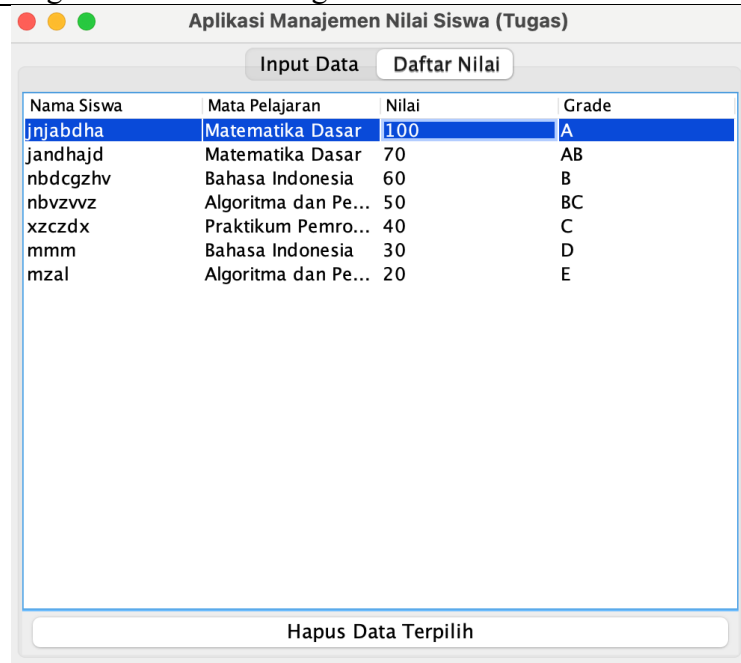


Bagian 2. Penjelasan Kode

Kode ini mengambil nilai angka (misal: 85), membaginya dengan 10. Karena tipe datanya integer, hasil desimal dibuang ($85 / 10 = 8$). Switch kemudian mengecek angka 8 tersebut dan memberikan grade "A".

Bagian 3. Hasil Running



Nama Siswa	Mata Pelajaran	Nilai	Grade
jnjabdha	Matematika Dasar	100	A
jandhajt	Matematika Dasar	70	AB
nbdcgzhv	Bahasa Indonesia	60	B
nbvzvzv	Algoritma dan Pe...	50	BC
xzczdx	Praktikum Pemro...	40	C
mmm	Bahasa Indonesia	30	D
mzal	Algoritma dan Pe...	20	E

Hapus Data Terpilih

Bagian 4. Penjelasan Hasil Running

Saat pengguna memasukkan nilai (misalnya "75") dan menekan Simpan, sistem menghitung di latar belakang. Pada tabel di tab "Daftar Nilai", kolom Grade akan otomatis terisi "AB". Secara visual bagi pengguna tidak ada bedanya dengan if-else, namun secara kode ini lebih terstruktur untuk logika diskrit.

Instruksi 2

Perintah Tugas:

Tambah tombol hapus di panel bagian bawah Tab Daftar Nilai. Anda dapat menggunakan method `tableData.getSelectedRow()` untuk mengetahui baris mana yang diklik pengguna. Jika hasilnya di atas -1 artinya ada yang dipilih, kemudian gunakan method `tableModel.removeRow(indeks)` untuk menghapusnya.

CPMK Terkait:

Kode	Uraian
IF21W0508-CPMK04	Mampu mengkontruksi solusi berbasis komputing menggunakan kakas pemrograman berorientasi objek

Bagian 1. Screenshot Semua Kode

```

JButton btnHapus = new JButton("Hapus Data Terpilih");
panel.add(btnHapus, BorderLayout.SOUTH);

btnHapus.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {

        int selectedRow = tableData.getSelectedRow();

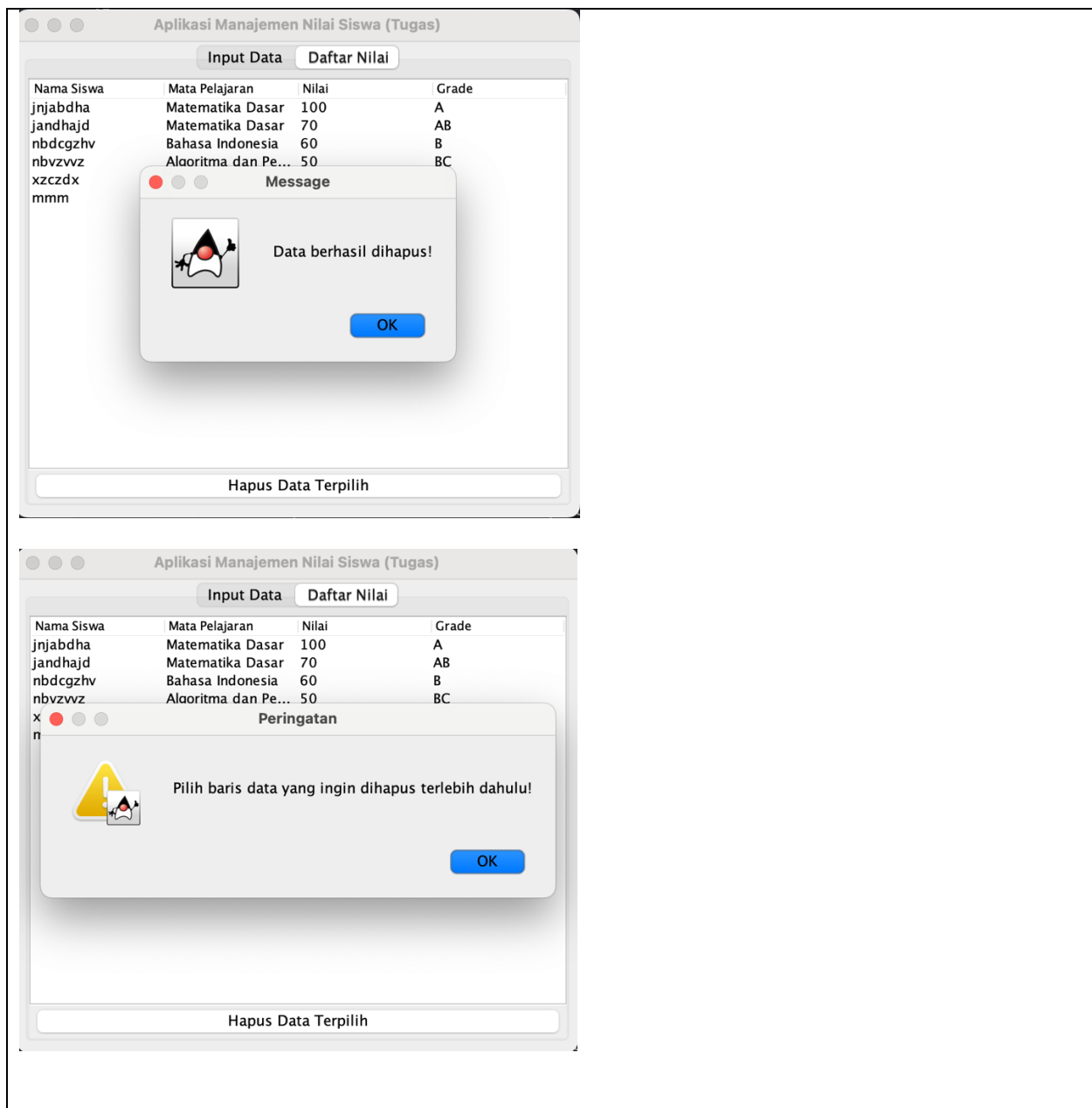
        if (selectedRow >= 0) {
            tableModel.removeRow(selectedRow);
            JOptionPane.showMessageDialog(null, "Data berhasil dihapus!");
        } else {
            JOptionPane.showMessageDialog(null, "Pilih baris data yang ingin dihapus terlebih dahulu!",
                                         "Peringatan", JOptionPane.WARNING_MESSAGE);
        }
    }
});

```

Bagian 2. Penjelasan Kode

Ini method `tableData.getSelectedRow()` untuk mengetahui baris ke-berapa yang sedang diklik pengguna. Jika hasilnya -1, berarti belum ada yang diklik. Jika ada (misal baris ke-0), perintah `tableModel.removeRow(selectedRow)` akan menghapus data tersebut dari memori table.

Bagian 3. Hasil Running



Bagian 4. Penjelasan Hasil Running

Di tab "Daftar Nilai", tepat di bawah tabel, muncul tombol baru bertuliskan "Hapus Data Terpilih".

- Jika pengguna langsung klik tombol tanpa memilih data: Muncul *popup* peringatan.
- Jika pengguna klik nama "Budi", lalu klik tombol: Baris data "Budi" hilang dari tabel

Instruksi 3

Perintah Tugas:

Tambahkan validasi untuk memastikan nama siswa minimal terdiri dari 3 karakter

CPMK Terkait:

Kode	Uraian
IF21W0508-CPMK04	Mampu mengkontruksi solusi berbasis komputing menggunakan kakas pemrograman berorientasi objek

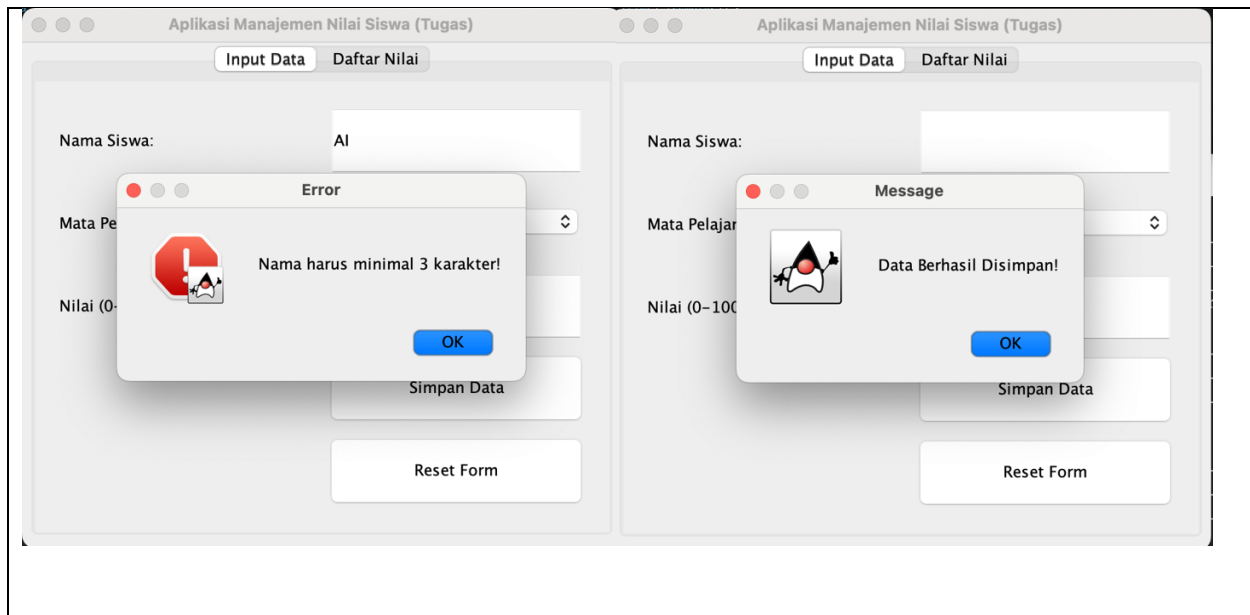
Bagian 1. Screenshot Semua Kode

```
if (nama.trim().isEmpty()) {
    JOptionPane.showMessageDialog(this, "Nama tidak boleh kosong!", "Error",
    JOptionPane.ERROR_MESSAGE);
    return;
} else if (nama.trim().length() < 3) {
    JOptionPane.showMessageDialog(this, "Nama harus minimal 3 karakter!", "Error",
    JOptionPane.ERROR_MESSAGE);
    return;
}
```

Bagian 2. Penjelasan Kode

Menggunakan method .length() pada string nama. Jika jumlah huruf kurang dari 3 (< 3), program memunculkan pesan error dan perintah return akan membatalkan proses penyimpanan.

Bagian 3. Hasil Running



Bagian 4. Penjelasan Hasil Running

- Jika pengguna mengetik "AI" di kolom Nama dan menekan Simpan.
- Akan muncul pesan *Pop-up* Error: "Nama harus minimal 3 karakter!".
- Data tidak akan masuk ke tabel sampai pengguna memperbaiki inputnya (misal menjadi "Ali").

Instruksi 4

Perintah Tugas:

Tambahkan tombol reset pada Tab Input Data yang berfungsi untuk membersihkan semua inputan (Nama dan Nilai)

CPMK Terkait:

Kode	Uraian
IF21W0508-CPMK04	Mampu mengkontruksi solusi berbasis komputing menggunakan kaskas pemrograman berorientasi objek

Bagian 1. Screenshot Semua Kode

```

JButton btnReset = new JButton("Reset Form");
panel.add(new JLabel(""));
panel.add(btnReset);

// 2. Logika Aksi Reset
btnReset.addActionListener(e -> {
    txtNama.setText("");
    txtNilai.setText("");
    cmbMatkul.setSelectedIndex(0);
    txtNama.requestFocus();
});

```

Bagian 2. Penjelasan Kode

Tombol ini memiliki *action listener* yang memerintahkan semua komponen input (`setText("")`) untuk kembali ke keadaan kosong atau *default*. `requestFocus()` adalah tambahan agar kursor langsung siap mengetik lagi di kolom nama.

Bagian 3. Hasil Running

Bagian 4. Penjelasan Hasil Running

- Di tab "Input Data", di sebelah tombol "Simpan Data", sekarang ada tombol "Reset Form".
- Jika pengguna sudah mengetik Nama "Siti", memilih Matkul, dan Nilai "90", lalu merasa salah dan ingin mengulang:

- Klik tombol "Reset Form" -> Semua isian menjadi bersih/kosong seketika.