**LAPORAN TUGAS BESAR**

**PEMROGRAMAN BERORIENTASI OBJEK**

**SISTEM PENCATATAN BUKU**

# HALAMAN JUDUL



**Disusun Oleh :**

**NAMA : Afsha Alifia Putri**

**NIM : 32602200031**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI**

**UNIVERSITAS ISLAM SULTAN AGUNG**

**SEMARANG**

**2022**

# BAB I Pendahuluan

## **Latar belakang**

Pengelolaan buku merupakan aspek penting dalam kehidupan sehari-hari, baik untuk keperluan pribadi maupun institusional. Pemanfaatan sistematis dalam mencatat dan melacak informasi buku menjadi suatu kebutuhan.

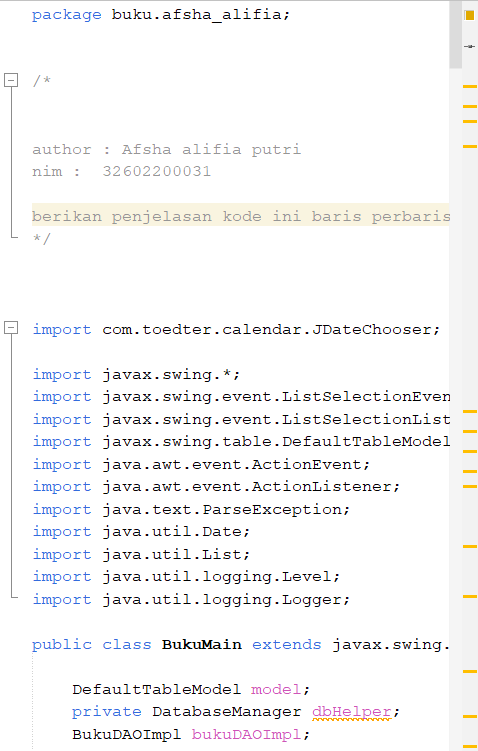
## **Tujuan**

Tujuan dari pengembangan program manajemen buku ini adalah meningkatkan efisiensi dan akurasi dalam pencatatan buku, memudahkan pengguna dalam mencari dan memanipulasi data buku, serta menyediakan basis data yang terorganisir dengan baik.

# BAB II Struktur Program

## **File Utama ( BukuMain )**

BukuMain merupakan file utama program yang berisi fungsi main untuk menjalankan aplikasi manajemen buku.



Gambar 1. 1 File Utama MainBuku



Gambar 1. 2 BukuMain 2



Gambar 1. 3 BukuMain 3



Gambar 1. 4 BukuMain 4



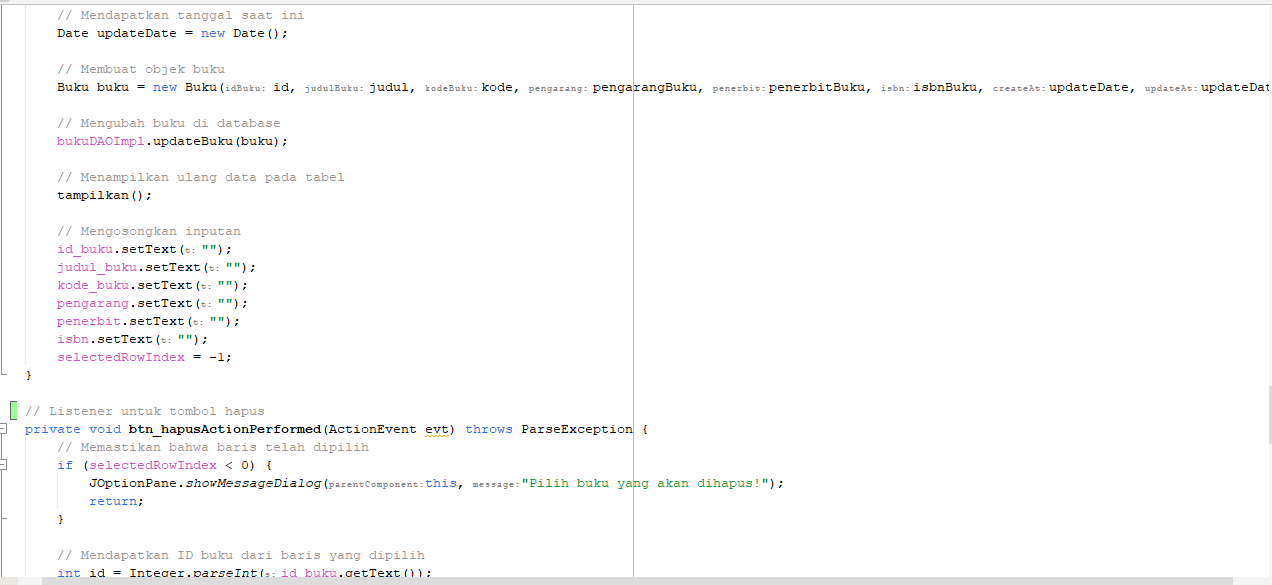
Gambar 1. 5 BukuMain 5



Gambar 1. 6 BukuMain 6



Gambar 1. 7 BukuMain 7



Gambar 1. 8 BukuMain 8



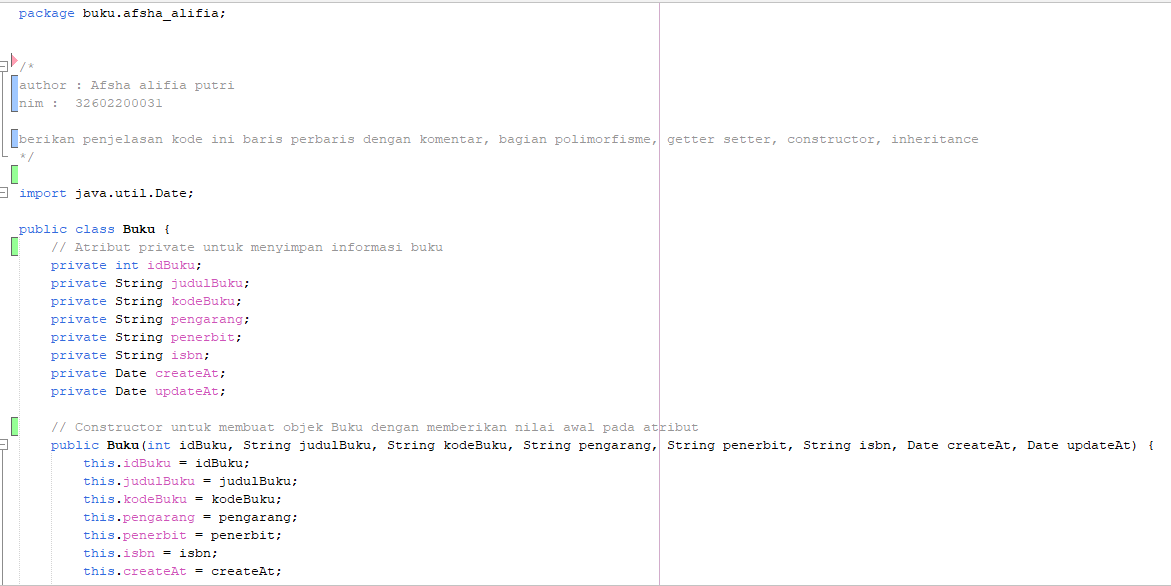
Gambar 1. 9 BukuMain 9



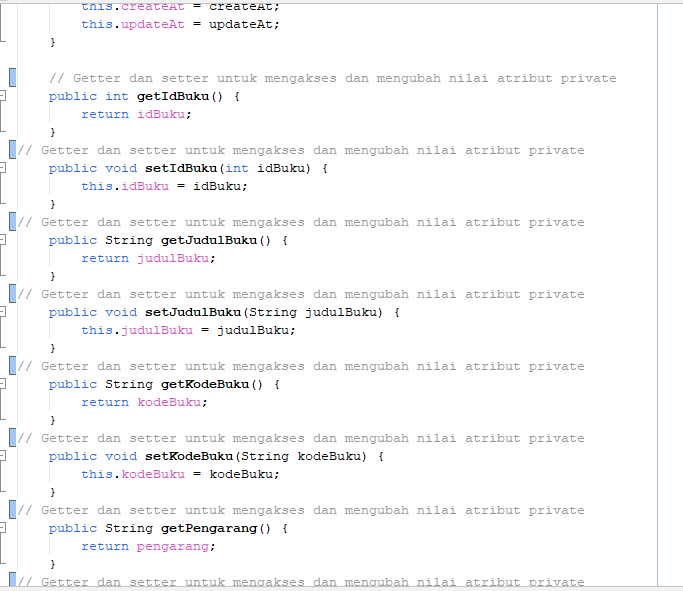
Gambar 1. 10 BukuMain 10

## **Kelas Buku**

Kelas Buku memiliki atribut judulBuku, kodeBuku, pengarang, penerbit, isbn, createAt, dan updateAt. Setiap atribut memiliki metode getter dan setter.



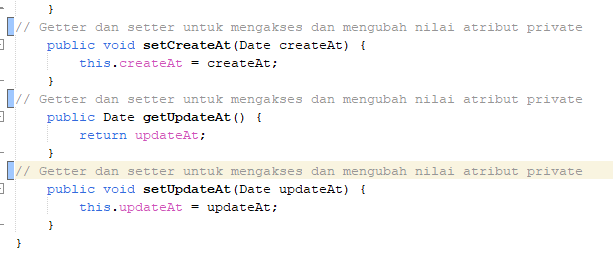
Gambar 1. 11 Kelas Buku 1



Gambar 1. 12 Kelas Buku 2



Gambar 1. 13 Buku 3



Gambar 1. 14 Buku 4

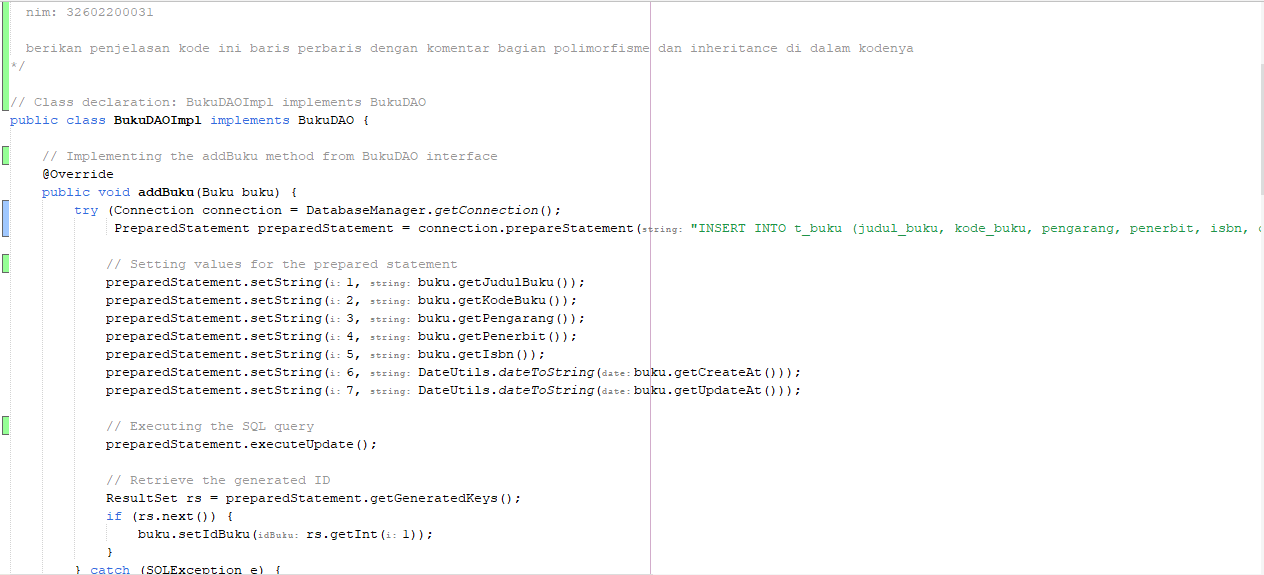
Penjelasan :

* 1. Constructor (public Buku(...)):

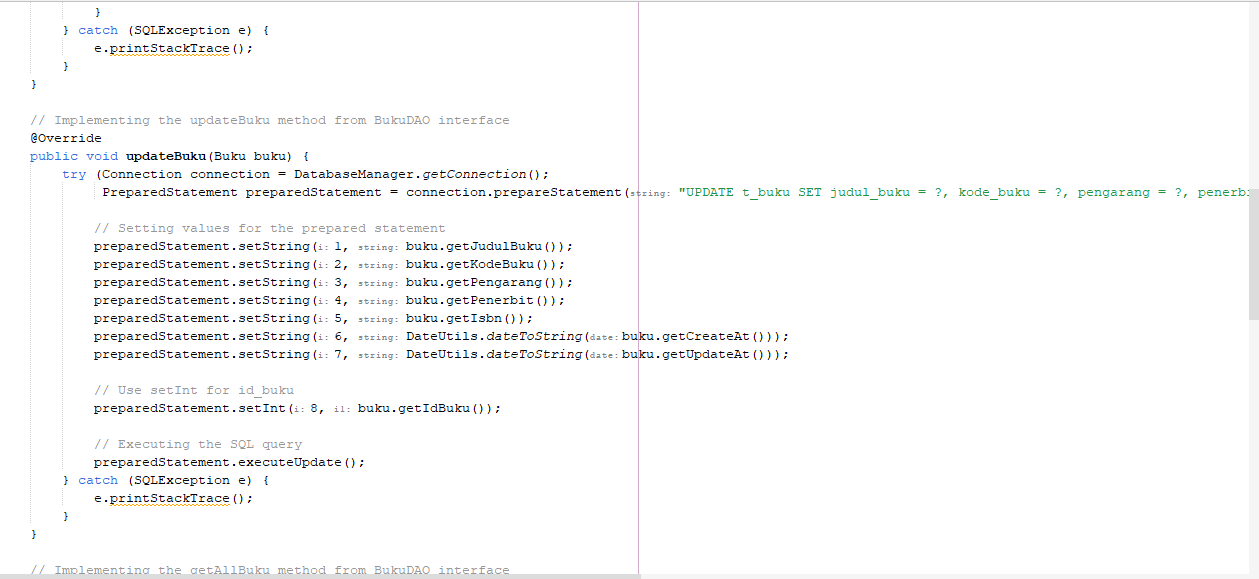
1. Membuat objek Buku dengan memberikan nilai awal pada atribut.
2. Digunakan saat pembuatan objek baru dari kelas Buku.
   1. Getter dan Setter:
3. Getter digunakan untuk mendapatkan nilai dari atribut private.
4. Setter digunakan untuk mengubah nilai dari atribut private.
5. Memungkinkan akses terkontrol ke atribut kelas Buku.

## **Kelas BukuDAOImpl**

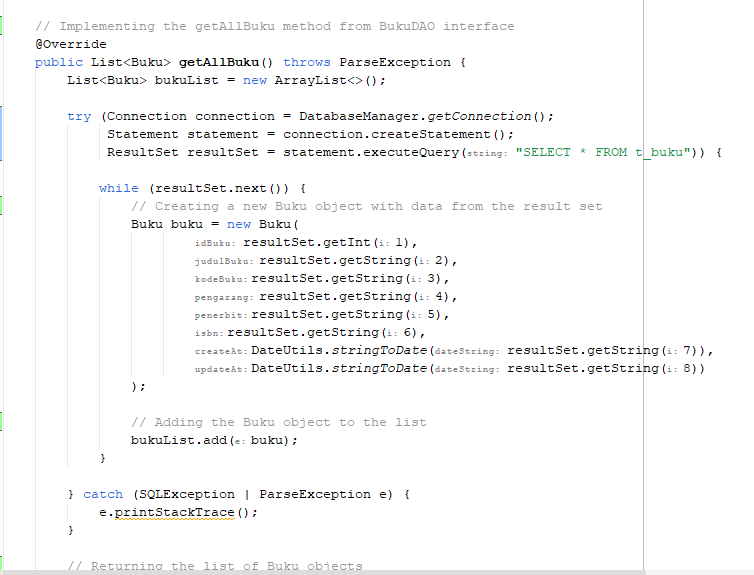
BukuImpl berisi implementasi logika bisnis terkait entitas Buku, termasuk penambahan, pengubahan, dan penghapusan data buku..



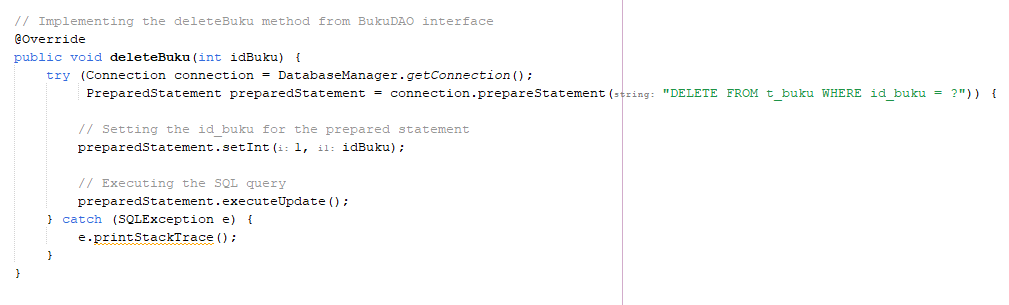
Gambar 1. 15 Kelas BukuDAOImpl 1



Gambar 1. 16 Kelas BukuDAOImpl 2



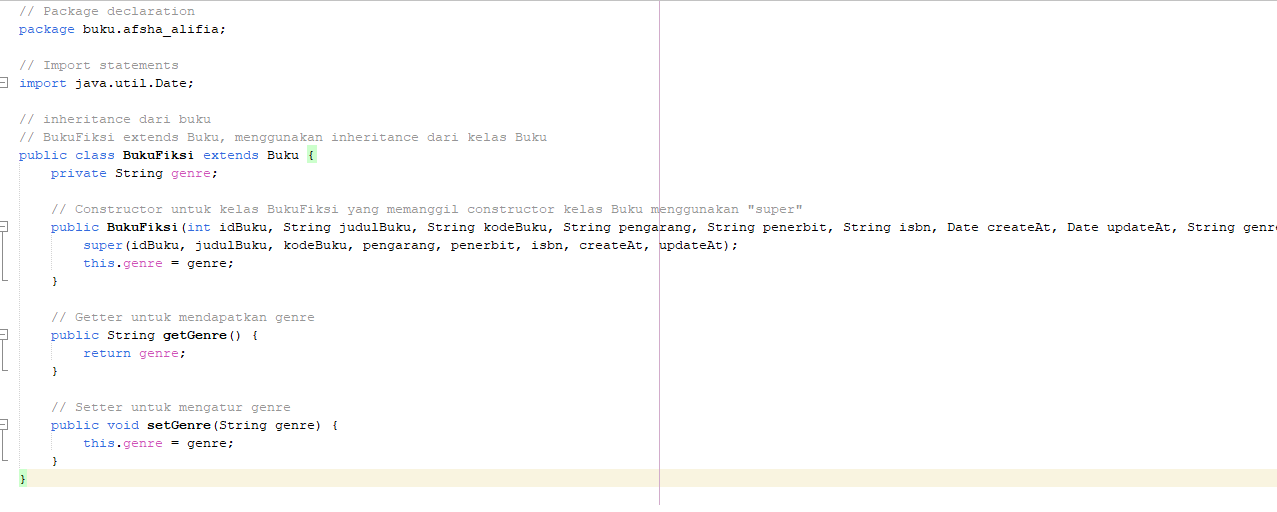
Gambar 1. 17 BukuDAOImpl 3



Gambar 1. 18 BukuDAOImpl 4

## **Buku Fiksi**

Sebuah class yang mendapatkan inheritance / pewarisan dari kelas buku.



Penjelasan :

* 1. Package Declaration:

- `package buku.afsha\_alifia;`: Mendefinisikan paket (package) dari kelas `BukuFiksi` dan menunjukkan lokasi tempat kelas ini berada dalam struktur proyek.

* 1. Import Statements:

- `import java.util.Date;`: Mengimpor kelas `Date` dari paket `java.util` untuk digunakan dalam deklarasi kelas.

* 1. Inheritance:

- `BukuFiksi extends Buku;`: Menunjukkan bahwa kelas `BukuFiksi` merupakan turunan dari kelas `Buku`, mengadopsi atribut dan metode dari kelas induknya.

* 1. Constructor:

- `public BukuFiksi(int idBuku, String judulBuku, String kodeBuku, String pengarang, String penerbit, String isbn, Date createAt, Date updateAt, String genre) { ... }`: Constructor kelas `BukuFiksi` yang memanggil constructor kelas `Buku` menggunakan kata kunci `super`, inisialisasi atribut kelas ini, serta atribut kelas induknya.

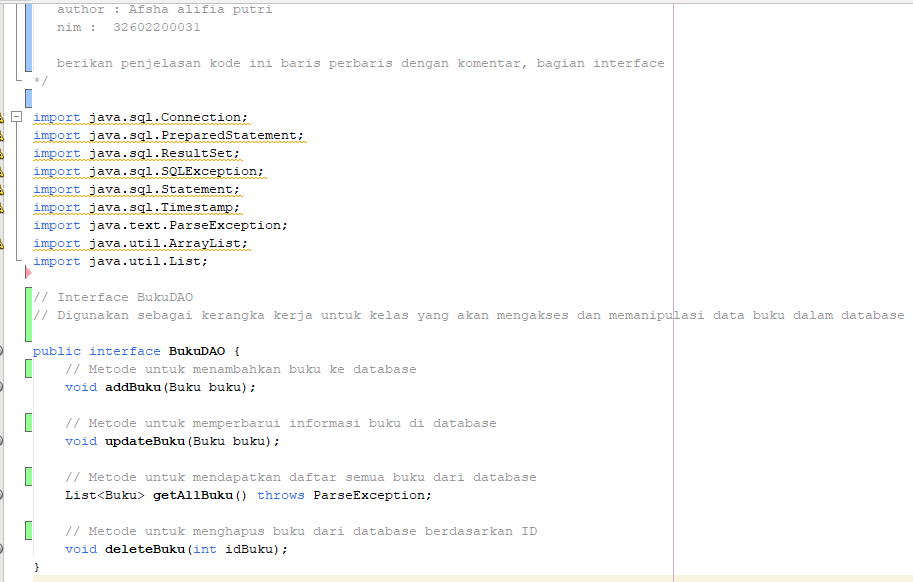
* 1. Getter and Setter:

- `public String getGenre() { return genre; }`: Getter untuk mendapatkan nilai dari atribut `genre`.

- `public void setGenre(String genre) { this.genre = genre; }`: Setter untuk mengatur nilai atribut `genre`.

## **Interface BukuDAO**

BukuDAO (Data Access Object) berfungsi sebagai perantara antara aplikasi dan database untuk operasi-operasi terkait buku..



Gambar 1. 19 BukuDAO Interface

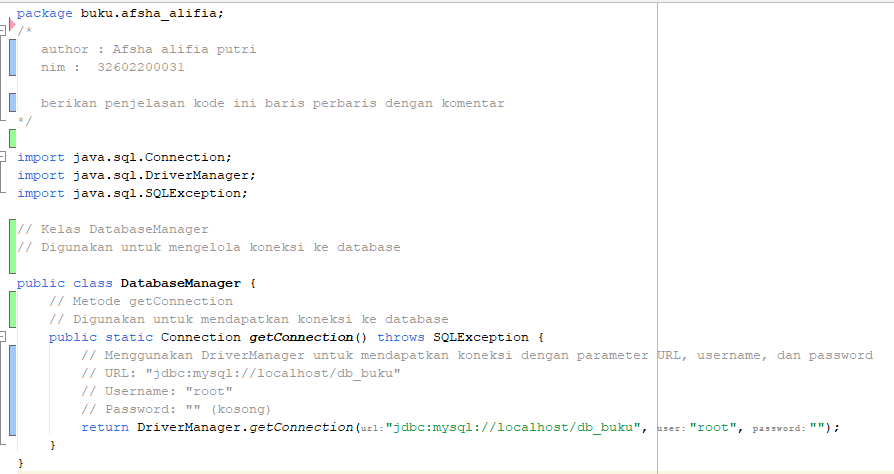
Penjelasan :

Interface:

1. Interface BukuDAO digunakan sebagai kontrak yang harus diikuti oleh kelas-kelas yang bertanggung jawab untuk berinteraksi dengan data buku di database.
2. Metode-metode di dalam interface ini mencakup operasi dasar seperti menambahkan, memperbarui, mengambil semua data, dan menghapus buku.
3. Interface memberikan kerangka kerja yang diperlukan untuk implementasi kelas-kelas DAO (Data Access Object) yang akan berkomunikasi dengan database. Implementasi dari interface ini dapat bervariasi tergantung pada jenis database atau penyimpanan data yang digunakan.

## **DbManager**

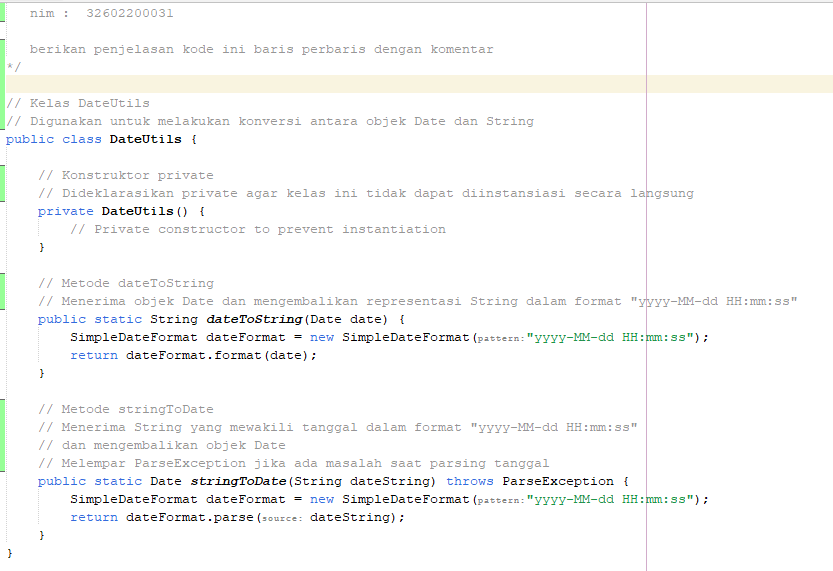
DbManager digunakan untuk mengelola koneksi ke database, sehingga aplikasi dapat berinteraksi dengan basis data..



Gambar 1. 20 DB Manager

## **DateUtils**

Date Utils menyediakan fungsi-fungsi untuk manipulasi tanggal seperti konversi dari Date ke String dan sebaliknya.



Gambar 1. 21 DateUtils

Penjelasan :

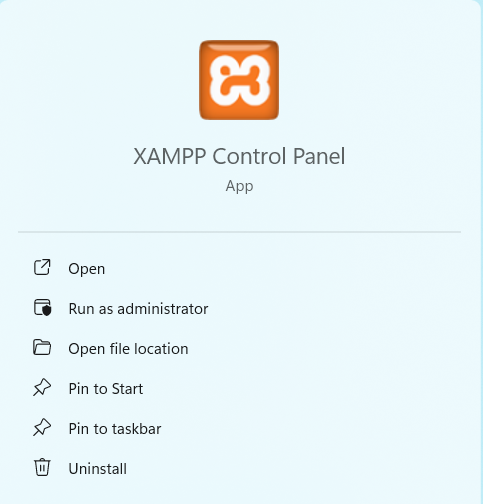
* 1. Utility Class:

1. Kelas DateUtils adalah kelas utilitas yang menyediakan metode-metode untuk konversi antara objek Date dan String dalam format tertentu.
2. Konstruktor dinyatakan sebagai private untuk mencegah instansiasi langsung dari kelas ini.

# BAB III Implementasi Program

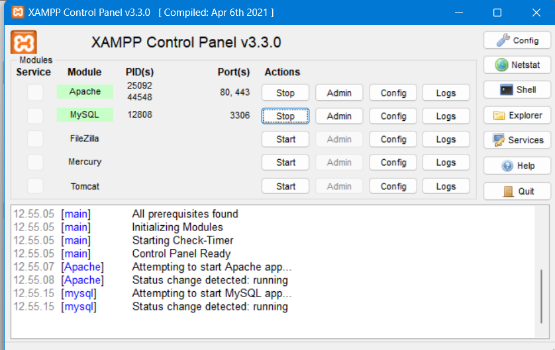
## **Menjalankan Program**

1. Buka Xampp



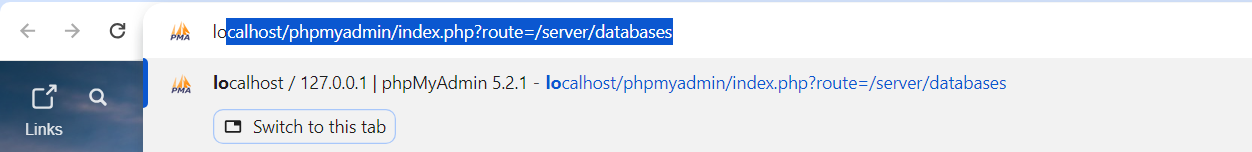
Gambar 1. 22 Open Xampp

1. Start apache & mysql, jika sudah berhasil sukses hijau



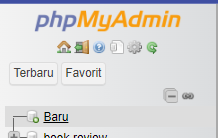
Gambar 1. 23 Sukses start xampp

1. Buka chrome, akses localhost



Gambar 1. 24 Akses localhost

1. Lalu klick baru, untuk buat db\_buku



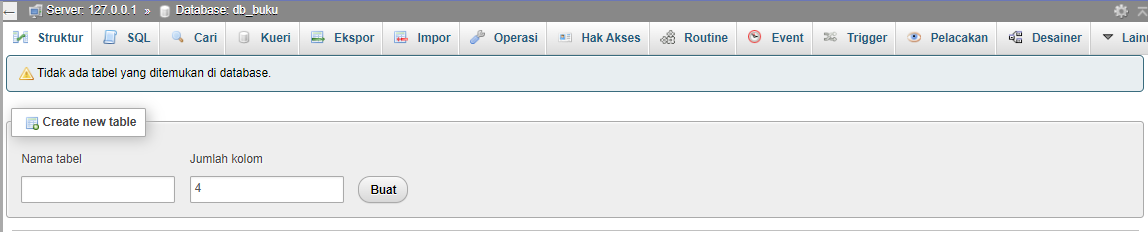
Gambar 1. 25 menuju ke buat db

1. Buat db baru dengan nama db\_buku, lalu klick buat



Gambar 1. 26 Buat db\_buku

1. klick impor table\_buku, jika berhasil masuk ke Langkah selanjutnya



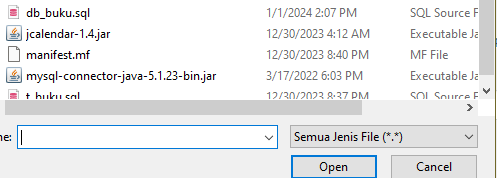
Gambar 1. 27 Menuju impor tabel

1. Lalu klick pilih file, untuk memilih



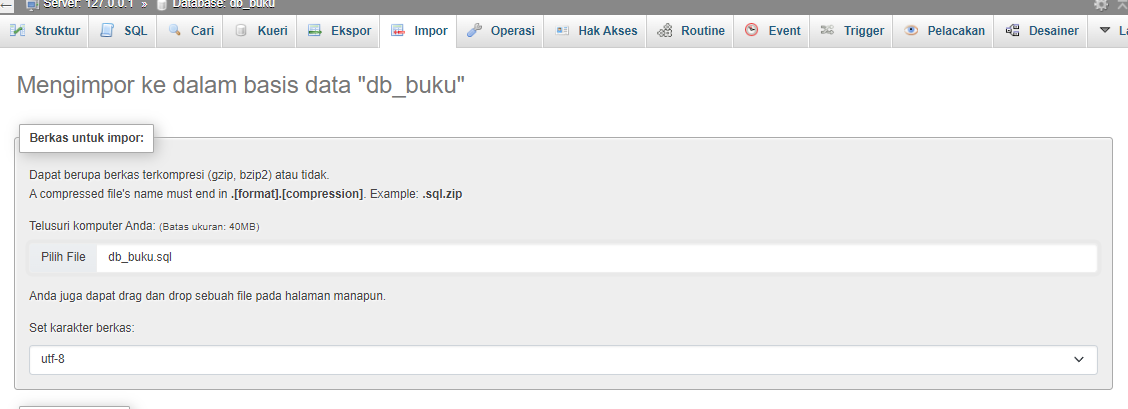
Gambar 1. 28 Klick file menuju direktori

1. Arahkan ke file db\_buku.sql, lalu klick Open,



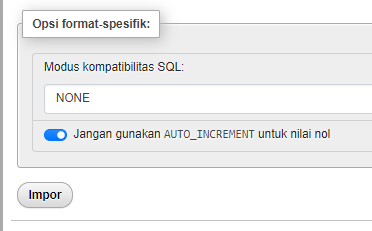
Gambar 1. 29 Pilih file, klick ope

1. Maka akan terpilih file db\_buku



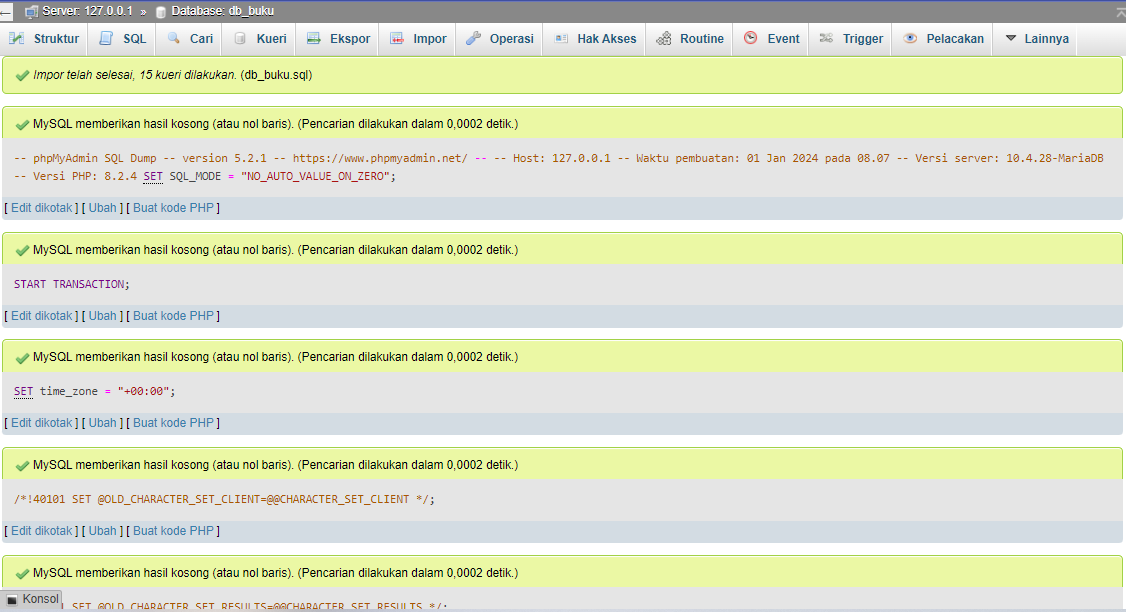
Gambar 1. 30 file terpilih

1. Setelah itu scroll kebawah untuk klick impor



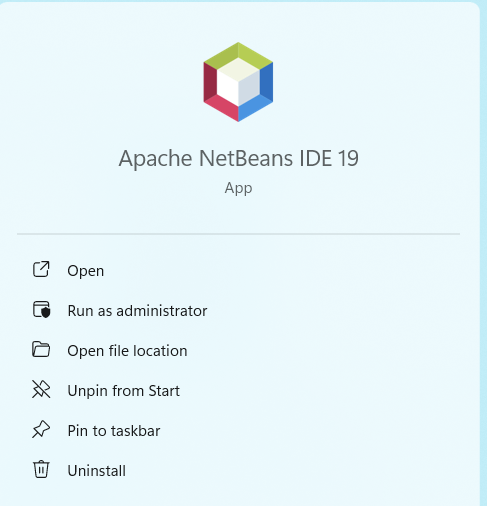
Gambar 1. 31 klick impor

1. Jika telah berhasil, maka import tabel\_buku dan pembuatan db buku berhasil sukses



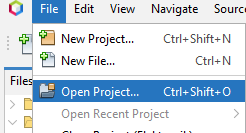
Gambar 1. 32 Sukses impor tabel dan buat db

1. Untuk menjalankan program klick neatbeans

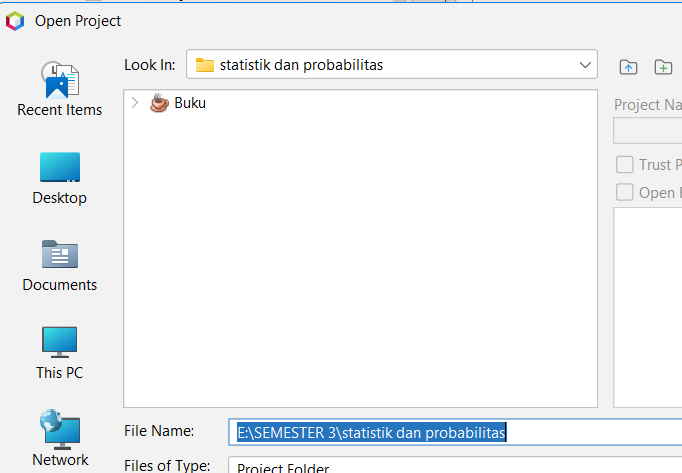


Gambar 2. 1 Open Neatbeans

1. Open new project, arahkan ke pencatatan buku

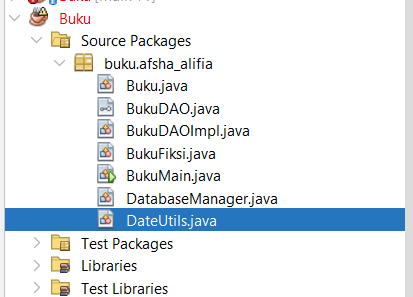


Gambar 2. 2 Open Projects



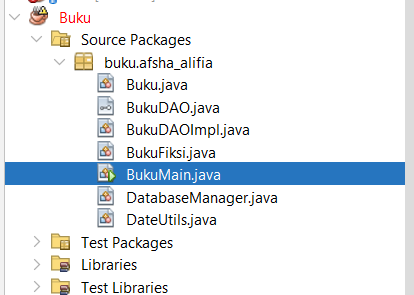
Gambar 2. 3 Arahkan ke projects

1. Buka src/buku/afsha\_alifia



Gambar 2. 4 Open struktur project

1. Klick file BukuMain.java, lalu klick kanan run file



Gambar 2. 5 Run program

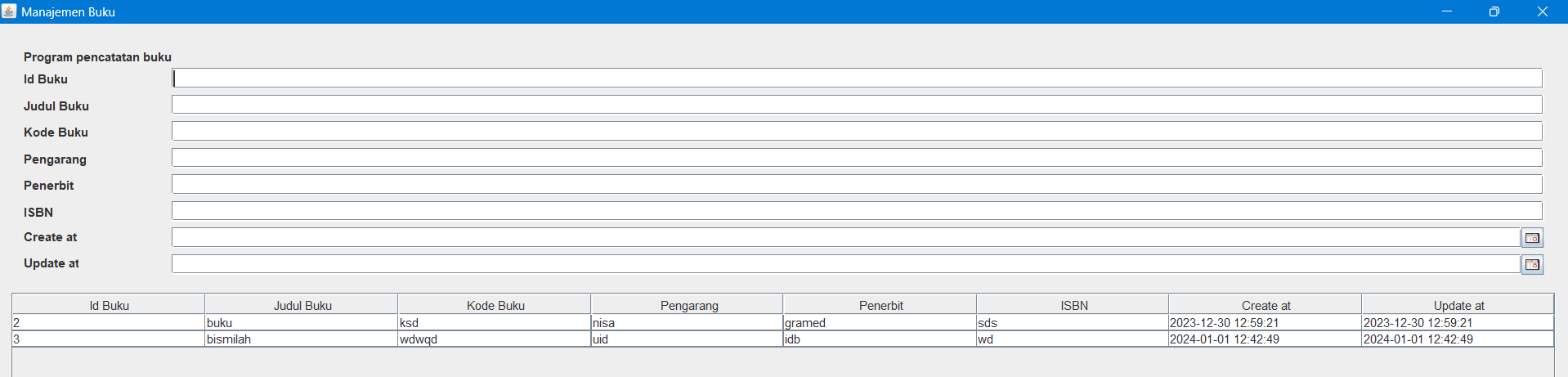
1. Tunggu program muncul



Gambar 2. 6 Tunggu program muncul

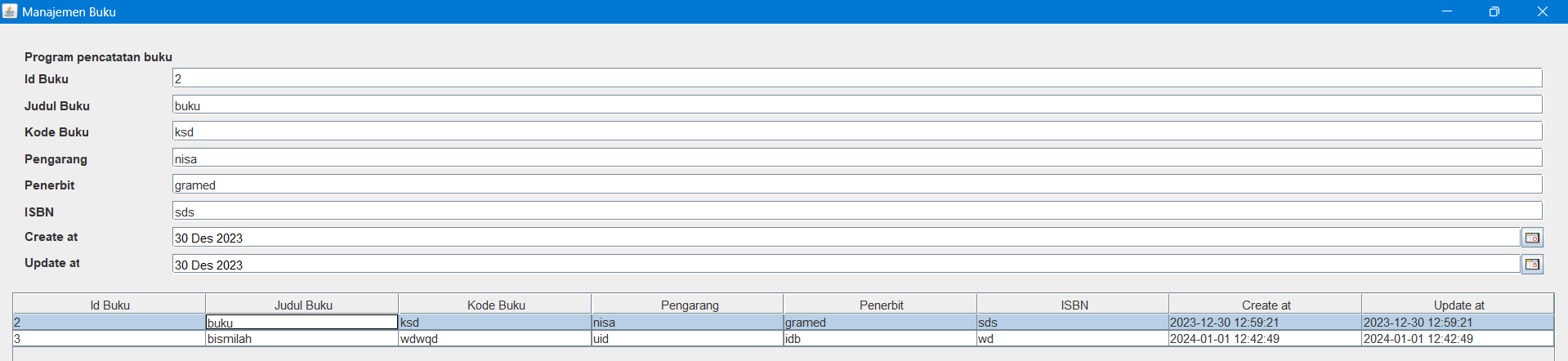
## **Menambahkan Buku**

1. Isi semua data, lalu klick tambah

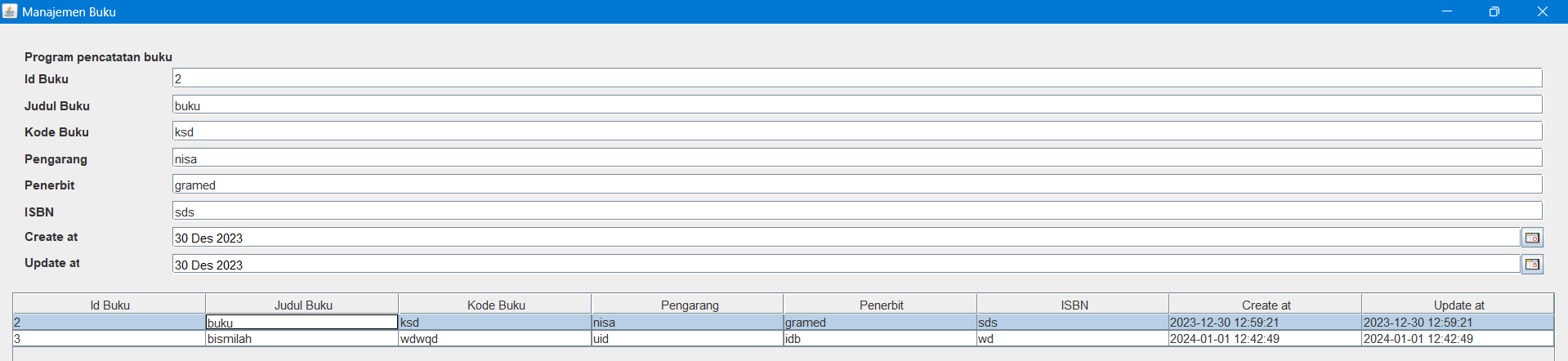


Gambar 2. 7 Tambah Buku

1. Jika sudah, klick salah satu item data dari tabel, coba edit di kolom text, lalu klick ubah

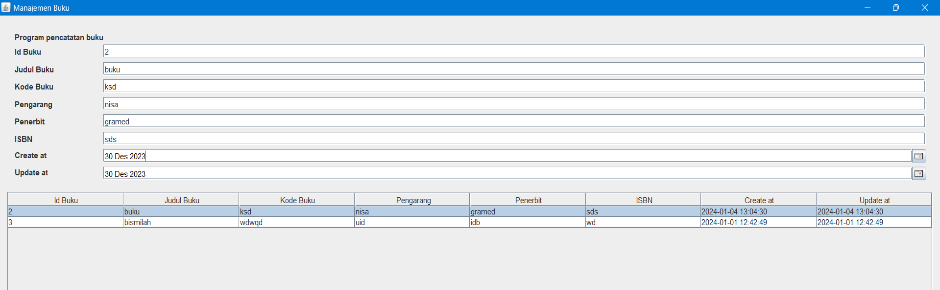


Gambar 2. 8 Buku sebelum dirubah



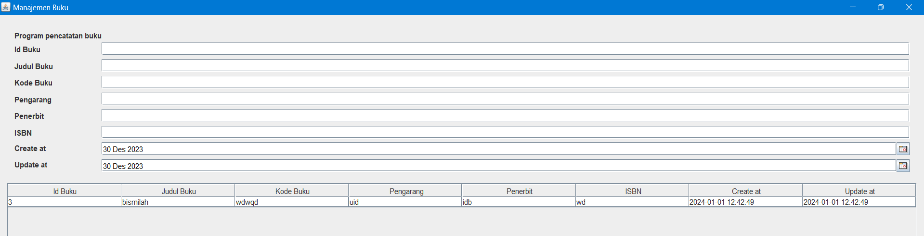
Gambar 1. 33 Buku yang akan dirubah

1. Buku di tabel akan berubah setelah diklick ubah



Gambar 1. 34 Buku berhasil dirubah

1. Jika ingin menghapus item buku, klick hapus maka buku akan terhapus dari tabel



Gambar 2. 9 Buku berhasil dihapus