**TUGAS PRAKTIKUM PEMOGRAMAN II**



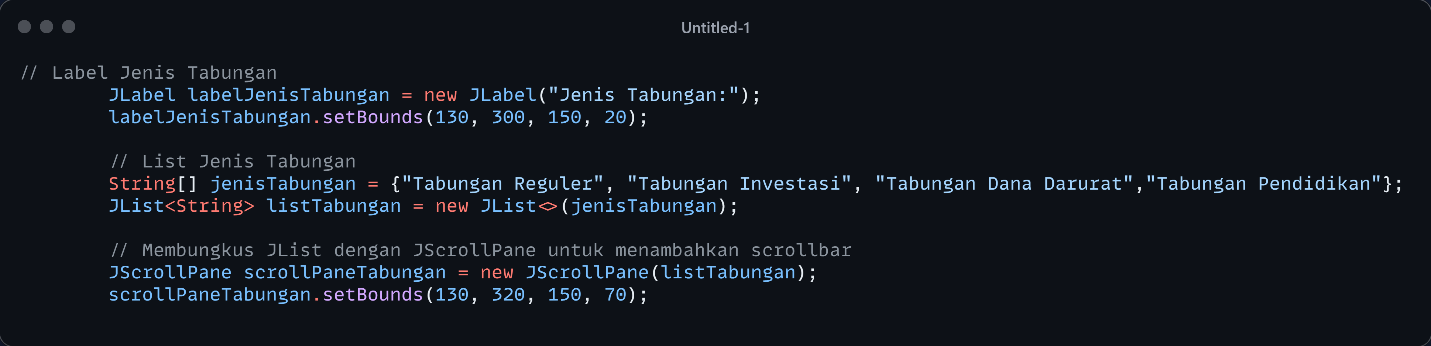
Novan Ramdan (223040031)

**FAKULTAS TEKNIK**

**UNIVERSITAS PASUNDAN**

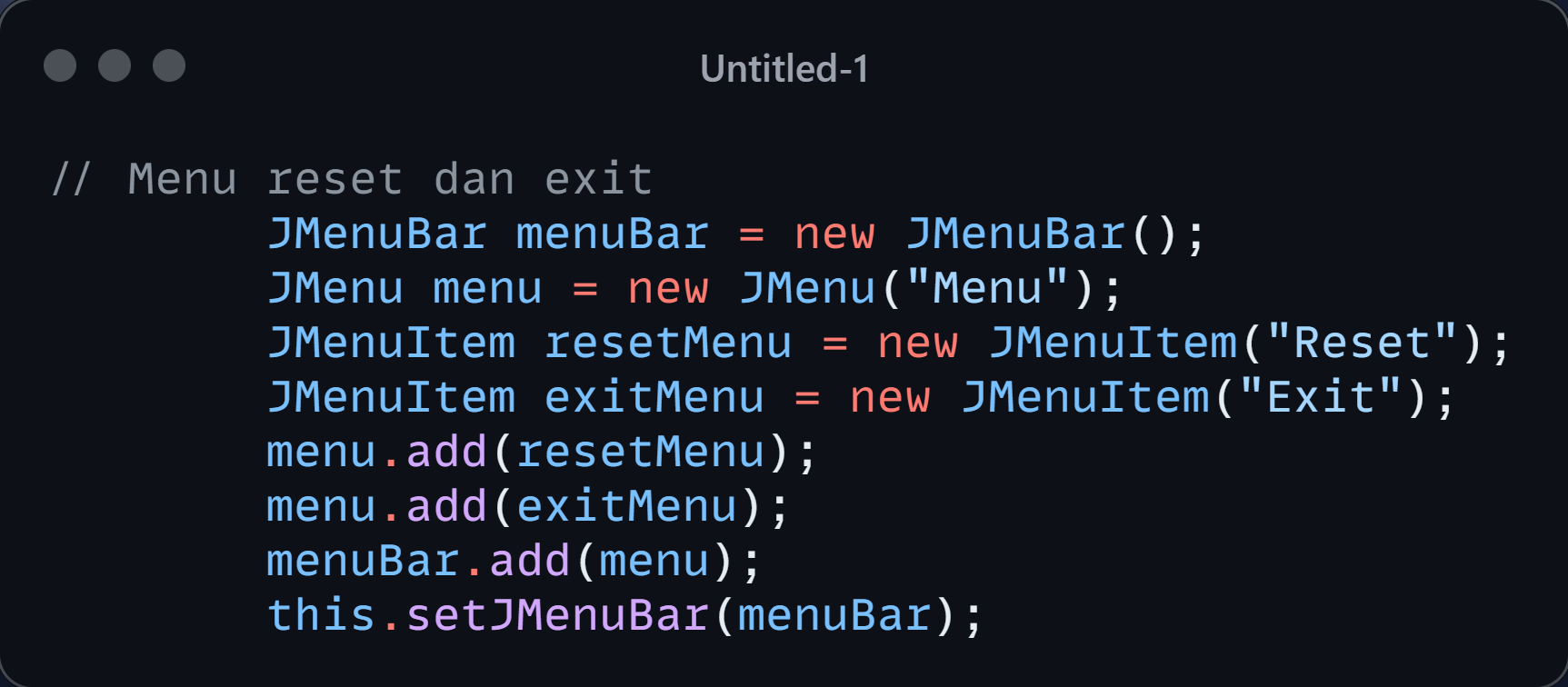
**BANDUNG 2024**

1. Tambahkan inputan untuk memilih jenis tabungan dengan menggunakan JList (minimal ada 4 pilihan tabungan)

****

* String[] jenisTabungan: Array yang menyimpan beberapa pilihan jenis tabungan. Dalam hal ini, terdapat empat pilihan: "Tabungan Reguler", "Tabungan Investasi", "Tabungan Dana Darurat", dan "Tabungan Pendidikan".
* JList<String> listTabungan: Membuat komponen list (daftar)yang berguna untuk memilih salah satu item
* JScrollPane: komponen yang digunakan untuk menambahkan kemampuan scrollbar (gulir) ke komponen lain.

1. Tambahkan menu yang minimal berisi menu Reset (menghapus semua inputan dan output) dan Exit (keluar dari aplikasi menggunakan kode System.exit(0);)



* Menu reset dan exit

1. JMenuBar adalah komponen yang digunakan untuk menambahkan menu bar di bagian atas jendela GUI.
2. JMenu adalah menu individual yang akan ditambahkan ke dalam menu bar.
3. JMenuItem adalah item pilihan dalam menu.
4. resetMenu: Opsi menu dengan label "Reset". Ketika diklik, membersihkan semua input dan output di aplikasi.
5. exitMenu: Opsi menu dengan label "Exit". Ketika diklik, ini akan memicu aksi keluar dari aplikasi dengan menggunakan System.exit(0).
6. Metode menu.add() digunakan untuk menambahkan item ke dalam menu.Item resetMenu dan exitMenu ditambahkan ke menu "Menu".
7. menuBar.add() menambahkan menu "Menu" ke dalam JMenuBar yang dibuat di awal.
8. this.setJMenuBar(menuBar): Bagian ini menghubungkan JMenuBar dengan jendela (JFrame) utama aplikasi. Menu bar ini akan tampil di bagian atas jendela GUI.



* resetMenu.addActionListener(): Menambahkan event listener untuk menangani aksi saat item Reset di menu diklik.
* Dalam metode actionPerformed(), berbagai komponen direset:

textFieldNama, textFieldNomor, passwordField, confirmPasswordField dihapus (diisi dengan string kosong "").

txtOutput dihapus (kosongkan area teks).

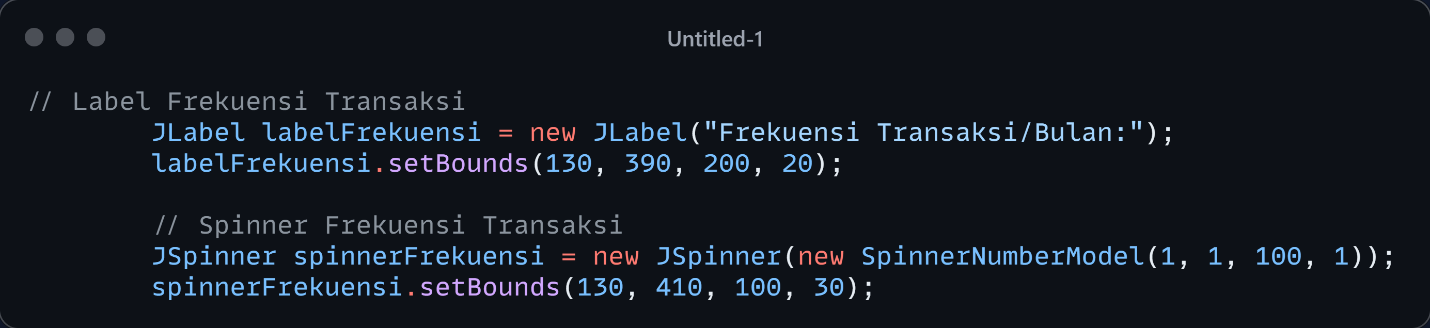
checkBox di-set ke tidak terpilih.

listTabungan dibersihkan (menghilangkan pilihan).

spinnerFrekuensi direset ke nilai default 1.

* exitMenu.addActionListener(): Menambahkan event listener untuk menangani aksi saat item Exit di menu diklik.
* System.exit(0): Keluar dari aplikasi dengan status kode 0, yang menunjukkan penghentian normal.

1. Tambahkan inputan jumlah frekuensi transaksi per bulan menggunakan JSpinner (1-100)



* + JSpinner adalah komponen yang memungkinkan pengguna untuk memilih atau mengubah nilai numerik menggunakan panah naik/turun.
  + new SpinnerNumberModel(1, 1, 100, 1) adalah model spinner yang menentukan batasan nilai:

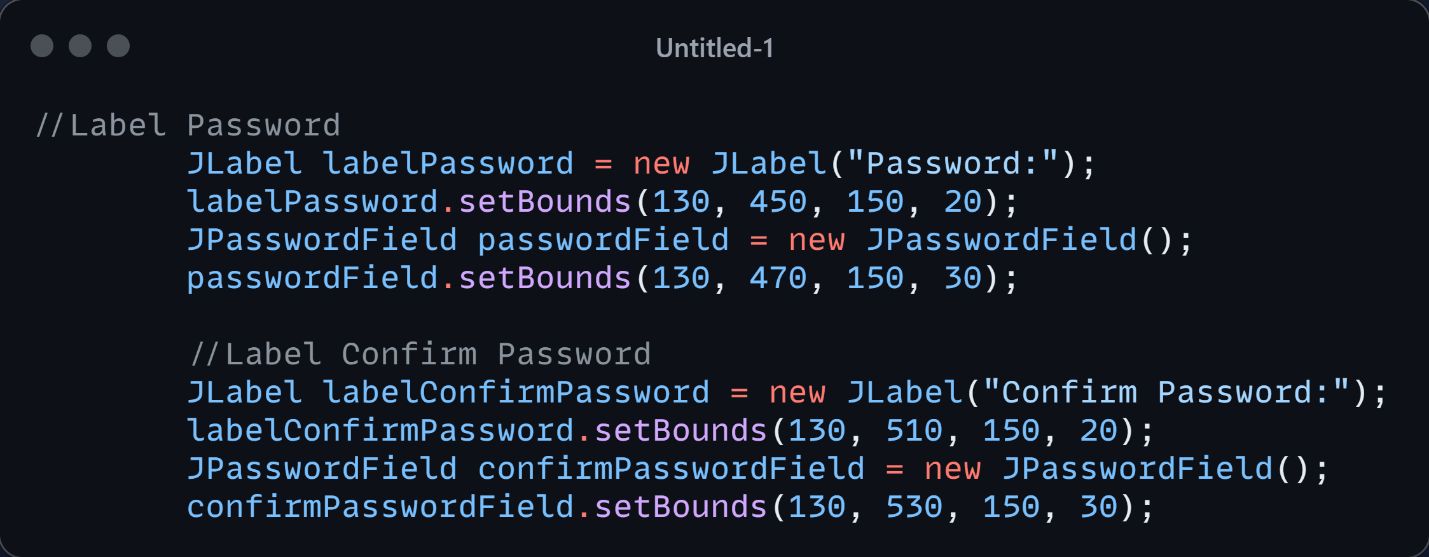
**1**: Nilai awal.

**1**: Nilai minimum.

**100**: Nilai maksimum.

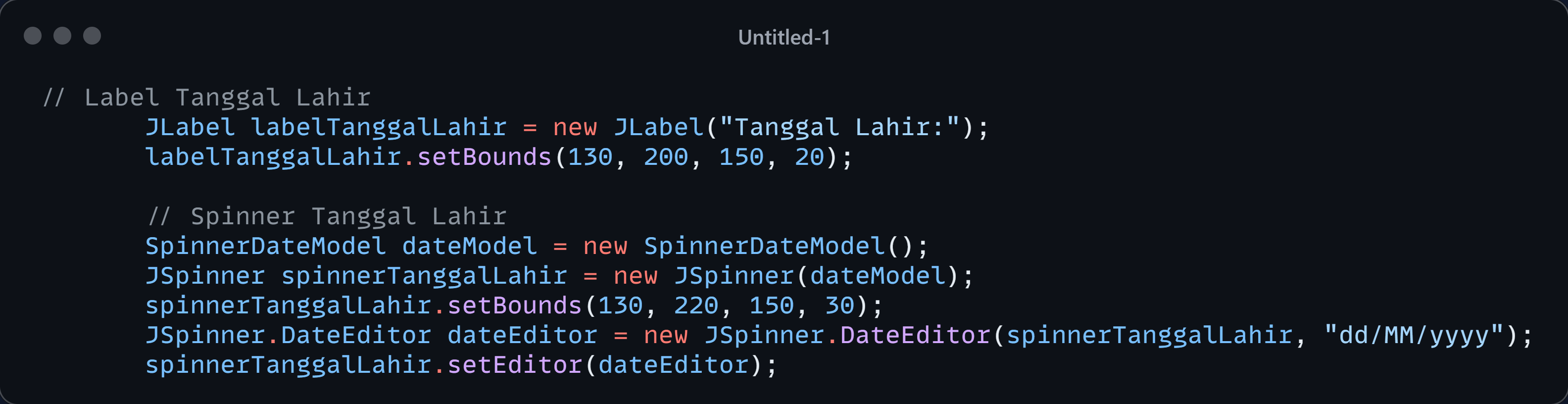
**1**: Inkrement, yaitu nilai yang akan bertambah/berkurang ketika pengguna menekan panah atas/bawah.

1. Tambahkan inputan password dan confirm password menggunakan JPasswordField (password tidak ditampilkan di output JTextArea, hanya ada output apakah password dan confirm password cocok atau tidak.



* JPasswordField adalah komponen input tetapi dengan tampilan karakter yang disembunyikan (biasanya ditampilkan sebagai titik atau bintang) agar pengguna dapat memasukkan password dengan aman.
* JLabel "Confirm Password:" digunakan untuk memberi petunjuk bahwa pengguna harus mengonfirmasi password dengan mengisi kolom di bawahnya.
* JPasswordField kedua digunakan untuk meminta pengguna memasukkan ulang password sebagai konfirmasi.

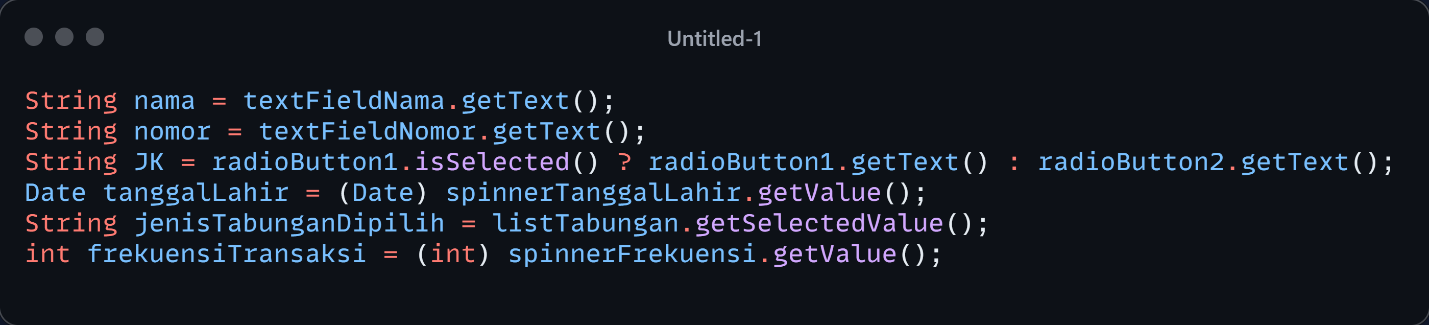
1. Tambahkan inputan tanggal lahir dengan menggunakan JSpinner.



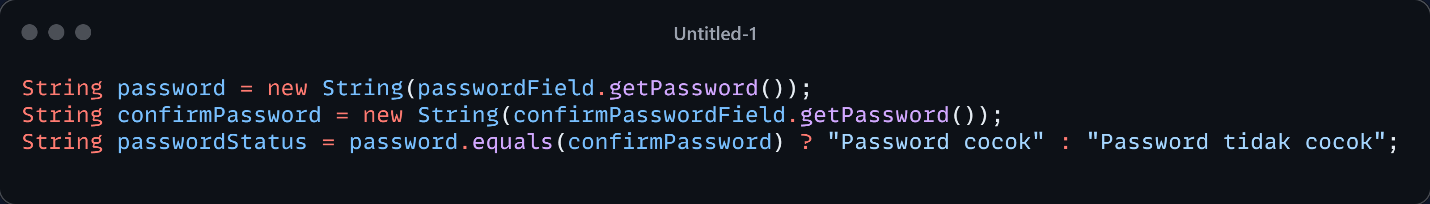
* SpinnerDateModel dateModel = new SpinnerDateModel();: Membuat SpinnerDateModel, yang berguna untuk memilih tanggal (sebagai nilai tanggal dalam spinner).
* JSpinner spinnerTanggalLahir = new JSpinner(dateModel);: Membuat JSpinner yang menggunakan SpinnerDateModel ini untuk menampilkan spinner yang dapat digunakan untuk memilih tanggal.
* JSpinner.DateEditor digunakan untuk menentukan format tampilan tanggal dalam spinner. Format yang digunakan adalah "dd/MM/yyyy",
* spinnerTanggalLahir.setEditor(dateEditor);: Mengatur editor dari spinnerTanggalLahir menjadi DateEditor dengan format yang telah ditentukan.

1. Bagian kode ini adalah aksi setelah kita menekan tombol simpan:





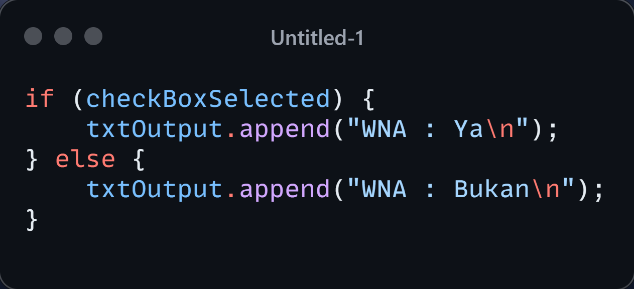
* nama dan nomor: Mendapatkan input teks dari textFieldNama dan textFieldNomor yang dimasukkan oleh pengguna.
* JK (Jenis Kelamin): Memeriksa apakah radioButton1 (Laki-laki) atau radioButton2 (Perempuan) yang dipilih, dan menyimpan teks yang sesuai.
* tanggalLahir: Mendapatkan nilai tanggal yang dipilih dari spinnerTanggalLahir.
* jenisTabunganDipilih: Mendapatkan pilihan yang dipilih pengguna dari JList yang berisi jenis tabungan.
* frekuensiTransaksi: Mendapatkan jumlah transaksi bulanan yang dipilih oleh pengguna dari JSpinner.



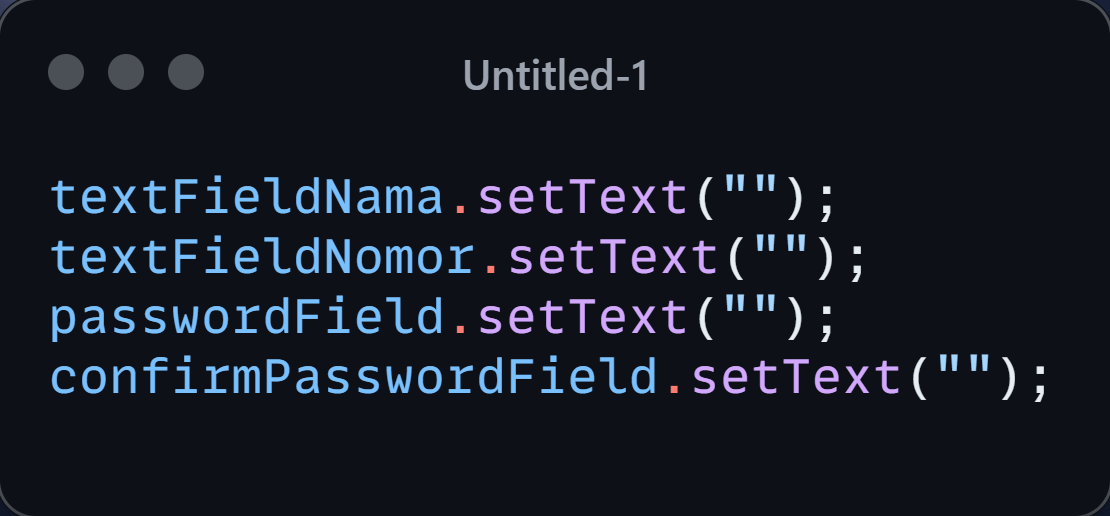
* password dan confirmPassword: Mendapatkan password dan konfirmasi password yang diinput pengguna dari JPasswordField.
* passwordStatus: Membandingkan kedua password apakah cocok. Jika cocok, menampilkan "Password cocok", jika tidak, menampilkan "Password tidak cocok".



* txtOutput.setText(""): Mengosongkan JTextArea sebelum menampilkan output baru.
* txtOutput.append(...): Menambahkan informasi yang dikumpulkan dari input pengguna ke txtOutput, termasuk nama, nomor telepon, jenis kelamin, tanggal lahir (diformat menggunakan dateEditor), jenis tabungan yang dipilih, frekuensi transaksi bulanan, dan status password.



* Memeriksa apakah checkbox untuk Warga Negara Asing dipilih atau tidak, dan menampilkan status "Ya" jika dipilih atau "Bukan" jika tidak.



* Setelah data ditampilkan, semua input pengguna di-reset atau dikosongkan untuk mempersiapkan input baru. Kolom nama, nomor, password, dan konfirmasi password dikosongkan.

Link Github: <https://github.com/NovanR/TugasPP2>