

UAS PEMROGRAMAN TERSTRUKTUR
MANAJEMEN DATA SEKOLAH
KELOMPOK 1

Anggota kelompok 1 :

- | | |
|---------------------------|------------|
| 1. Efraim urel palodang | 2310010093 |
| 2. Muhammad Ma'ruf | 2310010063 |
| 3. Muhammad Ahyat | 2310010077 |
| 4. Muhammad Rayhan Ariadi | 2310010097 |

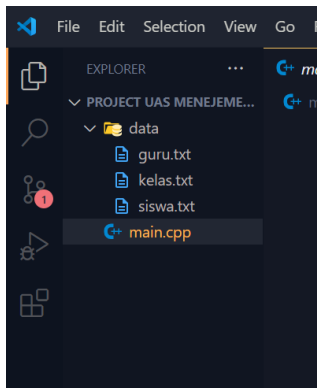
Deskripsi Program

Program manajemen data sekolah yang mengelola data kelas, siswa, dan guru. Program ini memiliki beberapa fitur seperti menambah, mengubah, menghapus, dan menampilkan data dari file. Data disimpan dalam file teks (kelas.txt, siswa.txt, dan guru.txt).

Struktur Data

- Kelas: Menyimpan data terkait kelas, seperti ID kelas, nama kelas, wali kelas, jumlah siswa, dan tahun ajaran.
- Siswa: Menyimpan data siswa, termasuk nama, NIS, jenis kelamin, tempat tanggal lahir (TTL), dan kelas yang diikuti.
- Guru: Menyimpan data guru, termasuk NIP, nama, jenis kelamin, TTL, dan nomor telepon.

1. Membuat folder data dan beberapa file txt untuk menyimpan data guru, kelas, dan siswa.



2. Bagian header dari program dan deklarasi struktur data (struct) dan vector untuk menyimpan informasi terkait kelas, siswa, dan guru dalam program.

```
1 #include <iostream>
2 #include <vector>
3 #include <fstream>
4 #include <iomanip> // Untuk std::setw
5
6 using namespace std;
7
8 ///! INSTANSIASI STRUCT
9
10 // DATA KELAS
11 struct Kelas {
12     string id, namaKelas, wali, jumlahSiswa, tahunAjaran;
13 };
14 vector<Kelas> dataKelas; // Membuat vector dari data kelas
15
16 // DATA SISWA
17 struct Siswa {
18     string nama, nis, jk, ttl, kelas;
19 };
20 vector<Siswa> dataSiswa; // Membuat vector dari data siswa
21
22 // DATA GURU
23 struct Guru {
24     string nip, nama, jk, ttl, nomor;
25 };
26 vector<Guru> dataGuru; // Membuat vector dari data guru
27
28
```

Funsi CRUD

1. Mengambil data dari file menggunakan fungsi `ambilDataKelas()`, `ambilDataSiswa()`, `ambilDataGuru()`, yang bertugas untuk membaca data dari file dan menyimpannya dalam vector sesuai dengan tipe data yang relevan. `getline()` digunakan di sini untuk membaca data dari file yang dipisahkan oleh tab (`'\t'`). Ini memungkinkan program untuk memisahkan dan mengolah data dengan format yang sudah ditentukan contohnya seperti tab.

```
1  ///! FUNGSI UNTUK MEMBACA DATA DARI FILE
2  void ambilDataKelas() {
3      ifstream data("data/kelas.txt"); // Membuka file untuk dibaca
4      if (!data.is_open()) { // Memastikan file berhasil dibuka
5          cout << "File tidak ditemukan!" << endl;
6          return;
7      }
8
9      string id, namaKelas, wali, jumlahSiswa, tahunAjaran;
10
11     // Membaca data dari file
12     while (data >> id >> ws) { // Membaca ID dan mengabaikan whitespace
13         getline(data, namaKelas, '\t'); // Membaca nama kelas hingga tab
14         getline(data, wali, '\t'); // Membaca wali hingga tab
15         getline(data, jumlahSiswa, '\t'); // Membaca jumlah siswa hingga tab
16         data >> tahunAjaran >> ws; // Membaca tahun ajaran dan mengabaikan whitespace
17
18         // Menambahkan data ke vector dataKelas
19         dataKelas.push_back({id, namaKelas, wali, jumlahSiswa, tahunAjaran});
20     }
21
22     data.close(); // Menutup file setelah selesai
23 }
24
25 void ambilDataSiswa() {
26     ifstream data("data/siswa.txt"); // Membuka file untuk dibaca
27     if (!data.is_open()) { // Memastikan file berhasil dibuka
28         cout << "File tidak ditemukan!" << endl;
29         return;
30     }
31
32     string nama, nis, jk, ttl, kelas;
33
34     // Membaca data dari file
35     while (getline(data, nama, '\t')) { // Membaca nama dan mengabaikan whitespace
36         getline(data, nis, '\t'); // Membaca nis kelas hingga tab
37         data >> jk >> ws; // Membaca jk kelas dan mengabaikan whitespace
38         data >> ttl >> ws; // Membaca ttl dan mengabaikan whitespace
39         getline(data, kelas); // Membaca kelas (sampai akhir baris)
40
41         // Menambahkan data ke vector dataKelas
42         dataSiswa.push_back({nama, nis, jk, ttl, kelas});
43     }
44
45     data.close(); // Menutup file setelah selesai
46 }
47
48 ///! FUNGSI UNTUK MEMBACA DATA GURU
49 void ambilDataGuru() {
50     ifstream data("data/guru.txt"); // Membuka file untuk dibaca
51     if (!data.is_open()) { // Memastikan file berhasil dibuka
52         cout << "File tidak ditemukan!" << endl;
53         return;
54     }
55
56     string nip, nama, jk, ttl, nomor;
57
58     // Membaca data dari file
59     while (getline(data, nip, '\t')) { // Membaca NIP dan mengabaikan whitespace
60         getline(data, nama, '\t'); // Membaca nama guru hingga tab
61         getline(data, jk, '\t'); // Membaca jenis kelamin hingga tab
62         getline(data, ttl, '\t'); // Membaca TTL hingga tab
63         getline(data, nomor); // Membaca nomor telepon
64
65         // Menambahkan data ke vector dataGuru
66         dataGuru.push_back({nip, nama, jk, ttl, nomor});
67     }
68
69     data.close(); // Menutup file setelah selesai
70 }
71
```

2. Fungsi cetak :

cetakDataKelas(), cetakDataSiswa(), dan cetakDataGuru() digunakan untuk menampilkan data yang ada di file. Semua fungsi menggunakan setw() untuk menetapkan lebar kolom dalam tabel, memastikan bahwa data yang dicetak terlihat rapi dan terstruktur. string(85, '-') digunakan untuk membuat garis pemisah yang memisahkan header dan data. Nomor Urut: Untuk setiap entri data, fungsi mencetak nomor urut (nomor++), yang akan bertambah seiring dengan bertambahnya data yang dicetak.

```
1
2 //! FUNGSI UNTUK CETAK DATA
3 // cetak data kelas
4 void cetakDataKelas() {
5     ifstream file("data/kelas.txt");
6
7     if (!file.is_open()) {
8         cout << "\nFILE data/kelas.txt TIDAK DITEMUKAN!!\n";
9         return;
10    }
11
12    cout << "\nDATA KELAS YANG SUDAH DI TAMBAHKAN !!\n";
13    cout << string(85, '-') << endl; // Garis pemisah
14    cout << left << setw(5) << "No."
15         << setw(7) << "ID"
16         << setw(20) << "Nama Kelas"
17         << setw(25) << "Wali Kelas"
18         << setw(10) << "Siswa"
19         << setw(10) << "Ajaran" << endl;
20    cout << string(85, '-') << endl; // Garis pemisah
21
22    string id, namaKelas, wali, jumlahSiswa, tahunAjaran;
23    int nomor = 1;
24
25    // Membaca dan mencetak data dari file
26    while (file >> id >> ws) {
27        getline(file, namaKelas, '\t'); // Membaca nama kelas hingga tab
28        getline(file, wali, '\t'); // Membaca wali kelas hingga tab
29        getline(file, jumlahSiswa, '\t'); // Membaca jumlah siswa hingga tab
30        file >> tahunAjaran >> ws; // Membaca tahun ajaran hingga akhir
31
32        // Mencetak data kelas yang dibaca dari file
33        cout << setw(5) << nomor++
34             << setw(7) << id
35             << setw(20) << namaKelas
36             << setw(25) << wali
37             << setw(10) << jumlahSiswa
38             << setw(10) << tahunAjaran << endl;
39    }
40
41    cout << string(85, '-') << endl; // Garis pemisah
42
43    file.close(); // Menutup file setelah selesai membaca
44 }
45
46 // cetak data siswa
47 void cetakDataSiswa(vector<Siswa>& data) {
48
49     if (data.empty()) {
50         cout << "\nDATA MASIH KOSONG, SILAHKAN MASUKKAN DATA DULU!!\n";
51         return;
52     }
53
54    cout << "\nDATA SISWA YANG SUDAH DI TAMBAHKAN !!\n";
55    cout << string(95, '-') << endl; // Garis pemisah
56    cout << left << setw(5) << "No."
57         << setw(25) << "Nama Siswa"
58         << setw(12) << "NIS"
59         << setw(15) << "Jenis Kelamin"
60         << setw(15) << "ttl"
61         << setw(15) << "Kelas" << endl;
62    cout << string(95, '-') << endl; // Garis pemisah
63
64    int nomor = 1;
65
66    for (const auto& Siswa : data) {
67        cout << setw(5) << nomor++
68             << setw(25) << Siswa.nama
69             << setw(12) << Siswa.nis
70             << setw(15) << Siswa.jk
71             << setw(15) << Siswa.ttl
72             << setw(15) << Siswa.kelas << endl;
73    }
74    cout << string(95, '-') << endl; // Garis pemisah
75 }
76
77 // cetak data guru
78 void cetakDataGuru() {
79     if (dataGuru.empty()) {
80         cout << "\nDATA GURU MASIH KOSONG, SILAHKAN MASUKKAN DATA DULU!!\n";
81         return;
82     }
83
84    cout << "\nDATA GURU YANG SUDAH DI TAMBAHKAN !!\n";
85    cout << string(95, '-') << endl; // Garis pemisah
86    cout << left << setw(5) << "No."
87         << setw(20) << "NIP"
88         << setw(25) << "Nama Guru"
89         << setw(15) << "Jenis Kelamin"
90         << setw(15) << "TTL"
91         << setw(15) << "Nomor" << endl;
92    cout << string(95, '-') << endl; // Garis pemisah
93
94    int nomor = 1;
95
96    for (const auto& guru : dataGuru) {
97        cout << setw(5) << nomor++
98             << setw(20) << guru.nip
99             << setw(25) << guru.nama
100            << setw(15) << guru.jk
101            << setw(15) << guru.ttl
102            << setw(15) << guru.nomor << endl;
103    }
104    cout << string(95, '-') << endl; // Garis pemisah
105 }
106
107
```

3. Fungsi Tambah Data:

fungsi

tambahDataKelas,

tambahDataSiswa, dan

tambahDataGuru

memiliki tujuan yang

sama, yaitu untuk

menambahkan data baru

(kelas, siswa, atau guru)

ke dalam sistem dan

menyimpannya ke

dalam file terkait. Setiap

fungsi membuat objek

baru (seperti Kelas,

Siswa, atau Guru) dan

meminta pengguna

untuk memasukkan data

melalui input. Setelah

data dimasukkan, data

ditambahkan ke dalam

vektor yang sesuai

(dataKelas, dataSiswa,

atau dataGuru) dan

kemudian disimpan langsung ke dalam file menggunakan fungsi penyimpanan masing-masing.

Fungsi-fungsi ini menggunakan `getline(cin, ...)` untuk membaca input karena metode ini memungkinkan pembacaan string yang mengandung spasi, seperti nama lengkap, jenis kelamin, atau alamat. Ini penting karena `cin >>` hanya membaca input hingga spasi pertama, sehingga `getline` lebih cocok untuk data yang terdiri dari kata-kata atau kalimat lengkap.

```
1
2  ///! FUNGSI UNTUK MENAMBAH DATA
3  // tambah data kelas
4  void tambahDataKelas() {
5      Kelas d; // membuat variable dari referensi objek kelas
6      cin.ignore(); // Mengabaikan newline yang tersisa di buffer
7      cout << "Masukkan ID Kelas: "; getline(cin, d.id);
8      cout << "Masukkan Nama Kelas: "; getline(cin, d.namaKelas);
9      cout << "Masukkan Wali Kelas: "; getline(cin, d.wali);
10     cout << "Masukkan Jumlah Siswa Kelas: "; cin >> d.jumlahSiswa;
11     cin.ignore(); // Mengabaikan newline yang tersisa di buffer
12     cout << "Masukkan Tahun Ajaran: "; getline(cin, d.tahunAjaran);
13     dataKelas.push_back(d);
14     simpanDataKelas(); // Menyimpan data secara langsung
15     cout << "\nData Kelas Berhasil Ditambahkan Dan Disimpan kedalam file data/kelas.txt !" << endl;
16 }
17
18 // tambah data siswa
19 void tambahDataSiswa() {
20     Siswa d; // membuat variable dari referensi objek Siswa
21     cin.ignore(); // Mengabaikan newline yang tersisa di buffer
22     cout << "Masukkan Nama : "; getline(cin, d.nama);
23     cout << "Masukkan NIS : "; getline(cin, d.nis);
24     cout << "Masukkan Jenis Kelamin : "; getline(cin, d.jk);
25     cout << "Masukkan TTL [tgl-bln-thn] : "; cin >> d.ttl;
26     cin.ignore(); // Mengabaikan newline yang tersisa di buffer
27     cout << "Masukkan Kelas Siswa : "; getline(cin, d.kelas);
28     dataSiswa.push_back(d);
29     simpanDataSiswa(); // Menyimpan data secara langsung
30     cout << "\nData Kelas Berhasil Ditambahkan Dan Disimpan kedalam file data/siswa.txt !" << endl;
31 }
32
33 // tambah data guru
34 void tambahDataGuru() {
35     Guru g; // membuat variable dari referensi objek Guru
36     cin.ignore(); // Mengabaikan newline yang tersisa di buffer
37     cout << "Masukkan NIP: "; getline(cin, g.nip);
38     cout << "Masukkan Nama Guru: "; getline(cin, g.nama);
39     cout << "Masukkan Jenis Kelamin: "; getline(cin, g.jk);
40     cout << "Masukkan TTL [tgl-bln-thn]: "; getline(cin, g.ttl);
41     cout << "Masukkan Nomor Telepon: "; getline(cin, g.nomor);
42
43     dataGuru.push_back(g);
44     simpanDataGuru(); // Menyimpan data secara langsung
45     cout << "\nData Guru Berhasil Ditambahkan Dan Disimpan kedalam file data/guru.txt !" << endl;
46 }
```

4. Fungsi ubah data :

Fungsi-fungsi di samping digunakan untuk mengubah data pada tiga kategori, yaitu kelas, siswa, dan guru. Setiap fungsi pertama-tama memeriksa apakah data yang ingin diubah tersedia, jika tidak, akan memberikan pesan bahwa data kosong. Setelah itu, fungsi menampilkan data yang ada dan meminta pengguna untuk memilih nomor data yang ingin diubah. Pengguna kemudian diminta untuk memasukkan informasi baru, seperti ID, nama, jumlah siswa, atau informasi terkait lainnya. Setelah data diperbarui, perubahan tersebut langsung disimpan menggunakan fungsi penyimpanan yang relevan (misalnya `simpanDataKelas()`, `simpanDataSiswa()`, `simpanDataGuru()`). Fungsi ini memastikan bahwa data yang dimasukkan pengguna disalin ke dalam objek yang sesuai dan perubahan tersimpan dengan baik.

```
1
2 //! FUNGSI UNTUK MENGUBAH DATA
3 // ubah data kelas
4 void ubahDataKelas() {
5     if (dataKelas.empty()) {
6         cout << "\nTidak ada data yang tersedia untuk diubah." << endl;
7         return;
8     }
9     cetakDataKelas();
10
11     int index;
12     cout << "Masukkan nomor data yang ingin diubah (mulai dari 1): ";
13     cin >> index;
14
15     if (index < 1 || index > dataKelas.size()) {
16         cout << "Nomor tidak Valid!" << endl;
17         return;
18     }
19
20     Kelas& d = dataKelas[index - 1];
21     cin.ignore(); // Mengabaikan newline yang tersisa di buffer
22     cout << "Masukkan ID Kelas: "; getline(cin, d.id);
23     cout << "Masukkan Nama Kelas: "; getline(cin, d.namaKelas);
24     cout << "Masukkan Wali Kelas: "; getline(cin, d.wali);
25     cout << "Masukkan Jumlah Siswa: "; cin >> d.jumlahSiswa;
26     cin.ignore(); // Mengabaikan newline yang tersisa di buffer
27     cout << "Masukkan Tahun Ajaran: "; getline(cin, d.tahunAjaran);
28     simpanDataKelas(); // Menyimpan data secara langsung
29     cout << "Data Kelas berhasil di Ubah!" << endl;
30 }
31
32 // ubah data siswa
33 void ubahDataSiswa() {
34     if (dataSiswa.empty()) {
35         cout << "\nTidak ada data yang tersedia untuk diubah." << endl;
36         return;
37     }
38     cetakDataSiswa(dataSiswa);
39
40     int index;
41     cout << "Masukkan nomor data yang ingin diubah (mulai dari 1): ";
42     cin >> index;
43
44     if (index < 1 || index > dataSiswa.size()) {
45         cout << "Nomor tidak Valid!" << endl;
46         return;
47     }
48
49     Siswa& d = dataSiswa[index - 1];
50     cin.ignore(); // Mengabaikan newline yang tersisa di buffer
51     cout << "Masukkan nama Siswa : "; getline(cin, d.nama);
52     cout << "Masukkan NIS : "; getline(cin, d.nis);
53     cout << "Masukkan Jenis Kelamin : "; getline(cin, d.jk);
54     cout << "Masukkan ttl [tgl-bln-thn] : "; getline(cin, d.ttl);
55     cout << "Masukkan Kelas: "; getline(cin, d.kelas);
56     simpanDataSiswa(); // Menyimpan data secara langsung
57     cout << "Data Siswa berhasil di Ubah!" << endl;
58 }
59
60 // ubah data guru
61 void ubahDataGuru() {
62     if (dataGuru.empty()) {
63         cout << "\nTidak ada data yang tersedia untuk diubah." << endl;
64         return;
65     }
66     cetakDataGuru();
67
68     int index;
69     cout << "Masukkan nomor data yang ingin diubah (mulai dari 1): ";
70     cin >> index;
71
72     if (index < 1 || index > dataGuru.size()) {
73         cout << "Nomor tidak Valid!" << endl;
74         return;
75     }
76
77     Guru& g = dataGuru[index - 1]; // membuat variable dari referensi objek Guru
78     cin.ignore(); // Mengabaikan newline yang tersisa di buffer
79     cout << "Masukkan NIP: "; getline(cin, g.nip);
80     cout << "Masukkan Nama Guru: "; getline(cin, g.nama);
81     cout << "Masukkan Jenis Kelamin: "; getline(cin, g.jk);
82     cout << "Masukkan TTL [tgl-bln-thn]: "; getline(cin, g.ttl);
83     cout << "Masukkan Nomor Telepon: "; getline(cin, g.nomor);
84     simpanDataGuru();
85     cout << "Data Guru berhasil di Ubah!" << endl;
86 }
87
```

5. Fungsi hapus data :

Setiap fungsi pertama-tama memeriksa apakah data yang ingin dihapus tersedia. Jika tidak ada data, maka sistem akan memberi tahu pengguna bahwa tidak ada data yang dapat dihapus. Setelah itu, fungsi menampilkan daftar data yang ada dan meminta pengguna untuk memilih nomor data yang ingin dihapus. Jika nomor yang dimasukkan tidak valid, sistem akan memberikan pesan kesalahan. Jika nomor valid, data pada indeks tersebut akan dihapus dari koleksi menggunakan fungsi `erase()`. Setelah penghapusan, perubahan langsung disimpan menggunakan fungsi penyimpanan yang relevan, seperti `simpanDataKelas()`, `simpanDataSiswa()`, atau `simpanDataGuru()`.

```
1
2  ///! FUNGSI UNTUK MENGHAPUS DATA
3  // hapus data kelas
4  void hapusDataKelas() {
5      if (dataKelas.empty()) {
6          cout << "Tidak ada data untuk dihapus!!" << endl;
7          return;
8      }
9      cetakDataKelas();
10     int index;
11     cout << "Masukkan data yang ingin dihapus (mulai dari 1): ";
12     cin >> index;
13     if (index < 1 || index > dataKelas.size()) {
14         cout << "Nomor tidak valid" << endl;
15         return;
16     }
17
18     dataKelas.erase(dataKelas.begin() + index - 1);
19     simpanDataKelas(); // Menyimpan data secara langsung
20     cout << "\nData Berhasil Dihapus!!\n" << endl;
21 }
22
23 // hapus data siswa
24 void hapusDataSiswa() {
25     if (dataSiswa.empty()) {
26         cout << "Tidak ada data untuk dihapus!!" << endl;
27         return;
28     }
29     cetakDataSiswa(dataSiswa);
30     int index;
31     cout << "Masukkan data yang ingin dihapus (mulai dari 1): ";
32     cin >> index;
33     if (index < 1 || index > dataSiswa.size()) {
34         cout << "Nomor tidak valid" << endl;
35         return;
36     }
37
38     dataSiswa.erase(dataSiswa.begin() + index - 1);
39     simpanDataSiswa(); // Menyimpan data secara langsung
40     cout << "\nData Berhasil Dihapus!!\n" << endl;
41 }
42
43 // hapus data guru
44 void hapusDataGuru() {
45     if (dataGuru.empty()) {
46         cout << "Tidak ada data untuk dihapus!!" << endl;
47         return;
48     }
49     cetakDataGuru();
50     int index;
51     cout << "Masukkan data yang ingin dihapus (mulai dari 1): ";
52     cin >> index;
53     if (index < 1 || index > dataGuru.size()) {
54         cout << "Nomor tidak valid" << endl;
55         return;
56     }
57
58     dataGuru.erase(dataGuru.begin() + index - 1);
59     simpanDataGuru(); // Menyimpan data secara langsung
60     cout << "\nData Berhasil Dihapus!!\n" << endl;
61 }
```


6. Simpan data :

Fungsi-fungsi di samping digunakan untuk menyimpan data kelas, siswa, dan guru ke dalam file teks. Setiap fungsi membuka file yang relevan untuk masing-masing kategori (misalnya kelas.txt, siswa.txt, guru.txt) menggunakan objek ofstream. Kemudian, fungsi tersebut melakukan iterasi terhadap koleksi data (seperti dataKelas, dataSiswa, atau dataGuru) dan menuliskan setiap elemen ke dalam file dengan format yang sudah ditentukan, memisahkan setiap atribut dengan tab (\t). Setelah seluruh data ditulis, file akan ditutup menggunakan file.close() untuk menyimpan perubahan tersebut. Fungsi-fungsi ini memastikan bahwa data yang telah diubah atau ditambahkan tersimpan dengan benar dalam file yang sesuai.

```
1
2  ///! FUNGSI UNTUK MENYIMPAN DATA KE FILE
3  // simpan data kelas
4  void simpanDataKelas() {
5      ofstream file("data/kelas.txt");
6
7      for (const auto& item : dataKelas) {
8          file << item.id << "\t"
9              << item.namaKelas << "\t"
10             << item.wali << "\t"
11             << item.jumlahSiswa << "\t"
12             << item.tahunAjaran << endl;
13      }
14
15      file.close();
16  }
17
18  // simpan data siswa
19  void simpanDataSiswa() {
20      ofstream file("data/siswa.txt");
21
22      for (const auto& item : dataSiswa) {
23          file << item.nama << "\t"
24              << item.nis << "\t"
25              << item.jk << "\t"
26              << item.ttl << "\t"
27              << item.kelas << endl;
28      }
29
30      file.close();
31  }
32
33  // simpan data guru
34  void simpanDataGuru() {
35      ofstream file("data/guru.txt");
36
37      for (const auto& item : dataGuru) {
38          file << item.nip << "\t"
39              << item.nama << "\t"
40              << item.jk << "\t"
41              << item.ttl << "\t"
42              << item.nomor << endl;
43      }
44
45      file.close();
46  }
47
```


7. Template Form

```
1
2 // form template
3 void templateForm(int a, string b) {
4     cout << string(a, '-') << endl;
5     cout << "Menu Pengelolaan Data " << b << endl;
6     cout << string(a, '-') << endl;
7     cout << "1. Tambah Data " << b <<
8         "\n2. Ubah Data " << b <<
9         "\n3. Hapus Data " << b <<
10        "\n4. Tampilkan Data " << b <<
11        "\n5. Kembali ke menu utama\n";
12 }
```

Fungsi `templateForm()` di atas digunakan untuk menampilkan format atau template menu yang sesuai dengan kategori data yang sedang dikelola (misalnya, kelas, siswa, atau guru). Fungsi ini menerima dua parameter: `a` (integer) yang menentukan panjang garis horizontal yang ditampilkan di atas dan bawah judul, serta `b` (string) yang merupakan nama kategori data (seperti "Kelas", "Siswa", atau "Guru").

Di dalam fungsi ini, pertama-tama akan ditampilkan garis horizontal sepanjang `a` karakter (`string(a, '-')`). Setelah itu, akan muncul judul "Menu Pengelolaan Data" diikuti dengan nama kategori data yang diteruskan melalui parameter `b`. Kemudian, akan ditampilkan daftar pilihan operasi yang bisa dilakukan untuk kategori data tersebut, seperti menambah, mengubah, menghapus, menampilkan data, atau kembali ke menu utama. Fungsi ini memastikan format menu tampil secara rapi dan konsisten.

FUNGSI UTAMA

Program utama di samping berfungsi untuk mengelola data sekolah, seperti data kelas, siswa, dan guru, dengan menggunakan menu interaktif yang memungkinkan pengguna memilih operasi yang ingin dilakukan. Pada awal program, data dari file (kelas, siswa, dan guru) diambil dan disimpan dalam variabel yang sesuai. Kemudian, program menampilkan menu utama dengan empat pilihan: mengelola data kelas, siswa, guru, atau keluar dari program. Ketika pengguna memilih salah satu menu, program akan menampilkan submenu untuk kategori yang dipilih (kelas, siswa, atau guru) dan memungkinkan pengguna untuk melakukan beberapa operasi seperti menambah, mengubah, menghapus, atau mencetak data. Setiap submenu memiliki pilihan untuk kembali ke menu utama. Fungsi `templateForm()` digunakan untuk menampilkan tampilan atau format tertentu sesuai kategori yang dipilih. Di dalam submenu, pengguna dapat memilih berbagai operasi dengan memasukkan angka yang sesuai. Setelah setiap operasi, program memastikan pengguna dapat kembali ke menu utama atau melanjutkan pengelolaan data yang sama. Proses ini berulang hingga pengguna memilih untuk keluar dari program (pilihan 4).

```
1
2 //! Program Utama
3 int main() {
4     int pilih;
5     bool kembaliKeMenuUtama = false;
6
7     ambilDataKelas(); // Mengambil data dari file saat program dimulai
8     ambilDataSiswa(); // Mengambil data dari file saat program dimulai
9     ambilDataGuru(); // Mengambil data dari file saat program dimulai
10
11     do {
12         // Menu utama
13         cout << "=====\\n";
14         cout << "PROJEK MANAJEMEN DATA SEKOLAH\\n";
15         cout << "=====\\n";
16         cout << "1. Data Kelas\\n";
17         cout << "2. Data Siswa\\n";
18         cout << "3. Data Guru\\n";
19         cout << "4. Keluar\\n";
20         cout << "Pilih Menu [1-4]: ";
21         cin >> pilih;
22
23         switch (pilih) {
24             case 1:
25                 // Mengelola data Kelas
26                 do {
27                     templateForm(28, "Kelas");
28                     int pilihKelas;
29                     cout << "Pilih menu: "; cin >> pilihKelas;
30                     switch (pilihKelas) {
31                         case 1: tambahDataKelas(); break;
32                         case 2: ubahDataKelas(); break;
33                         case 3: hapusDataKelas(); break;
34                         case 4: cetakDataKelas(); break;
35                         case 5: kembaliKeMenuUtama = true; break; // Kembali ke menu utama
36                         default: cout << "Pilihan tidak valid!" << endl;
37                     }
38                 } while (!kembaliKeMenuUtama); // Loop berlanjut hingga kembali ke menu utama
39                 break;
40
41             case 2:
42                 // Mengelola data Siswa
43                 do {
44                     templateForm(28, "Siswa");
45                     int pilihSiswa;
46                     cout << "Pilih menu: "; cin >> pilihSiswa;
47                     switch (pilihSiswa) {
48                         case 1: tambahDataSiswa(); break;
49                         case 2: ubahDataSiswa(); break;
50                         case 3: hapusDataSiswa(); break;
51                         case 4: cetakDataSiswa(dataSiswa); break;
52                         case 5: kembaliKeMenuUtama = true; break; // Kembali ke menu utama
53                         default: cout << "Pilihan tidak valid!" << endl;
54                     }
55                 } while (!kembaliKeMenuUtama); // Loop berlanjut hingga kembali ke menu utama
56                 break;
57
58             case 3:
59                 // Mengelola data Guru
60                 do {
61                     templateForm(28, "Guru");
62                     int pilihGuru;
63                     cout << "Pilih menu: "; cin >> pilihGuru;
64                     switch (pilihGuru) {
65                         case 1: tambahDataGuru(); break;
66                         case 2: ubahDataGuru(); break;
67                         case 3: hapusDataGuru(); break;
68                         case 4: cetakDataGuru(); break;
69                         case 5: kembaliKeMenuUtama = true; break;
70                         default: cout << "Pilihan tidak valid!" << endl;
71                     }
72                 } while (!kembaliKeMenuUtama);
73                 break;
74
75             case 4:
76                 cout << "Terima kasih! Program selesai.\\n";
77                 break;
78             default:
79                 cout << "Pilihan tidak valid!" << endl;
80         }
81
82         // Reset flag kembaliKeMenuUtama untuk loop berikutnya
83         kembaliKeMenuUtama = false;
84     } while (pilih != 4); // Loop kembali ke menu utama hingga pilih 4 untuk keluar
85
86     return 0;
87 }
```

HASIL PROGRAM

```
File Edit Selection View Go Run ... project uas menejement data sekolah
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\ASUS\OneDrive\Desktop\semester 3\project pemrograman terstruktur\project uas menejement data sekolah> cd "c:\Users\ASUS\OneDrive\Desktop\semester 3\project pemrograman terstruktur\project uas menejement data sekolah" ; if ($?) { g++ main.cpp -o main } ; if ($?) { .\main }
=====
PROJEK MANAJEMEN DATA SEKOLAH
=====
1. Data Kelas
2. Data Siswa
3. Data Guru
4. Keluar
Pilih Menu [1-4]:
```

```
File Edit Selection View Go Run ... project uas menejement data sekolah
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Pilih Menu [1-4]: 1
=====
Menu Pengelolaan Data Kelas
=====
1. Tambah Data Kelas
2. Ubah Data Kelas
3. Hapus Data Kelas
4. Tampilkan Data Kelas
5. Kembali ke menu utama
Pilih menu: 4

DATA KELAS YANG SUDAH DI TAMBAHKAN !!
=====
No. ID Nama Kelas Wali Kelas Siswa Ajaran
=====
1 B001 BISNIS KONSTRUKSI Azwirman.S.Pd 20 2023
2 D002 DESAIN PEMODELAN Saipul 24 2023
3 T003 TEKNIK KENDARAAN Iswardi.S.pd 30 2023
4 M004 MULTIMEDIA Helmiyati S.Pd 27 2023
5 A005 AGRIBISNIS Elpia Nengsih S.Pd 23 2023
=====

Menu Pengelolaan Data Kelas
=====
1. Tambah Data Kelas
2. Ubah Data Kelas
3. Hapus Data Kelas
4. Tampilkan Data Kelas
5. Kembali ke menu utama
Pilih menu:
```

```
File Edit Selection View Go Run ... project uas menejement data sekolah
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
4. Keluar
Pilih Menu [1-4]: 2
=====
Menu Pengelolaan Data Siswa
=====
1. Tambah Data Siswa
2. Ubah Data Siswa
3. Hapus Data Siswa
4. Tampilkan Data Siswa
5. Kembali ke menu utama
Pilih menu: 4

DATA SISWA YANG SUDAH DI TAMBAHKAN !!
=====
No. Nama Siswa NIS Jenis Kelamin ttl Kelas
=====
1 Sindy Alya 0069896650 Perempuan 12-12-2008 DESAIN PEMODELAN
2 Mutiara Kristiani 0063389195 Perempuan 05-11-2008 BISNIS KONTRUKSI
3 Andini Dwi 0051080894 Perempuan 07-01-2008 MULTIMEDIA
4 Bayu Samanta 0098764023 Laki-Laki 03-09-2007 TEKNIK KENDARAAN
=====

Menu Pengelolaan Data Siswa
=====
1. Tambah Data Siswa
2. Ubah Data Siswa
3. Hapus Data Siswa
4. Tampilkan Data Siswa
5. Kembali ke menu utama
Pilih menu:
```

```
File Edit Selection View Go Run ... project uas menejement data sekolah
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Pilih Menu [1-4]: 3
-----
Menu Pengelolaan Data Guru
-----
1. Tambah Data Guru
2. Ubah Data Guru
3. Hapus Data Guru
4. Tampilkan Data Guru
5. Kembali ke menu utama
Pilih menu: 4

DATA GURU YANG SUDAH DI TAMBAHKAN !!
-----
No. NIP Nama Guru Jenis Kelamin TTL Nomor
-----
1 1942731633200032 Azwirman.S.Pd Laki-Laki 07-08-1980 085709897876
2 3250747650110043 Saipul Laki-Laki 04-11-1985 087509837822
3 8534760661200032 Iswardi.S.Pd Laki-Laki 07-02-1989 086109998278
4 7540761664200012 Helmiyati S.Pd Perempuan 29-03-1995 081285769094
5 3737768668110002 Elpia Nengsih S.Pd Perempuan 19-10-1991 081509787367
-----

Menu Pengelolaan Data Guru
-----
1. Tambah Data Guru
2. Ubah Data Guru
3. Hapus Data Guru
4. Tampilkan Data Guru
5. Kembali ke menu utama
Pilih menu: 
```

```
File Edit Selection View Go Run ... project uas menejement data sekolah
EXPLORER main.cpp M guru.txt
PROJECT UAS MENEJEME... data > guru.txt
data guru.txt
1 1942731633200032 Azwirman.S.Pd Laki-Laki 07-08-1980 085709897876
2 3250747650110043 Saipul Laki-Laki 04-11-1985 087509837822
3 8534760661200032 Iswardi.S.Pd Laki-Laki 07-02-1989 086109998278
4 7540761664200012 Helmiyati S.Pd Perempuan 29-03-1995 081285769094
5 3737768668110002 Elpia Nengsih S.Pd Perempuan 19-10-1991 081509787367
6
```

```
File Edit Selection View Go Run ... project uas menejement data sekolah
EXPLORER main.cpp M kelas.txt
PROJECT UAS MENEJEME... data > kelas.txt
data kelas.txt
1 B001 BISNIS KONSTRUKSI Azwirman.S.Pd 20 2023
2 D002 DESAIN PEMODELAN Saipul 24 2023
3 T003 TEKNIK KENDARAAN Iswardi.S.pd 30 2023
4 M004 MULTIMEDIA Helmiyati S.Pd 27 2023
5 A005 AGRIBISNIS Elpia Nengsih S.Pd 23 2023
6
```

```
File Edit Selection View Go Run ... project uas menejement data sekolah
EXPLORER main.cpp M siswa.txt
PROJECT UAS MENEJEME... data > siswa.txt
data siswa.txt
1 Sindy Alya 0069896650 Perempuan 12-12-2008 DESAIN PEMODELAN
2 Mutiara Kristiani 0063389195 Perempuan 05-11-2008 BISNIS KONTRUKSI
3 Andini Dwi 0051080894 Perempuan 07-01-2008 MULTIMEDIA
4 Bayu Samanta 0098764023 Laki-Laki 03-09-2007 TEKNIK KENDARAAN
5
```