GUI dan Database

Nama : Rizqi Rohmatul Huda

Kelas : 2G-TI

No.Absen : 26

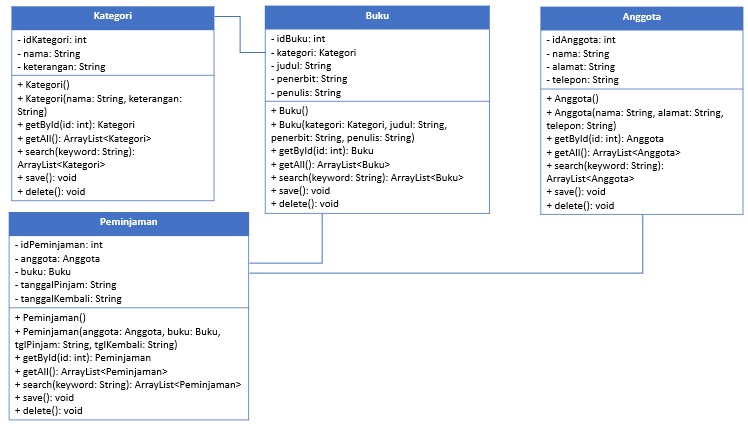
NIM : 2141720264

# Kompetensi

Setelah menempuh materi percobaan ini, mahasiswa mampu:

1. Menggunakan paradigma berorientasi objek untuk interaksi dengan database
2. Membuat Graphical User Interface (GUI)

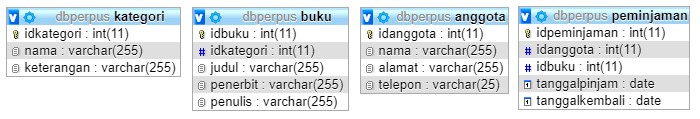
# Pendahuluan



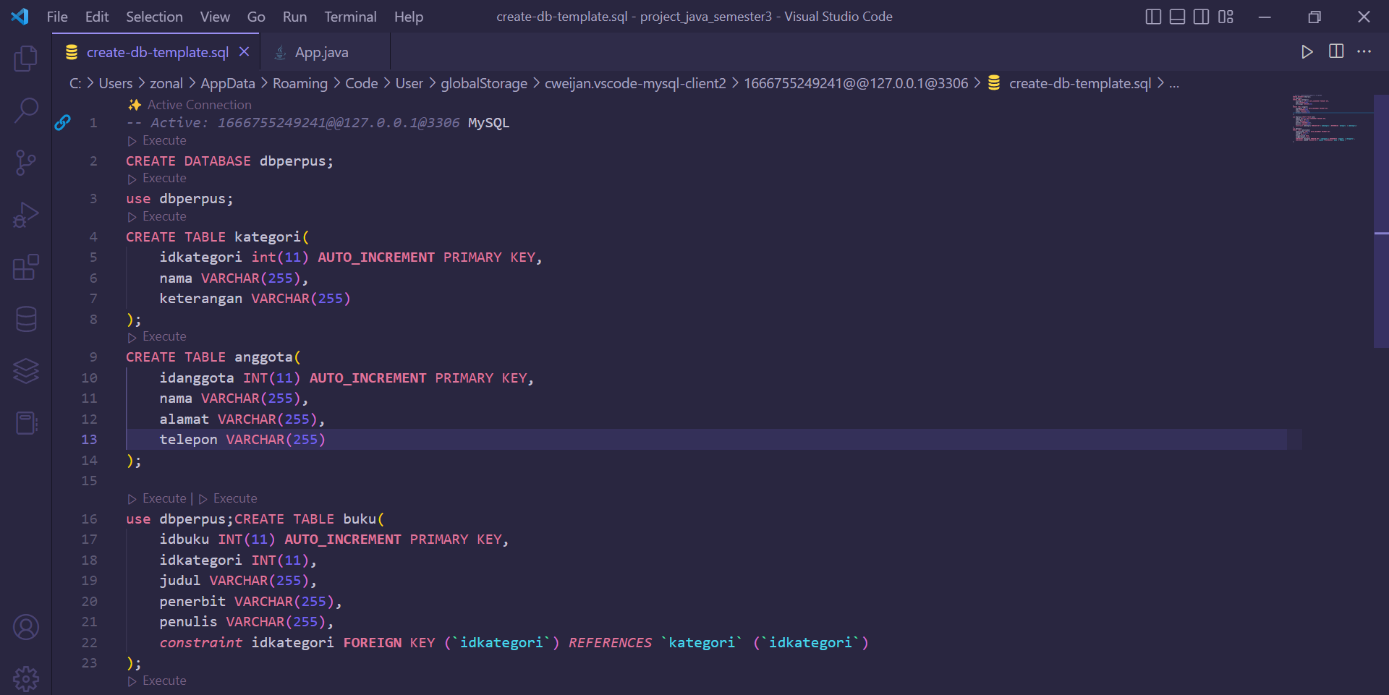
# Percobaan

## Percobaan 1

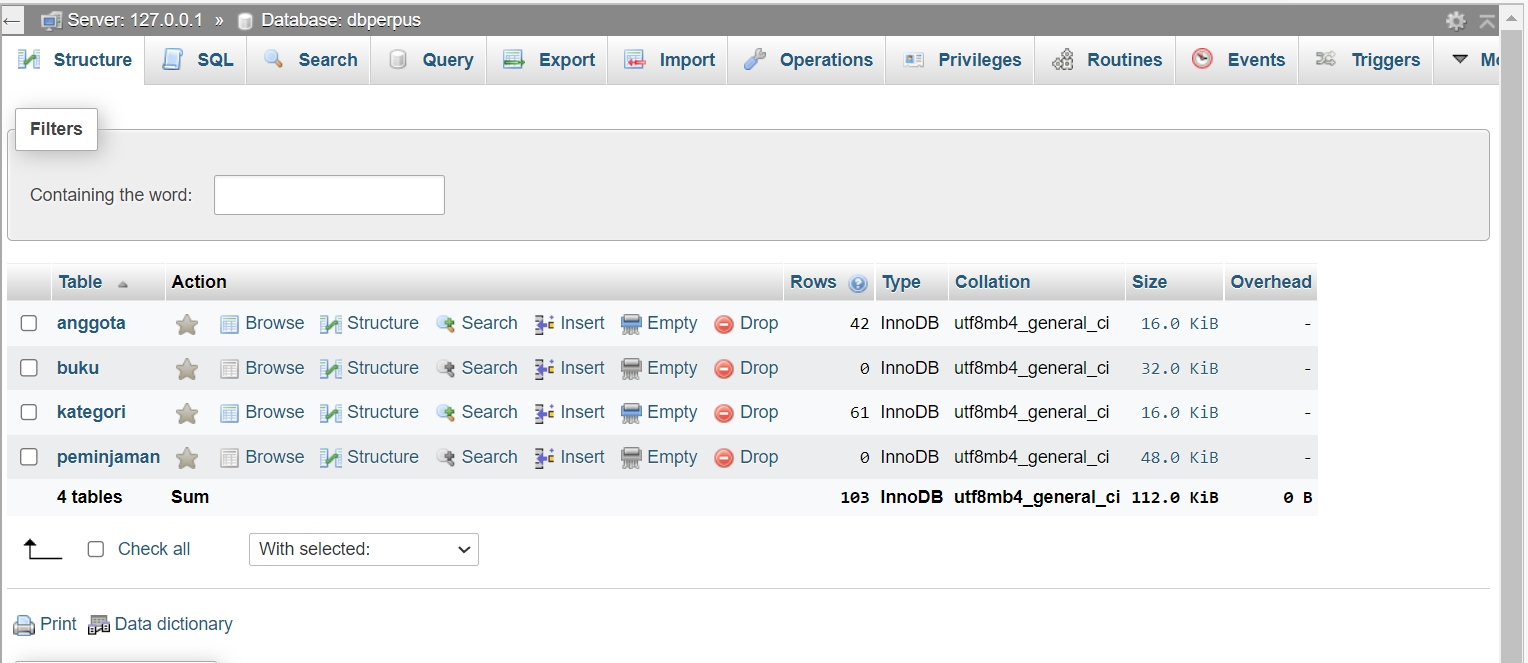
Membuat database.



**Sintaks database perpustakaan**



Database perpustakaan



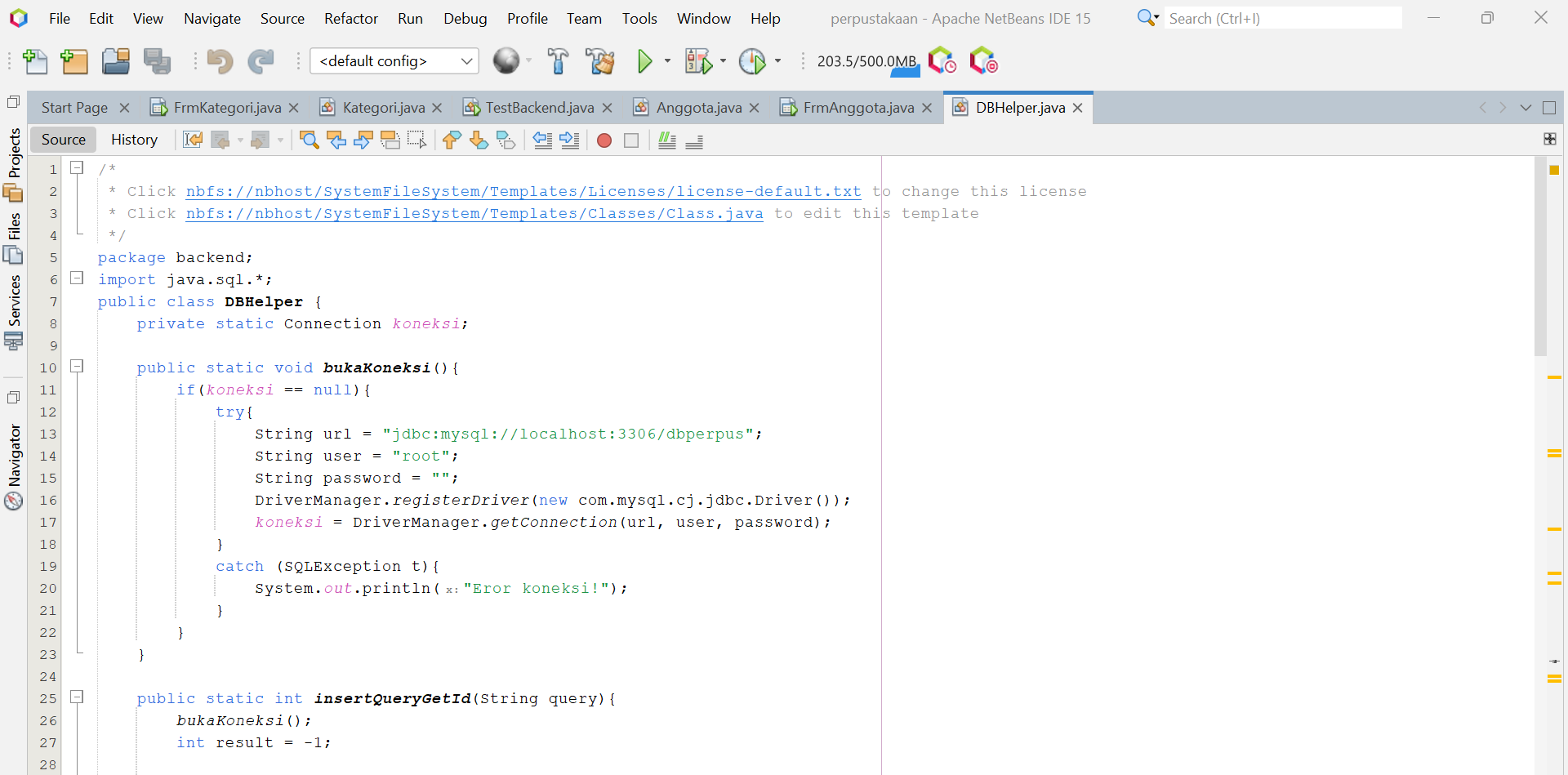
## Percobaan 2

Mempersiapkan project

1. Buat project baru, beri nama **Perpustakaan**.
2. Pada project explorer, klik kanan pada Libraries  Add Library, pilih MySQL JDBC Driver.
3. Jika tidak ada pilihan MySQL JDBC Driver:
   1. Buka link <https://dev.mysql.com/downloads/connector/j/>
   2. Pilih “Platform Independent”pada opsi Operating System
   3. Download ZIP Archive kemudian extract
   4. Pada project yang dibuat, klik kanan pada Libraries, kemudian add JAR/Folder… dan pilih file jar yang telah diextract sebelumnya
4. Buat package **frontend** dan **backend**. Cara membuat package adalah, pada project explorer, klik kanan pada Source Packages  New  Java Package, beri nama package nya (frontend, backend).

## Percobaan 3

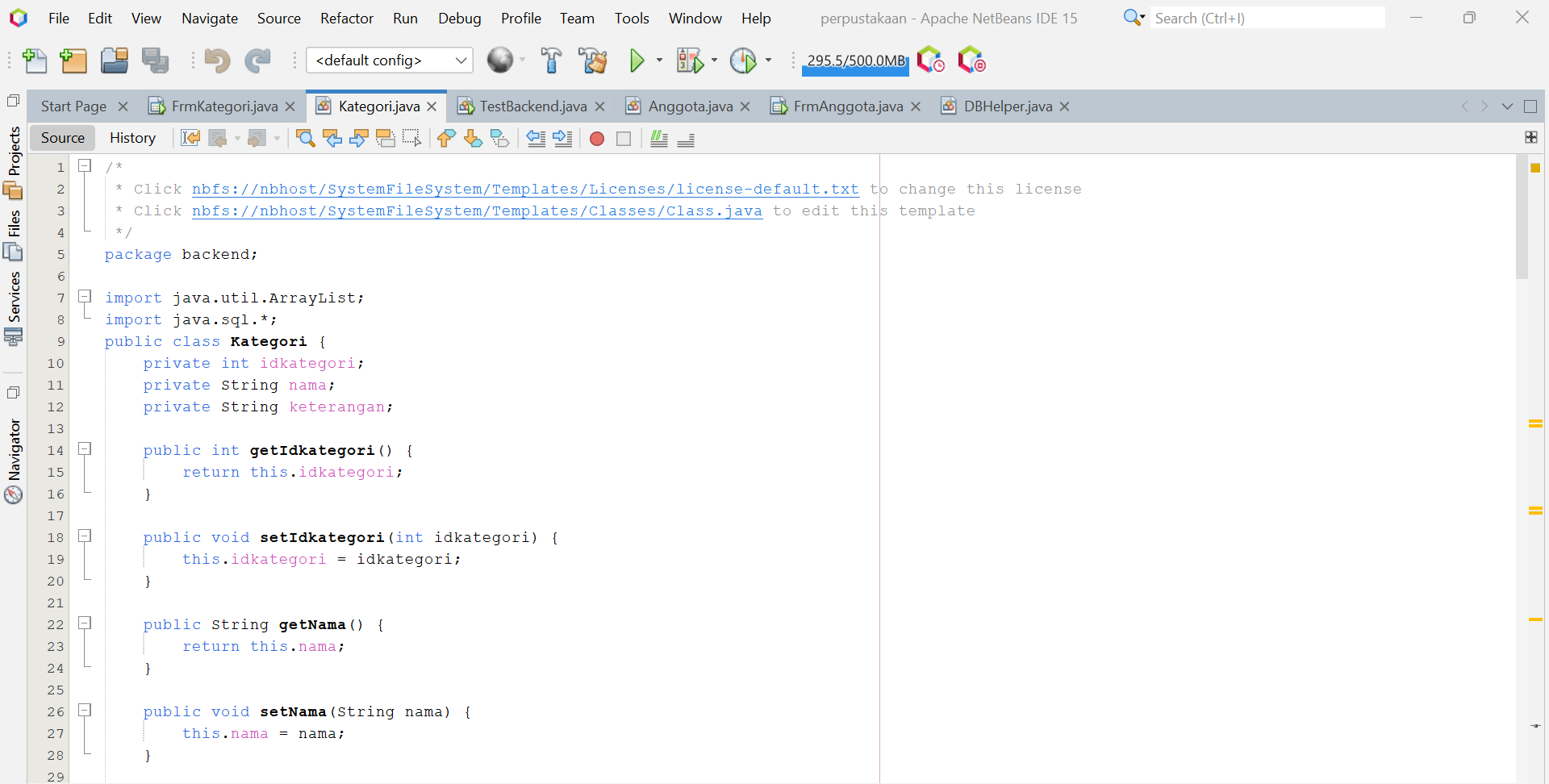
Membuat class helper untuk mengeksekusi query SQL.

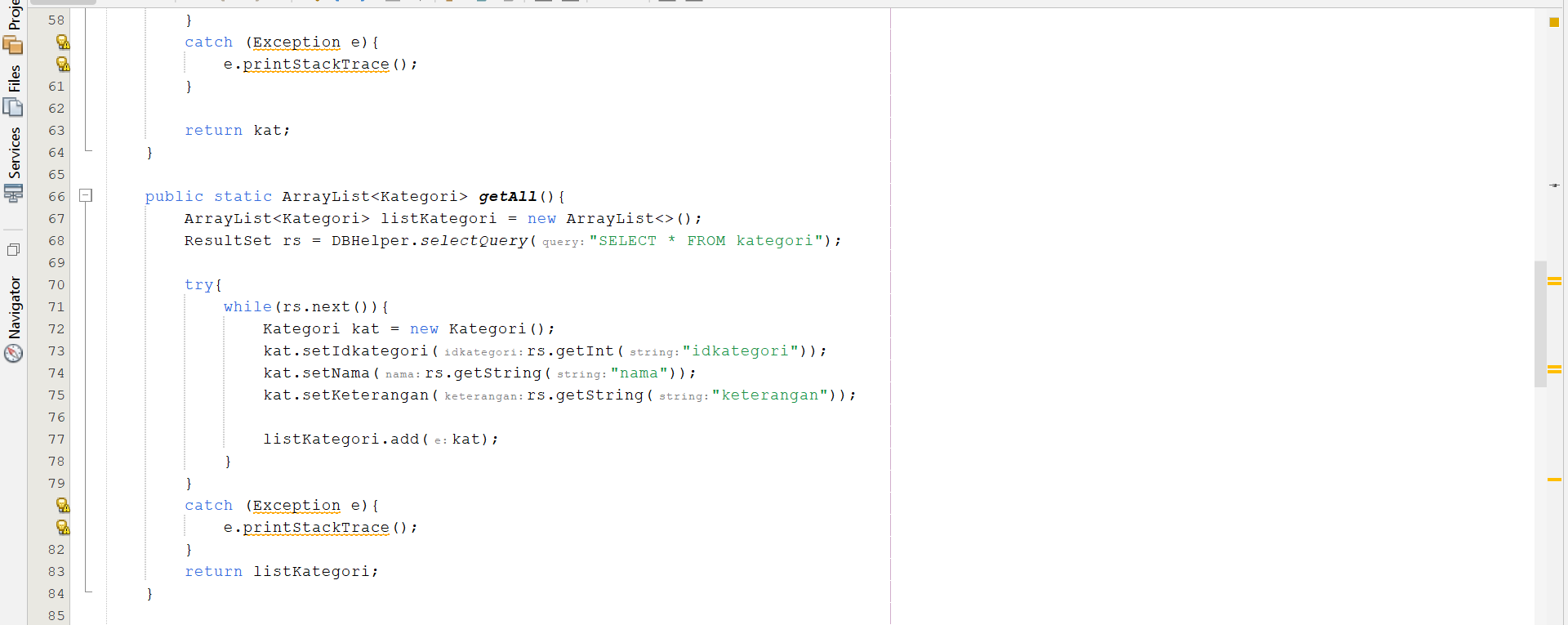




## Percobaan 4

Membuat class **Kategori** untuk menghandle CRUD pada tabel kategori.

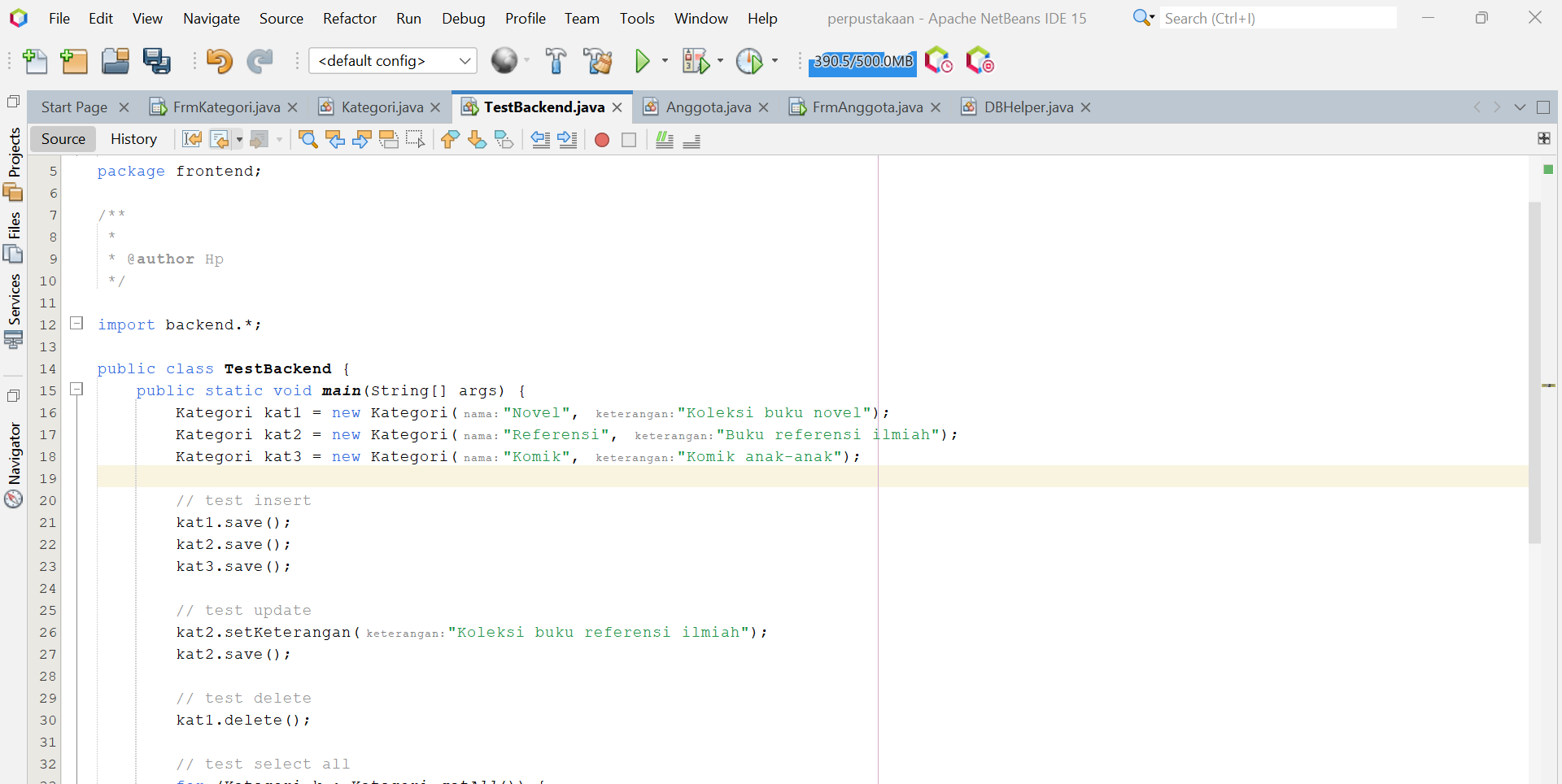


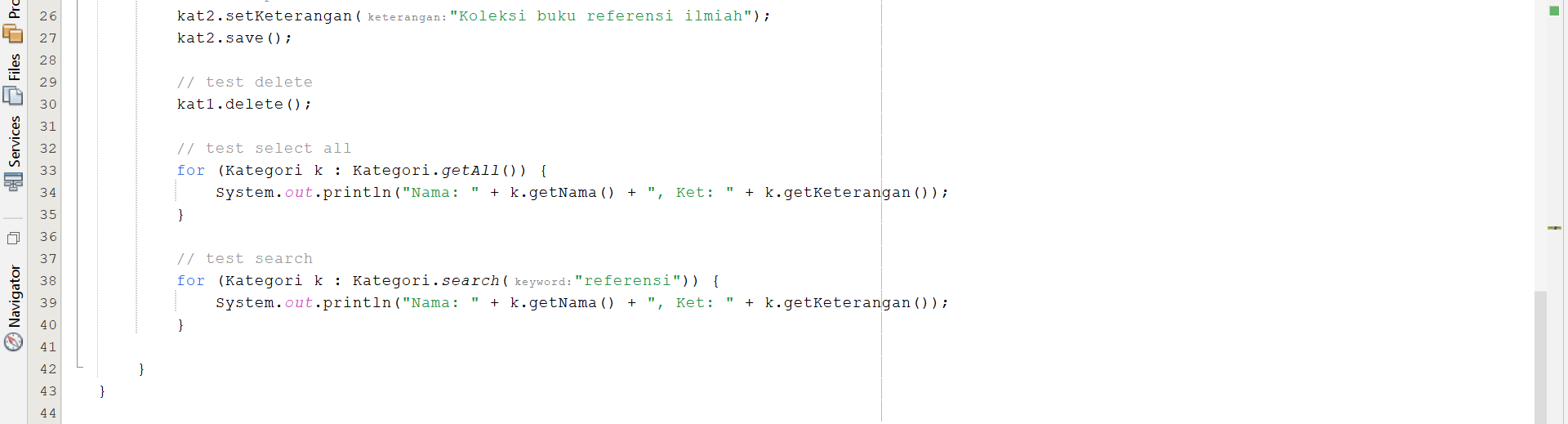


## Percobaan 5

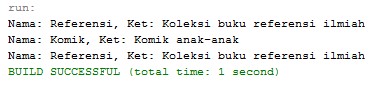
Mencoba backed yang sudah dibuat dengan mengoperasikannya lewat frontend berbasis teks (console). Percobaan ini dapat anda skip jika anda telah yakin bahwa backend yang anda buat sudah berfungsi dengan baik.

Code class TestBackend



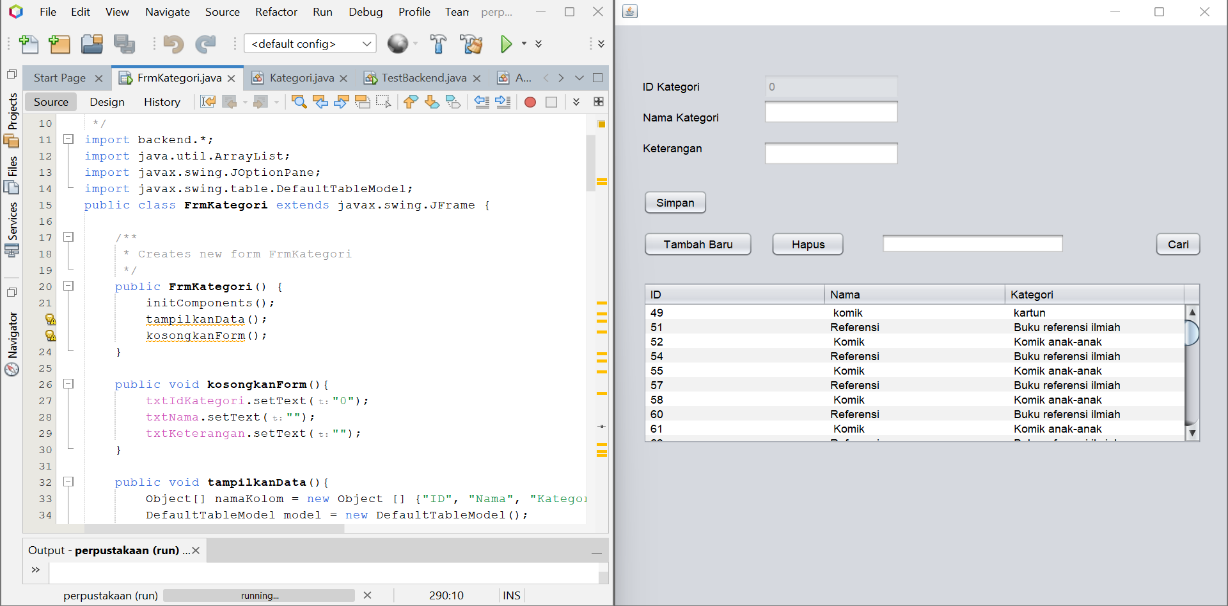


3. Jalankan TestBackend dengan klik kanan, Run File. Cocokkan outputnya:



## Percobaan 6

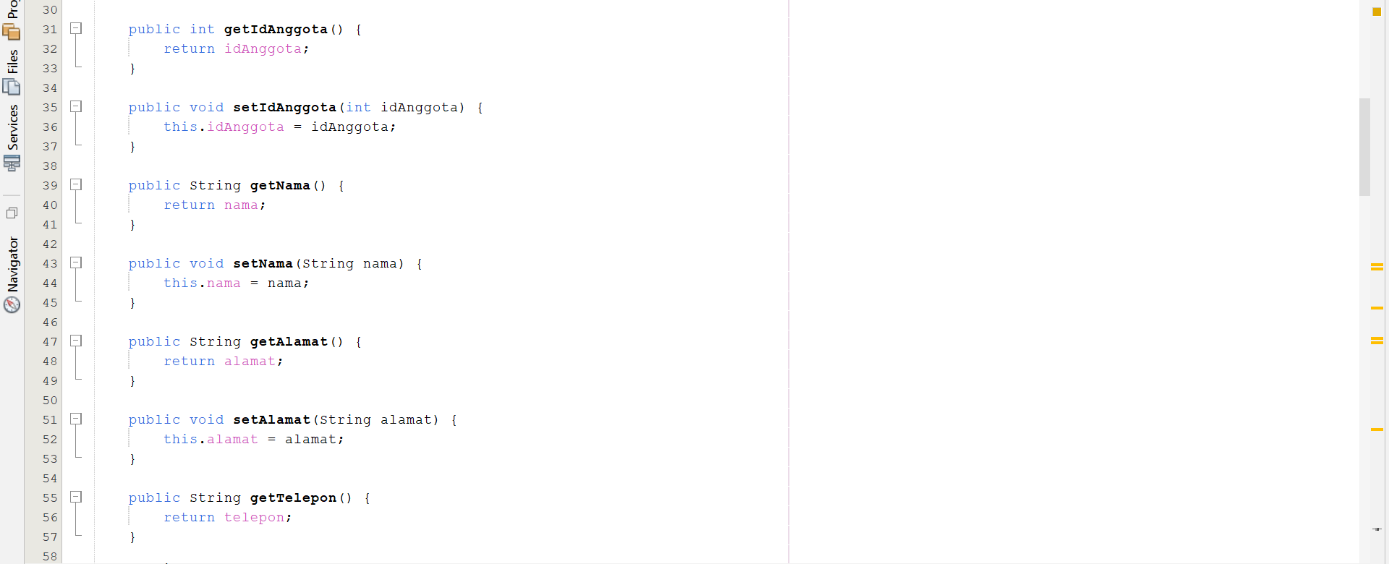
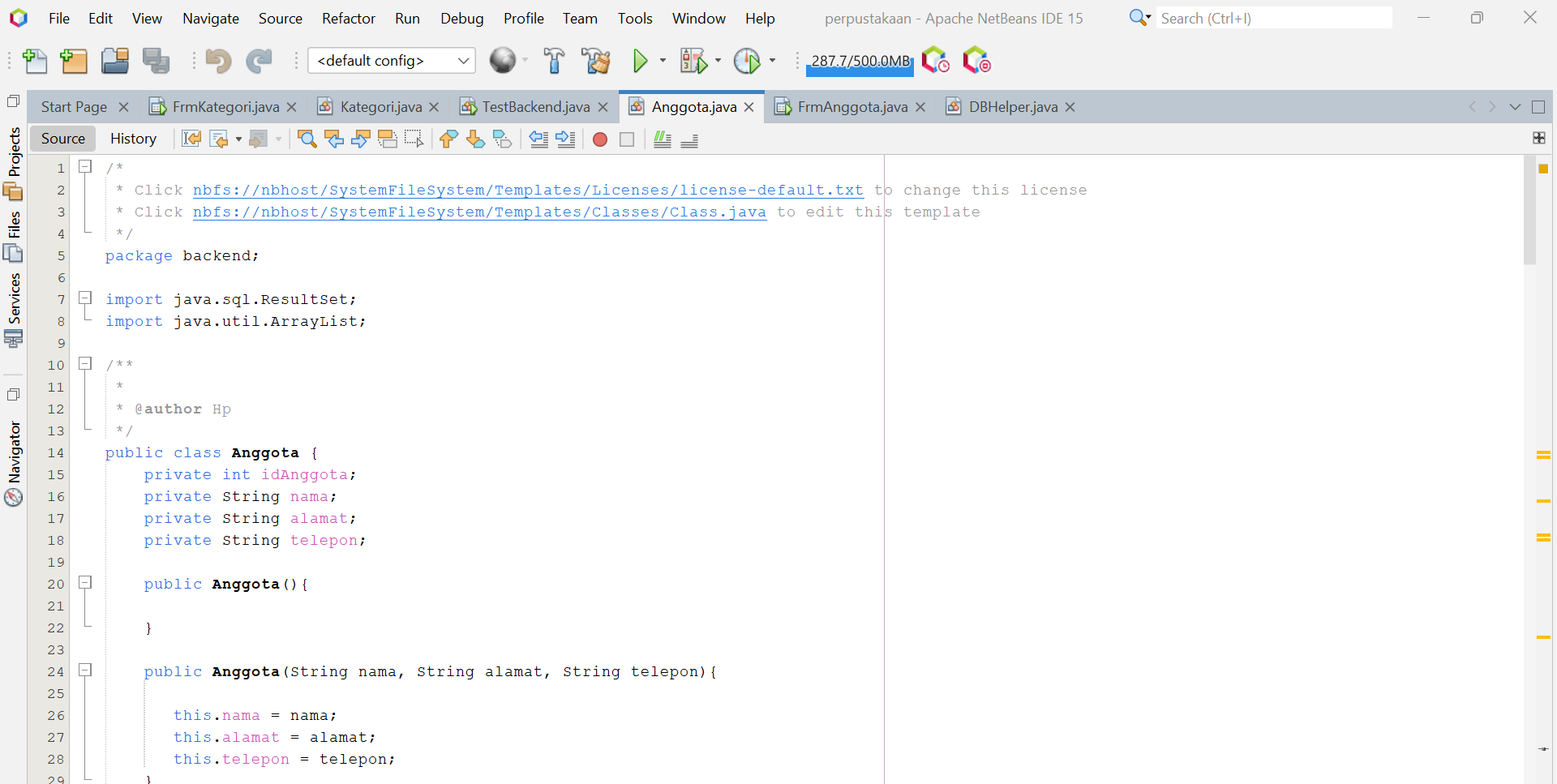
Pada percobaan ini kita akan membuat interface GUI untuk class **Kategori**.



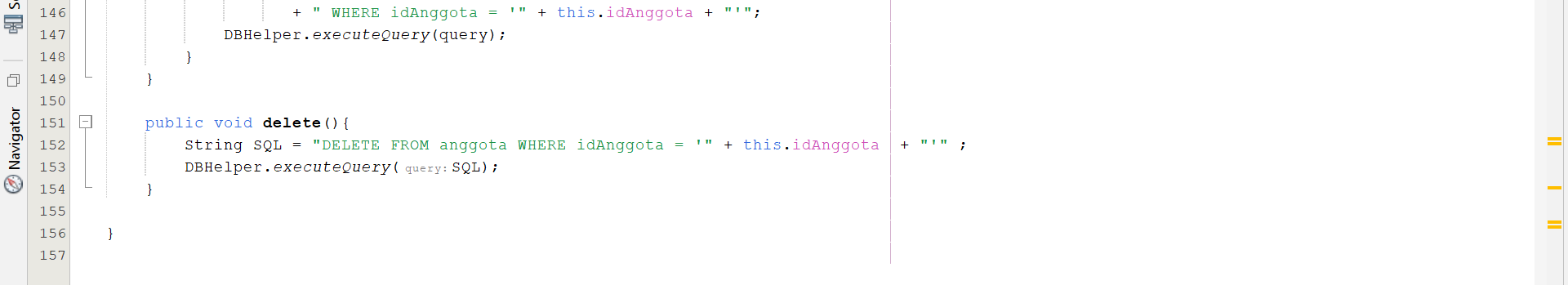
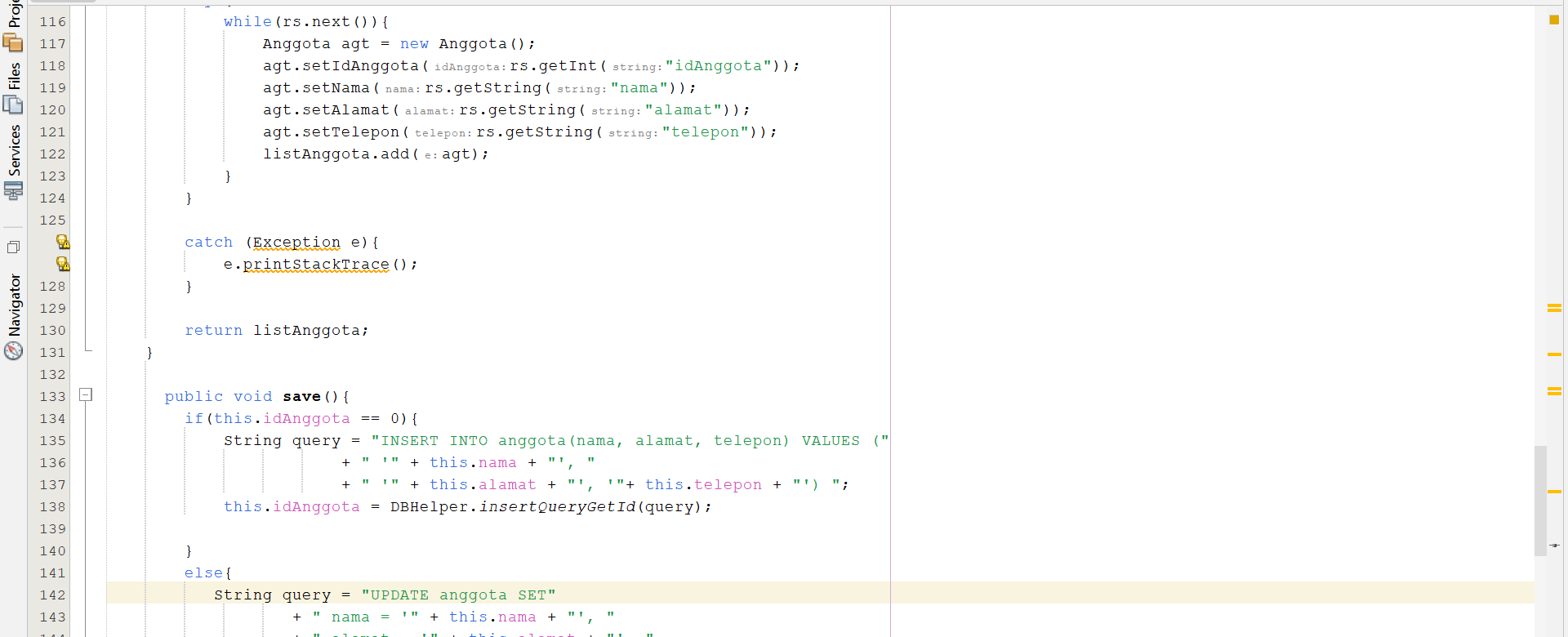
## Percobaan 7

Lakukan hal yang sama untuk data **Anggota!**

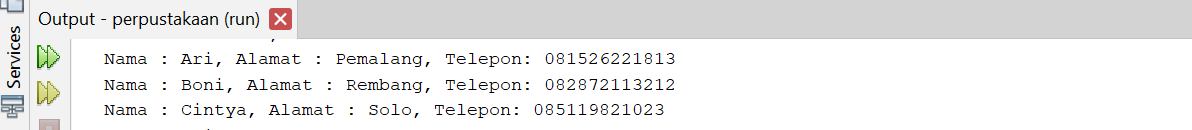
1. Buat class **Anggota** pada package **backend**, lengkapi atribut dan method-nya.







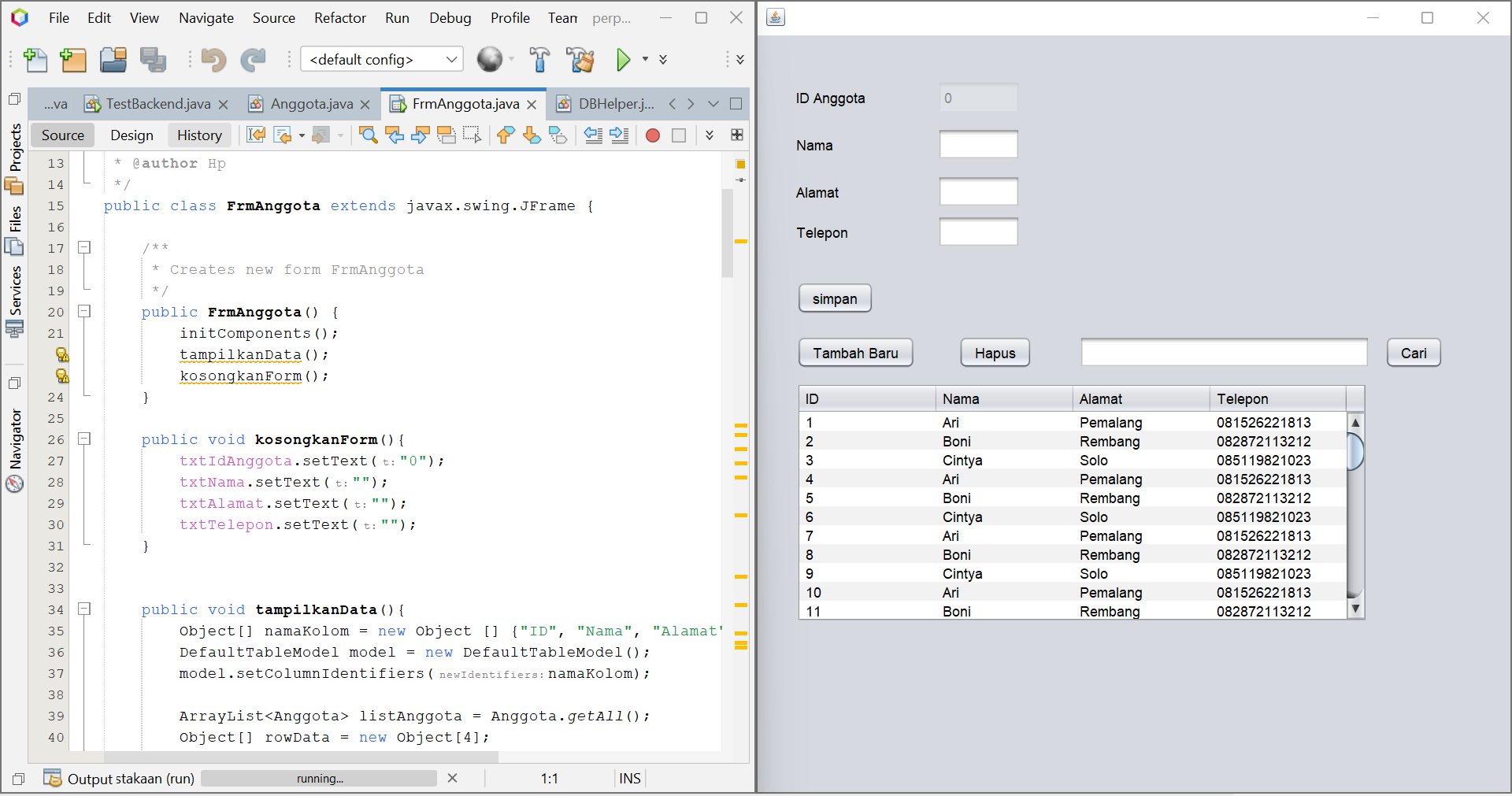
1. Lakukan test pada class TestBackend pada package **frontend**.



## Percobaan 8

Buat form untuk data **Anggota**.

1. Buat **FrmAnggota** pada package **frontend** dan lengkapi komponen, method serta event-nya.



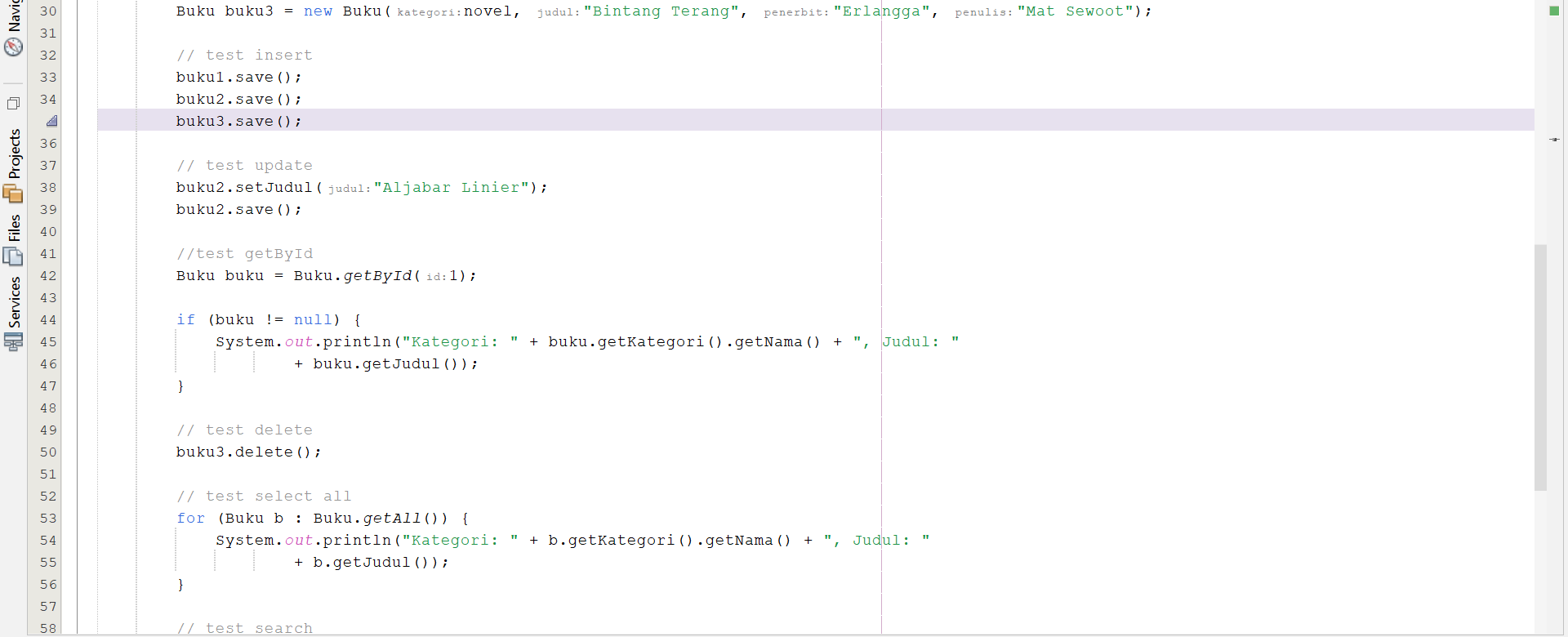
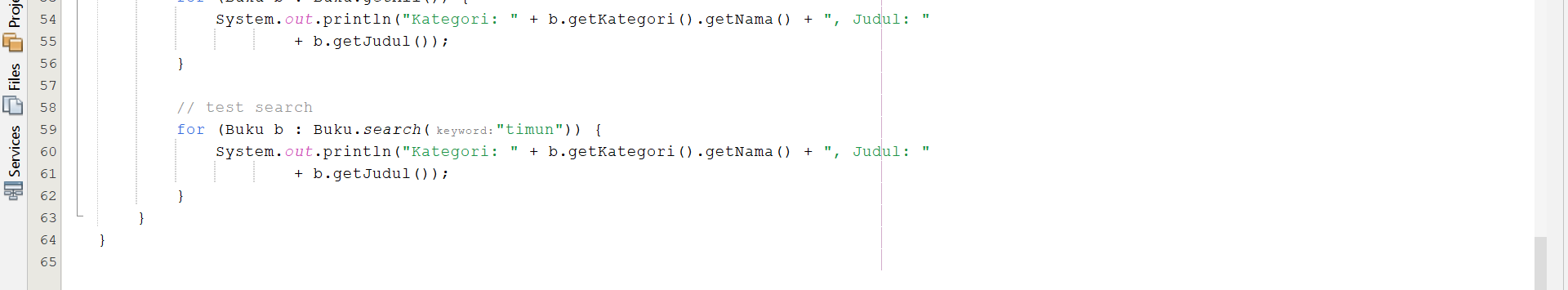
## Percobaan 9

Untuk data **Buku**, caranya kurang lebih sama seperti data Kategori dan Anggota. Hanya saja yang berbeda adalah:

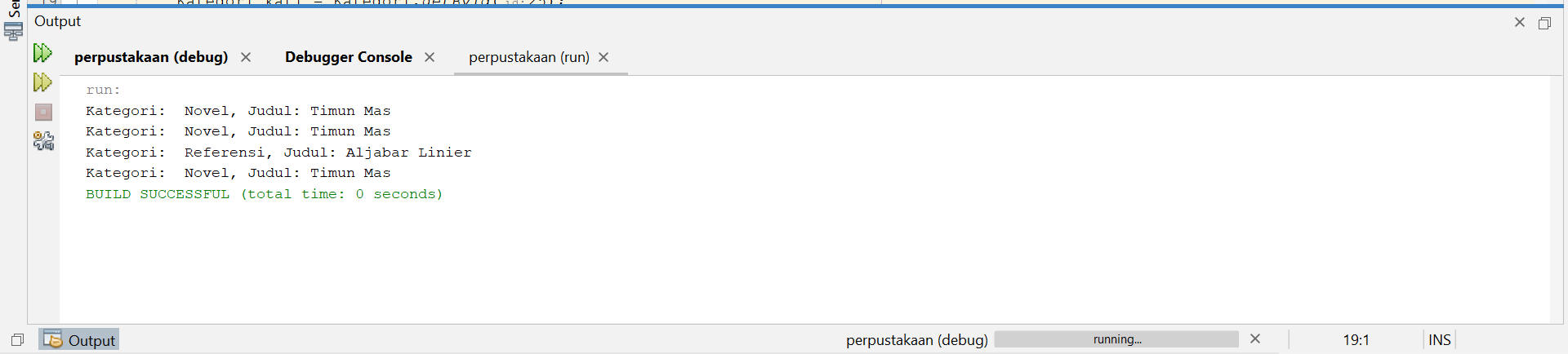
1. Pemanggilan **getKategori().getIdKategori()** pada query insert dan update untuk mengeset **idkategori** pada tabel **buku**
2. Query select yang melibatkan join table pada method getById(), getAll() dan search().

Kode lengkap class Buku dapat anda lihat di **Lampiran 1**. Untuk test buku pada **frontend**, bisa anda lihat di **Lampiran 2.**





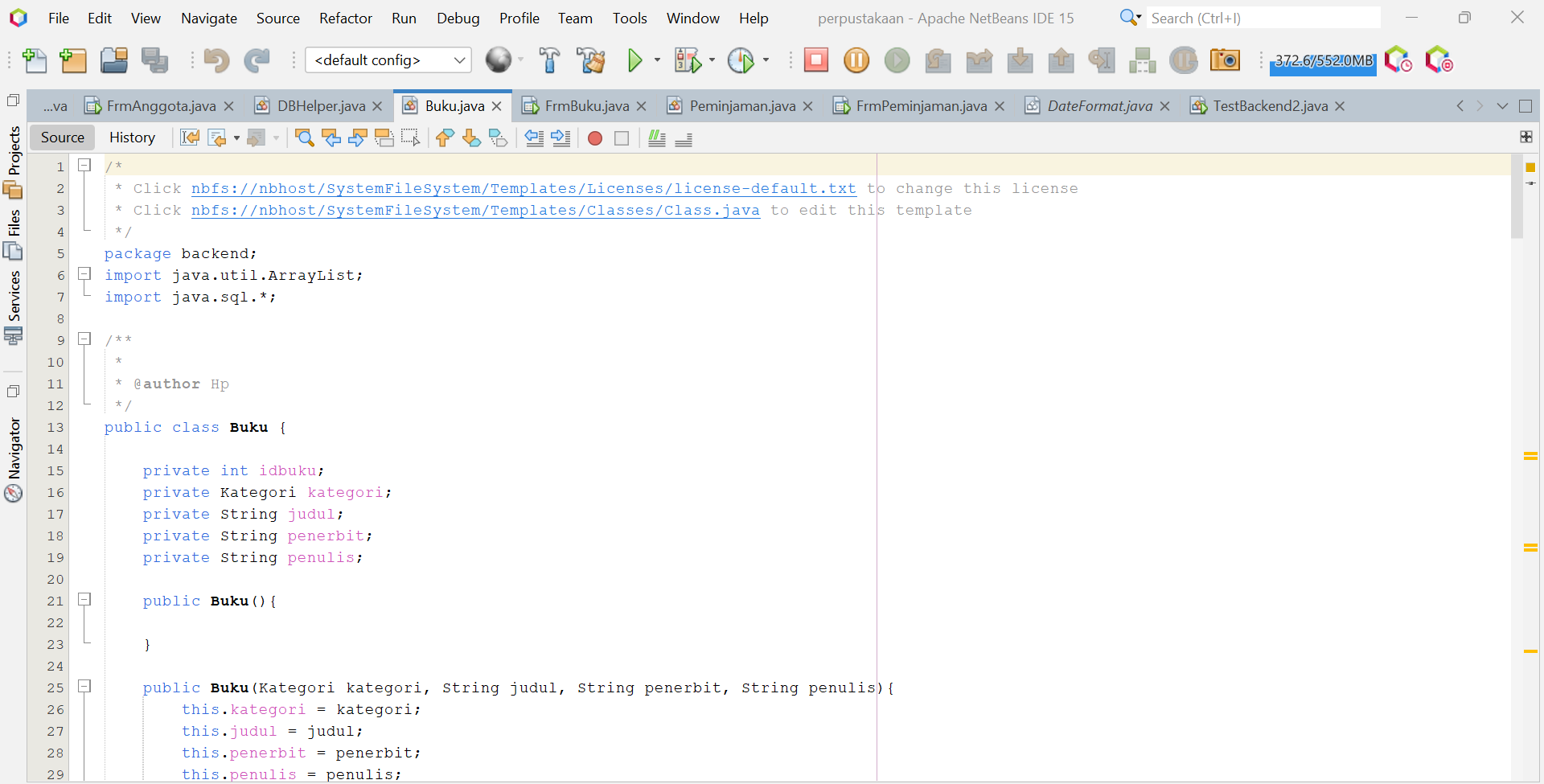
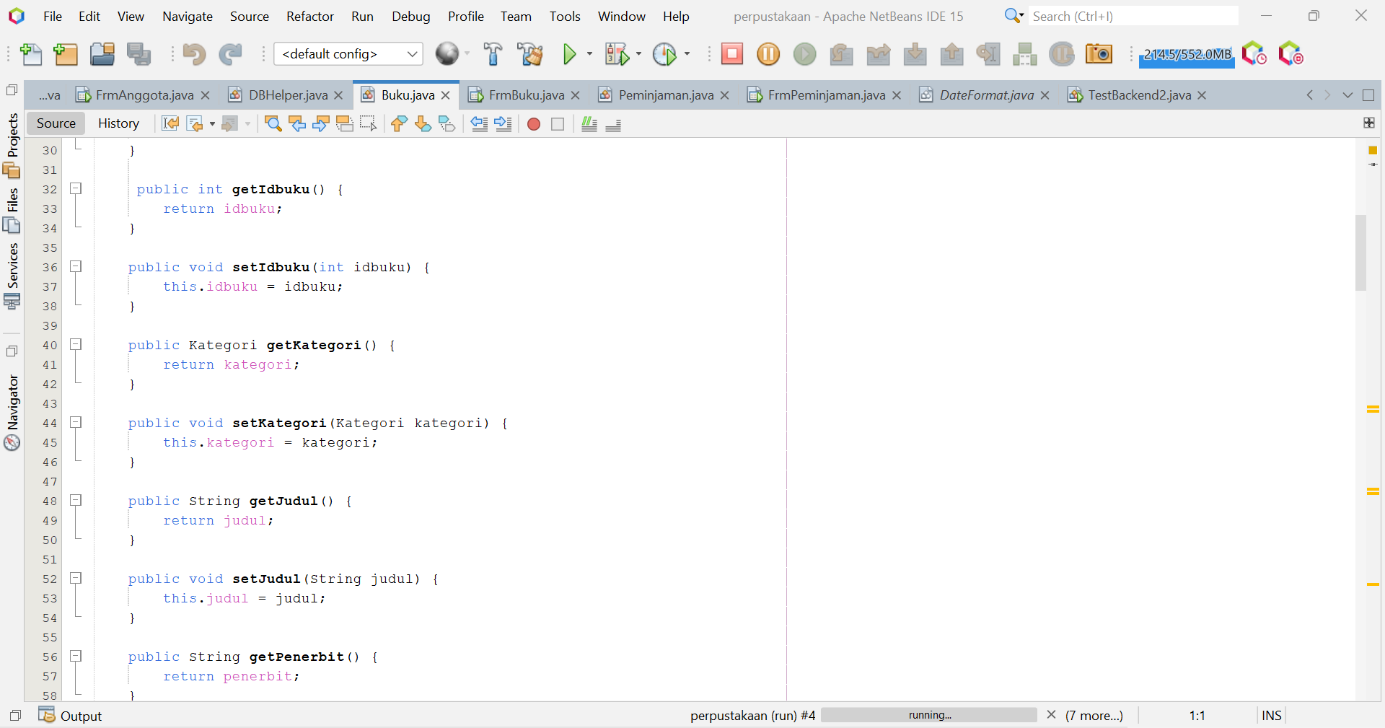
Hasil running :

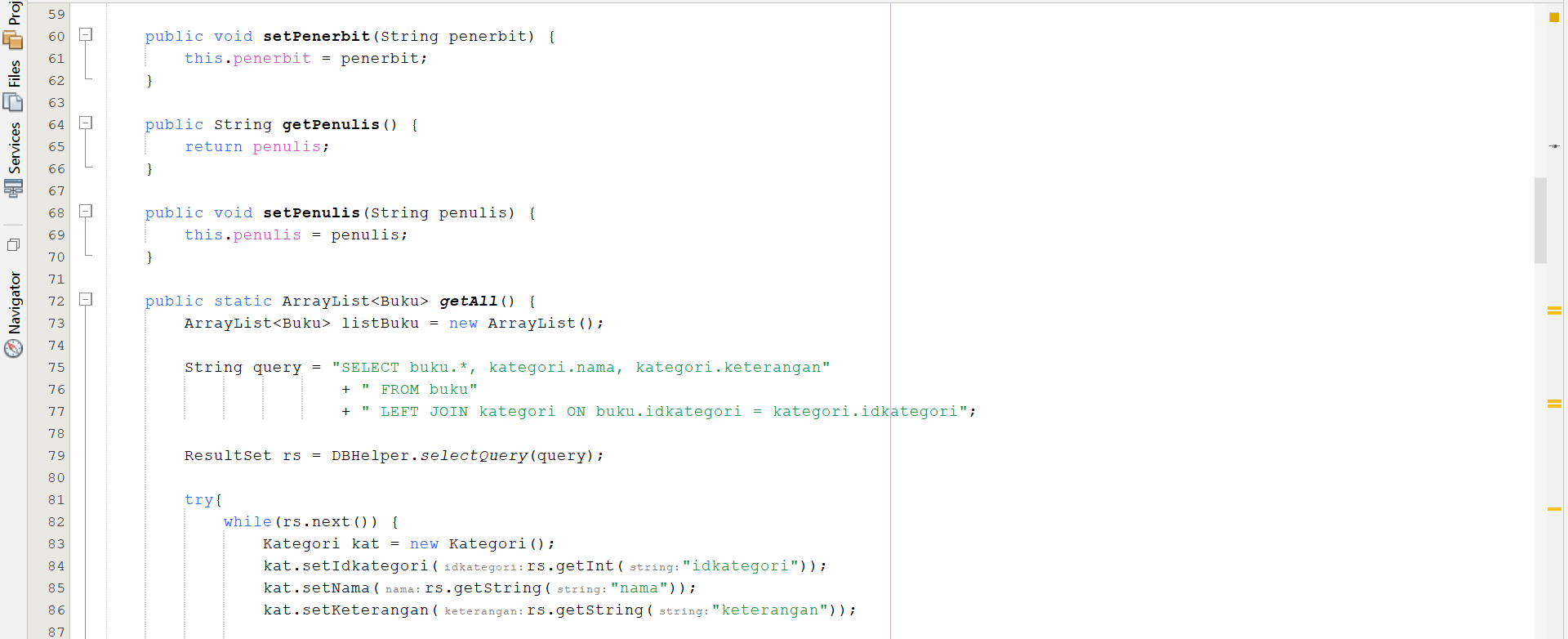


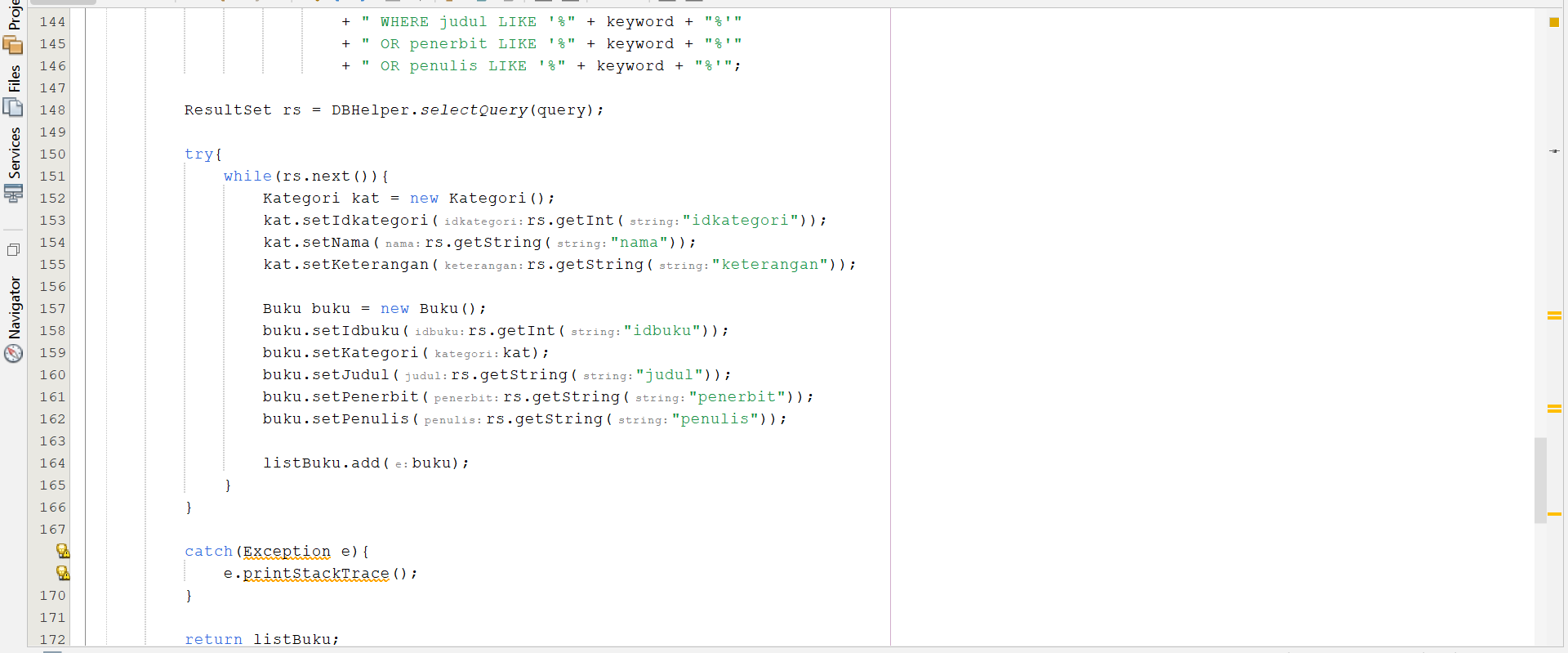
## Percobaan 10

Membuat GUI untuk data Buku, yang dilengkapi dengan combo box untuk memilih kategori yang terhubung dengan tabel kategori.

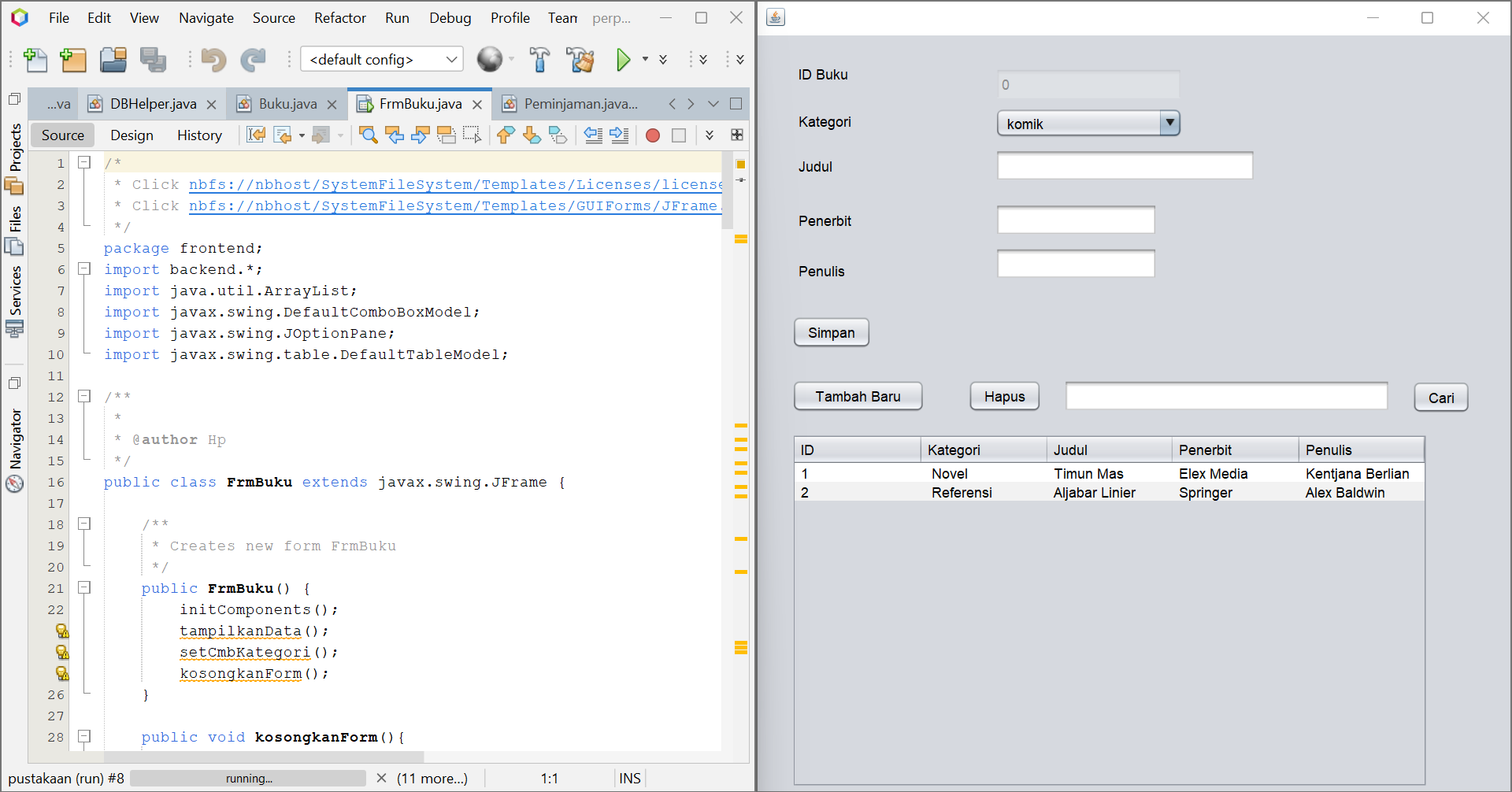
Code class Buku





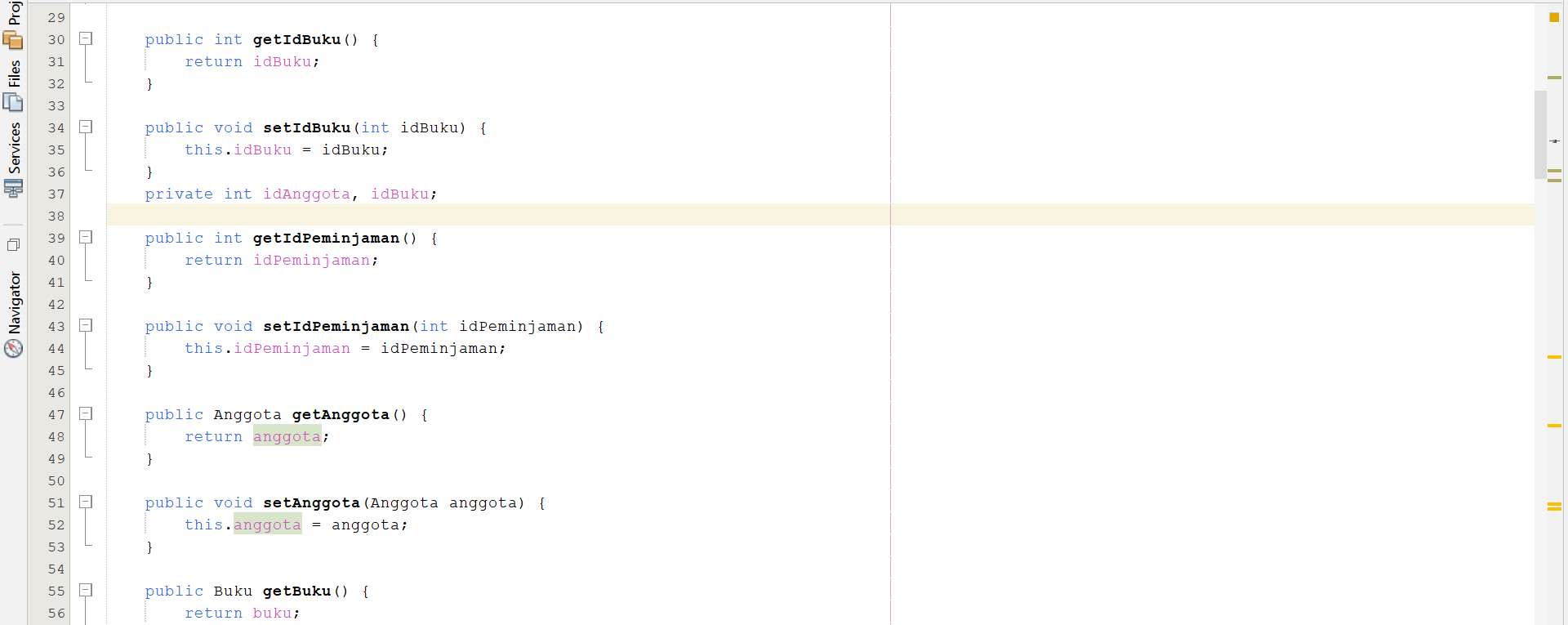
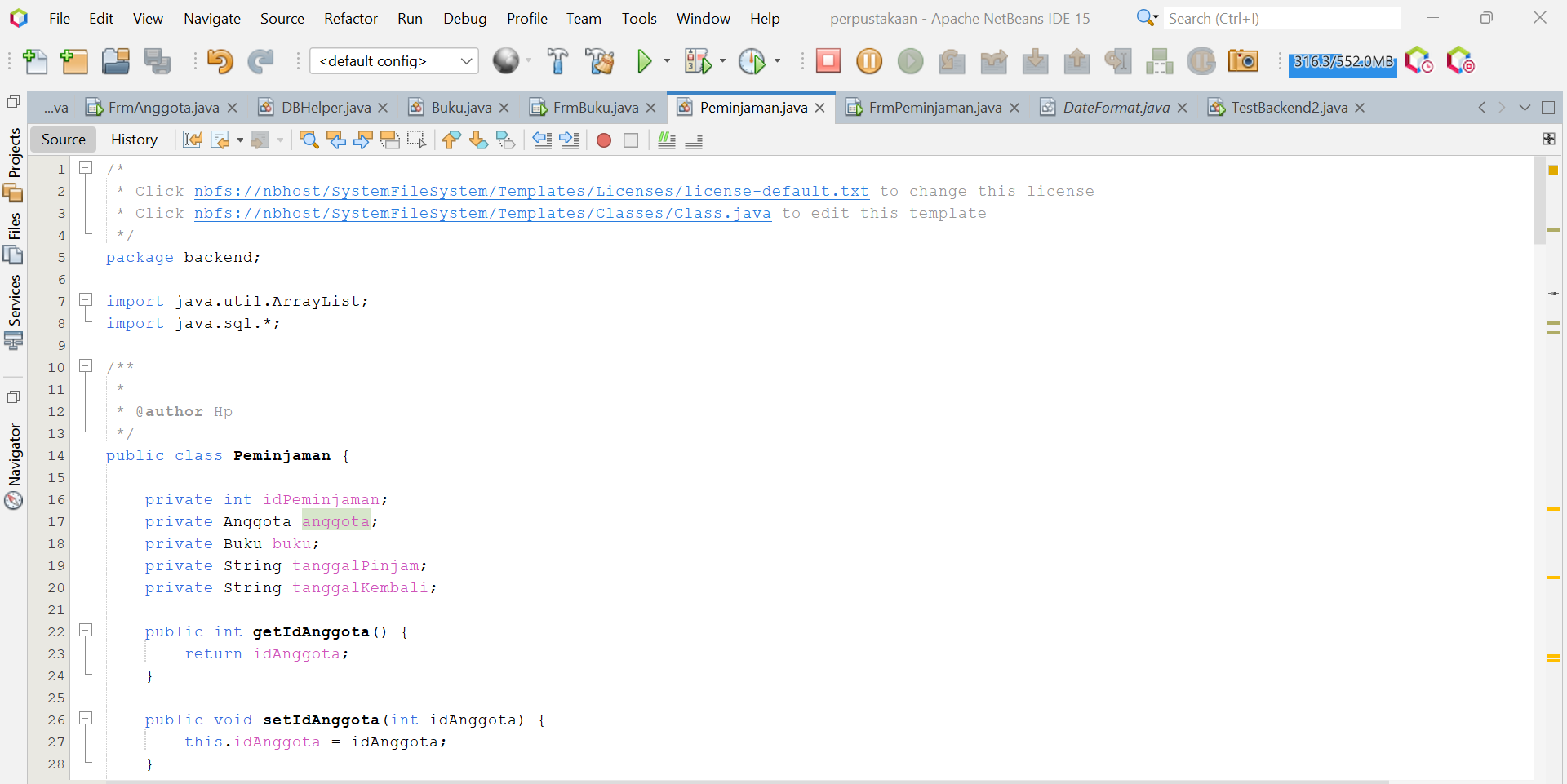


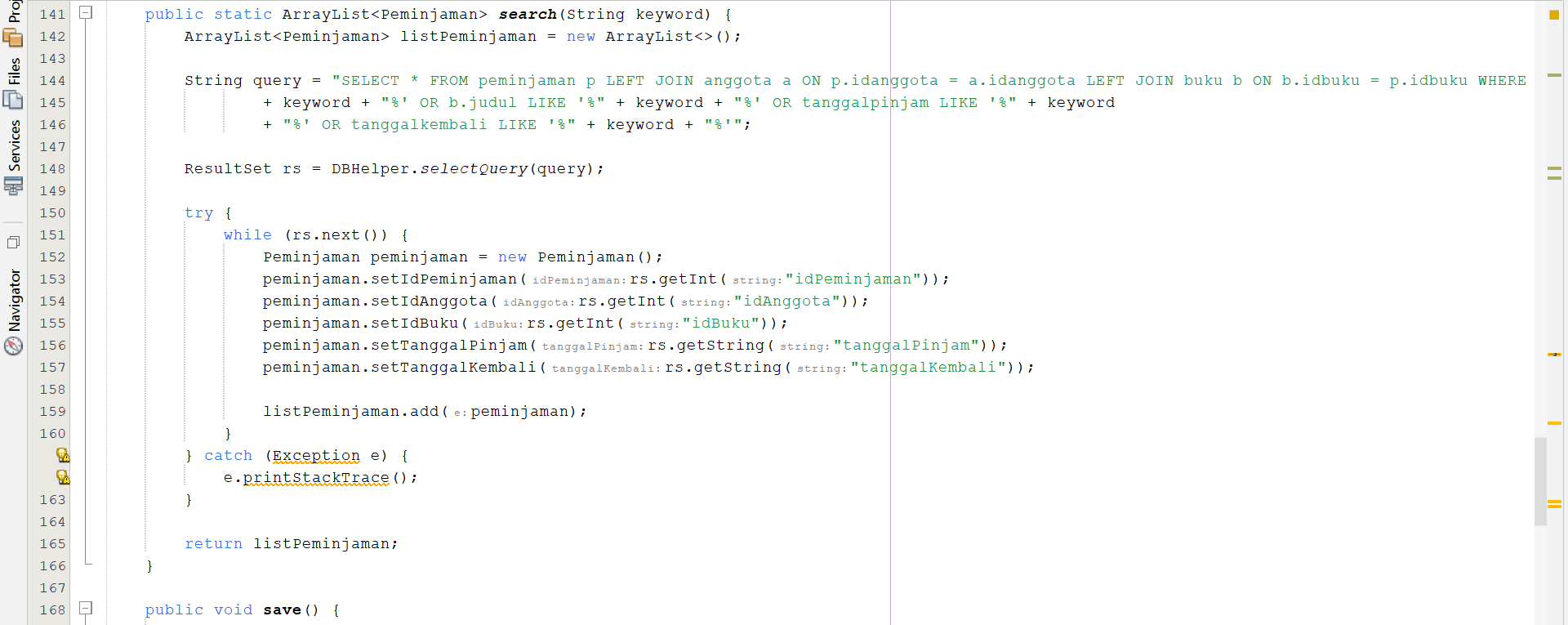
1. Pada package **frontend**, buat jFrame FrmBuku. Susun formnya sebagai berikut:



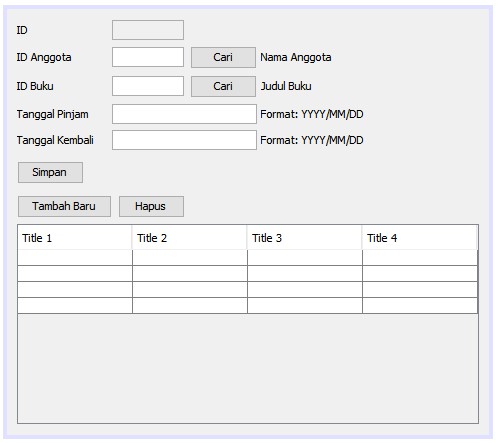
# Tugas

1. Buatlah class **Peminjaman.**





1. Buatlah form **FrmPeminjaman** dan susun sebagai berikut:



1. Atur kode program agar dapat menangani transaksi peminjaman dan pengembalian.

Note:

Pada textbox ID Anggota, pengguna tinggal memasukkan ID anggota, kemudian menekan tombol Cari. Jika ketemu, maka label **“Nama Anggota”** yang ada di samping tombol Cari tersebut akan menampilkan nama anggota dari ID yang dimasukkan tadi. Begitu juga dengan ID Buku.

**Hasil running FrmPeminjaman**

