LAPORAN PEMOGRAMAN BERORIENTASI OBJEK

****PEKAN 2

OLEH :

RAFI HAIKAL PRATAMA

NIM : 2411532002

MATA KULIAH PEMOGRAMAN BERORIENTASI OBJEK

DOSEN PENGAMPU : NURFIAH, S.ST, M.KOM

FAKULTAS TEKNOLOGI INFORMASI DEPARTEMEN INFORMATIKA

UNIVERSITAS ANDALAS

PADANG 2025

1. Pendahuluan

Pemrograman Berorientasi Objek (PBO) merupakan sebuah paradigma dalam pembuatan program komputer yang menggunakan kelas dan objek sebagai komponen utamanya untuk membangun sebuah perangkat lunak. Metode ini hadir sebagai penyempurnaan dari cara pemrograman lama, dengan ide utama untuk menempatkan informasi beserta tindakan yang relevan ke dalam satu lokasi terpusat, sehingga hanya tindakan tertentu yang dapat mengaksesnya. Dengan pendekatan ini, OOP secara efektif memastikan data menjadi lebih aman dan terorganisir, sekaligus membantu programmer untuk menciptakan program yang terpisah dan terstruktur dengan baik. Seiring waktu, OOP telah berkembang menjadi salah satu metode paling umum dalam pemrograman modern, terutama pada bahasa Java yang mendukung penuh ide-ide krusial seperti penjagaan privasi data, pewarisan, dan penggunaan kembali kode. Berbagai manfaat tersebut membuat proses pengembangan perangkat lunak menjadi lebih mudah untuk ditangani, diperbarui, dan diubah sesuai kebutuhan, sehingga pemahaman terhadap dasar-dasar OOP menjadi sangat penting untuk membangun program yang mengikuti aturan pengembangan perangkat lunak modern.

1. Tujuan

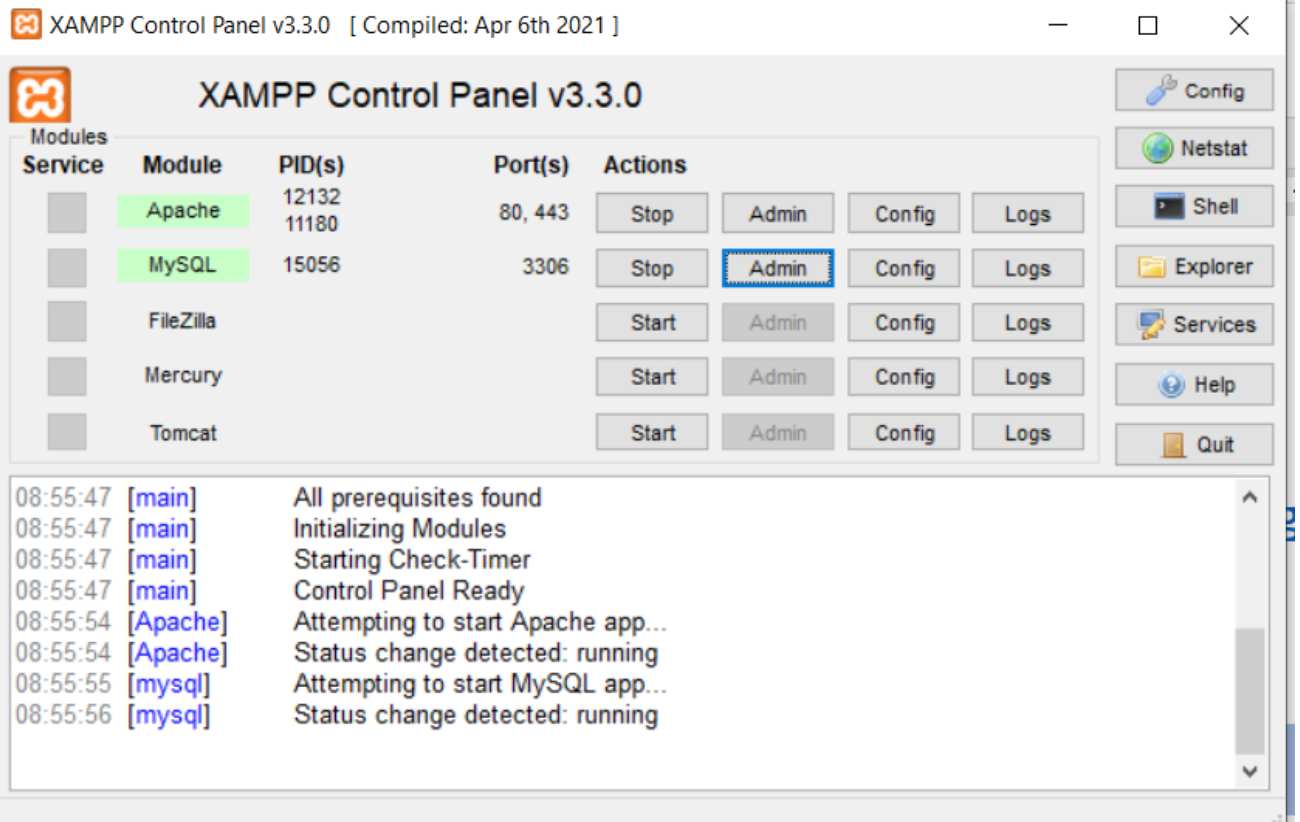
**1. Koneksi Database:** Mahasiswa mampu membuat tabel user pada database MySQL dan berhasil menghubungkannya ke dalam aplikasi Java.

**2. Desain Antarmuka (GUI):** Mahasiswa mampu merancang dan membangun tampilan antarmuka pengguna grafis (GUI) untuk memfasilitasi operasi data user, yang mencakup proses **Create, Read, Update, dan Delete (CRUD)**, dengan menerapkan konsep Pemrograman Berorientasi Objek.

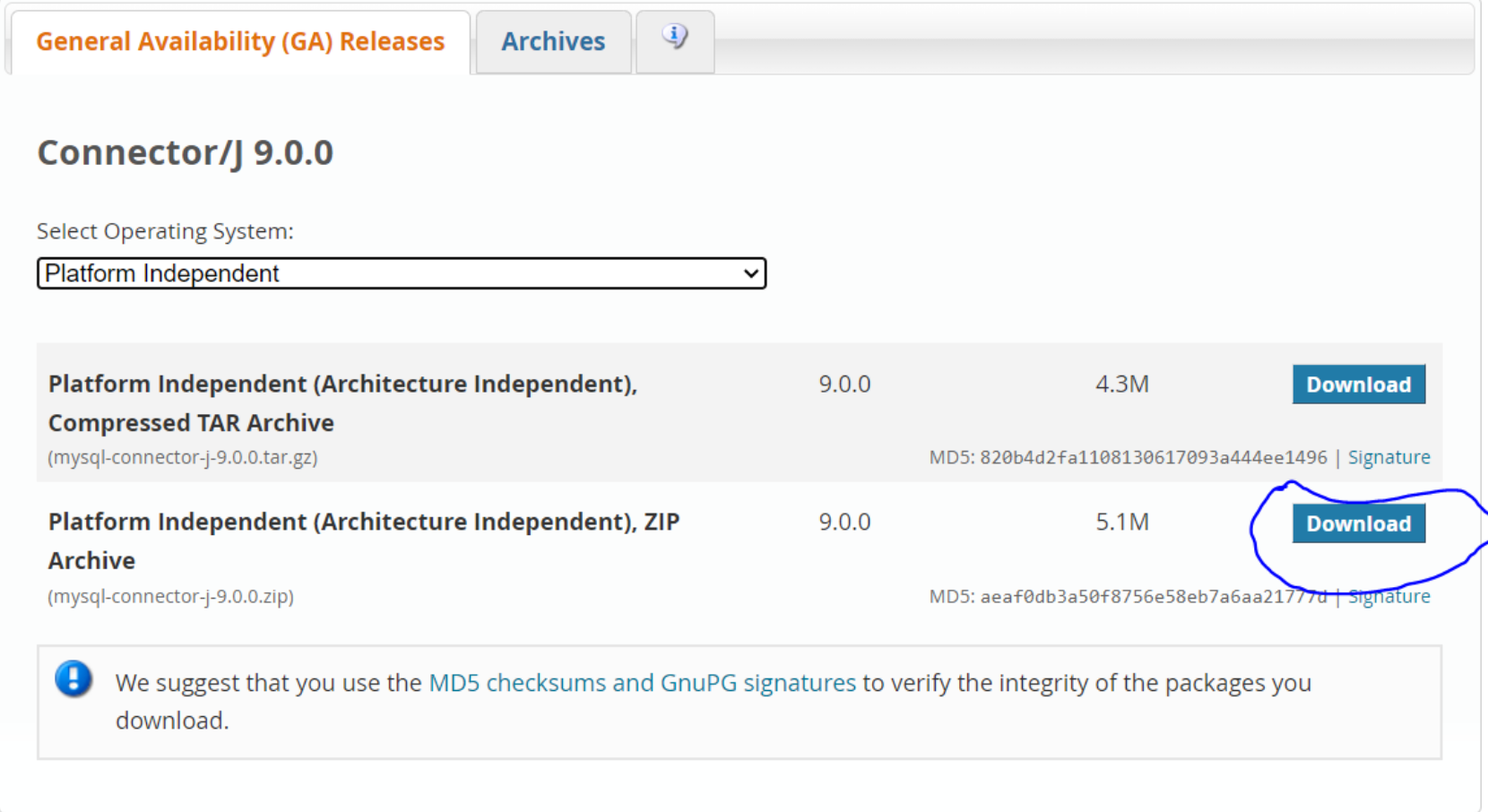
**3. Implementasi Pola DAO:** Mahasiswa mampu membuat serta mengimplementasikan *interface* dan fungsi dari *Data Access Object* (DAO) untuk mengelola logika akses data secara terstruktur, terpisah dari logika antarmuka, dan modular.

1. Langkah Pengerjaan
2. Instal Xampp pada link berikut : <https://www.apachefriends.org/>

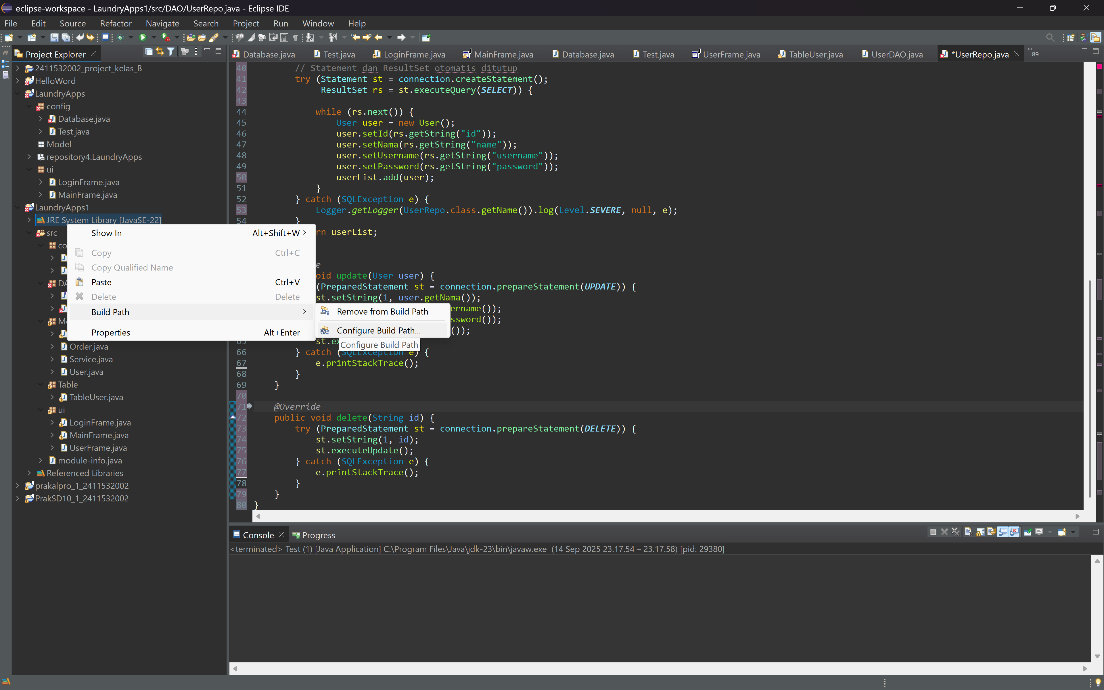
Lalu jalankan apache dan mysql

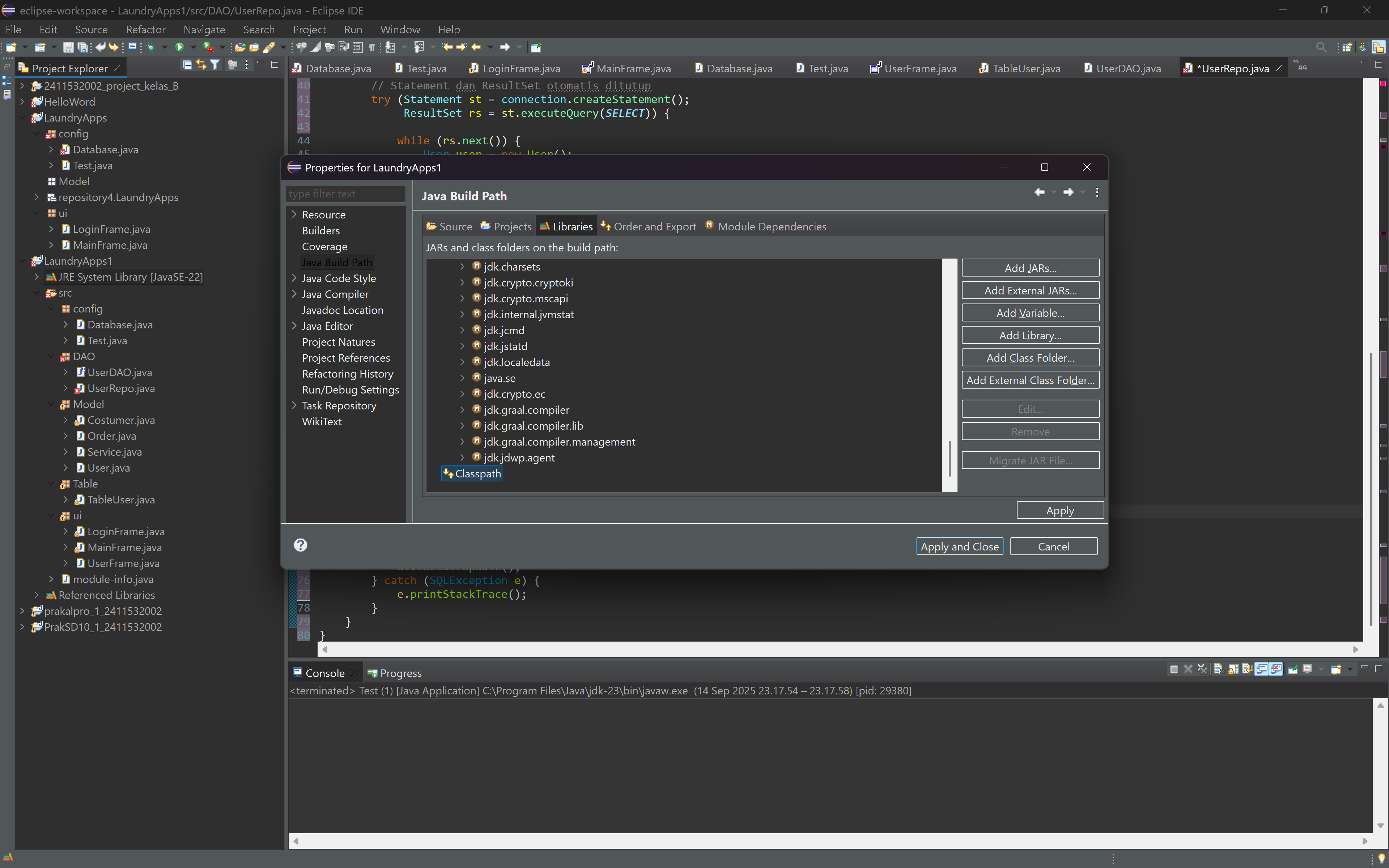


1. Download mysql connection pada link berikut : <https://dev.mysql.com/downloads/connector/j/>

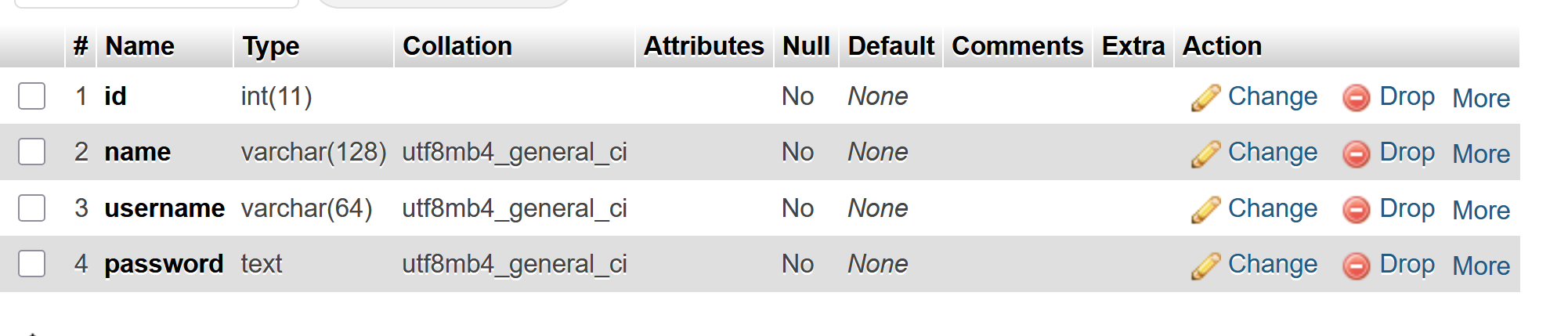


1. Tambahkan mysql connector kedalam project





1. Membuat Database dan tabel user



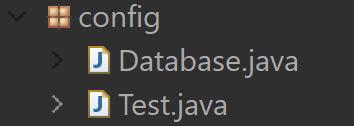
Ini adalah hasilnya

1. Membuat koneksi ke databese mysql

Ada beberapa langkah yang harus dilakukan yaitu kita buat package config



Lalu buat class baru dengan nama Database



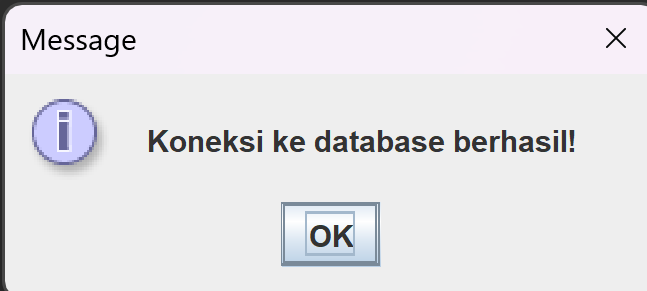
Ini adalah kode programnya



Untuk mengetes apakah sudah connect dengan database saya disini membuat class baru dengan nama test

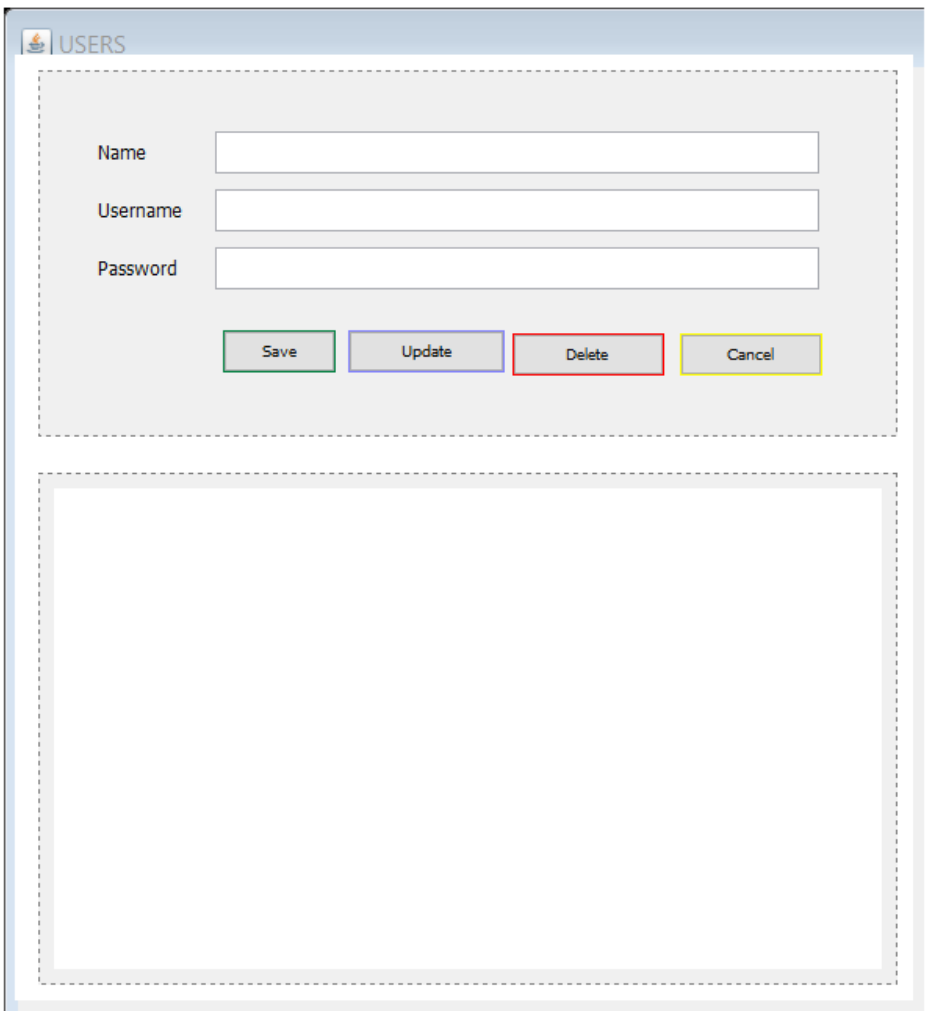


Ini gambaran apabila bisa connecct ke database



1. Membuat tampilan Crud User

Langkah awal kita perlu membuat Jframe pada package dengan nama UserFrame

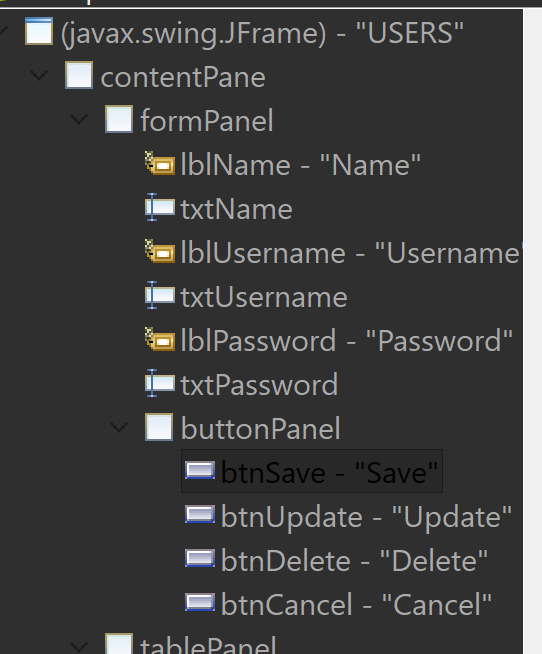


Dengan keterangan sebagai berikut

Atur nama JFrame menjadi "USERS" untuk judul pada frame. Di dalam JFrame UserFrame, tambahkan sebuah panel utama dengan nama contentPane sebagai wadah semua komponen.

Tambahkan panel bernama formPanel ke dalam contentPane. Panel ini berfungsi sebagai tempat input data user. Komponen-komponen yang ditambahkan di formPanel antara lain:

* JLabel lblName dan JTextField txtName untuk menginput nama.
* JLabel lblUsername dan JTextField txtUsername untuk menginput username.
* JLabel lblPassword dan JTextField txtPassword untuk menginput password.



Selanjutnya tambahkan panel bernama buttonPanel di bawah formPanel. Panel ini berisi 4 buah tombol (JButton) sebagai aksi utama:

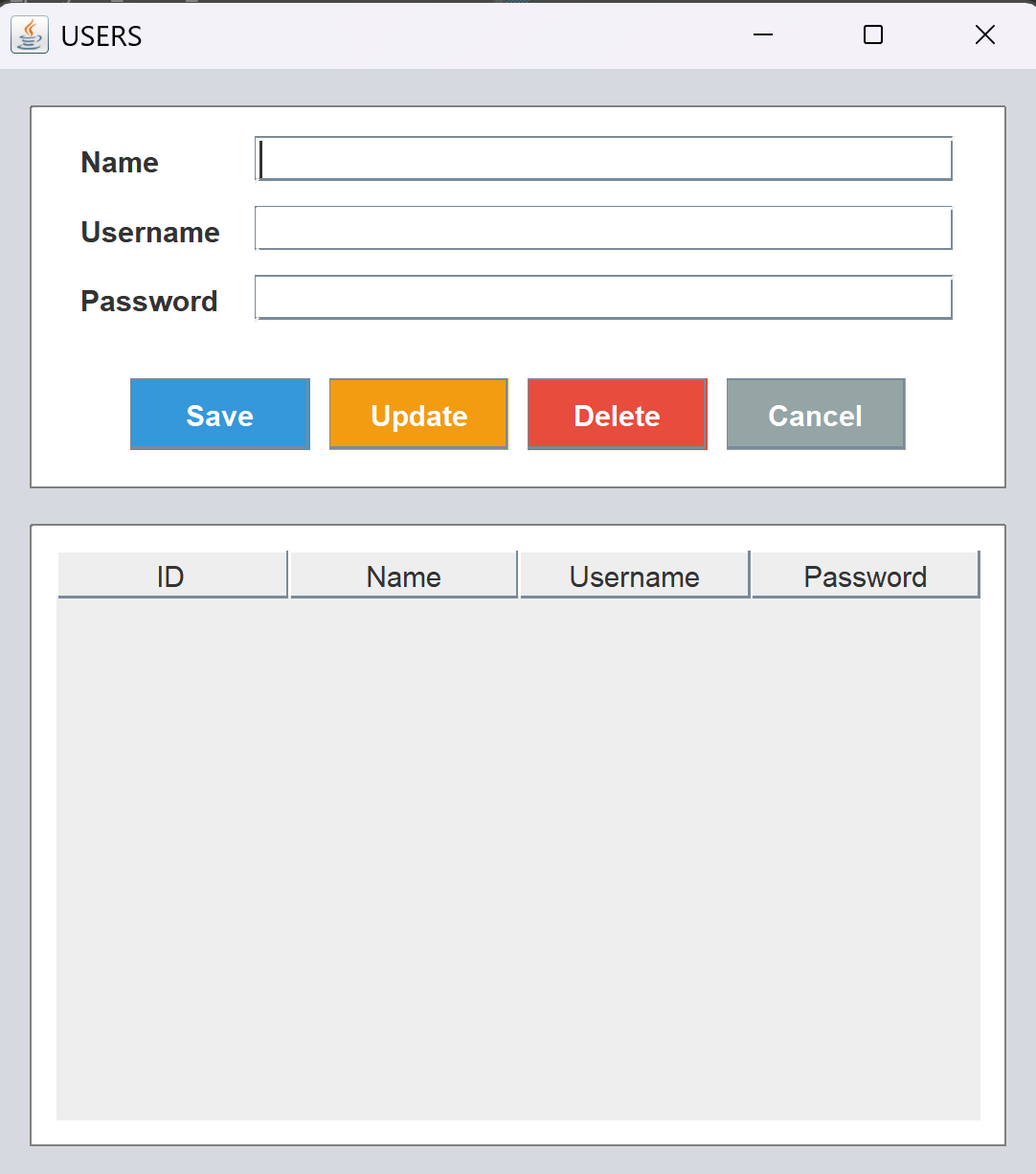
* btnSave untuk menyimpan data baru ke database.
* btnUpdate untuk memperbarui data yang dipilih dari tabel.
* btnDelete untuk menghapus data yang dipilih dari tabel.
* btnCancel untuk membatalkan input atau mengosongkan field.

Selanjutnya masing-masing tombol diatur warnanya untuk membedakan fungsi secara visual. Sebagai contoh warna yang saya gunakan:

* Hijau → Save (tambah data)
* Magenta → Update (ubah data)
* Merah → Delete (hapus data)
* Kuning → Cancel (reset/batal)

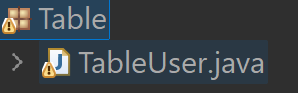
Tambahkan panel bernama tablePanel di bagian bawah contentPane. Panel ini berisi sebuah JTable bernama tableUser untuk menampilkan data user dalam bentuk tabel

Hasil akhir akan seperti berikut

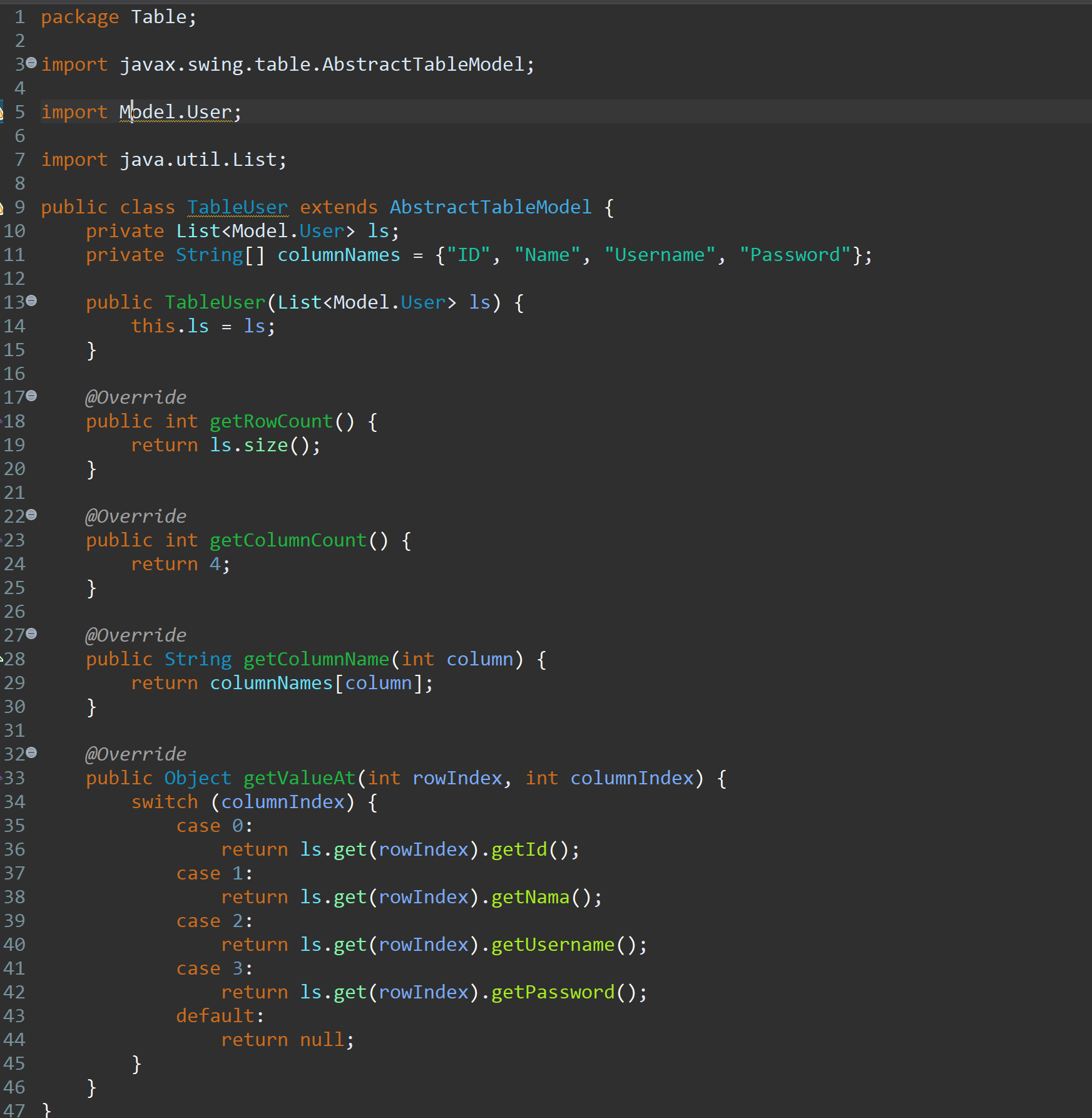


1. Membuat table model

Buat package baru dengan judul Table



Lalu buat class dengan judul tabel user

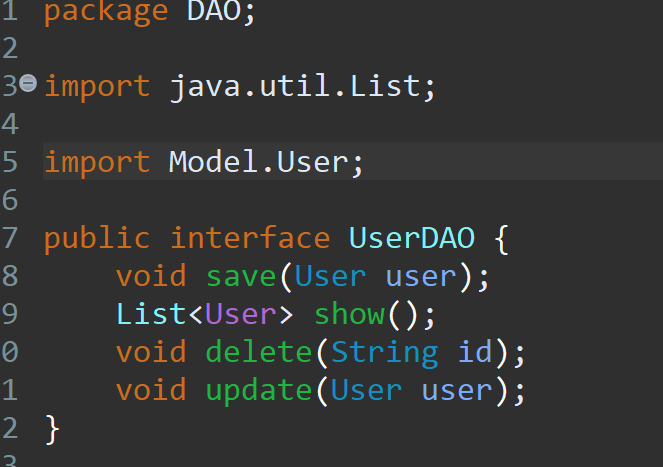


1. Membuat fungsi DAO

Buat package baru dengan nama DAO



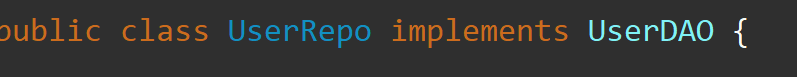
Lalu buat class baru dengan nama user dao



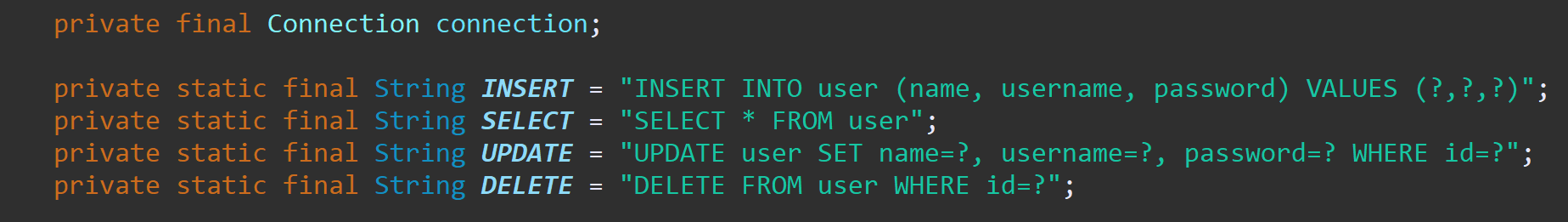
1. Menggunakan Fungsi DAO

Buat Class baru dengan nama UserRepo

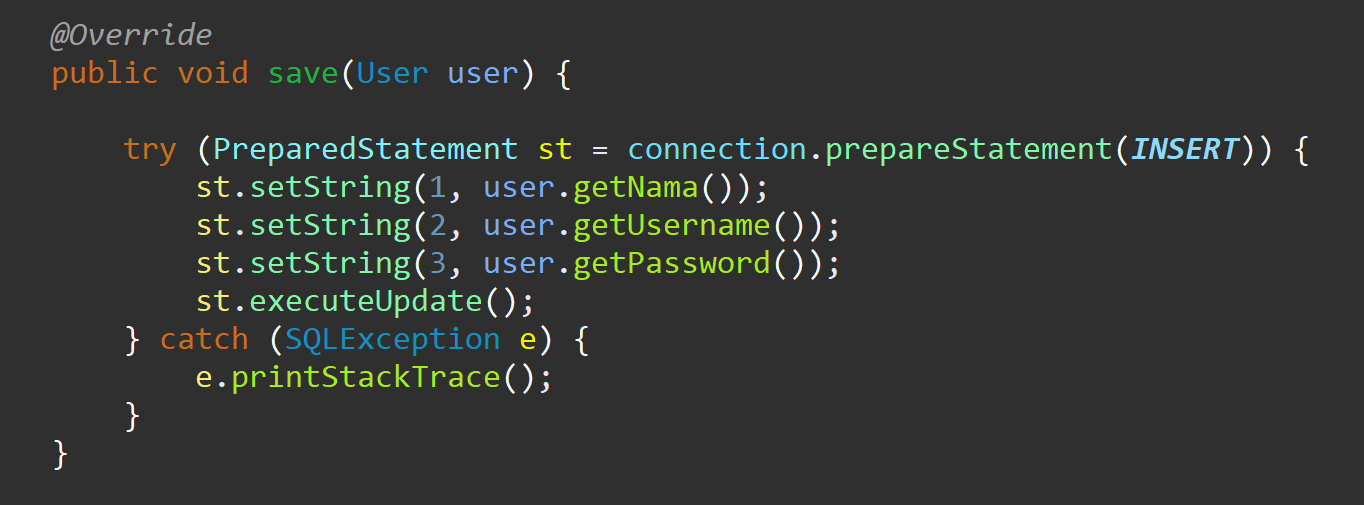
* Lalu implementasikan user dao dengan kata kunci implements



* Selanjutnya istanisasi connction, membuat construktor dan membuat string untuk melakukan manipulasi database

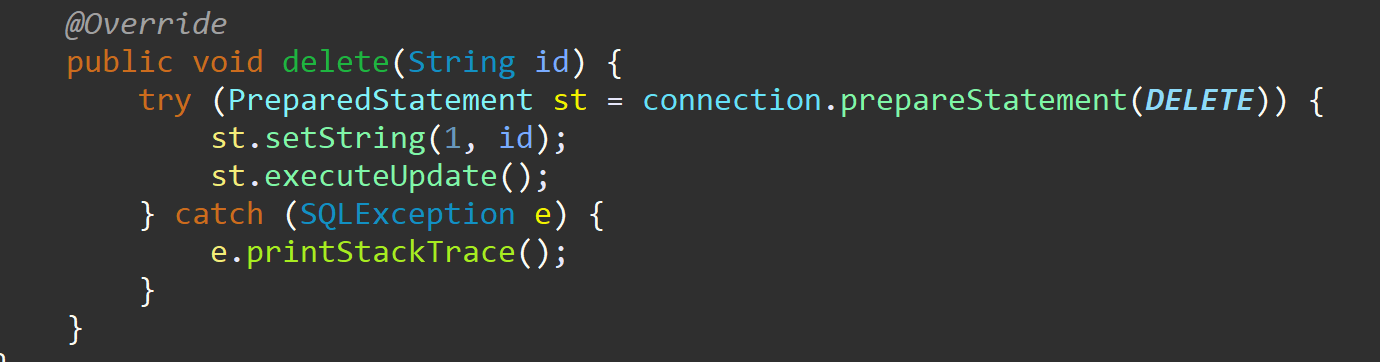


* Selanjutnya kita akan membuat method save, mehod show, method update, dan method delete.









1. Kesimpulan

Praktikum ini telah membekali saya dengan pemahaman mendalam serta kemampuan praktis untuk membangun sistem **CRUD (Create, Read, Update, Delete)** pada data user menggunakan bahasa pemrograman Java yang terintegrasi dengan database MySQL. Saya mendapatkan pengalaman langsung dalam mempersiapkan infrastruktur database, mulai dari membuat tabel user di MySQL hingga membangun koneksi yang fungsional dengan aplikasi Java. Selain itu, keterampilan saya dalam merancang antarmuka pengguna (GUI) juga berkembang, di mana saya berhasil menerapkan konsep **Pemrograman Berorientasi Objek (OOP)** untuk mendukung proses CRUD. Pembelajaran krusial lainnya adalah implementasi *interface* dan fungsi **DAO (Data Access Object)**, yang saya terapkan untuk menciptakan arsitektur sistem yang modular dan terstruktur. Pendekatan ini terbukti efektif dalam mengelola proses CRUD, menghasilkan sistem yang tidak hanya efisien tetapi juga mudah untuk dikelola dan dikembangkan di masa depan.