

# **MAKALAH PEMROGRAMAN BERORIENTASI OBJEK**

## **PROGRAM KANTIN**



DISUSUN OLEH :

KOM C

BETSYEDA VALENTINA TARIGAN (201402033)

YEFTHA EL IMANI PARDEDE (201402075)

M. FADIL RAMADHAN (201402078)

M. FAREL KINANDUNG (201402132)

**Diajukan sebagai syarat untuk memenuhi tugas besar mata kuliah Pemrograman  
Berorientasi Objek**

**Dosen : Dani Gunawan**

**FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI  
PROGRAM STUDI TEKNOLOGI INFORMASI  
UNIVERSITAS SUMATERA UTARA  
JUNI 2021**

## **KATA PENGANTAR**

Puji dan syukur kehadiran tuhan yang maha esa yang telah memberikan kesempatan, kesehatan serta kenikmatan sehingga kami dapat menyelesaikan makalah ini dengan baik. Makalah ini berjudul “Program Kasir Kantin Pancen Oye” yang disusun sebagai tugas mata kuliah Dasar Pemrograman.

Banyak pihak yang memberikan bimbingan dan dukungan dalam penyelesaian makalah ini. Oleh karena itu dengan kerendahan hati kami mengucapkan terima kasih kepada Bapak Dani Gunawan, ST., Mt. selaku dosen pembimbing yang sudah membimbing dan memberikan masukan dalam penulisan makalah ini.

Alasan pemilihan judul ini adalah agar kita dapat mengetahui secara jelas bagaimana program kantin yang dijalankan dengan menggunakan komputer. Dan juga agar kita mengetahui bahwa program yang dijalankan ini lebih efisien daripada dengan menggunakan cara manual atau mencatat transaksi jual beli.

Kami juga menyadari bahwa makalah ini masih jauh dari kata sempurna tentunya banyak terdapat kesalahan. oleh sebab itu, kami mengharapkan saran dan kritik yang membangun dari pembaca agar makalah ini dapat menjadi sempurna. Kami berharap semoga makalah ini dapat bermanfaat bagi pembaca dan masyarakat luas.

Medan, 12 June 2021

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	<b>i</b>
<b>DAFTAR ISI.....</b>	<b>ii</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
Judul.....	1
Deskripsi Tugas Besar.....	1
<b>BAB II HASIL.....</b>	<b>2-17</b>
Setup Awal.....	2-4
Penjelasakan Source Code.....	5-17
<b>BAB III PENUTUPAN.....</b>	<b>18</b>
Kesimpulan.....	18
Saran.....	18

# **BAB I**

## **PENDAHULUAN**

### **Judul**

Kelompok kami memilih judul **“Program Kasir Kantin Pancen Oye”** dengan alasan agar kita semua dapat menambah wawasan mengenai program yang dijalankan dengan komputer untuk bisa menghemat waktu dibandingkan dengan menggunakan cara manual.

### **Deskripsi Judul Tugas Besar**

Aplikasi Program kantin dibuat oleh kelompok 8 untuk memenuhi tugas besar mata kuliah Pemrograman Berorientasi Objek. Aplikasi ini dibuat dengan cara dibentuk dengan menerapkan konsep- konsep Pemrograman Berorientasi Objek, yaitu: abstraksi, enkapsulasi, inheritance, penyimpanan data ( file atau database ), dan polimorfisme. Program ini juga menerapkan berbagai library c++ seperti fstream, vektor, sstream, iomanip. Aplikasi ini dibuat untuk mempermudah atau mempercepat transaksi jual beli dibandingkan dengan secara manual dan sebagai alat yang memudahkan dalam manajemen kasir Kantin Pancen Oye. Setiap menu yang tersedia memiliki fungsi masing-masing yang membangun Aplikasi Program menjadi lebih sempurna.

## BAB II

### HASIL

#### Setup Awal

##### a. Tampilan menu login aplikasi

```
+-----+
|               |
|  APLIKASI KASIR  |
| ~KANTIN PANCEN OYE~ |
|               |
+-----+
|               |
|  +-----+  |
|  | LOGIN |  |
|  +-----+  |
|               |
Masukkan username: _
```

Di aplikasi ini , pertama sekali admin diwajibkan Login dengan menginput username dan password. Jika terjadi kesalahan login maka tidak dapat masuk kedalam aplikasi. Jika terjadi kesalahan 3x pada menginput username dan password maka aplikasi keluar secara terpaksa.

##### b. Menu Utama

```
+-----+
|               |
|  SELAMAT DATANG DI  |
|  CAFE KITA - KITA  |
|               |
+-----+
|               |
|  Anda Masuk Sebagai Administrator  |
|               |
|  +-----+  |
|  | -- Main Menu -- |  |
|  | 1. Kasir       |  |
|  | 2. Logout      |  |
|  +-----+  |
|               |
Masukkan Pilihan Anda : _
```

Jika berhasil login, maka bisa mengakses menu admin. Didalam menu utama, admin bisa mengakses dua input yaitu pada input yang pertama masuk kedalam menu kasir dan yang kedua logout untuk keluar dari aplikasi.

##### c. Menu kasir

```
=====
||  SELAMAT DATANG DI PROGRAM KANTIN  ||
=====
1. Beli
2. Tambahkan Menu
3. Edit Menu
4. Hapus Menu
5. Cari Data
6. Exit
Masukkan Pilihan Anda : _
```

Jika dari menu utama, kita menginput 1, maka kita akan terhubung pada menu kasir. Dimenu kasir terdapat 6 inputan yaitu Beli, tambahkan menu, edit menu, hapus menu, cari data dan exit.

#### d. Menu Beli

```
=====
|   Id   | Nama Makanan |   Harga   |
|-----|-----|-----|
1         | Nasi Goreng  | 20000     |
2         | Ikan Asin    | 500       |
3         | Pisang Bakar | 15000     |
4         | Mie Soto     | 10000     |
5         | Bakso Aci    | 12000     |
6         | Samgyeopsal  | 127000    |
|-----|-----|-----|
ID Pembeli      : 10
Nama Pembeli    : _
```

Jika kita menginput 1, maka menu beli akan muncul. Disini ada 6 menu makanan yang dijual yaitu ada Nasi goreng, Ikan asin, Pisang bakar, Mie soto, Bakso aci, dan Samgyeopsal.

#### e. Menu Tambah Menu

```
=====
|   Id   | Nama Makanan |   Harga   |
|-----|-----|-----|
1         | Nasi Goreng  | 20000     |
2         | Ikan Asin    | 500       |
3         | Pisang Bakar | 15000     |
4         | Mie Soto     | 10000     |
5         | Bakso Aci    | 12000     |
6         | Samgyeopsal  | 127000    |
|-----|-----|-----|
Masukkan Menu Makanan Tambahan
Id          : 7
Nama Makanan : _
```

Jika kita menginput 2 dari menu kasir, maka menu tambahkan menu akan muncul. Dimenu ini kita dapat menambahkan menu makanan.

#### f. Menu Edit Menu

Jika kita menginput 3 dari menu kasir, maka menu edit menu akan muncul. Dimenu ini kita dapat mengubah menu makanan.

### g. Menu Hapus Menu

```
=====
|   Id   | Nama Makanan |  Harga  |
=====
1         | Nasi Goreng  | 20000   |
2         | Ikan Asin    | 500     |
3         | Pisang Bakar | 15000   |
4         | Mie Soto     | 10000   |
5         | Bakso Aci    | 12000   |
6         | Samgyeopsal  | 127000  |
=====

Masukkan Id Menu yang akan dihapus : _
```

Jika kita menginput 4 dari menu kasir, maka menu hapus menu akan muncul. Disini kita dapat menghapus daftar menu.

### h. Menu Cari Data

```
=====
|   Id   | Nama Pembeli | Menu      | Jumlah |  Harga  |
=====
1         | Yeftha      | Nasi Gorengs | 2       | 40000   |
2         | Jose        | Bakso Aci    | 3       | 36000   |
3         | Kiara       | Mie Soto     | 1       | 10000   |
4         | Yeftha      | Pisang Bakar | 2       | 30000   |
5         | Betsyeda    | Pisang Bakar | 2       | 30000   |
6         | Jose        | Samgyeopsal  | 1       | 127000  |
7         | Obed        | Pisang Bakar | 2       | 30000   |
8         | Isyana      | Ikan Asin    | 20      | 10000   |
9         | wk          | Nasi Goreng  | 0       | 40000   |
10        | 1           | Nasi Goreng  | 0       | 20000   |
=====

Masukkan nama pembeli : Obed

=====
| No Pembelian | Nama Makanan | Jumlah |  Harga  |
=====
| 7            | Pisang Bakar | 2       | 30000   |
=====

Apakah Anda ingin mencari data kembali? (y/n) : _
```

Jika kita menginput 5 dari menu kasir, maka menu cari data akan muncul. Disini kita dapat mencari atau mengetahui data dari pembeli, setra kita juga dapat melakukan transaksi.

### i. Menu Exit

```
+-----+
|  -- Main Menu --  |
| 1. Kasir          |
| 2. Logout         |
+-----+
Masukkan Pilihan Anda :
```

Jika kita menginput 6 dari menu kasir, maka menu exit akan muncul. Setelah kita memilih menu exit, kita akan dibawa kembali ke menu utama.

## Penjelasan Mengenai Kode program

Berikut source code yang digunakan pada program :

### a. Class Autentifikasi

#### 1. Fungsi Login

```
class Autentikasi : public Main
{
private :
    char username[20], password[20];

public :
    void login()
    {
        // variabel banyak percobaan
        int status = 0;
        system("cls");

        // warna background
        system("color 0a");

        for (int i = 0; i < 3; i++)
        {
            /*----Tampilan Aplikasi Pembuka----*/
            cout << " + " << setfill('.') << setw(40) << "+" << "\n";
            cout << " | " << setfill(' ') << setw(40) << "| " << "\n";
            cout << " | " << setfill(' ') << setw(25) << "APLIKASI KASIR" << setfill(' ') << setw(15) << "| " << "\n";
            cout << " | " << setfill(' ') << setw(27) << "~KANTIN PANCEN OYE~" << setfill(' ') << setw(13) << "| " << "\n";
            cout << " | " << setfill(' ') << setw(40) << "| " << "\n";
            cout << " + " << setfill('.') << setw(40) << "+" << "\n";

            cout << "\n\t +-----+\n";
            cout << "\t |          LOGIN          |\n";
            cout << "\t +-----+\n";
            cout << " Masukkan username: ";
            cin >> username;
            cout << " Masukkan password: ";
            cin >> password;

            if (strcmp(username, "admin") == 0)
            {
                // memastikan password yang dimasukkan valid
                if (strcmp(password, "12345") == 0)
                {
                    break;
                }
                else
                {
                    system("cls");
                    cout << " Password salah!\n";
                    status++;
                }
            }
            else
            {
                system("cls");
                cout << " Username Anda salah\n";
                status++;
            }
        }
        if (status == 3)
        {
            cout << " Anda Sudah melebihi batas login, tidak bisa login kembali\n";
            getch();
            // menutup program apabila sudah mencapai 3 percobaan
            exit(0);
        }
    }
}
```

Di Bagian awal, kami membuat fungsi login di dalam class autentikasi, dengan memakai inheritance pada class main agar keamanan admin terjaga. Kami menggunakan fungsi void(). Variabel username dan password kami menggunakan tipe data char agar menghindari terjadinya kesalahan inputan yang tidak sesuai dengan tipe data. Kami juga membuat program keluar paksa jika 3x kesalahan input saat login.



## 2. Tampilan Awal Program

```
void greeting()
{
    system("cls");
    /*----Tampilan Aplikasi Kasir-----*/
    cout << " + " << setfill('-') << setw(40) << "+" << "\n";
    cout << " | " << setfill(' ') << setw(40) << "| " << "\n";
    cout << " | " << setfill(' ') << setw(27) << "SELAMAT DATANG DI" << setfill(' ') << setw(13) << "| " << "\n";
    cout << " | " << setfill(' ') << setw(26) << "CAFE KITA - KITA" << setfill(' ') << setw(14) << "| " << "\n";
    cout << " | " << setfill(' ') << setw(40) << "| " << "\n";
    cout << " + " << setfill('-') << setw(40) << "+" << "\n";

    cout << "\n Anda Masuk Sebagai Administrator\n";
}

void menuPembuka()
{
    cout << "\n";
    cout << " + " << setfill('-') << setw(21) << "+" << "\n";
    cout << " | " << setfill(' ') << setw(17) << "-- Main Menu --" << setfill(' ') << setw(4) << "| " << "\n";
    cout << " | " << setfill(' ') << setw(6) << "1. Kasir" << setfill(' ') << setw(13) << "| " << "\n";
    cout << " | " << setfill(' ') << setw(6) << "2. Logout" << setfill(' ') << setw(12) << "| " << "\n";
    cout << " + " << setfill('-') << setw(21) << "+" << "\n";
}
```

### b. Class Main

Class main berisi fungsi untuk membuat data, menampilkan menu, menampilkan menu makanan, menampilkan daftar pembelian, dan menghitung jumlah data.

## 1. Fungsi Membuat File

```
class Main
{
protected :
    int jlh_lines;

public :
    void fileData (fstream &data) // Berfungsi untuk membuat dataMakanan.csv dan dataPembeli.csv
    {
        data.open("dataMakanan.csv", ios::out | ios::in | ios::binary);

        if (!data.is_open()) {
            data.close();
            data.open("dataMakanan.csv", ios::trunc | ios::out | ios::in | ios::binary);
        }
        data.close();

        data.open("dataPembeli.csv", ios::out | ios::in | ios::binary);

        if (!data.is_open()) {
            data.close();
            data.open("dataPembeli.csv", ios::trunc | ios::out | ios::in | ios::binary);
        }
        data.close();
    }
}
```

Pada fungsi ini kami menggunakan fungsi fstream. Fungsi akan membuka file dataMakanan.csv dan dataPembeli.csv. Apabila pada komputer tidak ditemukan data tersebut maka fungsi akan membuat file tersebut.

## 2. Fungsi menu

```
void menuUtama() // Berfungsi untuk menampilkan menu utama
{
    cout << "===== " << endl;
    cout << "||      SELAMAT DATANG DI PROGRAM KANTIN      ||" << endl;
    cout << "===== " << endl;

    cout << "1. Beli" << endl;
    cout << "2. Tambahkan Menu" << endl;
    cout << "3. Edit Menu" << endl;
    cout << "4. Hapus Menu" << endl;
    cout << "5. Cari Data" << endl;
    cout << "6. Exit" << endl;
    cout << "Masukkan Pilihan Anda : ";
}

void menuMakanan() // Berfungsi untuk menampilkan menu makanan
{
    ifstream data("dataMakanan.csv");
    string row, col[3];
    cout << "\t===== \n";
    cout << "\t|      Id      | Nama Makanan |      Harga      | \n";
    cout << "\t===== \n";
    while (getline(data, row))
    {
        cout << "\t";
        stringstream s(row);
        for (int i = 0; i < 3; i++)
            getline(s, col[i], ',');
        for (int i = 0; i < 3; i++)
            cout << left << setfill(' ') << setw(15) << col[i] << '|';
        cout << endl;
    }
    cout << "\t===== \n";
    data.close();
}

void daftarPembelian() // Berfungsi untuk menampilkan daftar pembelian
{
    ifstream data("dataPembeli.csv");
    string row, col[5];
    cout << "===== \n";
    cout << "\t|      Id      | Nama Pembeli |      Menu      |      Jumlah      |      Harga      | \n";
    cout << "\t===== \n";
    while (getline(data, row))
    {
        stringstream s(row);
        for (int i = 0; i < 5; i++)
            getline(s, col[i], ',');
        for (int i = 0; i < 5; i++)
            cout << left << setfill(' ') << setw(15) << col[i] << '|';
        cout << endl;
    }
    cout << "===== \n";
    data.close();
}
```

Fungsi void menuUtama() berfungsi untuk menampilkan menu utama program. Fungsi void menuMakanan() dan void daftarPembelian() berfungsi untuk menampilkan file dari csv. Dimana fungsi pertama kali akan membuka file dan mengambil data di dalamnya per line dan setelah itu menutup file.

### c. Class Makanan

Class makanan meng-inheritance class Main untuk mengambil fungsi yang berada di class main, seperti menuMakanan().

#### 1. Template ambil harga

```
class Makanan : public Main
{
    private :
        string pembeli, menuPembeli, id, namaMakanan, hargaMakanan, idNew, namaNew;
        int id_pembeli, jlhDibeli, total_harga, hargaNew, count = 0;

    public :

        template <class T>
        T ambilHarga(string s, int col)
        {
            T val;
            int counter = 0;
            string str;
            stringstream ss(s);

            while (counter++ < col)
            {
                getline(ss, str, ',');
            }

            stringstream sstr(str);
            sstr >> val;
            return val;
        }
}
```

Fungsi ini digunakan untuk menyederhanakan fungsi dalam hal mengambil data dari file csv tetapi dalam bentuk integer. Fungsi ini akan mengambil data dari file csv pada baris tertentu dan mengambil nilainya ( harga makanan ) dalam bentuk integer.

## 2. Fungsi beli

```
void beli ()
{
    menuMakanan();
    jlhDataPembeli();

    string id_item;
    int id_harga, jlh, harga, count = 0;
    char yn;

    //menulis data ke "dataPembeli.csv"
    ofstream data("dataPembeli.csv", ios::app);

    //memasukkan input yang dibutuhkan
    awal:
    cout << "\nID Pembeli\t\t\t : ";
    cout << jlh_lines + 1 << endl;
    id_pembeli = jlh_lines + 1;

    cout << "Nama Pembeli\t\t\t : ";
    cin.ignore();
    getline(cin, pembeli);

    input:
    cout << "Masukan Id Menu yang mau dibeli\t : ";
    cin >> id_item;

    //mengambil data dari "dataMakanan.csv" bertipe string
    fstream data2;
    data2.open("dataMakanan.csv", ios::in);
    vector<string> row;
    string line, word, temp;

    while (getline(data2, line))
    {
        row.clear();
        istringstream s(line);

        while (getline(s, word, ','))
        {
            row.push_back(word);
        }

        if (row[0] == id_item)
        {
            count = 1;
            cout << "Nama Makanan\t\t\t : " << row[1] << endl;
            menuPembeli = row[1];
        }
    }

    if (count == 0)
    {
        cout << "Menu tidak Ditemukan" << endl;
        goto input;
    }

    //menguubah string id_item ke int id_harga
    string s;
    stringstream geek(id_item);
    geek >> id_harga;

    //mengambil data dari (dataMakanan.csv) bertipe int
    ifstream bacaMakanan("dataMakanan.csv", ios::binary);
    while (getline(bacaMakanan, s, '\n'))
    {
        if (ambilHarga<int>(s, 1) == id_harga)
        {
            cout << "Harga Makanan\t\t\t : Rp";
            cout << ambilHarga<int>(s, 3) << endl;
        }
    }
}
```

```

        harga = ambilHarga<int>(s, 3);

        cout << "Masukkan Jumlah Pesanan\t\t : ";
        cin >> jlh;
        jlhDibeli = jlh;
        total_harga = harga * jlh;
    }
}

cout << "Total Harga adalah\t\t : Rp" << total_harga << endl;
tanya:
cout << "\nAnda yakin ingin memasukkan data tersebut? (y/n) : ";
cin >> yn;

if (yn == 'Y' | yn == 'y')
{
    data << id_pembeli << "," << pembeli << "," << menuPembeli << "," << jlhDibeli << "," << total_harga << endl;
    cout << "\nData pemesanan telah berhasil dimasukkan!\n" << endl;
    system("pause");
}
else if (yn == 'N' | yn == 'n')
{
    cout << "\nData pemesanan dibatalkan!\n" << endl;
    system("pause");
}
else
{
    cout << "Input yang anda masukkan salah!" << endl;
    goto tanya;
}

//menutup kedua data
data.close();
data2.close();
}

```

Fungsi beli pada program ini adalah user akan menginput data nama pembeli, memilih makanan, dan memasukan jumlah yang dibeli lalu memasukkannya ke dalam file csv. Awal dari fungsi ini akan menampilkan menu makanan yang diambil dari fungsi pada class main dan secara otomatis akan memasukan id pembeli berdasarkan dari jumlah data yang ada di file sehingga diharapkan dapat menghindari data yang double. Setelah itu user akan memasukan pilihan menu yang akan dibeli dan akan secara otomatis tertampilkan. Lalu user akan memasukan jumlah makanan yang akan dibeli dan secara otomatis total harga akan keluar. Setelah itu, user akan diminta untuk memasukan data tersebut atau tidak. Apabila iya maka data akan dimasukan kedalam file csv, apabila tidak maka program akan kembali ke menu utama.

### 3. Fungsi tambah menu

```
void tambahMenu ()
{
    menuMakanan();
    jlhDataMakanan();

    ofstream data("dataMakanan.csv", ios::app);
    fstream data2;
    data2.open("dataMakanan.csv", ios::in);

    vector<string> row;
    string line, word, temp;
    int count = 0;

    cout << "\nMasukkan Menu Makanan Tambahan " << endl;
    cout << "Id\t\t : ";
    cout << jlh_lines + 1 << endl;
    id = jlh_lines + 1;

    int id_tambah = jlh_lines + 1;

    while (getline(data2, line))
    {
        row.clear();
        istringstream s(line);

        while (getline(s, word, ','))
        {
            row.push_back(word);
        }

        if (row[0] != id)
        {
            count = 1;

            cout << "Nama Makanan\t : ";
            cin.ignore();
            getline(cin, namaMakanan);

            cout << "Harga\t\t : Rp";
            cin >> hargaMakanan;
            break;
        }

        else
        {
            count = 0;
            cout << "Maaf, Id yang anda masukkan sudah tersedia. Silahkan Masukkan Id kembali" << endl;
            getchar();
            break;
        }
    }

    tanya:
    char yn;
    cout << "\nAnda yakin ingin memasukkan data tersebut? (y/n) : ";
    cin >> yn;

    if (yn == 'Y' | yn == 'y')
    {
        data << id_tambah << "," << namaMakanan << "," << hargaMakanan << endl;
        cout << "\nPenambahan menu telah berhasil dimasukkan!\n" << endl;
        system("pause");
    }
    else if (yn == 'N' | yn == 'n')
    {
        cout << "\nPenambahan menu dibatalkan!\n" << endl;
        system("pause");
    }
    else
    {
        cout << "Input yang anda masukkan salah!" << endl;
        goto tanya;
    }

    data2.close();
}
```

Fungsi ini berguna untuk menambahkan menu makanan. Sama seperti fungsi sebelumnya, setelah menampilkan menu makanan maka secara otomatis id menu makanan yang baru akan ditampilkan berdasarkan jumlah data makanan pada data file csv lalu ditambah 1. Setelah itu, user akan diminta memasukkan input nama makanan dan harga lalu akan muncul validasi setelahnya untuk memastikan kembali apakah data tersebut ingin dimasukkan kedalam file atau tidak. Apabila iya maka data akan dimasukkan, apabila tidak maka data tidak akan dimasukkan dan kembali ke menu awal.

#### 4. Fungsi edit

```
void edit ()
{
    menuMakanan();

    fstream data_in, data_out;
    data_in.open("dataMakanan.csv", ios::in);
    data_out.open("dataMakananNew.csv", ios::out);

    awal:
    vector<string> row;
    string id_items, line, word;

    cout << "\nMasukkan Id Menu yang akan diubah : ";
    cin >> id_items;

    while (getline(data_in, line))
    {
        row.clear();

        stringstream s(line);

        while (getline(s, word, ','))
        {
            row.push_back(word);
        }

        if (row[0] == id_items)
        {
            count = 1;

            idNew = row[0];
            cout << "Masukkan Nama Menu Yang Baru\t : ";
            cin.ignore();
            getline(cin, namaNew);
            cout << "Masukkan Harga Menu Yang Baru\t : Rp ";
            cin >> hargaNew;
        }
    }
}
```

```

else
{
    if (!data_out.eof())
    {
        for (int i = 0; i < row.size() - 1; i++)
        {
            data_out << row[i] << ",";
        }
        data_out << row[row.size() - 1] << endl;
    }
}

if (data_in.eof())
    break;
}

if (count == 0)
{
    cout << "Id Menu tidak Ditemukan" << endl;
    getchar();
    getchar();
    goto akhir;
}

tanya:
char yn;
cout << "\nAnda yakin ingin memasukkan data tersebut? (y/n) : ";
cin >> yn;

if (yn == 'Y' | yn == 'y')
{
    data_out << idNew << "," << namaNew << "," << hargaNew << endl;
    cout << "\nData Menu telah berhasil di Ubah\n" << endl;
    system("pause");
}

else if (yn == 'N' | yn == 'n')
{
    cout << "\nPenambahan menu dibatalkan!\n" << endl;
    system("pause");
}
else
{
    cout << "Input yang anda masukkan salah!" << endl;
    goto tanya;
}

akhir:
data_in.close();
data_out.close();

remove("dataMakanan.csv");
rename("dataMakananNew.csv", "dataMakanan.csv");
}

```

Fungsi ini digunakan untuk mengedit data makanan yang sudah tersedia. Pada



awalnya fungsi ini akan menampilkan menu makanan lalu akan meminta user untuk menginput id makanan yang mau di edit atau ubah. Fungsi ini akan menggunakan 2 file data yaitu, data makanan awal dan data makanan baru yang nantinya akan diubah namanya menjadi file data awal. Setelah memilih id makanan mana yang ingin diubah maka user akan memasukkan nama yang baru dan juga harga yang baru. Setelah itu akan muncul validasi dan apabila user memilih untuk memasukkan data, data tersebut akan dimasukkan ke dataMakananNew.csv. setelah itu file tersebut akan di rename menjadi dataMakanan.csv, dan data yang lama akan dihapus.

#### 5. Fungsi hapus

```
void hapus ()
{
    menuMakanan();

    fstream data_in, data_out;
    data_in.open("dataMakanan.csv", ios::in);
    data_out.open("dataMakananNew.csv", ios::out);

    vector<string> row;
    string id_items, line, word;

    cout << "\nMasukkan Id Menu yang akan dihapus : ";
    cin >> id_items;

    while (getline(data_in, line))
    {
        row.clear();

        stringstream s(line);

        while (getline(s, word, ','))
        {
            row.push_back(word);
        }

        if (row[0] != id_items)
        {
            if (!data_in.eof())
            {
                for (int i = 0; i < row.size() - 1; i++)
                {
                    data_out << row[i] << ",";
                }
                data_out << row[row.size() - 1] << endl;
            }
        }
    }
}
```

```

        else { count = 1; }

        if (data_in.eof())
            break;
    }

    if (count == 1)
    {
        cout << "Data telah berhasil Dihapus" << endl;
        getchar();
    }
    else
    {
        cout << "Data tidak ditemukan" << endl;
        getchar();
    }

    data_in.close();
    data_out.close();

    remove("dataMakanan.csv");
    rename("dataMakananNew.csv", "dataMakanan.csv");
}

```

Sama seperti fungsi edit(), fungsi hapus ini juga menggunakan 2 file data yaitu file data yang lama dan yang baru yang dimana akan di rename menjadi file data yang lama. Setelah itu, user akan diminta untuk memasukkan id yang ingin dihapus. Berikutnya adalah program akan mencari id tersebut dan apabila ditemukan maka data tersebut akan dihapus dan dibuat menjadi dataMakananNew.csv dan akan di rename menjadi dataMakanan.csv yang dimana file data yang lama sudah di remove.

## 6. Fungsi cari

```

void cari ()
{
    daftarPembelian();

    string data_search, col[4];
    int count = 0;

    input:
    cout << "\nMasukkan nama pembeli\t : ";
    cin >> data_search;

    fstream data;
    data.open("dataPembeli.csv", ios::in);
    vector<string> row;
    string line, word, temp;

    cout << "\t=====\\n";
    cout << "\t| No Pembelian | Nama Makanan | Jumlah | Harga |\\n";
    cout << "\t=====\\n";

    while (getline(data, line))
    {
        row.clear();
        istringstream s(line);

        while (getline(s, word, ','))
        {
            row.push_back(word);
        }

        if (row[1] == data_search)
        {
            count = 1;
            cout << "\t";
            cout << "| " << left << setfill(' ') << setw(7) << row[0] << '|';
            cout << " " << left << setfill(' ') << setw(13) << row[2] << '|';
            cout << " " << left << setfill(' ') << setw(10) << row[3] << '|';
            cout << " " << left << setfill(' ') << setw(10) << row[4] << '|';
            cout << "\\n";
        }
    }
    if (count == 0)
    {
        cout << "\\t\\t\\t Data pembeli tidak Ditemukan" << endl;
    }
    cout << "\\t=====\\n" << endl;

    tanya:
    char yn;
    cout << "\\nApakah Anda ingin mencari data kembali? (y/n) : ";
    cin >> yn;

    if (yn == 'Y' | yn == 'y')
    {
        goto input;
    }
    else if (yn == 'N' | yn == 'n')
    {
        system("pause");
    }
    else
    {
        cout << "Input yang anda masukkan salah!" << endl;
        goto tanya;
    }
    data.close();
}
};

```

Fungsi cari ini digunakan untuk mencari data pembeli. Awalnya user akan diminta untuk memasukkan nama pembeli yang ingin dicari datanya. Setelah itu, program akan membuka dataPembeli.csv lalu akan mencari baris yang sesuai dengan nama yang diinputkan. Setelah itu data dari pembeli tersebut akan muncul seperti nomor pembelian, nama makanan, jumlah, dan total harga. Setelah itu program akan menanyakan apakah user ingin mencari data kembali atau tidak. Apabila iya maka user dapat mencari kembali dan apabila tidak maka akan kembali ke menu awal.

## **BAB III**

### **PENUTUP**

#### **Kesimpulan**

Sering kali kita dapati kantin yang masih menggunakan cara manual untuk mencatat transaksi penjualan. Pada pengerjaan tugas besar Pemrograman Berorientasi Objek kali ini, kami sekelompok mencoba untuk membuat program kantin untuk mempermudah pekerjaan dan menghemat waktu. Dengan begitu kita tidak perlu repot-repot untuk menggunakan catatan dan menghitung lagi produk. Dengan menggunakan metode Pemrograman Berorientasi Objek ini, program akan lebih terstruktur dan mudah untuk dikelola.

#### **Saran**

Untuk menghemat waktu dan tenaga, ada baiknya kantin mulai menjalankan program aplikasi menggunakan computer. Selain dapat memanfaatkan teknologi, kita juga dapat mempermudah pekerjaan kita dan mungkin tidak ada terjadi kesalahan dalam mengingat dan menghitung pembayaran. Program kantin ini masih jauh dari kata sempurna kami memohon maaf apabila masih terjadi kesalahan pada program yang kami buat ini. Kami berharap program ini dapat berguna dan dikembangkan menjadi lebih baik lagi.