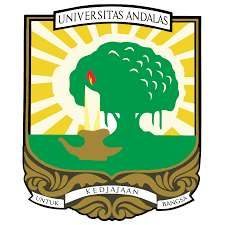
**LAPORAN PRAKTIKUM 2 PBO**

**MEMBUAT FUNGSI CRUD (CREATE, READ, UPDATE, DELETE) USER**

**DENGAN DATABASE MYSQL**

****

**Disusun Oleh:**

**INDAH SYAHFITRI**

**2311532016**

**Dosen Pembimbing: Nurfiah, S. ST., M. Kom.**

**DEPARTEMEN INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS ANDALAS**

1. **PENDAHULUAN**

XAMPP yaitu paket software yang terdiri dari Apache HTTP Server, MySQL,PHP dan Perl yang bersifat open source, XAMPP biasanya digunakan sebagai *development environment* dalam pengembangan aplikasi berbasis web secara *localhost*. Apache berfungsi sebagai web server yang digunakan untuk menjalankan halaman web, MySQL digunakan untuk manajemen basis data dalam melakukan manipulasi data, PHP digunakan sebagai bahasa pemrograman untuk membuat aplikasi berbasis web.

MySQL adalah sebuah *relational database management system* (RDBMS) *open-source* yang digunakan dalam pengelolaan database suatu aplikasi, MySQL ini dapat digunakan untuk menyimpan, mengelola, dan mengambil data dalam format table.

MySQL Connection/j adalah driver yang digunakan untuk menghubungkan aplikasi berbasis java dengan database MySQL sehingga dapat berinteraksi seperti menyimpan, mengubah, mengambil dan menghapus data. Beberapa fungsi MySQL connector yaitu:

• Membuka koneksi ke database MySQL.

• Mengirimkan permintaan SQL ke server MySQL.

• Menerima hasil dari permintaan SQL.

• Menutup koneksi ke database MySQL.

DAO (Data Access Object) merupakan object yang menyediakan abstract interface terhadap beberapa method yang berhubungan dengan database seperti mengambil data (read), menyimpan data (create), menghapus data (delete), mengubah data(update). Tujuan penggunaan DAO yaitu:

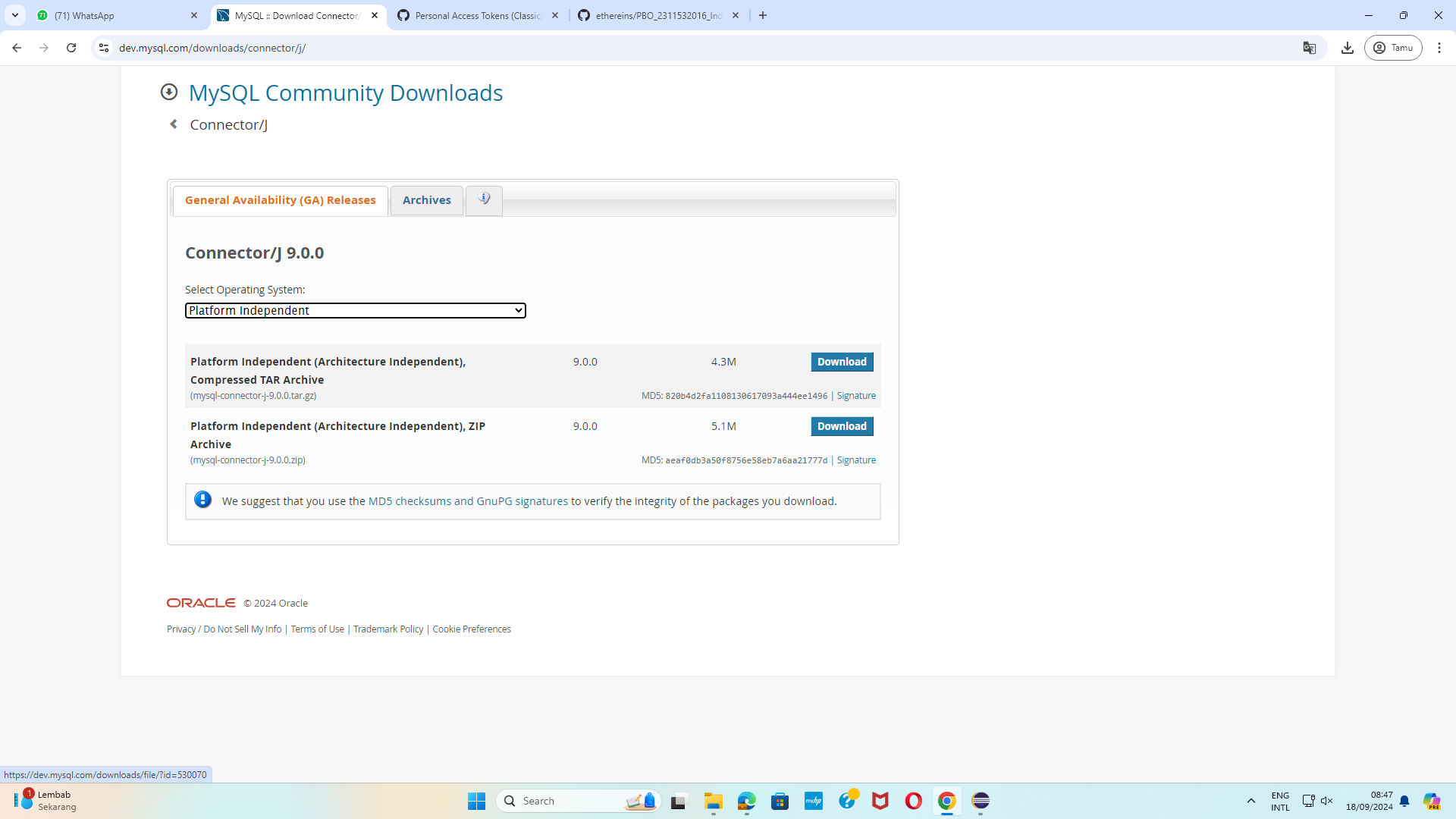
• Meningkatkan modularitas yaitu memisahkan logika akses data dengan logika bisnis sehingga memudahkan untuk dikelola.

• Meningkatkan reusabilitas yaitu DAO dapat digunakan kembali.

• Perubahan pada logika akses data dapat dilakukan tanpa mempengaruhi logika bisnis.

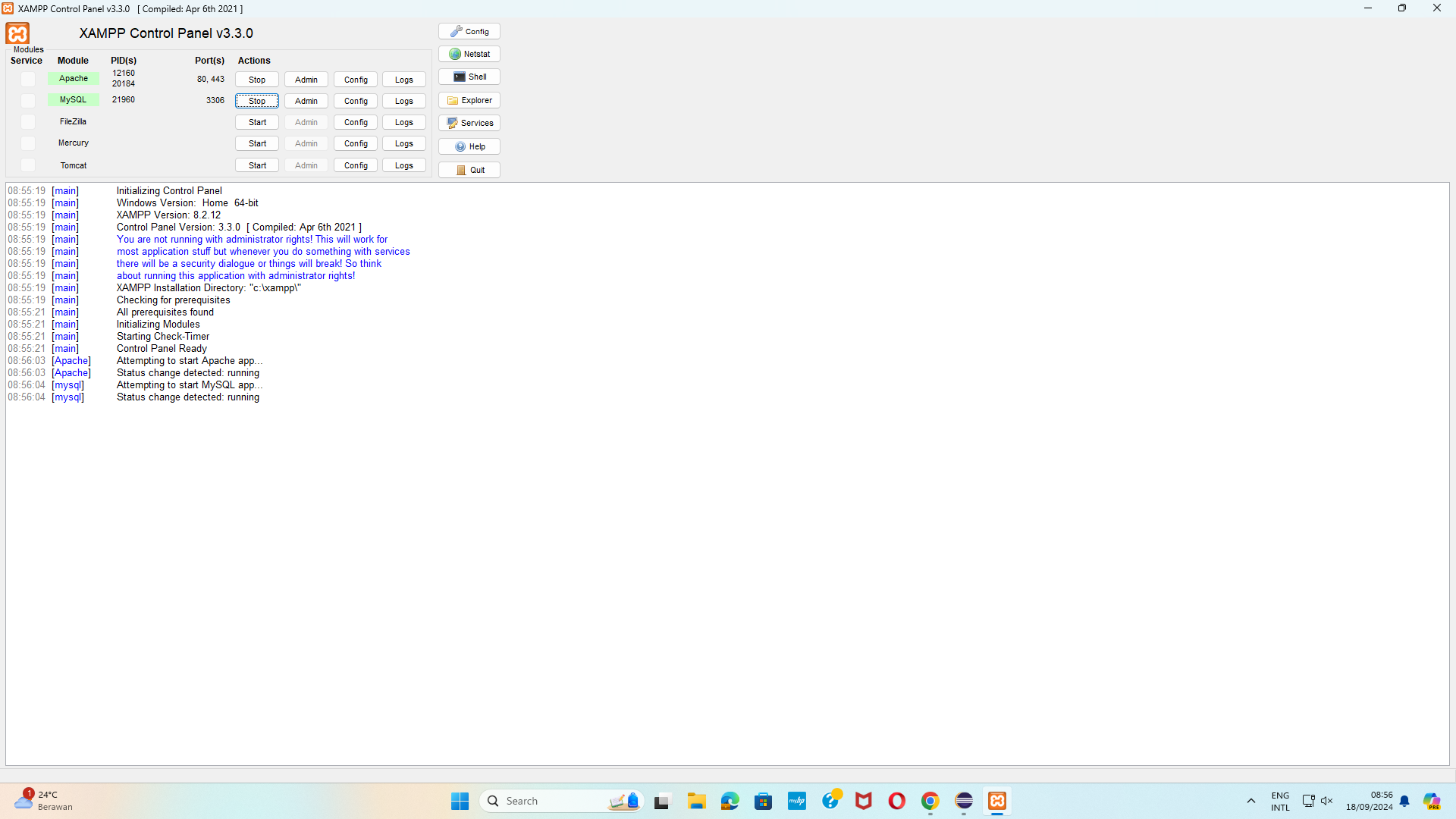
Interface dalam Bahasa java yaitu mendefinisikan beberapa method abstrak yang harus diimplementasikan oleh class yang akan menggunakannya. Sedangkan CRUD (Create, Read, Update, Delete) merupakan fungsi dasar atau umum yang ada pada sebuah aplikasi yang mana fungsi ini dapat membuat, membaca, mengubah dan menghapus suatu data pada database aplikasi.

1. **TUJUAN PRAKTIKUM**
2. Mahasiswa mampu membuat tabel user pada database MySQL.
3. Mahasiswa mampu membuat koneksi Java dengan database MySQL.
4. Mahasiswa mampu membuat tampilan GUI CRUD user.
5. Mahasiswa mampu membuat dan mengimplementasikan interface.
6. Mahasiswa mampu membuat fungsi DAO (Data Access Object) dan mengimplementasikannya.
7. Mahasiswa mampu membuat fungsi CRUD dengan menggunakan konsep Pemrograman Berorientasi Objek.
8. **LANGKAH-LANGKAH**
9. Aplikasi java agar dapat terhubung dengan database MySQL membutuhkan sebuah driver yaitu MySQL Connection. Untuk menghubungkan program dengan database, kita bisa mengunduh MySQL terlebih dahulu di link berikut: <https://dev.mysql.com/downloads/connector/j/>

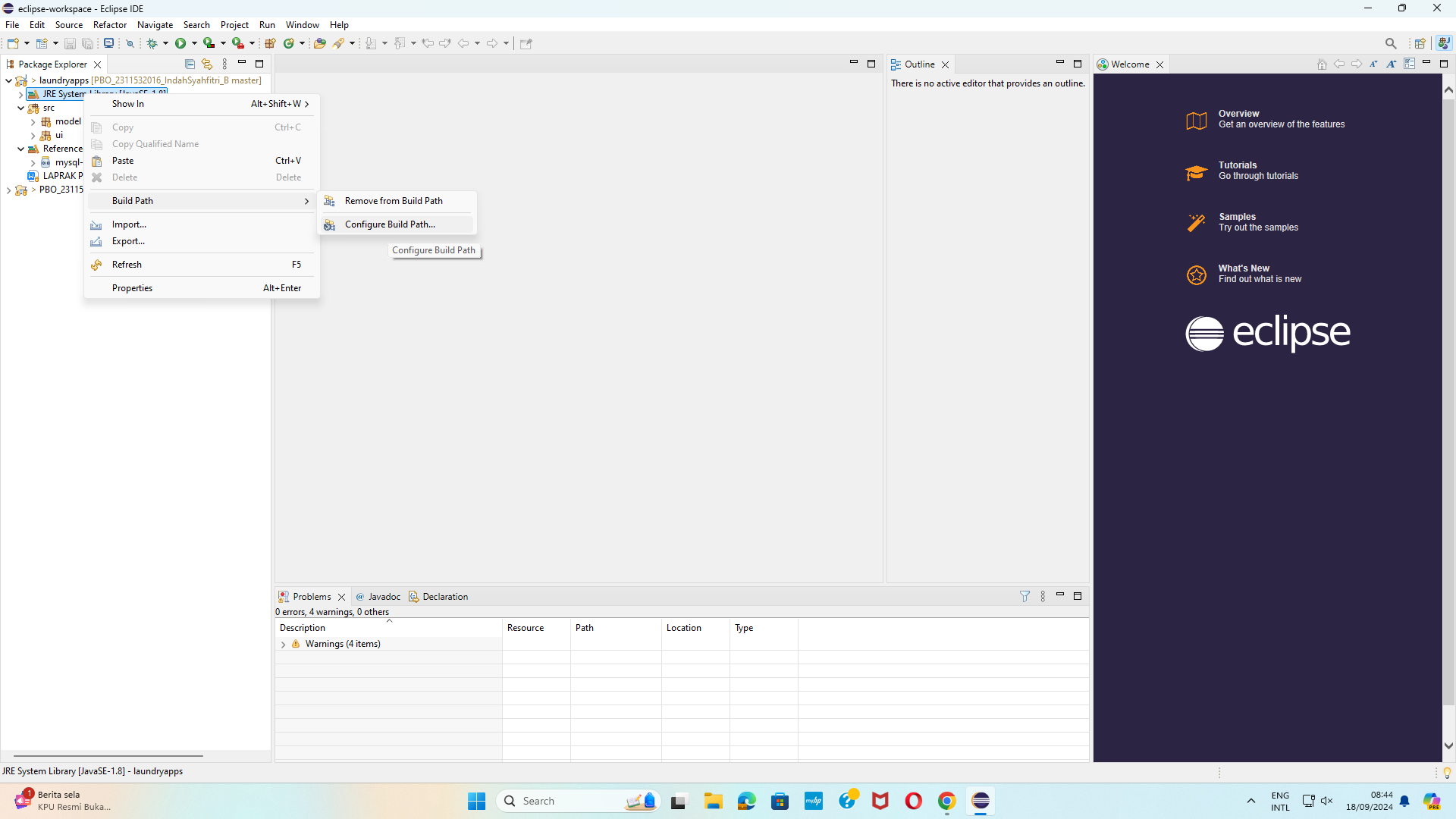


Pilih yang berekstensi zip.

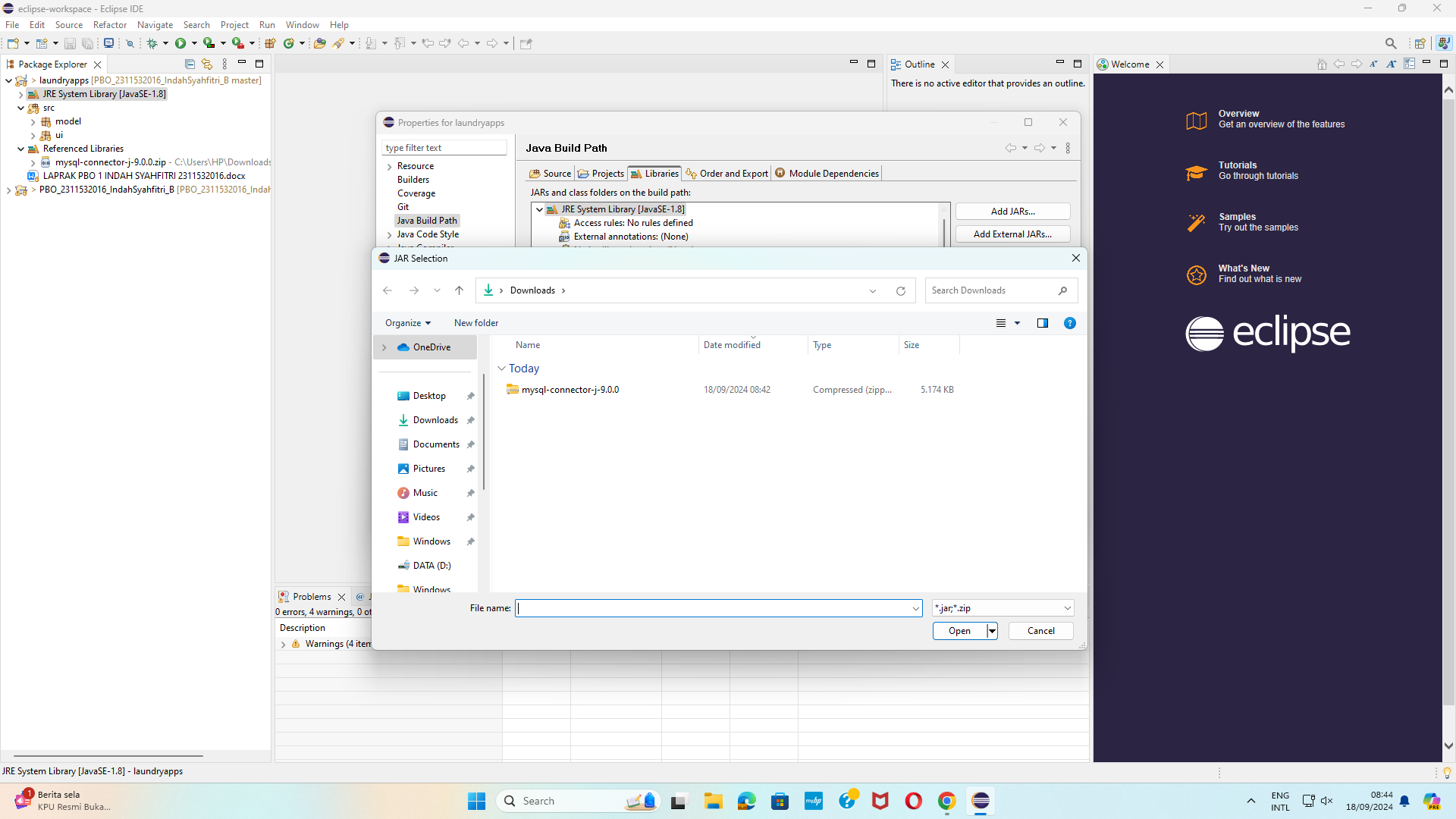
1. Setelah terinstall, buka aplikasi MySQL lalu tekan start pada Apache dan MySQL agar terkoneksi dengan program java yang akan dibuat.

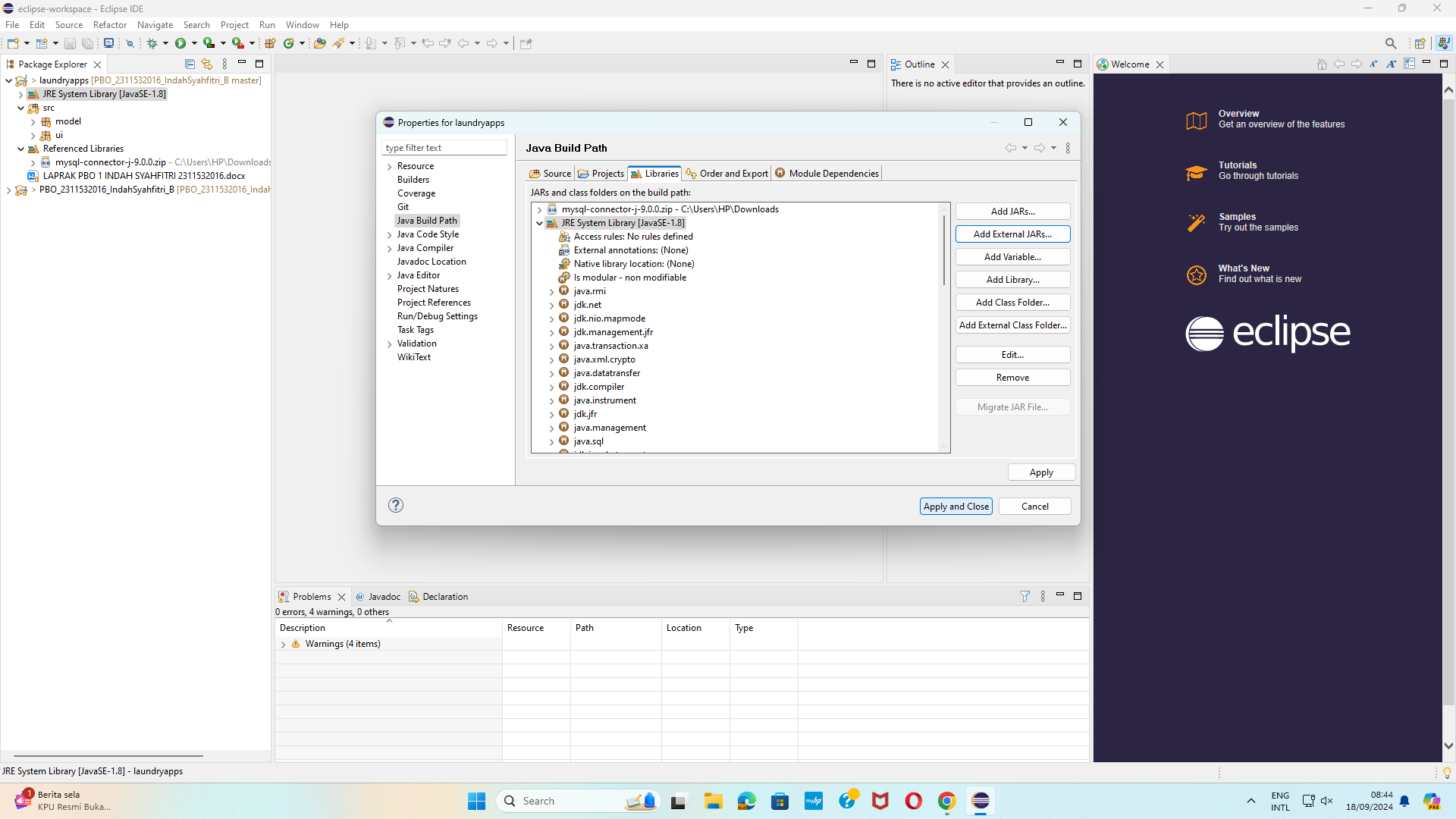


1. Tambahkan MySQL Connector kedalam project dengan cara klik kanan directory JRE. Klik kanan, pilih System Library → Built Path → Configure Build Path.

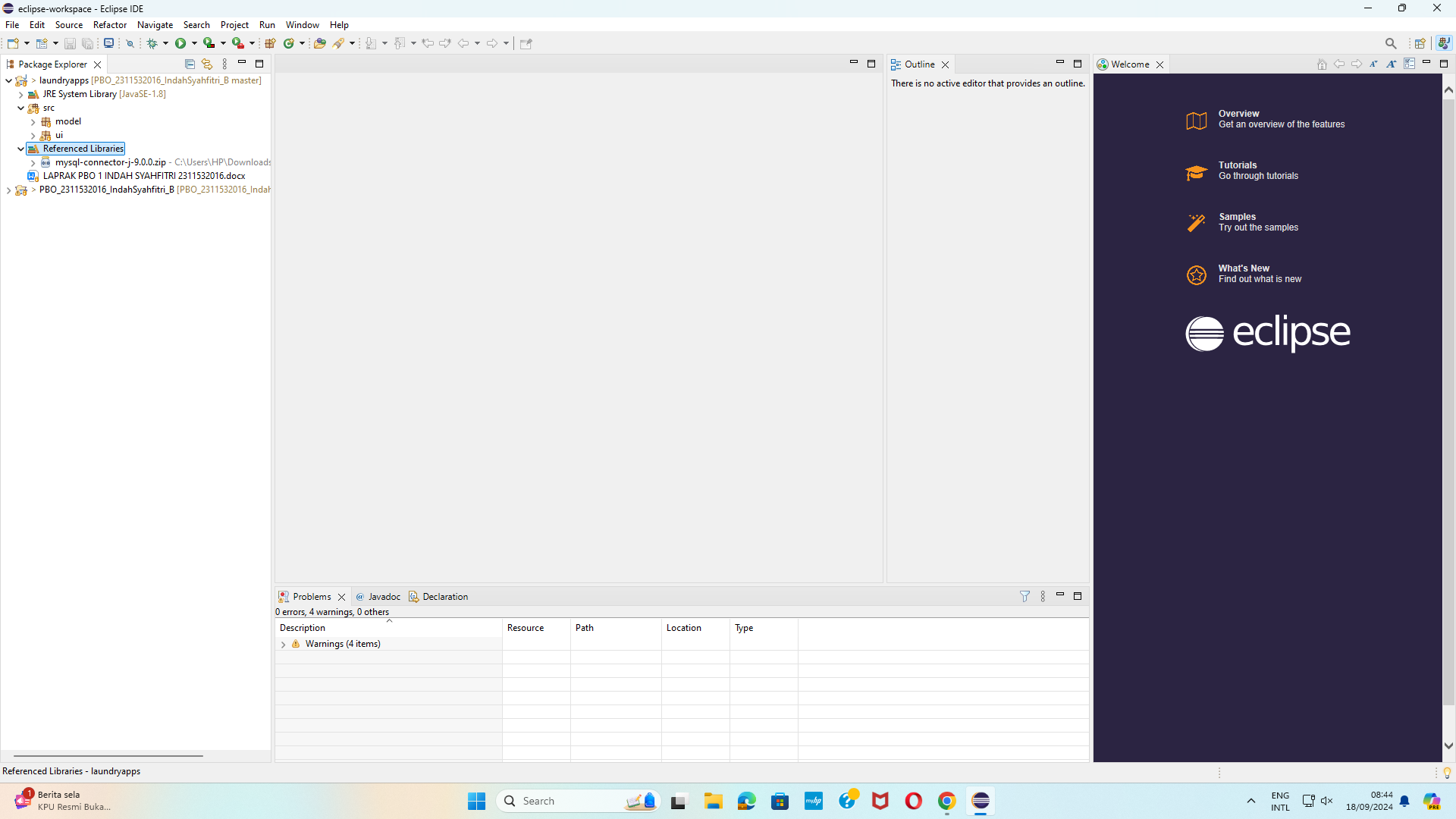


Pilih Libraries → Classpath. Tambahkan file MySQL Connector dengan cara klik Add External JARs dan pilih file yang telah didownlod dan pilih Apply and Close.

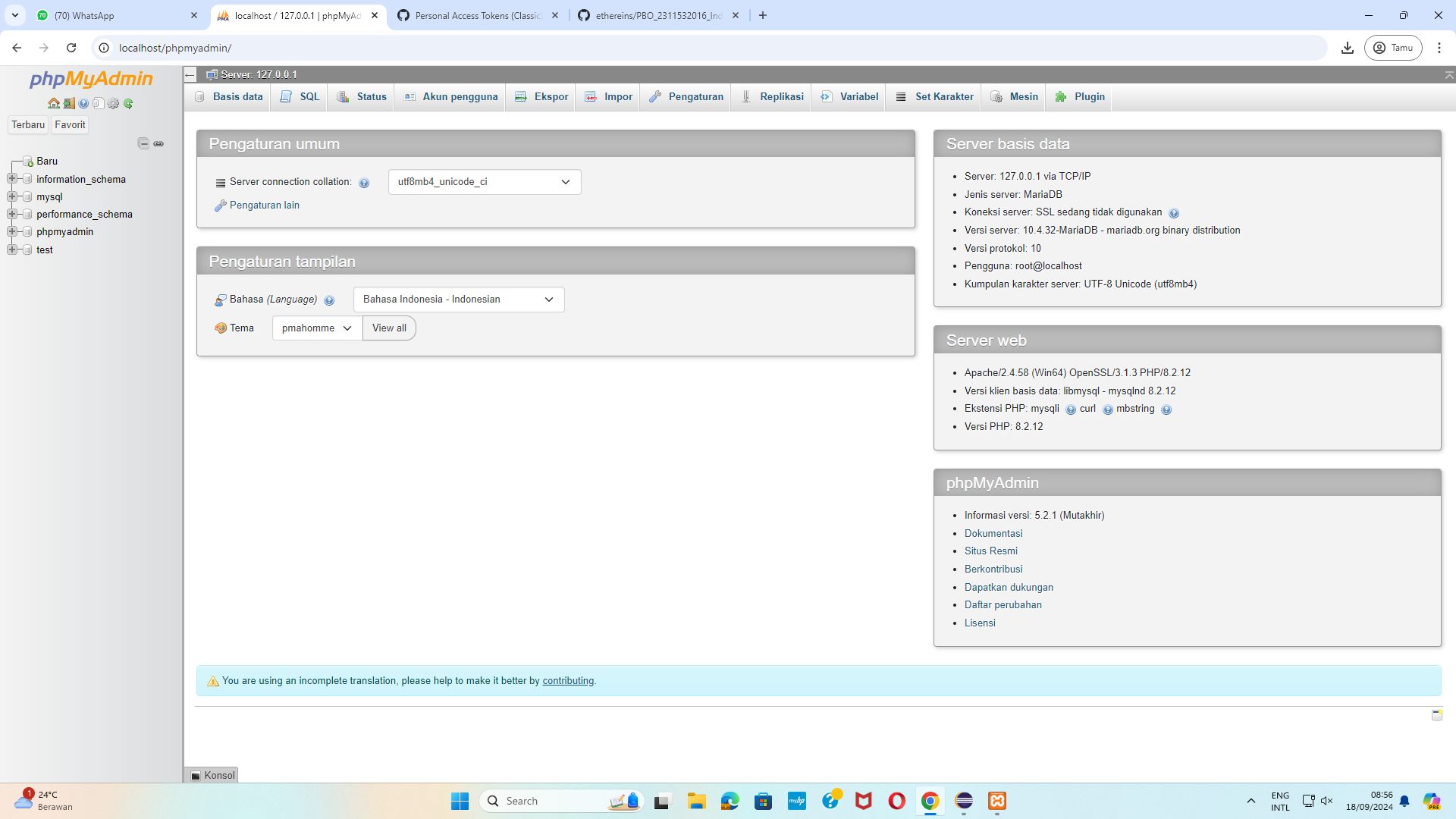


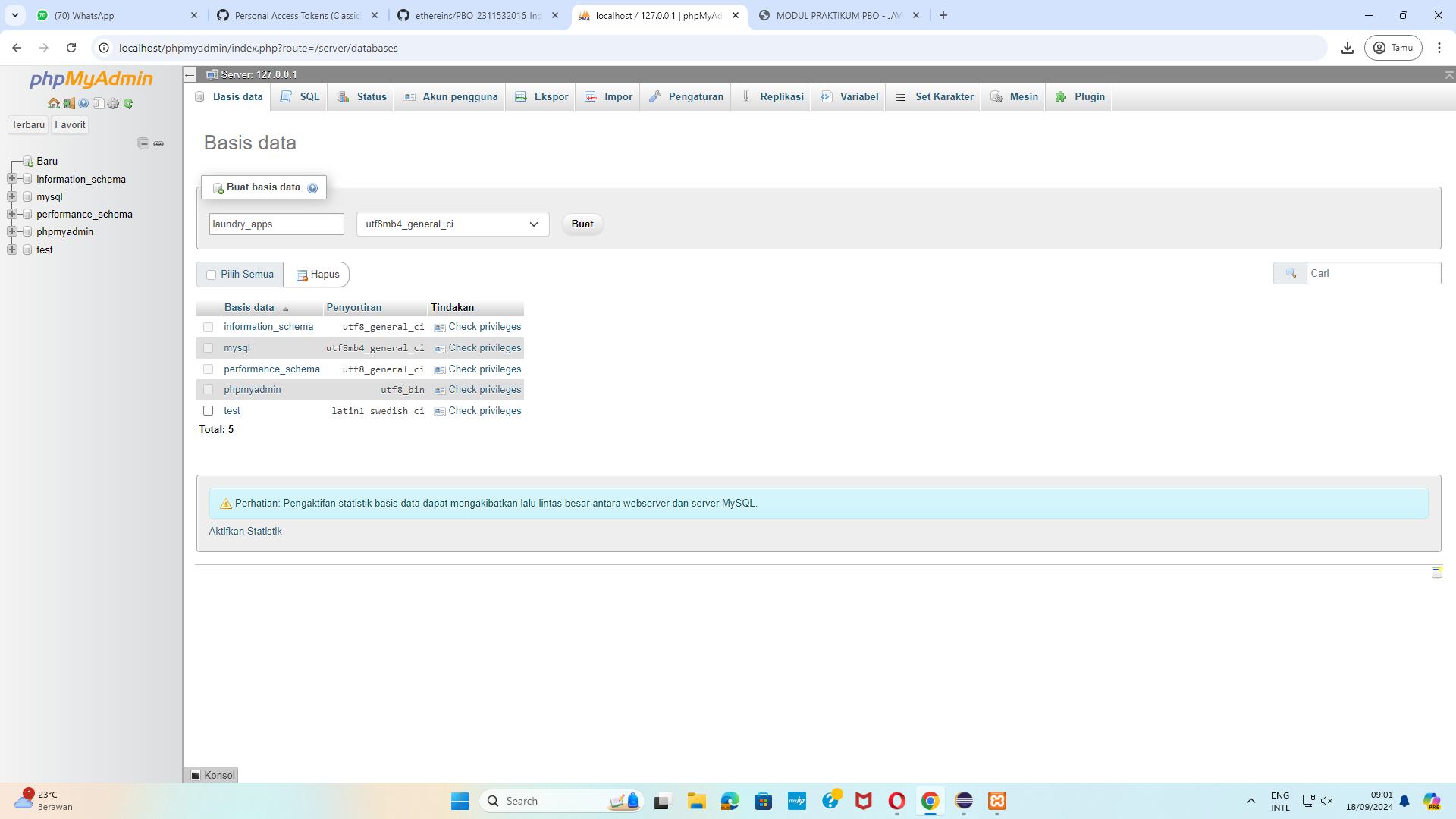


Jika berhasil menambahkan MySQL Connector maka akan generate folder Referenced Libraries pada project yang berisi MySQL Connector.

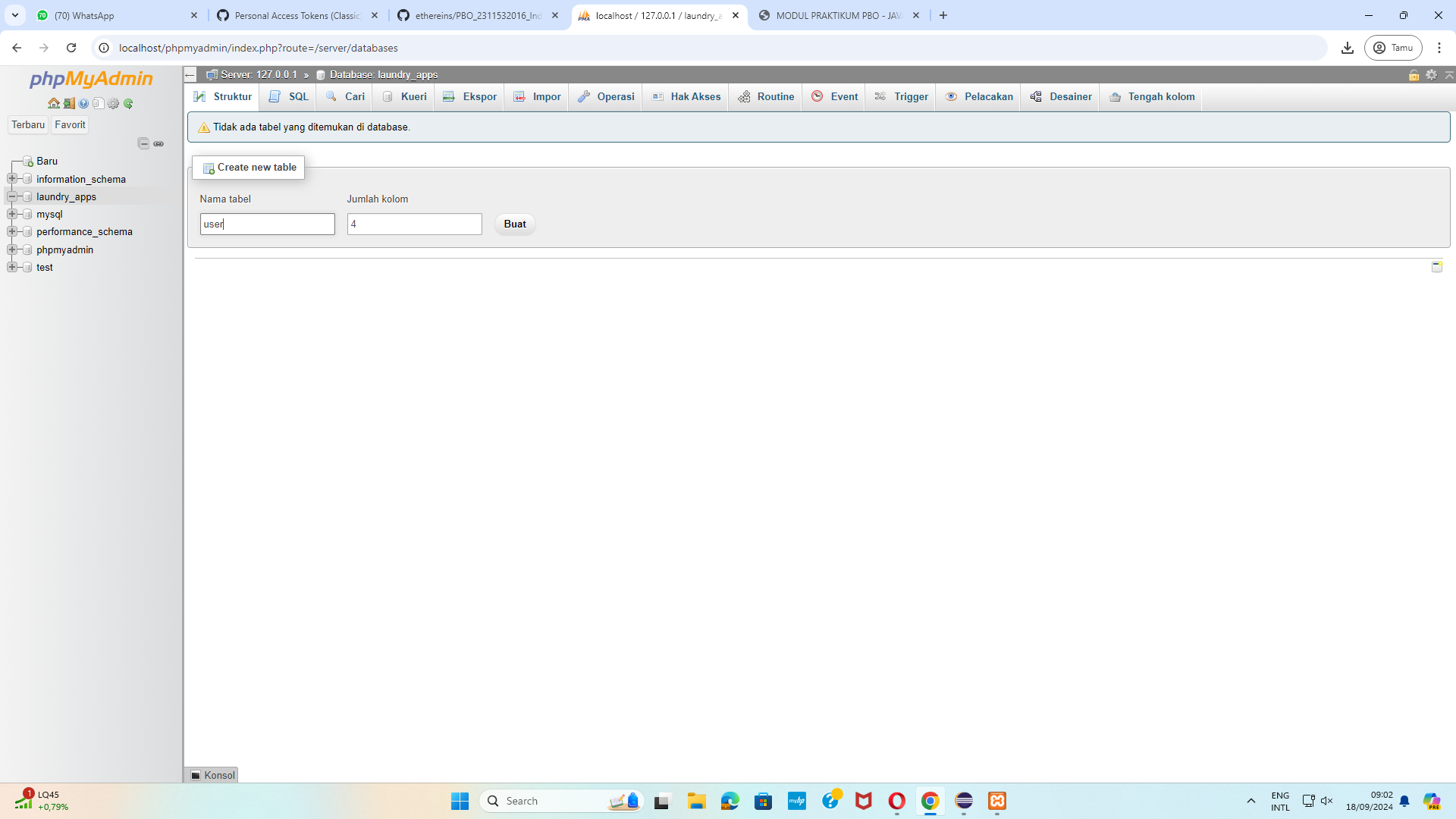


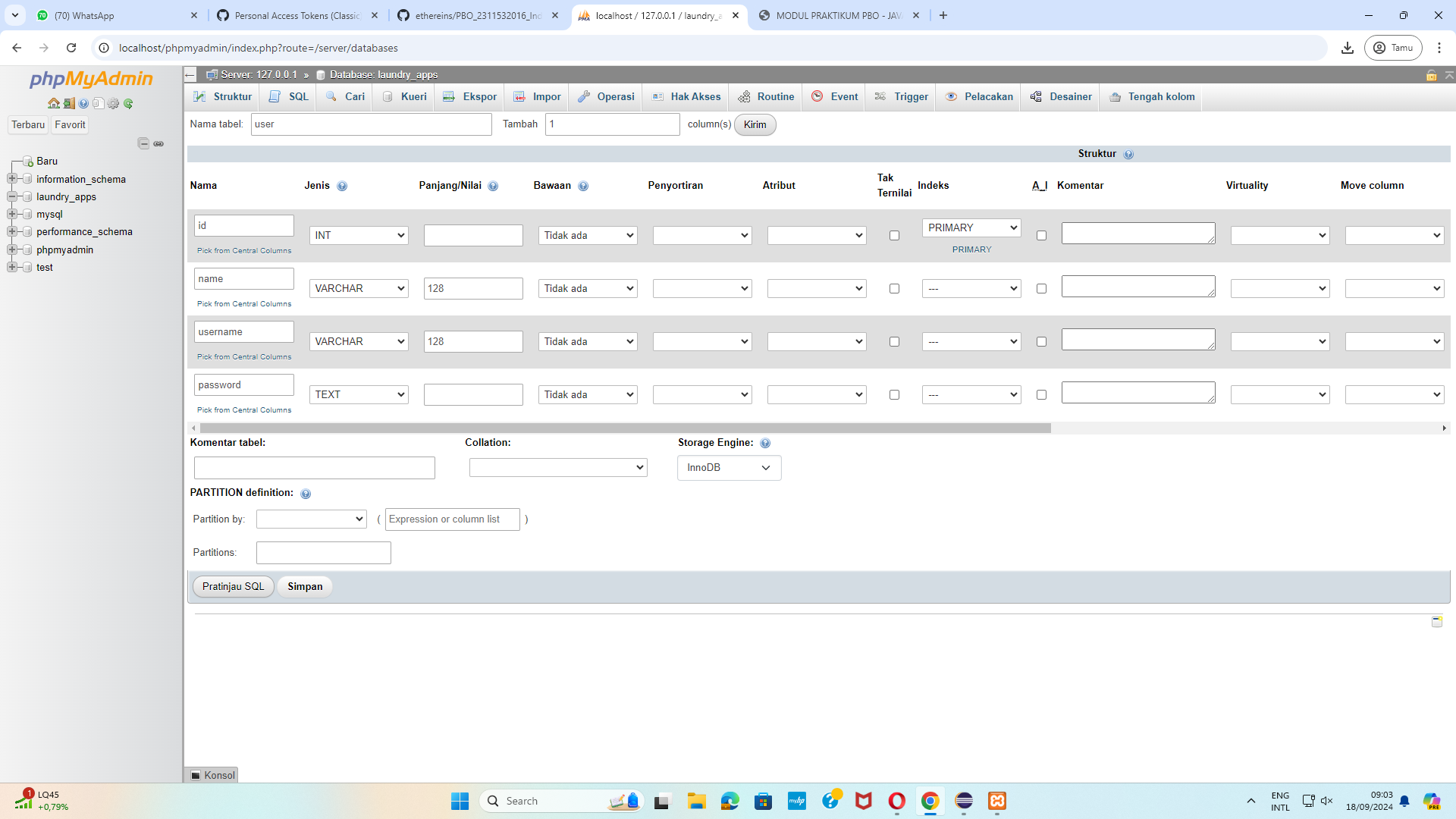
1. Untuk membuat database dan table user, buka <http://localhost/phpmyadmin> . Klik baru/new dan buat database dengan nama “laundry\_apps”.

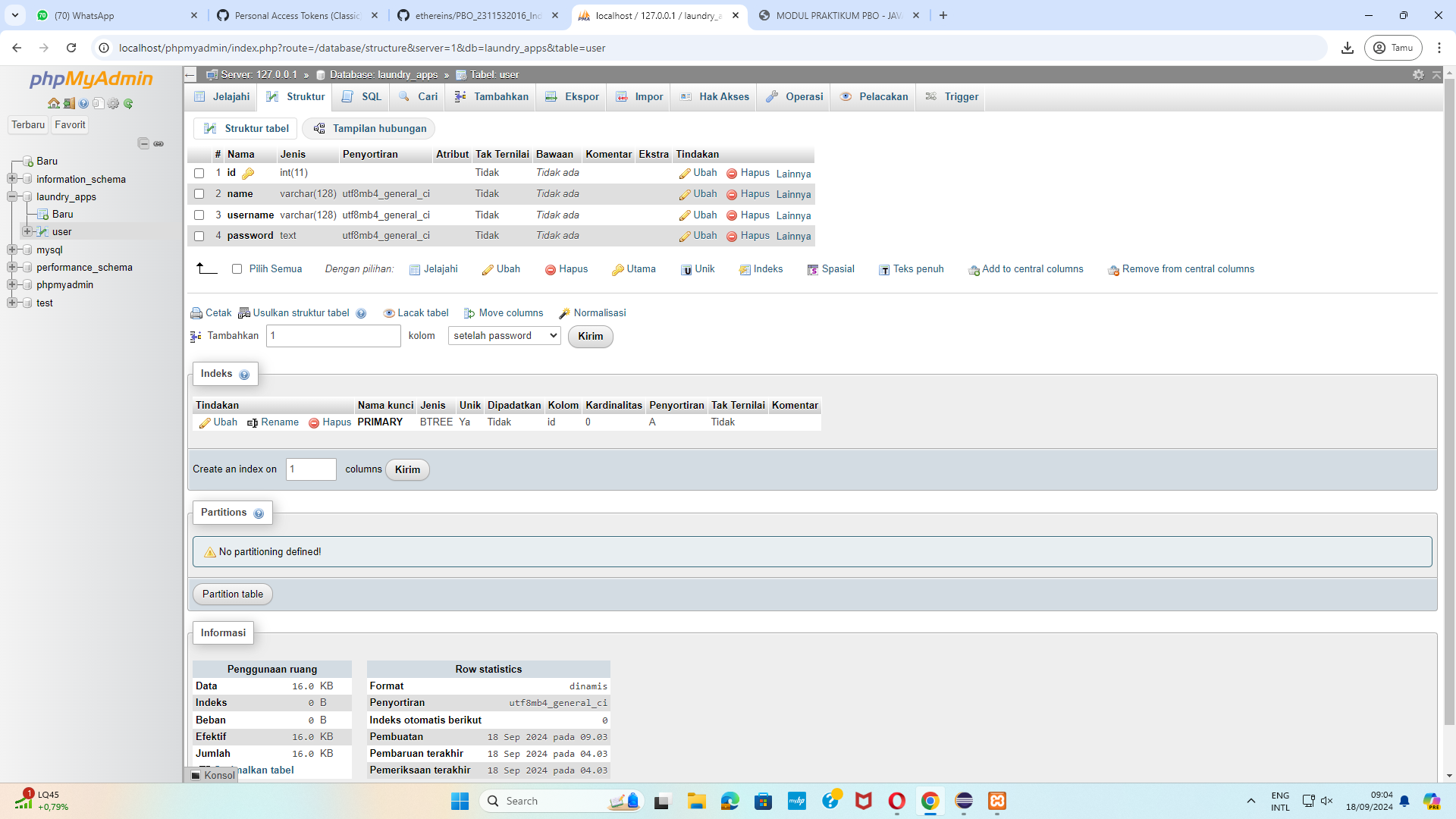




Tentukan jumlah kolom sesuai keinginan pengguna.

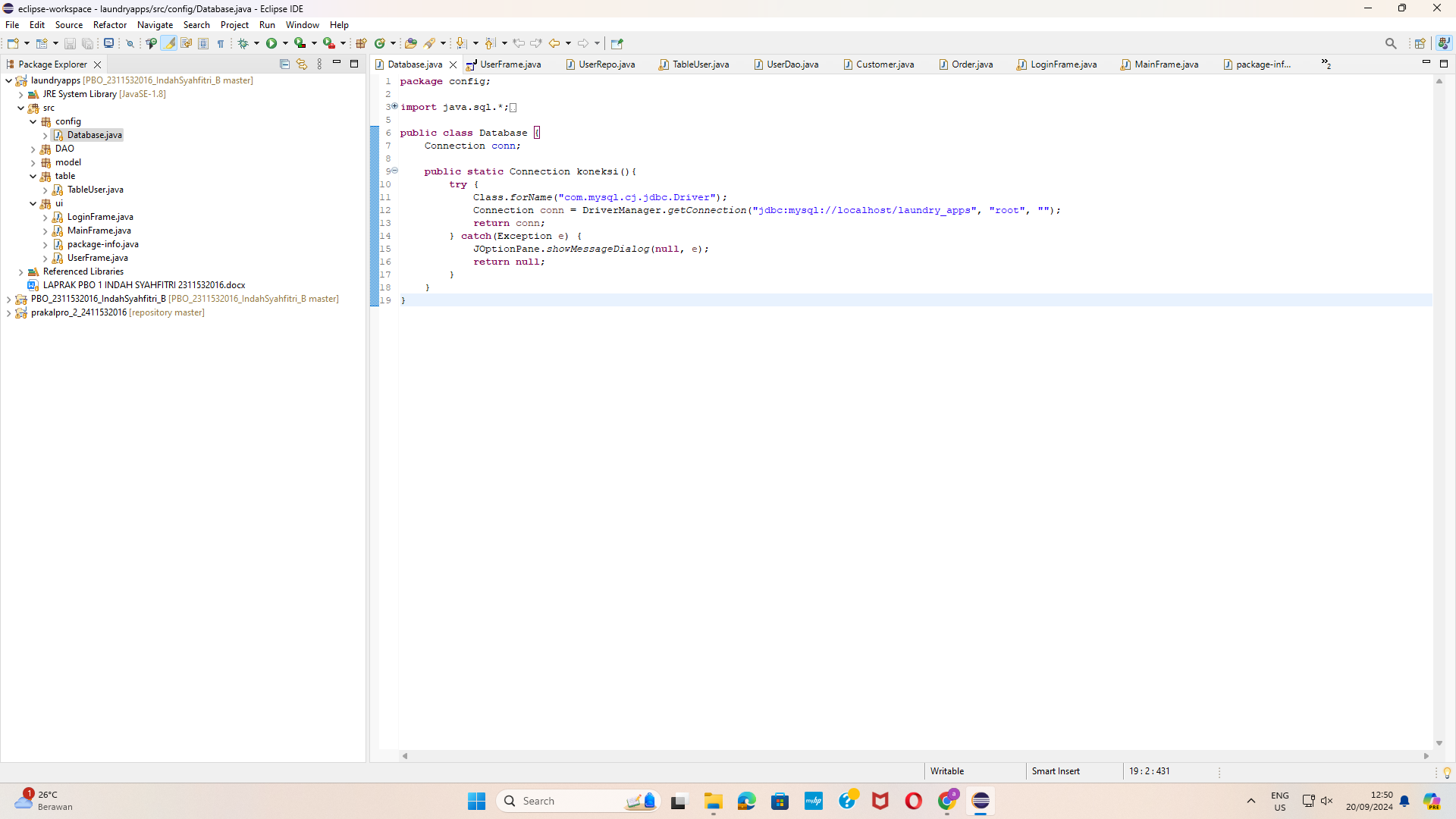






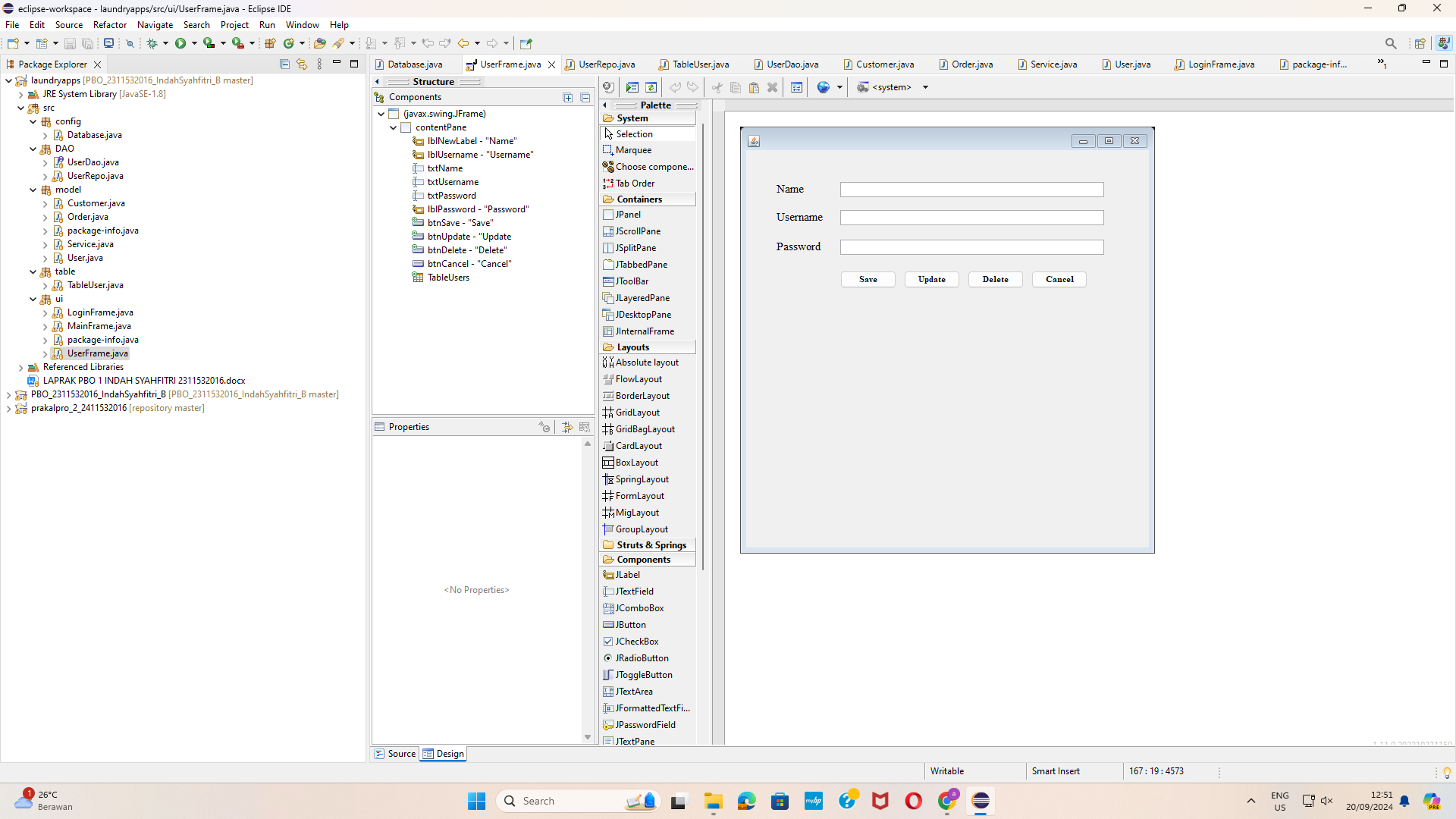
Isi seperti gambar di atas dan klik save. Setelah berhasil menambahkan MySQL Connector maka dapat membuat koneksi ke database MySQL.

1. Buat package baru dengan nama **config**, package ini yang akan digunakan untuk membuat konfigurasi aplikasi yang akan dibuat termasuk dengan konfigurasi database. Buat class baru dengan nama Database, kemudian konfigurasi sesuai dengan kode program.

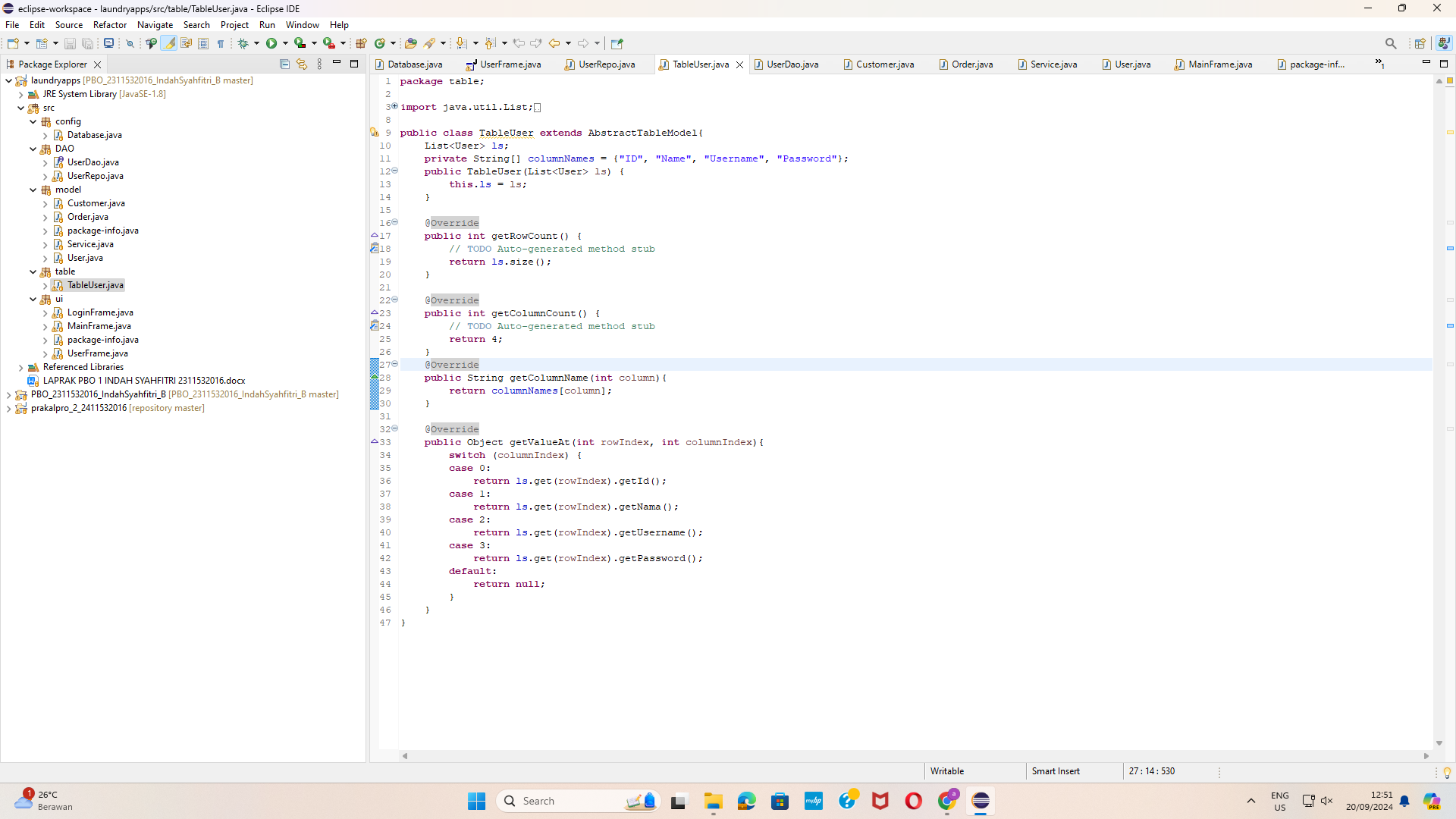


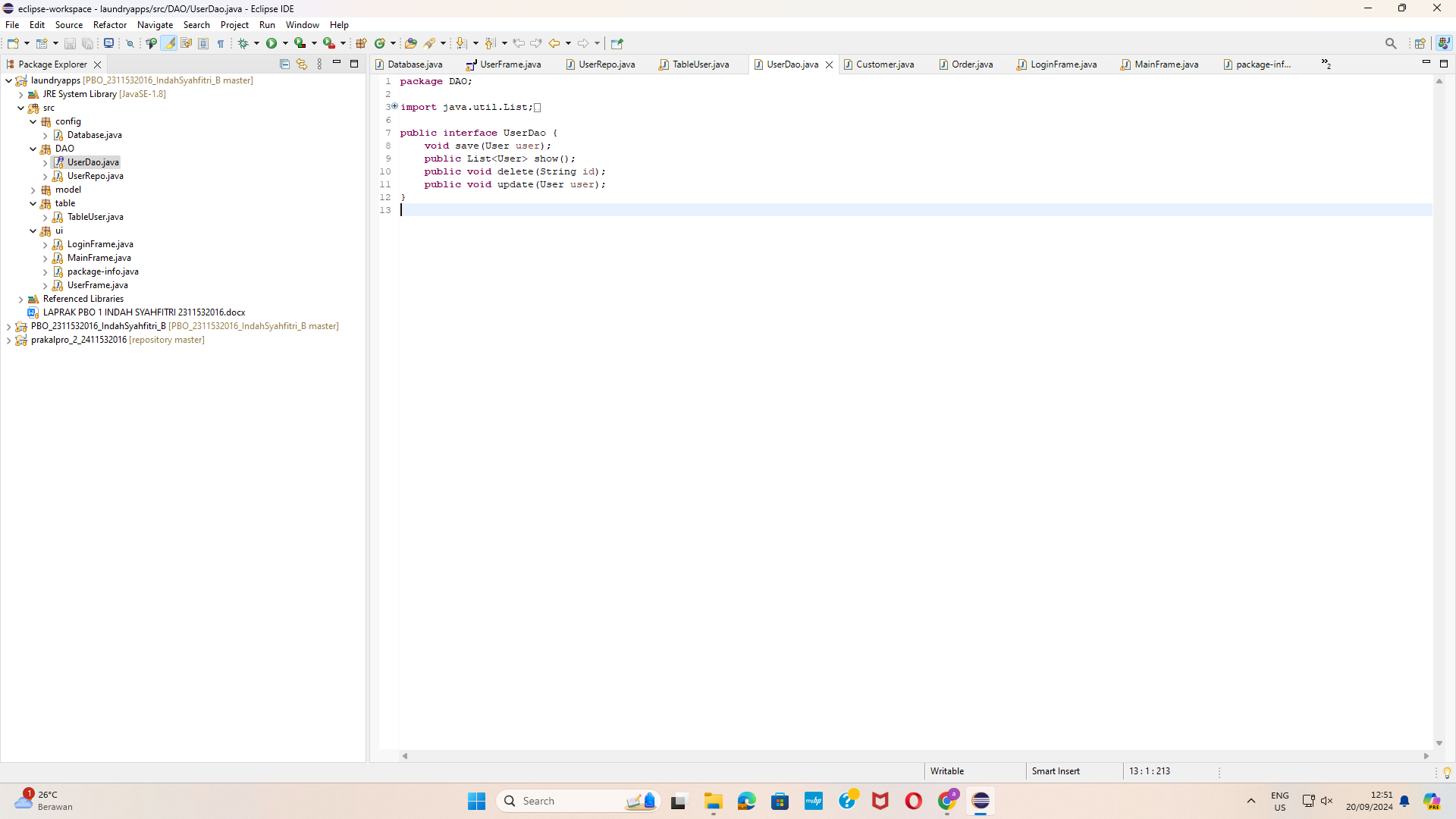
Import java.sql.\* digunakan untuk import seluruh fungsi-fungsi SQL. Line 9 membuka method Connection dengan nama koneksi, yang mana method ini akan digunakan untuk membuka koneksi ke database. Line 11-14 membuat koneksi database, jika koneksi berhasil maka akan mengembalikan nilai Connection. Line 15-16 jika koneksi gagal maka akan ditampilkan pesan error menggunakan JOptionPane.

1. Untuk membuat tampilan CRUD user, buat file baru menggunakan JFrame pada package ui dengan nama UserFrame seperti gambar berikut ini. Buat nama elemen sesuai dengan elemen yang dipakai. Misal txtName, txtUsername, btnSave, dll.



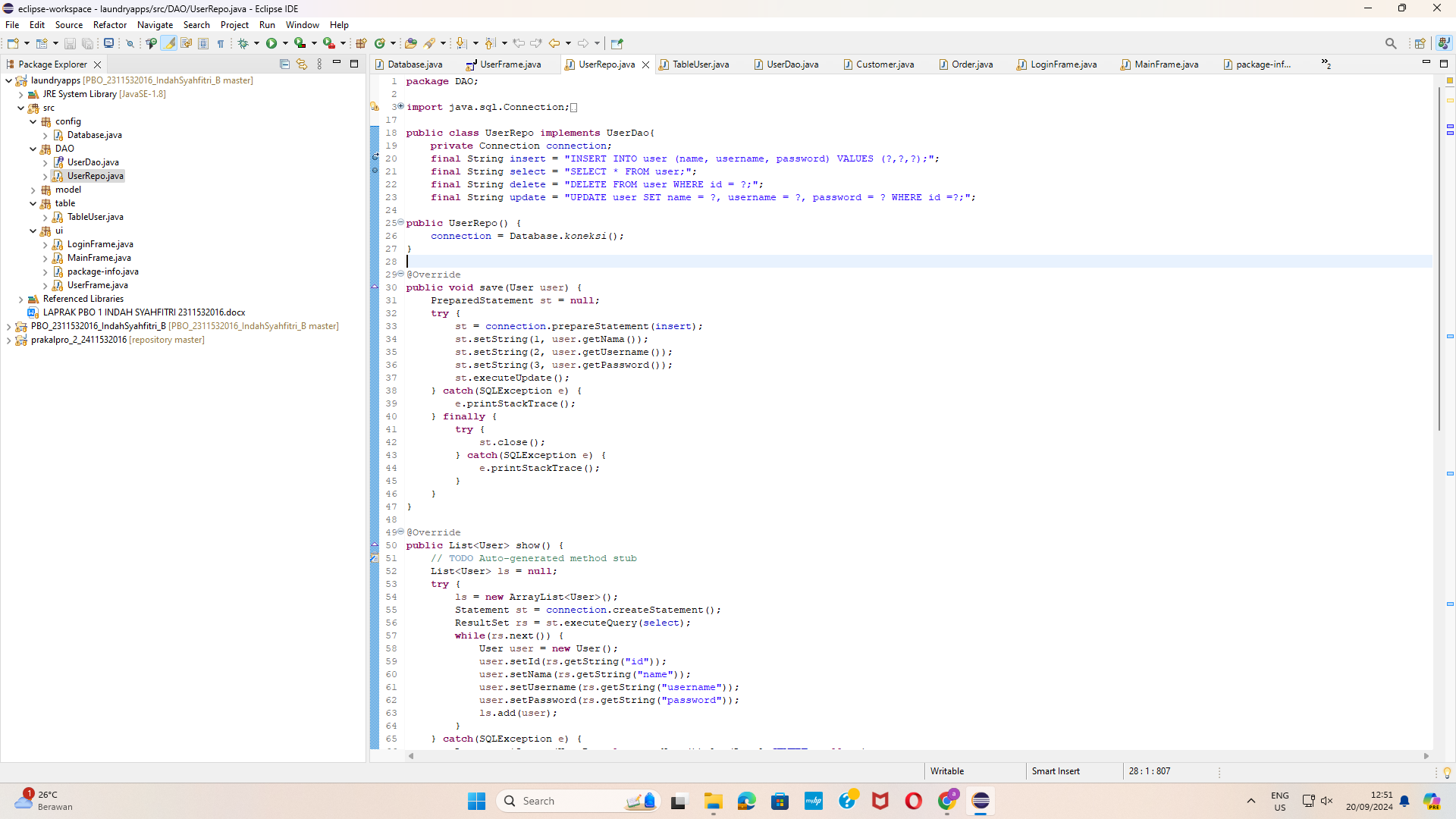
1. Lalu, buat package baru dengan nama table. Table model user ini berguna untuk mengambil data dari database dan ditampilkan ke dalam table. Buat file baru didalam package table dengan nama TableUser, kemudian isikan dengan kode program berikut.

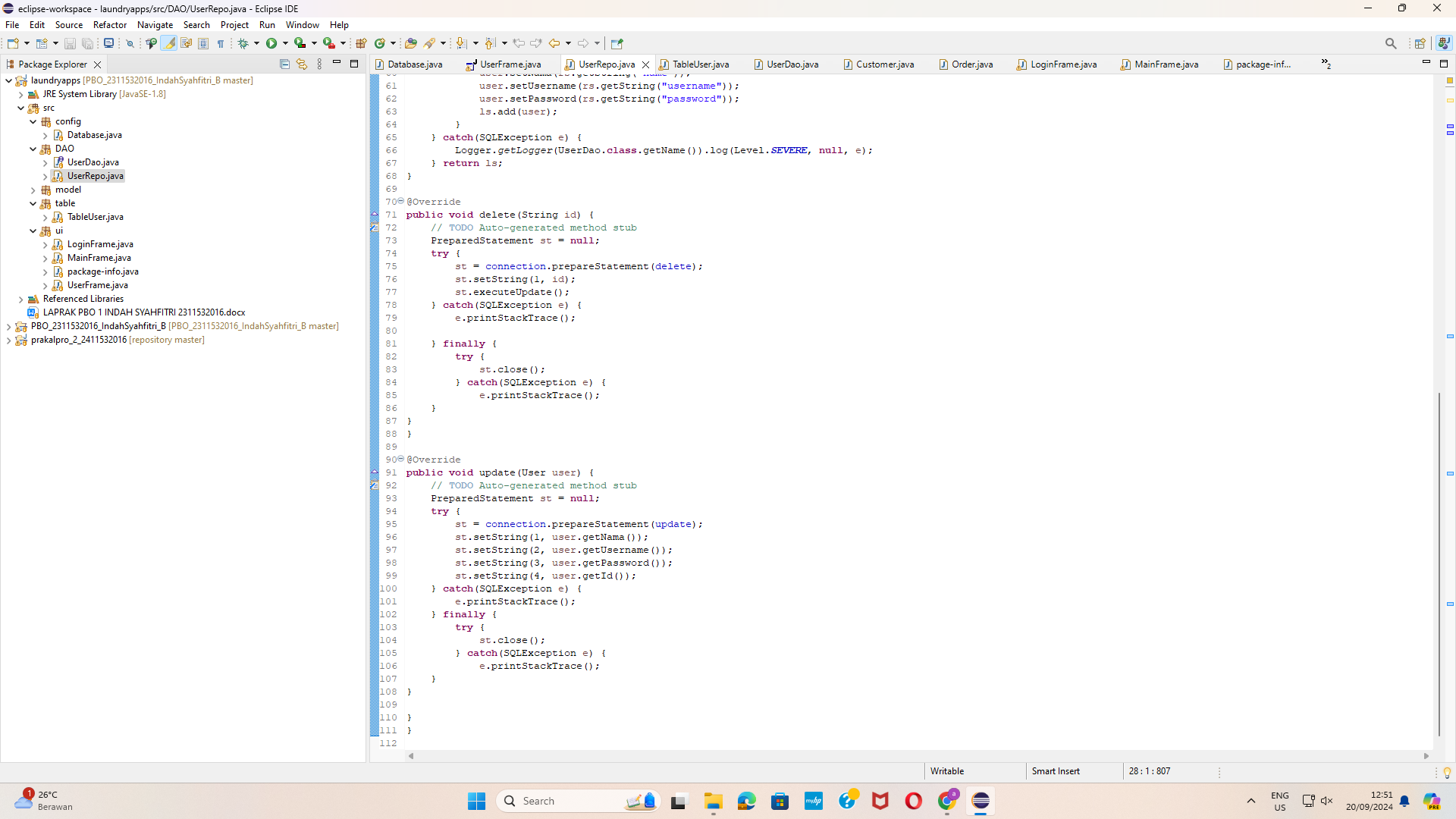


1. Kemudian, buat package baru dengan nama DAO. Buat class Interface baru dengan nama UserDAO, kemudian isikan dengan kode program berikut. 

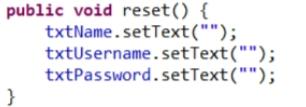
Terdapat method save, show, delete dan update. Method pada class interface digunakan sebagai method utama yang wajib diimplementasikan pada class yang menggunakannya.

1. Setelahnya, buat class baru pada package DAO dengan nama UserRepo yang mana akan digunakan untuk mengimplementasikan DAO yang telah dibuat. Implementasikan UserDao dengan kata kunci implements. Membuat instanisasi Connection, membuat constructor dan membuat String untuk melakukan manipulasi database. Lanjutkan nanti dengan membuat method save, show, delete dan update untuk diterapkan kepada data yang diinginkan.

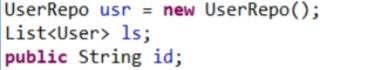




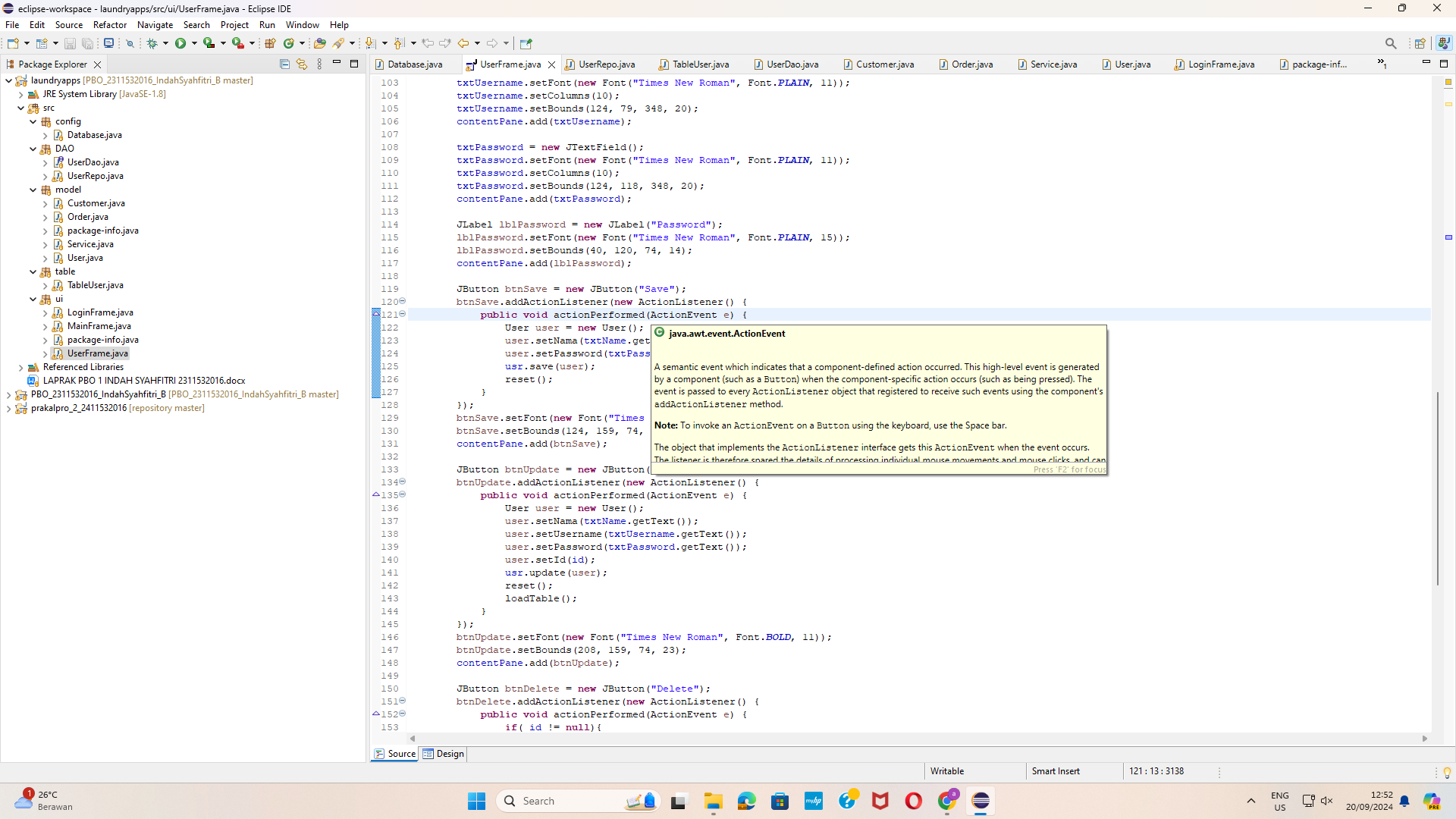
1. Ketika suatu proses berhasil dilakukan, buat method reset pada JFrame seperti kode program di bawah ini.

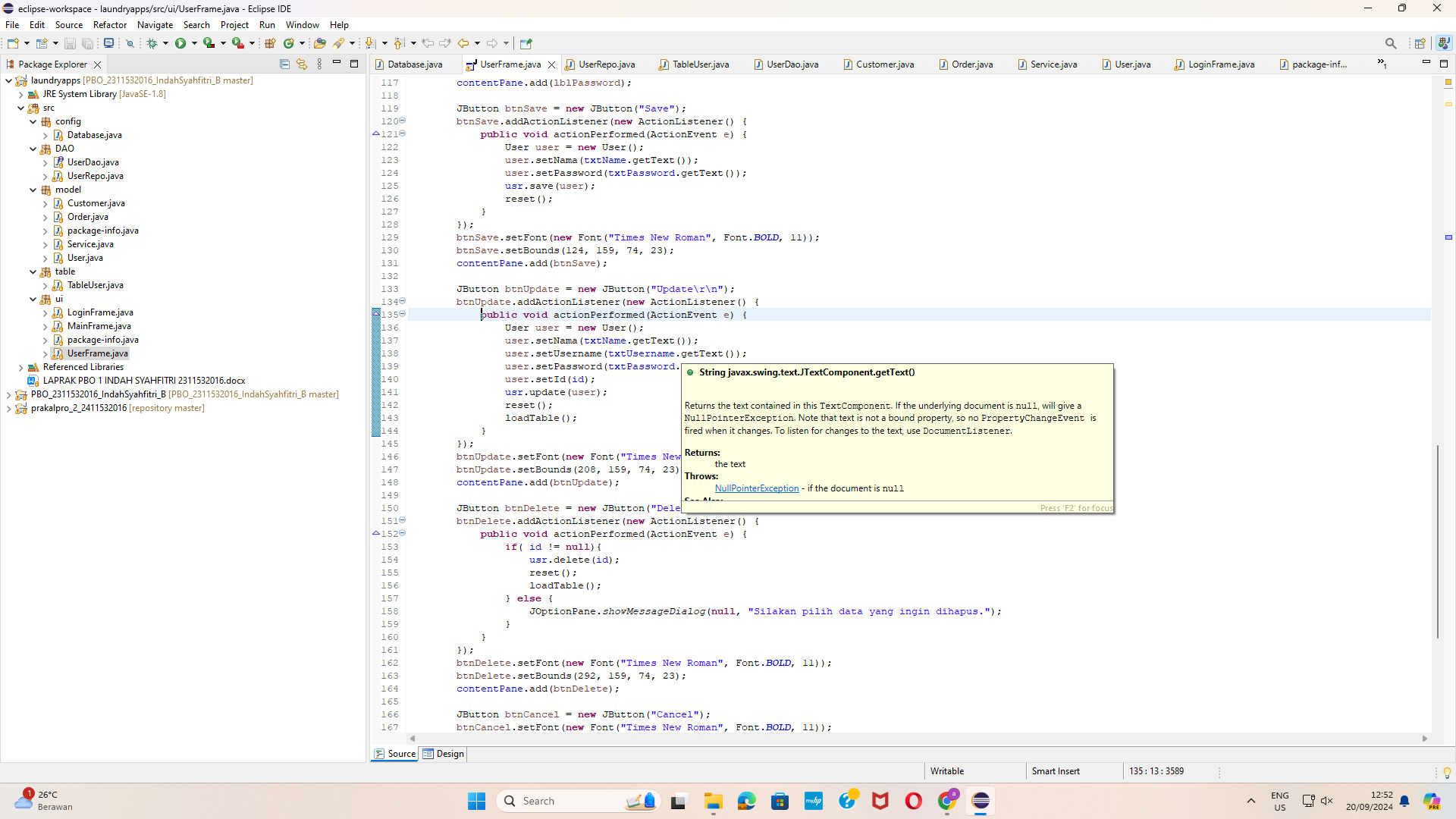


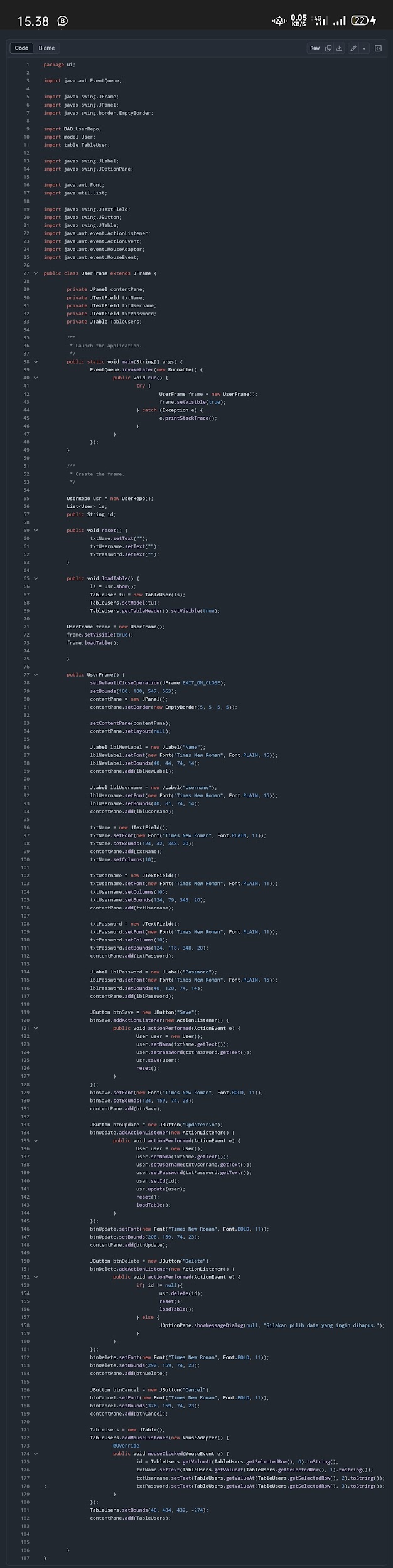
1. Buat instance pada class UserFrame.



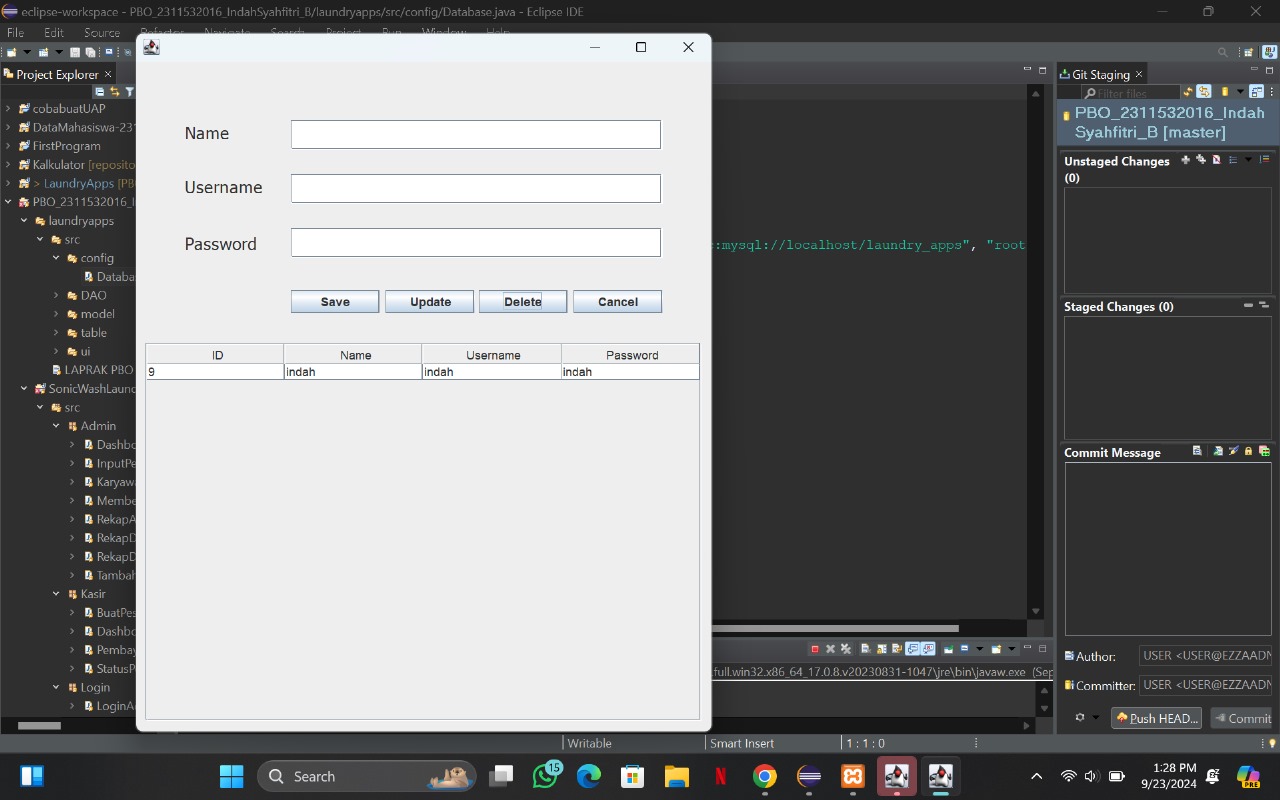
1. Lalu, klik kanan pada tombol save → add event handlers → actionPerformed untuk membuat program CRUD.



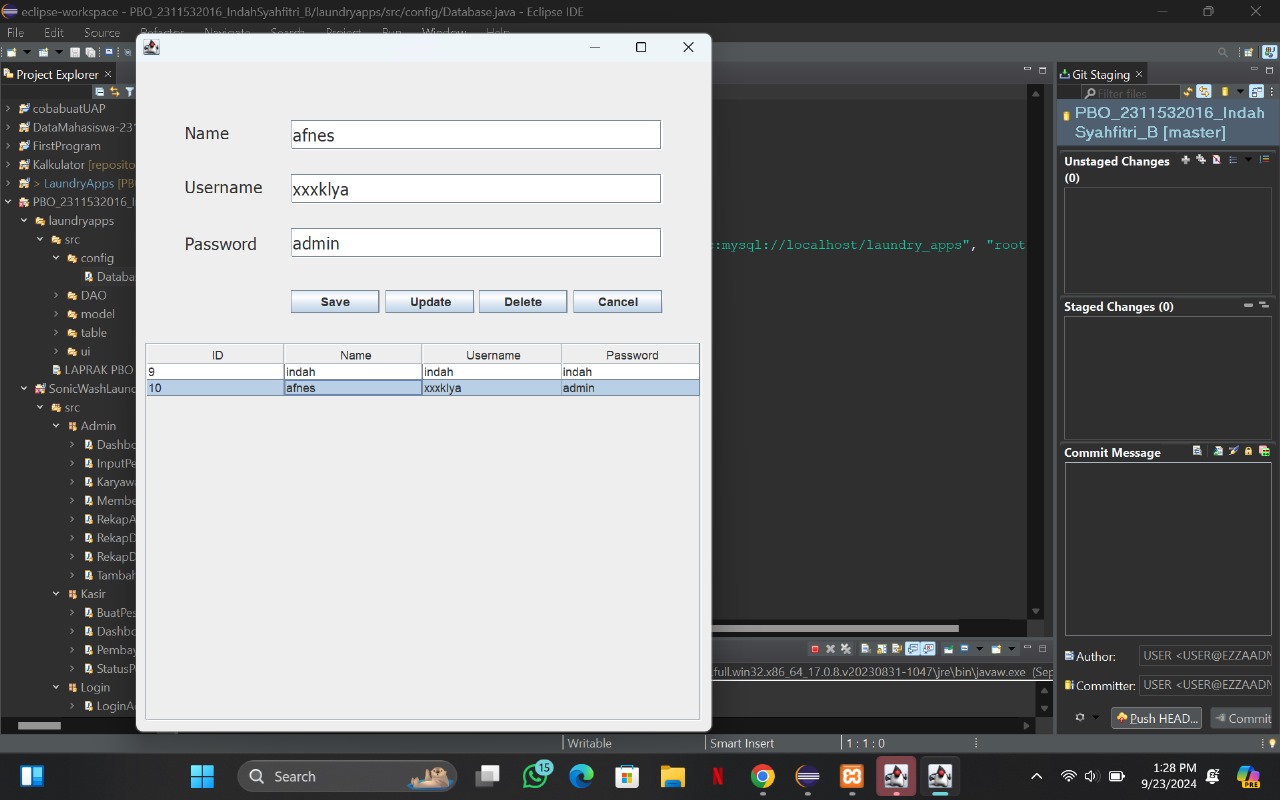




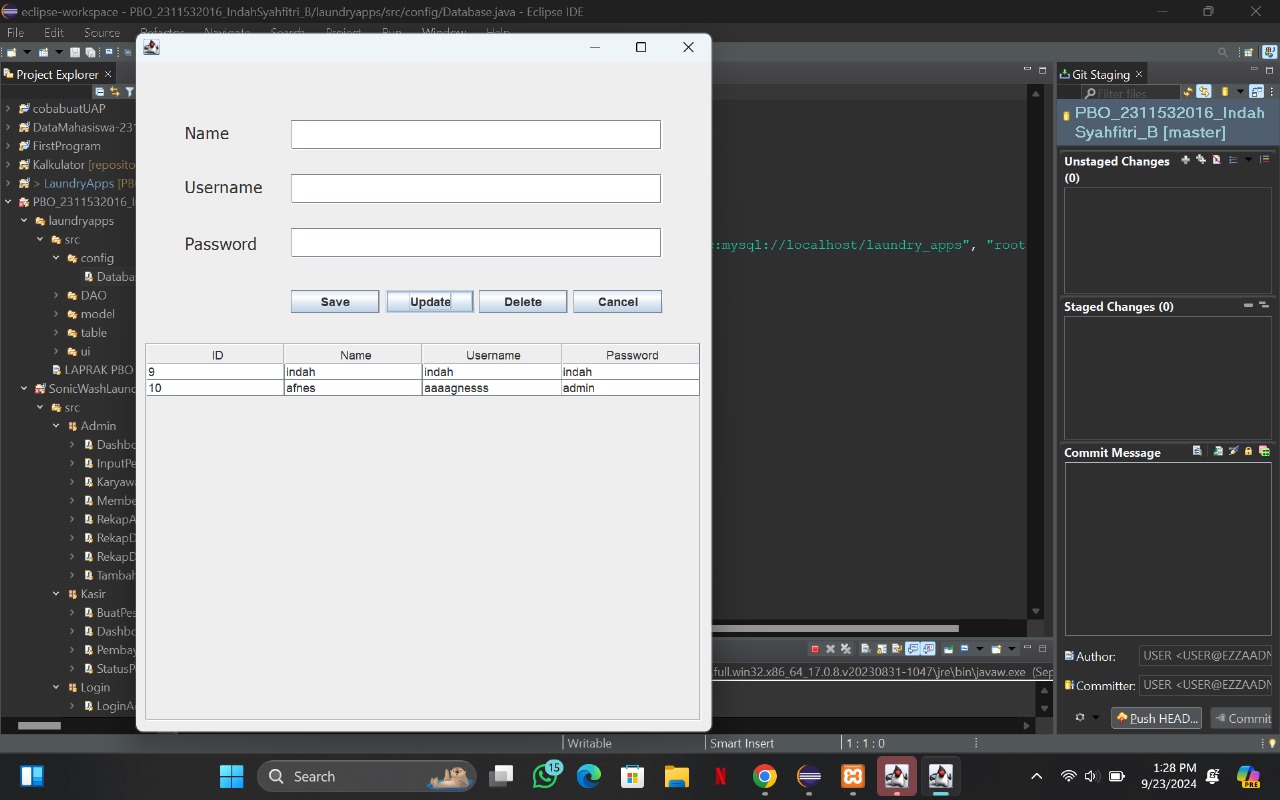
1. Jika sudah selesai, program bisa dijalankan. Akan muncul tampilan seperti ini.



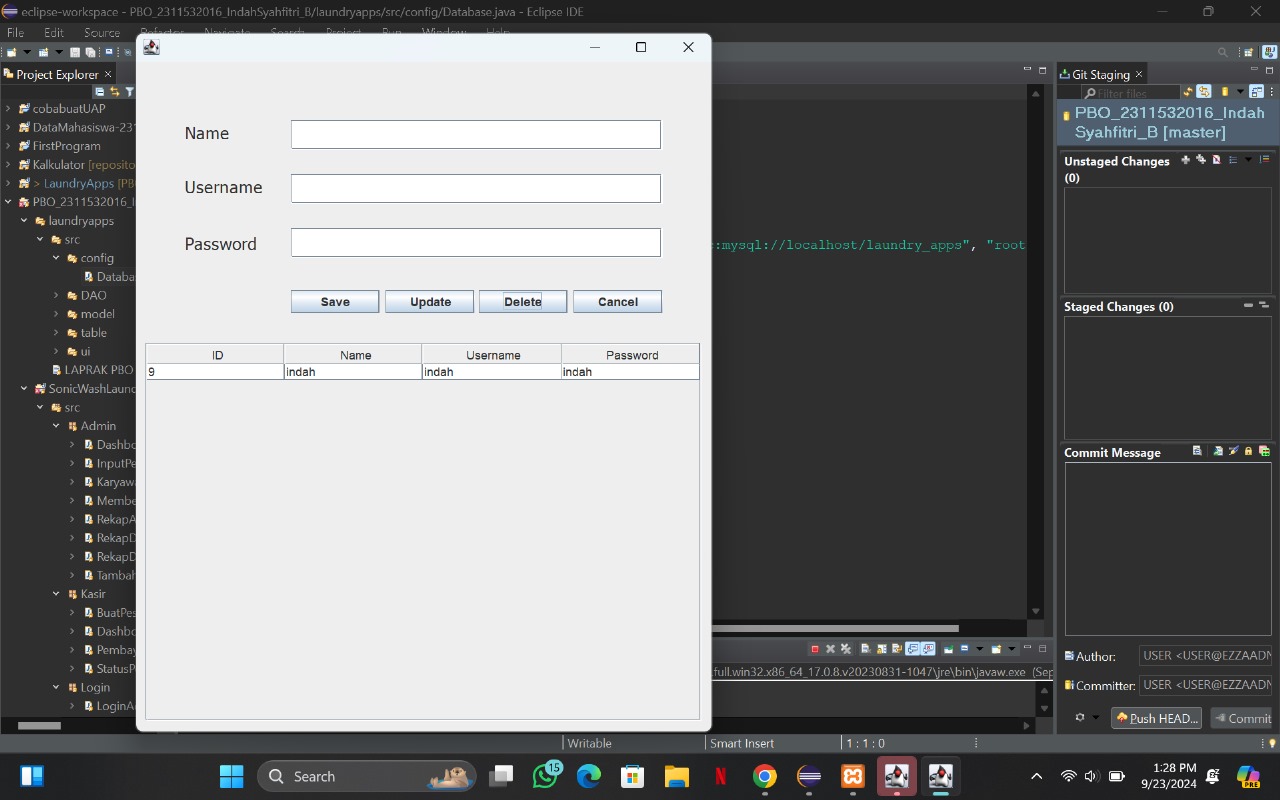
Untuk memasukkan data, isi nama, username, dan password lalu tekan save. Data baru akan masuk ke dalam tabel.



1. Untuk memperbarui data, ubah bagian yang diinginkan lalu tekan tombol update. Maka data di tabel akan terperbarui.



1. Untuk menghapus data, pilih data yang ingin dihapus lalu tekan tombol delete, maka data terpilih akan terhapus dari dalam tabel.



1. Untuk menghapus data yang belum sempat tersubmit dan ingin mengisinya dengan yang baru, tekan cancel. Maka akan terhapus.
2. **SIMPULAN**

Praktikum ini telah mengimplementasikan konsep dasar dari pemrograman berorientasi objek menggunakan bahasa pemrograman Java. Kelas pertama adalah kelas User sebagai pengguna dalam aplikasi manajemen laundry, dengan atribut seperti id, nama, username, dan password. Program ini juga memiliki metode getter dan setter untuk mengakses dan mengubah nilai atribut, serta metode login yang menggunakan parameter username dan password untuk memverifikasi identitas pengguna.

Untuk antarmuka pengguna (UI) telah diimplementasikan pada kelas LoginFrame dan MainFrame. Program ini menghubungkan tampilan visual dengan ptogram yang telah ada pada kelas User. Pengguna dapat login dengan memasukkan username dan password, yang akan dicek melalui metode login pada kelas User. Setelah login berhasil, pengguna akan diarahkan ke halaman utama yang berisi fitur-fitur aplikasi seperti layanan, pelanggan, dan pesanan. Setiap fitur diwakili oleh kelas-kelas terpisah seperti Service, Customer, dan Order. Kelas-kelas ini memanfaatkan konsep getter dan setter.