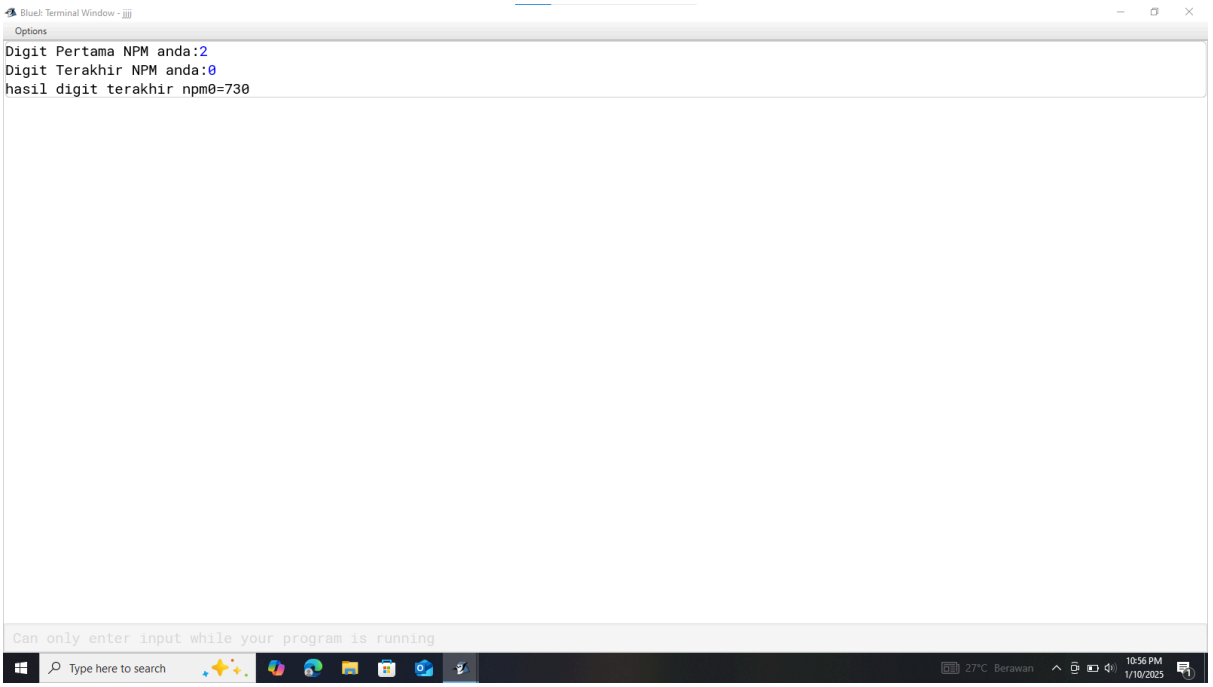
Proses Perhitungan
. Vas (a.b) = 3. Vas (a.b-1) - 2
b = 6 hingga b = 0

Base case (b=0)
Vas (a.0) = 2

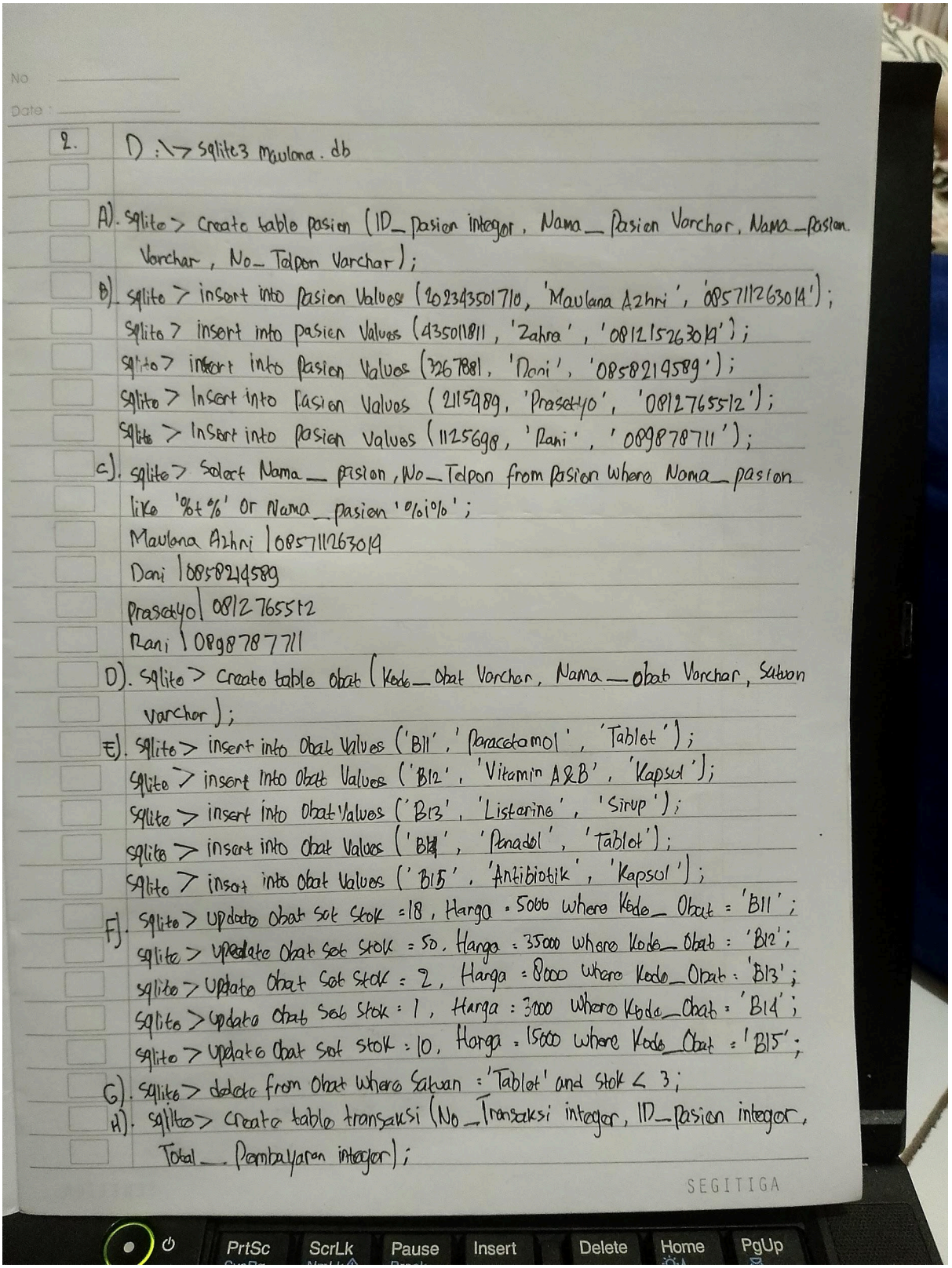
Rekursi ke atas :
- Vas (a.1) = 3. Vas (a.0) - 2 = 3.2 - 2 = 6 - 2 = 4
- Vas (a.2) = 3. Vas (a.1) - 2 = 3.4 - 2 = 12 - 2 = 10
- Vas (a.3) = 3. Vas (a.3) - 2 = 3 . 28 - 2 = 84 - 2 = 82

- Vas (a.5) = 3. Vas (a.4) - 2 = 3.82 - 2 = 246 - 2 = 244
- Vas (a.6) = 3. Vas (a.5) - 2 = 3.244 - 2 = 732 - 2
= 730 //

Output Programnya



2. **Tulis Tangan**



2. D:\> sqlite3 Maulana.db

A). sqlite> Create table pasien (ID_pasien integer, Nama_Pasien Varchar, Nama_pasien Varchar, No_Telpon Varchar);

B). sqlite> insert into pasien Values (10234350710, 'Maulana Azhri', '085711263014');
sqlite> insert into pasien Values (435011811, 'Zahra', '081215263014');
sqlite> insert into pasien Values (326788l, 'Dani', '08582l4589');
sqlite> Insert into Pasien Values (2115489, 'Prasetyo', '0812765512');
sqlite> Insert into pasien Values (1125698, 'Rani', '08987878711');

c). sqlite> Select Nama_pasien, No_Telpon from pasien where Nama_pasien like '%t%' or Nama_pasien '%i%';
Maulana Azhri | 085711263014
Dani | 08582l4589
Prasetyo | 0812765512
Rani | 08987878711

D). sqlite> Create table obat (Kode_Obat Varchar, Nama_obat Varchar, Satuan varchar);

E). sqlite> insert into Obat Values ('B11', 'Paracetamol', 'Tablet');
sqlite> insert into Obat Values ('B12', 'Vitamin A&B', 'Kapsul');
sqlite> insert into Obat Values ('B13', 'Listerine', 'Sirup');
sqlite> insert into obat Values ('B14', 'Panadol', 'Tablet');
sqlite> insert into obat Values ('B15', 'Antibiotik', 'Kapsul');

F). sqlite> Update obat set stok = 18, Harga = 5000 where Kode_Obat = 'B11';
sqlite> Update obat set stok = 50, Harga = 35000 where Kode_Obat = 'B12';
sqlite> Update obat set stok = 2, Harga = 8000 where Kode_Obat = 'B13';
sqlite> Update obat set stok = 1, Harga = 3000 where Kode_Obat = 'B14';
sqlite> Update obat set stok = 10, Harga = 15000 where Kode_Obat = 'B15';

G). sqlite> delete from Obat where Satuan = 'Tablet' and stok < 3;

H). sqlite> Create table transaksi (No_Transaksi integer, ID_pasien integer, Total_Pembayaran integer);

A.

```
sqlite> create table pasien (ID_Pasien integer, Nama_Pasien Varchar, No_Telpon Varchar);
```

sqlite> create table pasien (ID_Pasien integer, Nama_Pasien Varchar, No_Telpon Varchar);

B.

```
sqlite> insert into pasien values (202343501710,'Maulana Azhri','085711263014');
sqlite> insert into pasien values (435011811,'Zahra','081215263014');
sqlite> insert into pasien values (3267881,'Dani','0858214589');
sqlite> insert into pasien values (2115489,'Prasetyo','0812765512');
sqlite> insert into pasien values (1125698,'Rani','0898787711');
```

sqlite> insert into pasien values (202343501710,'Maulana Azhri','085711263014');
sqlite> insert into pasien values (435011811,'Zahra','081215263014');
sqlite> insert into pasien values (3267881,'Dani','0858214589');
sqlite> insert into pasien values (2115489,'Prasetyo','0812765512');
sqlite> insert into pasien values (1125698,'Rani','0898787711');

C.

```
sqlite> select Nama_Pasien,No_Telpon from pasien where Nama_Pasien like '%t%' or Nama_Pasien like '%i%';
Maulana Azhri|085711263014
Dani|0858214589
Prasetyo|0812765512
Rani|0898787711
```

sqlite> select Nama_Pasien,No_Telpon from pasien where Nama_Pasien like '%t%' or Nama_Pasien like '%i%';
Maulana Azhri|085711263014
Dani|0858214589
Prasetyo|0812765512
Rani|0898787711

D.

```
sqlite> create table obat(Kode_Obat varchar, Nama_Obat varchar, Satuan Varchar);
```

sqlite> create table obat(Kode_Obat varchar, Nama_Obat varchar, Satuan Varchar);

E.

```
sqlite> insert into obat values ('B11', 'Paracetamol', 'Tablet');
sqlite> insert into obat values ('B12', 'Vitamin A&B', 'Kapsul');
sqlite> insert into obat values ('B13', 'Listerine', 'Sirup');
sqlite> insert into obat values ('B14', 'panadol', 'Tablet');
sqlite> insert into obat values ('B15', 'Antibiotik', 'Kapsul');
```

sqlite> insert into obat values ('B11', 'Paracetamol', 'Tablet');
sqlite> insert into obat values ('B12', 'Vitamin A&B', 'Kapsul');
sqlite> insert into obat values ('B13', 'Listerine', 'Sirup');

sqlite> insert into obat values ('B14', 'panadol', 'Tablet');
sqlite> insert into obat values ('B15', 'Antibiotik', 'Kapsul');

F.

```
sqlite> update obat set Stok =18, Harga=5000 where Kode_Obat='B11';
sqlite> update obat set Stok =50, Harga=35000 where Kode_Obat='B12';
sqlite> update obat set Stok =2, Harga=8000 where Kode_Obat='B13';
sqlite> update obat set Stok =1, Harga=3000 where Kode_Obat='B14';
sqlite> update obat set Stok =10, Harga=15000 where Kode_Obat='B15';
```

sqlite> update obat set Stok =18, Harga=5000 where Kode_Obat='B11';
sqlite> update obat set Stok =50, Harga=35000 where Kode_Obat='B12';
sqlite> update obat set Stok =5, Harga=8000 where Kode_Obat='B13';
sqlite> update obat set Stok =15, Harga=4000 where Kode_Obat='B14';
sqlite> update obat set Stok =10, Harga=15000 where Kode_Obat='B15';

G.

```
sqlite> delete from obat where Satuan ='Tablet' and Stok <3;
```

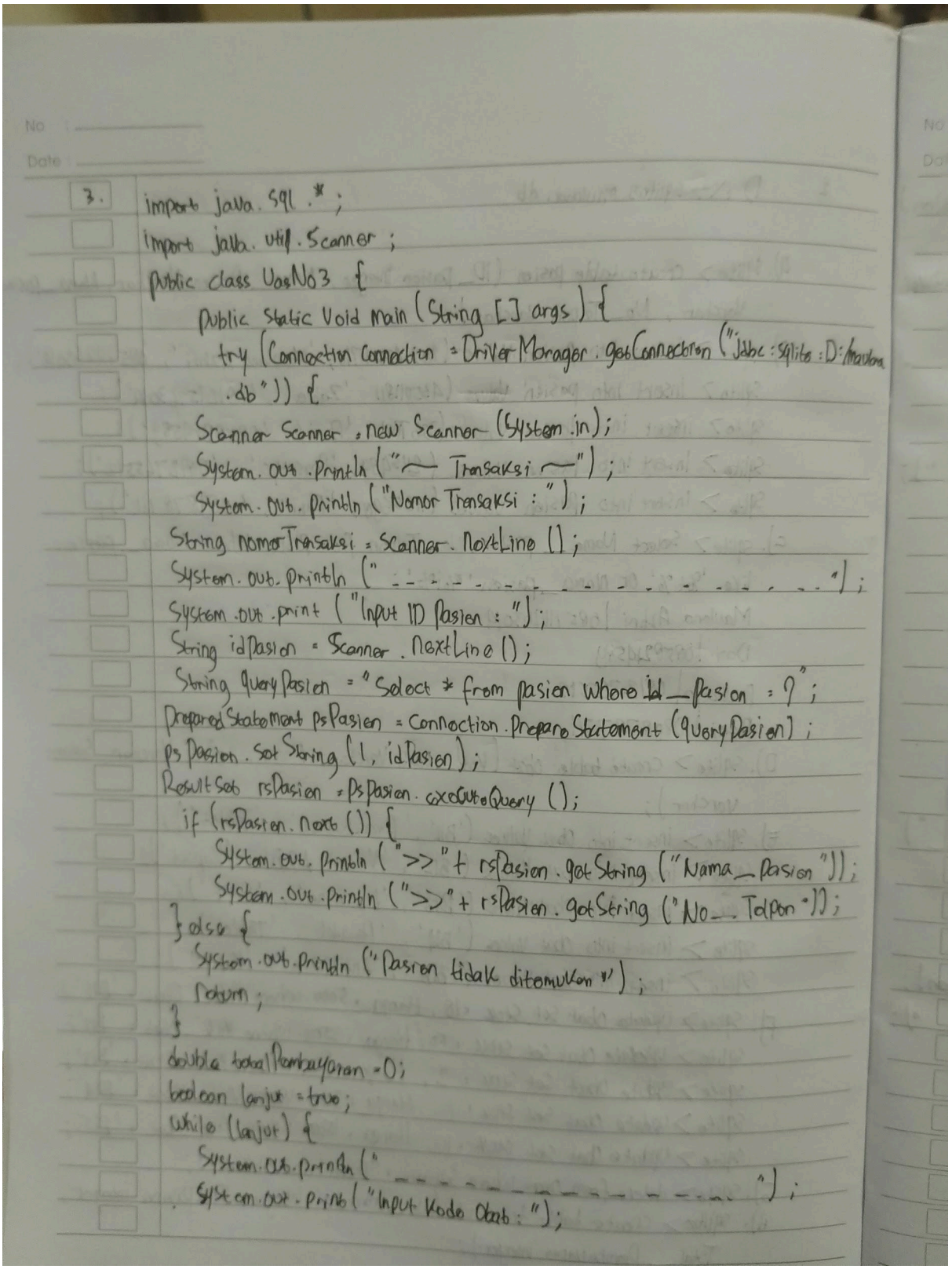
sqlite> delete from obat where Satuan ='Tablet' and Stok < 3;

H.

```
sqlite> Create table transaksi (No_Transaksi integer, ID_Pasien integer, Total_Pembayaran integer);
```

sqlite> Create table transaksi (No_Transaksi integer, ID_Pasien integer, Total_Pembayaran integer);

3. **Tulis Tangan Program**



```
3.
import java.sql.*;
import java.util.Scanner;
public class UasNo3 {
    public static void main(String[] args) {
      try (Connection connection = DriverManager.getConnection("jdbc:sqlite:D:/kuliah
        .db")) {
      Scanner scanner = new Scanner(System.in);
      System.out.println("~ Transaksi ~");
      System.out.println("Nomor Transaksi :");
    String nomorTransaksi = scanner.nextLine();
    System.out.println("-----------------");
    System.out.print("Input ID Pasien :");
    String idPasien = scanner.nextLine();
    String queryPasien = "Select * from pasien where id_pasien = ?";
    PreparedStatement psPasien = connection.prepareStatement(queryPasien);
    psPasien.setString(1, idPasien);
    ResultSet rsPasien = psPasien.executeQuery();
      if (rsPasien.next()) {
        System.out.println(">>" + rsPasien.getString("Nama_Pasien"));
        System.out.println(">>" + rsPasien.getString("No_Telpon"));
    } else {
      System.out.println("Pasien tidak ditemukan");
      return;
    }
    double totalPembayaran = 0;
    boolean lanjut = true;
    while (lanjut) {
      System.out.println("-----------------");
      System.out.print("Input Kode Obat :");
```

No
Date

```
String KodeObat = Scanner.nextLine();
String QueryObat = "Select * from obat where Kode_Obat = ?";
PreparedStatement psObat = Connection.PreparedStatement (QueryObat);
psObat.SetString (1, KodeObat);
ResultSet rsObat = psObat.executeQuery();
    if (rsObat.next()) {
        String nama = rsObat.getString ("Nama_Obat");
        double harga = rsObat.getDouble ("Harga ");
        int stok = rsObat.getString ("Satuan");
String jenis = rsObat.getString ("Satuan ");
System.out.println (">> "+ nama);
System.out.println(">> Rp "+ harga);
System.out.println(">> Tersedia "+ stok +" "+ jenis);
System.out.println("- - - - - - - - - - - - - - - - - - - -");
System.out.print ("Input jumlah : ");
int jumlah = Scanner.nextInt();
Scanner.nextLine();
    if (jumlah > stok) {
     System.out.println ("jumlah melebihi stok tersedia!");
} else {
    double subtotal = jumlah * harga;
   total pembayaran += subtotal;
   System.out.println(" Subtotal >> Rp "+ subtotal);
String UpdateStokQuery = "Update Obat set stok = stok - ? where Kode_Obat
   = ?";
PreparedStatement psUpdateStok = Connection.PrepareStatement (UpdateStok
Query);
psUpdatStok.SetInt (1, jumlah);
psUpdateStok.SetString (2, KodeObat);
psUpdateStok.executeUpdate ();
```

SEGITIGA

```
String insertTransaksiQuery = "Insert Into Transaksi (No_Transaksi, ID_Pasien)
Values (?,?)";
PreparedStatement psInsertTransaksi = Connection.prepareStatement (insertTransaksi
Query);
psInsertTransaksi.setString (1, nomorTransaksi);
psInsertTransaksi.setString (2, ~~Nomor~~ IDPasien);
psInsertTransaksi.executeUpdate();
System.out.println("_____");
    }
} else {
    System.out.println ("Obat tidak ditemukan");
}
System.out.print ("[?] lanjut Pembayaran lain ? [Ya/Tidak]:");
String Pilihan = Scanner.nextLine ();

if (pilihan.equalsIgnoreCase ("Tidak")) {
    lanjut = false;
    System.out.println ("_____");
}

}
System.out.println ("Total Pembayaran >> Rp " + total Pembayaran);
System.out.println("_____");
} catch (SQLException e) {
    System.out.println ("Kesalahan :" + e.getMassage ());
  }
 }
}
```

SEGITIGA

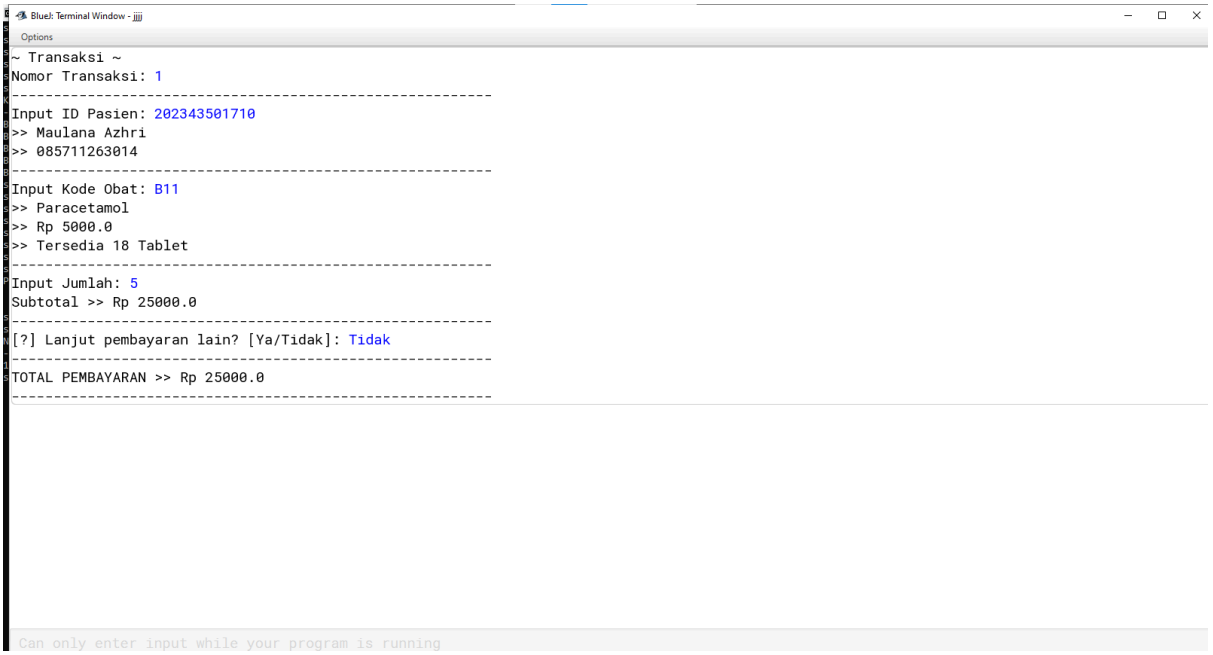
Copy Paste Program

```
import java.sql.*;
import java.util.Scanner;
public class UasNo3 {
    public static void main(String[] args) {
        try (Connection connection =
DriverManager.getConnection("jdbc:sqlite:D:/maulana.db")) {
            Scanner scanner = new Scanner(System.in);
            System.out.println("~ Transaksi ~");
            System.out.print("Nomor Transaksi: ");
            String nomorTransaksi = scanner.nextLine();
            System.out.println("-----------------------------------------------------");
            System.out.print("Input ID Pasien: ");
            String idPasien = scanner.nextLine();
            String queryPasien = "SELECT * FROM pasien WHERE id_pasien = ?";
            PreparedStatement psPasien = connection.prepareStatement(queryPasien);
            psPasien.setString(1, idPasien);
            ResultSet rsPasien = psPasien.executeQuery();
            if (rsPasien.next()) {
                System.out.println(">> " + rsPasien.getString("Nama_Pasien"));
                System.out.println(">> " + rsPasien.getString("No_Telpon"));
            } else {
                System.out.println("Pasien tidak ditemukan.");
                return;
            }
            double totalPembayaran = 0;
            boolean lanjut = true;
            while (lanjut) {
                System.out.println("-----------------------------------------------------");
                System.out.print("Input Kode Obat: ");
                String kodeObat = scanner.nextLine();
                String queryObat = "SELECT * FROM obat WHERE kode_obat = ?";
                PreparedStatement psObat = connection.prepareStatement(queryObat);
                psObat.setString(1, kodeObat);
                ResultSet rsObat = psObat.executeQuery();
                if (rsObat.next()) {
                    String nama = rsObat.getString("Nama_Obat");
                    double harga = rsObat.getDouble("Harga");
                    int stok = rsObat.getInt("Stok");
                    String jenis = rsObat.getString("Satuan");
                    System.out.println(">> " + nama);
                    System.out.println(">> Rp " + harga);
                    System.out.println(">> Tersedia " + stok + " " + jenis);
                    System.out.println("-----------------------------------------------------");
                    System.out.print("Input Jumlah: ");
```

```
                int jumlah = scanner.nextInt();
                scanner.nextLine();
                if (jumlah > stok) {
                    System.out.println("Jumlah melebihi stok tersedia!");
                } else {
                    double subtotal = jumlah * harga;
                    totalPembayaran += subtotal;
                    System.out.println("Subtotal >> Rp " + subtotal);
                    String updateStokQuery = "UPDATE obat SET stok = stok - ? WHERE
kode_obat = ?";
                    PreparedStatement psUpdateStok =
connection.prepareStatement(updateStokQuery);
                    psUpdateStok.setInt(1, jumlah);
                    psUpdateStok.setString(2, kodeObat);
                    psUpdateStok.executeUpdate();
                    String insertTransaksiQuery = "INSERT INTO transaksi (No_Transaksi,
ID_pasien) VALUES (?, ?)";
                    PreparedStatement psInsertTransaksi =
connection.prepareStatement(insertTransaksiQuery);
                    psInsertTransaksi.setString(1, nomorTransaksi);
                    psInsertTransaksi.setString(2, idPasien);
                    psInsertTransaksi.executeUpdate();
                    System.out.println("-----------------------------------------------------");
                }
            } else {
                System.out.println("Obat tidak ditemukan.");
            }
            System.out.print("[?] Lanjut pembayaran lain? [Ya/Tidak]: ");
            String pilihan = scanner.nextLine();

            if (pilihan.equalsIgnoreCase("Tidak")) {
                lanjut = false;
                System.out.println("-----------------------------------------------------");
            }
        }
        System.out.println("TOTAL PEMBAYARAN >> Rp " + totalPembayaran);
        System.out.println("-----------------------------------------------------");
    } catch (SQLException e) {
    System.out.println("Kesalahan: " + e.getMessage());
 }
 }
}
```

Hasil Output Program


BlueJ: Terminal Window - jjjj
Options
~ Transaksi ~
Nomor Transaksi: 1
Input ID Pasien: 202343501710
>> Maulana Azhri
>> 085711263014
Input Kode Obat: B11
>> Paracetamol
>> Rp 5000.0
>> Tersedia 18 Tablet
Input Jumlah: 5
Subtotal >> Rp 25000.0
[?] Lanjut pembayaran lain? [Ya/Tidak]: Tidak
TOTAL PEMBAYARAN >> Rp 25000.0
Can only enter input while your program is running