

LAPORAN PROYEK AKHIR SEMESTER GASAL



NAMA : NATASYA DELFITA RISTANIA
KELAS : X PPLG 1
NIS : 258724
JUDUL PROYEK : APLIKASI
“ MANAJEMEN KEUANGAN PERUSAHAAAN”

PENGEMBANGAN PERANGKAT LUNAK DAN GIM

SMK NEGERI 1 KANDEMAN

TAHUN PELAJARAN 2025 / 2026

DESKRIPSI PROYEK

Aplikasi Manajemen Keuangan Perusahaan Adalah sebuah program berbasis C# Console Application yang dirancang untuk membantu Perusahaan dalam melakukan pencatatan dan pengelolahan transaksi keuangan secara sederhana namun terstruktur. Aplikasi ini dibuat sebagai bagian dari proyek akhir semester gasal pada mata Pelajaran Pemrograman Terstruktur, dengan tujuan menerapkan konsep dasar seperti :

- Variable
- Operator
- Percabangan
- Perulangan
- array
- serta fungsi atau method

Dalam aplikasi ini, pengguna dapat melakukan berbagai aktivitas seperti pengelolaan keuangan, seperti :

- menambahkan transaksi pemasukan
- pengeluaran
- melihat daftar transaksi
- menghitung total saldo
- serta menghapus seluruh data transaksi

Data transaksi disimpan sementara menggunakan struktur array, sehingga memudahkan proses penyimpanan, perhitungan , dan pencarian dalam aplikasi.

Aplikasi ini dirancang dengan tampilan sederhana berupa menu utama yang memudahkan pengguna memilih fitur fitur yang diinginkan. Setiap fungsi dijalankan melalui metode terpisah untuk mencerminkan prinsip pemrograman tersusunan dan modular. Dengan demikian proyek ini tidak hanya menghasilkan aplikasi yang fungsional, tetapi juga memberikan pengalaman nyata dalam membangun Solusi perangkat lunak yang dapat digunakan di lingkungan Perusahaan.

Secara keseluruhan , proyek ini merupakan implementasi nyata dari materi pembelajaran pemrograman, sekaligus memberikan Gambaran mengenai bagaimana sebuah aplikasi dapat membantu proses pengelolaan keuangan secara lebih efisien dan terorganisir.

RINGKASAN TEORI C# YANG DIGUNAKAN

1. Variabel dan Tipe Data

Variabel digunakan untuk menyimpan data yang akan digunakan dalam program.

Contoh tipe data yang sering digunakan dalam aplikasi manajemen keuangan Perusahaan.

Contoh:

```
static string[] namaTransaksi = new string[100];  
static int[] jumlahTransaksi = new int[100];  
static string[] jenisTransaksi = new string[100];  
static int jumlahData = 0;
```

- int → bilangan bulat (ID transaksi)
- string → teks (nama transaksi , jenis transaksi)

2. Operator

Operator digunakan untuk perhitungan atau perbandingan.

Contoh:

```
if (jenisTransaksi[i].ToLower() == "pemasukan")  
    totalPemasukan += jumlahTransaksi[i];  
else if (jenisTransaksi[i].ToLower() == "pengeluaran")  
    totalPengeluaran += jumlahTransaksi[i];
```

Operator yang muncul:

- Aritmetika: +, +=
- Perbandingan: ==(di bagian menu)

3. Percabangan

Percabangan yang digunakan.

```
// PERCABANGAN IF-ELSE
if (jenisTransaksi[i].ToLower() == "pemasukan")
    totalPemasukan += jumlahTransaksi[i];
else if (jenisTransaksi[i].ToLower() == "pengeluaran")
    totalPengeluaran += jumlahTransaksi[i];
}

// =====
// 4 PERCABANGAN SWITCH (menjalankan menu)
// =====

switch (pilihan)
{
    case 1:
        TambahTransaksi();
        break;
    case 2:
        LihatTransaksi();
        break;
    case 3:
        HitungSaldo();
        break;
    case 4:
        HapusData();
        break;
    case 5:
        Console.WriteLine("Terima kasih telah menggunakan program ini!");
        break;
    default:
        Console.WriteLine("Pilihan tidak valid. Coba lagi!");
        break;
}
```

4. Perulangan

```
// PERULANGAN FOR untuk menampilkan semua data
    for (int i = 0; i < jumlahData; i++)
    {
        Console.WriteLine($"{i + 1}. {namaTransaksi[i]} - 
{jenisTransaksi[i]} Rp{jumlahTransaksi[i]}");
    }
    Console.WriteLine();

// PERULANGAN FOR untuk menghitung total
for (int i = 0; i < jumlahData; i++)
{
    // PERCABANGAN IF-ELSE
    if (jenisTransaksi[i].ToLower() == "pemasukan")
        totalPemasukan += jumlahTransaksi[i];
    else if (jenisTransaksi[i].ToLower() == "pengeluaran")
        totalPengeluaran += jumlahTransaksi[i];
}

int saldoAkhir = totalPemasukan - totalPengeluaran;
Console.WriteLine("== LAPORAN KEUANGAN PERUSAHAAN ==");
Console.WriteLine($"Total Pemasukan : Rp{totalPemasukan}");
Console.WriteLine($"Total Pengeluaran : Rp{totalPengeluaran}");
Console.WriteLine($"Saldo Akhir : Rp{saldoAkhir}\n");

}

// PERULANGAN WHILE untuk menghapus isi array
while (i < jumlahData)
{
    namaTransaksi[i] = "";
    jumlahTransaksi[i] = 0;
    jenisTransaksi[i] = "";
    i++;
}

jumlahData = 0; // Reset jumlah data
Console.WriteLine("Semua data transaksi berhasil
dihapus!\n");
```

```

        }
    else
    {
        Console.WriteLine("Penghapusan dibatalkan.\n");
    }
}

```

5. Array

```

// [5] FUNGSI: TAMBAH TRANSAKSI
// =====
static void TambahTransaksi()
{
    Console.Write("Masukkan nama transaksi: ");
    namaTransaksi[jumlahData] = Console.ReadLine();

    Console.Write("Masukkan jumlah uang: ");
    jumlahTransaksi[jumlahData] = int.Parse(Console.ReadLine());

    Console.Write("Jenis transaksi (pemasukan/pengeluaran): ");
    jenisTransaksi[jumlahData] = Console.ReadLine();

    jumlahData++;
    Console.WriteLine("Transaksi berhasil disimpan!\n");
}

```

6. Fungsi atau Method

```

// =====
// [5] FUNGSI: TAMBAH TRANSAKSI
// =====

static void TambahTransaksi()
{
    Console.Write("Masukkan nama transaksi: ");
    namaTransaksi[jumlahData] = Console.ReadLine();

    Console.Write("Masukkan jumlah uang: ");
    jumlahTransaksi[jumlahData] = int.Parse(Console.ReadLine());

    Console.Write("Jenis transaksi (pemasukan/pengeluaran): ");

```

```

jenisTransaksi[jumlahData] = Console.ReadLine();

jumlahData++;
Console.WriteLine("Transaksi berhasil disimpan!\n");
}

// =====
// 6 FUNGSI: LIHAT SEMUA TRANSAKSI
// =====

static void LihatTransaksi()
{
    // PERCABANGAN IF-ELSE
    if (jumlahData == 0)
    {
        Console.WriteLine("Belum ada transaksi yang tercatat.\n");
    }
    else
    {
        Console.WriteLine("==== DAFTAR TRANSAKSI ===");

        // PERULANGAN FOR untuk menampilkan semua data
        for (int i = 0; i < jumlahData; i++)
        {
            Console.WriteLine($"{i + 1}. {namaTransaksi[i]} - {jenisTransaksi[i]}"
Rp{jumlahTransaksi[i]}");

        }
        Console.WriteLine();
    }
}

```

7. Input dan Output

```

// =====
// 3 INPUT & OUTPUT (menampilkan menu ke layar)

```

```

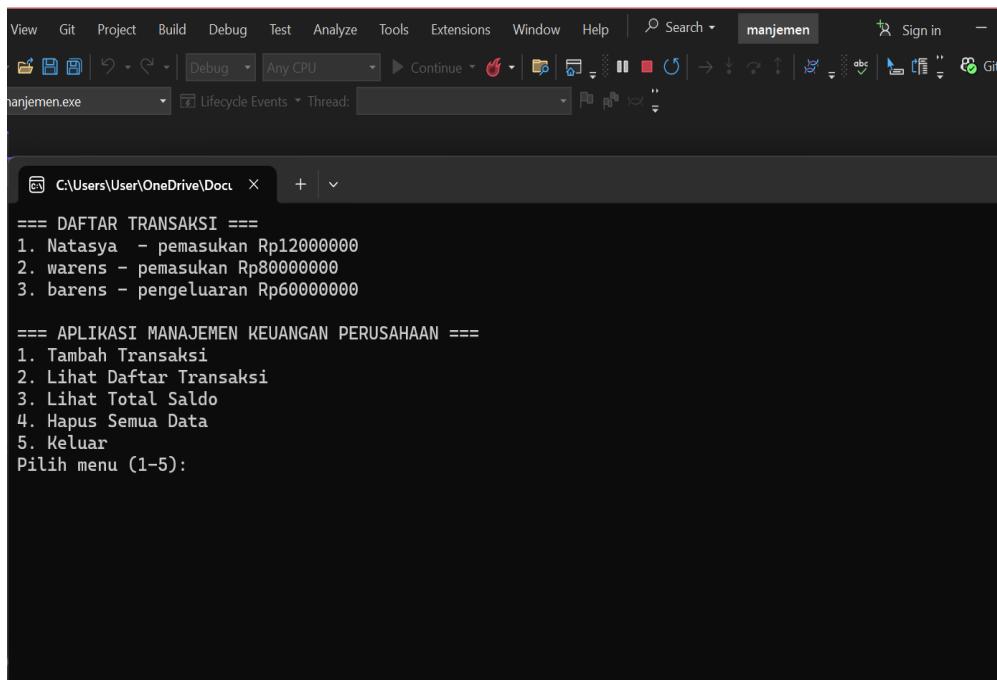
// =====
Console.WriteLine("== APLIKASI MANAJEMEN KEUANGAN PERUSAHAAN ==");
Console.WriteLine("1. Tambah Transaksi");
Console.WriteLine("2. Lihat Daftar Transaksi");
Console.WriteLine("3. Lihat Total Saldo");
Console.WriteLine("4. Hapus Semua Data");
Console.WriteLine("5. Keluar");
Console.Write("Pilih menu (1-5): ");
pilihan = int.Parse(Console.ReadLine());

Console.Clear(); // Membersihkan layar agar lebih rapi

```

PERANCANGAN PROGRAM

Rancangan Aplikasi manajemen Persuahaan.



```

C:\Users\User\OneDrive\Docx > +
== DAFTAR TRANSAKSI ==
1. Natasya - pemasukan Rp12000000
2. warens - pemasukan Rp80000000
3. barens - pengeluaran Rp60000000

== APLIKASI MANAJEMEN KEUANGAN PERUSAHAAN ==
1. Tambah Transaksi
2. Lihat Daftar Transaksi
3. Lihat Total Saldo
4. Hapus Semua Data
5. Keluar
Pilih menu (1-5):

```

IMPLEMENTASI PROGRAM

Kode utama:

```
using System;
```

```
using System.Collections.Generic;
```

```
using System.Linq;
```

```
using System.Text;
using System.Threading.Tasks;

namespace manjemen
{
    internal class Program
    {
        // =====
        // 1 VARIABEL DAN ARRAY

        // =====
        static string[] namaTransaksi = new string[100];
        static int[] jumlahTransaksi = new int[100];
        static string[] jenisTransaksi = new string[100];
        static int jumlahData = 0;

        static void Main()
        {
            int pilihan; // Variabel untuk menyimpan pilihan menu pengguna

            // =====
            // 2 PERULANGAN DO-WHILE untuk menu utama
            // =====

            do
            {
```

```
// =====  
  
// 3 INPUT & OUTPUT (menampilkan menu ke layar)  
  
// =====  
  
Console.WriteLine("==== APLIKASI MANAJEMEN KEUANGAN  
PERUSAHAAN ===");  
  
Console.WriteLine("1. Tambah Transaksi");  
  
Console.WriteLine("2. Lihat Daftar Transaksi");  
  
Console.WriteLine("3. Lihat Total Saldo");  
  
Console.WriteLine("4. Hapus Semua Data");  
  
Console.WriteLine("5. Keluar");  
  
Console.Write("Pilih menu (1-5): ");  
  
pilihan = int.Parse(Console.ReadLine());  
  
  
Console.Clear(); // Membersihkan layar agar lebih rapi  
  
// =====  
  
// 4 PERCABANGAN SWITCH (menjalankan menu)  
  
// =====  
  
switch (pilihan)  
{  
    case 1:  
        TambahTransaksi();  
        break;  
  
    case 2:  
        break;  
}
```

```
LihatTransaksi();  
break;  
  
case 3:  
  
    HitungSaldo();  
    break;  
  
case 4:  
  
    HapusData();  
    break;  
  
case 5:  
  
    Console.WriteLine("Terima kasih telah menggunakan program ini!");  
    break;  
  
default:  
  
    Console.WriteLine("Pilihan tidak valid. Coba lagi!");  
    break;  
}  
  
}  
  
} while (pilihan != 5); // PERULANGAN berhenti jika pengguna memilih keluar  
}  
  
// ======  
  
// 5 FUNGSI: TAMBAH TRANSAKSI  
// ======  
  
static void TambahTransaksi()  
{
```

```
Console.Write("Masukkan nama transaksi: ");

namaTransaksi[jumlahData] = Console.ReadLine();

Console.Write("Masukkan jumlah uang: ");

jumlahTransaksi[jumlahData] = int.Parse(Console.ReadLine());

Console.Write("Jenis transaksi (pemasukan/pengeluaran): ");

jenisTransaksi[jumlahData] = Console.ReadLine();

jumlahData++;

Console.WriteLine("Transaksi berhasil disimpan!\n");

}
```

```
// =====

// 6 FUNGSI: LIHAT SEMUA TRANSAKSI

// =====

static void LihatTransaksi()

{

    // PERCABANGAN IF-ELSE

    if (jumlahData == 0)

    {

        Console.WriteLine("Belum ada transaksi yang tercatat.\n");

    }

    else
```

```
{  
    Console.WriteLine("==== DAFTAR TRANSAKSI ===");  
  
    // PERULANGAN FOR untuk menampilkan semua data  
  
    for (int i = 0; i < jumlahData; i++)  
  
    {  
        Console.WriteLine($"{i + 1}. {namaTransaksi[i]} - {jenisTransaksi[i]}  
Rp{jumlahTransaksi[i]}");  
  
    }  
  
    Console.WriteLine();  
  
}  
  
// ======  
  
// 7 FUNGSI: HITUNG TOTAL SALDO  
  
// ======  
  
static void HitungSaldo()  
  
{  
    int totalPemasukan = 0;  
  
    int totalPengeluaran = 0;  
  
    // PERULANGAN FOR untuk menghitung total  
  
    for (int i = 0; i < jumlahData; i++)  
  
    {  
        // PERCABANGAN IF-ELSE
```

```

        if (jenisTransaksi[i].ToLower() == "pemasukan")
            totalPemasukan += jumlahTransaksi[i];

        else if (jenisTransaksi[i].ToLower() == "pengeluaran")
            totalPengeluaran += jumlahTransaksi[i];

    }

int saldoAkhir = totalPemasukan - totalPengeluaran;

Console.WriteLine("== LAPORAN KEUANGAN PERUSAHAAN ==");

Console.WriteLine($"Total Pemasukan : Rp {totalPemasukan}");

Console.WriteLine($"Total Pengeluaran : Rp {totalPengeluaran}");

Console.WriteLine($"Saldo Akhir : Rp {saldoAkhir}\n");

}

// =====
// 8 FUNGSI: HAPUS SEMUA DATA TRANSAKSI
// =====

static void HapusData()

{
    Console.Write("Apakah yakin ingin menghapus semua data? (y/n): ");

    string konfirmasi = Console.ReadLine();

    // PERCABANGAN IF-ELSE

    if (konfirmasi.ToLower() == "y")
    {

```

```
int i = 0;

// PERULANGAN WHILE untuk menghapus isi array

while (i < jumlahData)

{
    namaTransaksi[i] = "";
    jumlahTransaksi[i] = 0;
    jenisTransaksi[i] = "";
    i++;
}

jumlahData = 0; // Reset jumlah data

Console.WriteLine("Semua data transaksi berhasil dihapus!\n");

}

else

{
    Console.WriteLine("Penghapusan dibatalkan.\n");
}

}

}

}
```

PENJELASAN KODE PROGRAM:

1. Deklarasi Namespace dan Class

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace manjemen
{
    internal class Program
```

- Using System : → digunakan untuk memerintahkan dasar C# seperti Console.WriteLine() dan Console.ReadLine().
- Namespace manajemen → ruang lingkup program agar tidak bentrok dengan program lain.
- Internal class Program → Kelas utama sebagai tempat kode program dijalankan.

2. Fungsi hitung total()

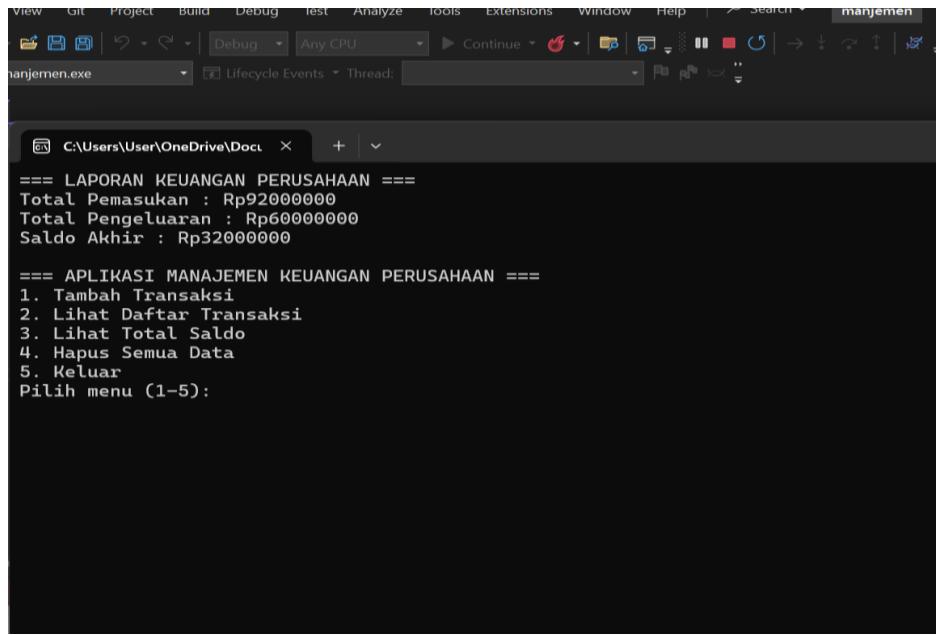
```
static void HitungSaldo()
{
    int totalPemasukan = 0;
    int totalPengeluaran = 0;

    // PERULANGAN FOR untuk menghitung total
    for (int i = 0; i < jumlahData; i++)
    {
        // PERCABANGAN IF-ELSE
        if (jenisTransaksi[i].ToLower() == "pemasukan")
            totalPemasukan += jumlahTransaksi[i];
        else if (jenisTransaksi[i].ToLower() == "pengeluaran")
            totalPengeluaran += jumlahTransaksi[i];
    }

    int saldoAkhir = totalPemasukan - totalPengeluaran;
    Console.WriteLine("== LAPORAN KEUANGAN PERUSAHAAN ==");
    Console.WriteLine($"Total Pemasukan : Rp{totalPemasukan}");
    Console.WriteLine($"Total Pengeluaran : Rp{totalPengeluaran}");
    Console.WriteLine($"Saldo Akhir : Rp{saldoAkhir}\n");
}
```

- Fungsi ini berfungsi menghitung total pemasukan dan pengeluaran dan sisa saldo.
- Saldo dan sisa saldo akan muncul disaldo akhir .

Contoh:



3. Percabangan SWITCH (menjalankan menu)

```
// =====
// 4 PERCABANGAN SWITCH (menjalankan menu)
// =====

switch (pilihan)
{
    case 1:
        TambahTransaksi();
        break;
    case 2:
        LihatTransaksi();
        break;
    case 3:
        HitungSaldo();
        break;
    case 4:
        HapusData();
        break;
    case 5:
```

```

        Console.WriteLine("Terima kasih telah menggunakan program ini!");
        break;
    default:
        Console.WriteLine("Pilihan tidak valid. Coba lagi!");
        break;
    }

} while (pilihan != 5); // PERULANGAN berhenti jika pengguna memilih keluar
}

```

- Program ini akan menjalaankan menu yang dipilih

4. Input & Output (menampilkan menu ke layer)

```

// =====
// 3 INPUT & OUTPUT (menampilkan menu ke layar)
// =====

Console.WriteLine("== APLIKASI MANAJEMEN KEUANGAN
PERUSAHAAN ==");
Console.WriteLine("1. Tambah Transaksi");
Console.WriteLine("2. Lihat Daftar Transaksi");
Console.WriteLine("3. Lihat Total Saldo");
Console.WriteLine("4. Hapus Semua Data");
Console.WriteLine("5. Keluar");
Console.Write("Pilih menu (1-5): ");
pilihan = int.Parse(Console.ReadLine());

Console.Clear(); // Membersihkan layar agar lebih rapi

```

- Program ini yang akan ditampilkan jika start dimulai.
- Pengguna bisa memilih salah satu.

5. Fungsi tambah transaksi

```

// =====
// [5] FUNGSI: TAMBAH TRANSAKSI
// =====
static void TambahTransaksi()
{
    Console.WriteLine("Masukkan nama transaksi: ");
    namaTransaksi[jumlahData] = Console.ReadLine();

    Console.WriteLine("Masukkan jumlah uang: ");
    jumlahTransaksi[jumlahData] = int.Parse(Console.ReadLine());

    Console.WriteLine("Jenis transaksi (pemasukan/pengeluaran): ");
    jenisTransaksi[jumlahData] = Console.ReadLine();

    jumlahData++;
    Console.WriteLine("Transaksi berhasil disimpan!\n");
}

```

- Program ini akan menampilkan menu tampilan .
- Dapat menambah transaksi .

6. Fungsi lihat semua transaksi

```

// =====
// [6] FUNGSI: LIHAT SEMUA TRANSAKSI
// =====

static void LihatTransaksi()
{
    // PERCABANGAN IF-ELSE
    if (jumlahData == 0)
    {
        Console.WriteLine("Belum ada transaksi yang tercatat.\n");
    }
    else
    {
        Console.WriteLine("==== DAFTAR TRANSAKSI ===");

        // PERULANGAN FOR untuk menampilkan semua data
        for (int i = 0; i < jumlahData; i++)
        {

```

```

Console.WriteLine($" {i + 1}. {namaTransaksi[i]} - {jenisTransaksi[i]}}
Rp {jumlahTransaksi[i]}");
}

Console.WriteLine();

```

- Program ini akan menampilkan seluruh transaksi yang sudah dibuat.

7. Hapus semua data transaksi

```

// =====
// 8 FUNGSI: HAPUS SEMUA DATA TRANSAKSI
// =====

static void HapusData()
{
    Console.Write("Apakah yakin ingin menghapus semua data? (y/n): ");
    string konfirmasi = Console.ReadLine();

    // PERCABANGAN IF-ELSE
    if (konfirmasi.ToLower() == "y")
    {
        int i = 0;

        // PERULANGAN WHILE untuk menghapus isi array
        while (i < jumlahData)
        {
            namaTransaksi[i] = "";
            jumlahTransaksi[i] = 0;
            jenisTransaksi[i] = "";
            i++;
        }

        jumlahData = 0; // Reset jumlah data
    }
}

```

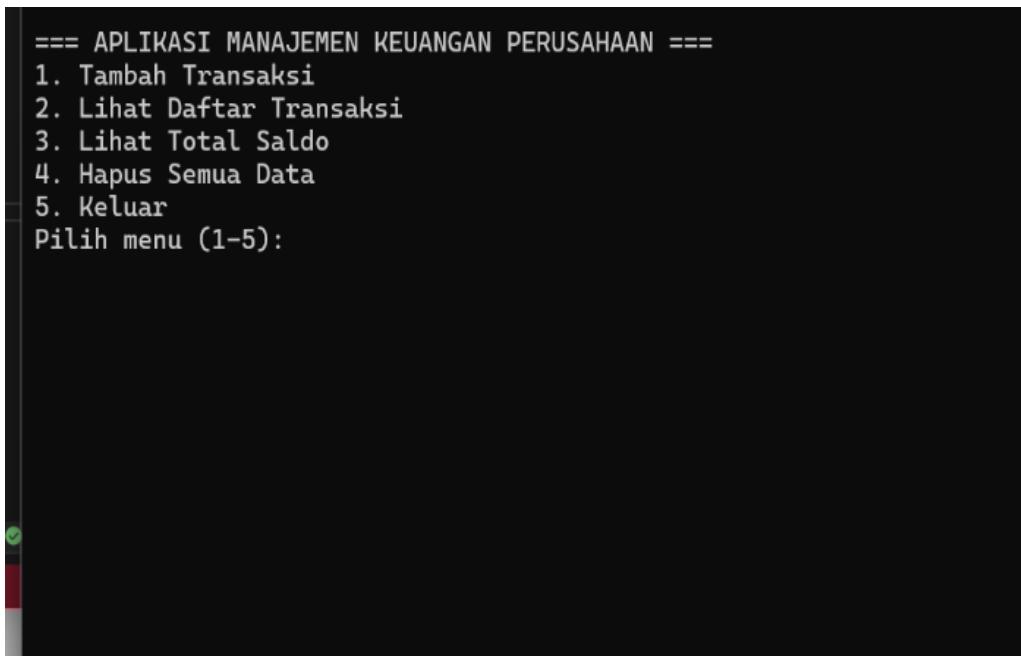
```

        Console.WriteLine("Semua data transaksi berhasil dihapus!\n");
    }
    else
    {
        Console.WriteLine("Penghapusan dibatalkan.\n");
    }
}
}

```

- Program ini berfungsi menghapus data transaksi.
- Pengguna dapat memilih jawaban.
- Program tersebut juga sebagai penanda akhir pengguna dapat menekan tombol bebas.

TAMPILAN HASIL PROGRAM



```

== APLIKASI MANAJEMEN KEUANGAN PERUSAHAAN ==
1. Tambah Transaksi
2. Lihat Daftar Transaksi
3. Lihat Total Saldo
4. Hapus Semua Data
5. Keluar
Pilih menu (1-5):

```

```
0] manjemen.exe Lifecycle Events Thread: 
x
C:\Users\User\OneDrive\Docl + ▾

Masukkan nama transaksi: warens
Masukkan jumlah uang: 80000000
Jenis transaksi (pemasukan/pengeluaran): pemasukan
Transaksi berhasil disimpan!

== APLIKASI MANAJEMEN KEUANGAN PERUSAHAAN ==
1. Tambah Transaksi
2. Lihat Daftar Transaksi
3. Lihat Total Saldo
4. Hapus Semua Data
5. Keluar
Pilih menu (1-5):
```

```
^
x
C:\Users\User\OneDrive\Docl + ▾

Masukkan nama transaksi: barens
Masukkan jumlah uang: 60000000
Jenis transaksi (pemasukan/pengeluaran): pengeluaran
Transaksi berhasil disimpan!

== APLIKASI MANAJEMEN KEUANGAN PERUSAHAAN ==
1. Tambah Transaksi
2. Lihat Daftar Transaksi
3. Lihat Total Saldo
4. Hapus Semua Data
5. Keluar
Pilih menu (1-5):
```

```
^
x
C:\Users\User\OneDrive\Docl + ▾

Masukkan nama transaksi: Natasya
Masukkan jumlah uang: 120000000
Jenis transaksi (pemasukan/pengeluaran): pemasukan
Transaksi berhasil disimpan!

== APLIKASI MANAJEMEN KEUANGAN PERUSAHAAN ==
1. Tambah Transaksi
2. Lihat Daftar Transaksi
3. Lihat Total Saldo
4. Hapus Semua Data
5. Keluar
Pilih menu (1-5):
```

```
Program.cs X
manjemen C:\Users\User\OneDrive\Docx + ▾
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96

    === DAFTAR TRANSAKSI ===
    1. Natasya - pemasukan Rp12000000
    2. warens - pemasukan Rp80000000
    3. barens - pengeluaran Rp60000000

    === APLIKASI MANAJEMEN KEUANGAN PERUSAHAAN ===
    1. Tambah Transaksi
    2. Lihat Daftar Transaksi
    3. Lihat Total Saldo
    4. Hapus Semua Data
    5. Keluar
Pilih menu (1-5):
```

```
× C:\Users\User\OneDrive\Docx + ▾
    === LAPORAN KEUANGAN PERUSAHAAN ===
    Total Pemasukan : Rp92000000
    Total Pengeluaran : Rp60000000
    Saldo Akhir : Rp32000000

    === APLIKASI MANAJEMEN KEUANGAN PERUSAHAAN ===
    1. Tambah Transaksi
    2. Lihat Daftar Transaksi
    3. Lihat Total Saldo
    4. Hapus Semua Data
    5. Keluar
Pilih menu (1-5):
```

```
× C:\Users\User\OneDrive\Docx + ▾
Apakah yakin ingin menghapus semua data? (y/n): y
Semua data transaksi berhasil dihapus!

    === APLIKASI MANAJEMEN KEUANGAN PERUSAHAAN ===
    1. Tambah Transaksi
    2. Lihat Daftar Transaksi
    3. Lihat Total Saldo
    4. Hapus Semua Data
    5. Keluar
Pilih menu (1-5):
```

LAMPIRAN:

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace APLIKASI_KELUARGA
8  {
9      internal class Program
10     {
11         // =====
12         // 10 VARIABEL DAN ARRAY
13         // =====
14         static string[] namaTransaksi = new string[100];
15         static int[] jumlahTransaksi = new int[100];
16         static string[] jenisTransaksi = new string[100];
17         static int jumlahData = 0;
18
19         static void Main()
20         {
21             int pilihan; // Variabel untuk menyimpan pilihan menu pengguna
22
23             // =====
24             // 20 PERULANGAN DO-WHILE untuk menu utama
25             // =====
26             do
27             {
28                 // =====
29                 // 30 INPUT & OUTPUT (menampilkan menu ke layar)
30                 // =====
31                 Console.WriteLine("== APLIKASI MANAJEMEN KELUARGA PERUSAHAAN ==");
32                 Console.WriteLine("1. Tambah Transaksi");
33                 Console.WriteLine("2. Lihat Daftar Transaksi");
34                 Console.WriteLine("3. Lihat Total Saldo");
35                 Console.WriteLine("4. Hapus Semua Data");
36                 Console.WriteLine("5. Keluar");
37                 Console.Write("Pilih menu (1-5): ");
38                 pilihan = int.Parse(Console.ReadLine());
39
40                 Console.Clear(); // Membersihkan layar agar lebih rapi
41
42                 // =====
43                 // 40 PERCABANGAN SWITCH (menjalankan menu)
44                 // =====
45                 switch (pilihan)
46                 {
47                     case 1:
48                         TambahTransaksi();
49                         break;
50                     case 2:
51                         LihatTransaksi();
52                         break;
53                     case 3:
54                         HitungSaldo();
55                         break;
56                     case 4:
57                         HapusData();
58                         break;
59                     case 5:
60                         Console.WriteLine("Terima kasih telah menggunakan program ini!");
61                         break;
62                     default:
63                         Console.WriteLine("Pilihan tidak valid. Coba lagi!");
64                         break;
65                 }
66
67             } while (pilihan != 5); // PERULANGAN berhenti jika pengguna memilih keluar
68
69             // =====
70             // 50 FUNGSI: TAMBAH TRANSAKSI
71             // =====
72             static void TambahTransaksi()
73             {
74                 Console.Write("Masukkan nama transaksi: ");
75                 namaTransaksi[jumlahData] = Console.ReadLine();
76
77                 Console.Write("Masukkan jumlah uang: ");
78                 jumlahTransaksi[jumlahData] = int.Parse(Console.ReadLine());
79
80                 Console.Write("Jenis transaksi (pemasukan/pengeluaran): ");
81             }
82         }
83     }
84 }
```

```

77         Console.WriteLine("masukkan jumlah uang: ");
78         jumlahTransaksi[jumlahData] = int.Parse(Console.ReadLine());
79
80         Console.WriteLine("Jenis transaksi (pemasukan/pengeluaran): ");
81         jenisTransaksi[jumlahData] = Console.ReadLine();
82
83         jumlahData++;
84         Console.WriteLine("Transaksi berhasil disimpan\n");
85     }
86
87     // -----
88     // ED FUNGSI: LIHAT SEMUA TRANSAKSI
89     // -----
90     // reference
91     static void LihatTransaksi()
92     {
93         // PERCABANGAN IF-ELSE
94         if (jumlahData == 0)
95         {
96             Console.WriteLine("Belum ada transaksi yang tercatat.\n");
97         }
98         else
99         {
100            Console.WriteLine("==== DAFTAR TRANSAKSI ====");
101
102            // PERULANGAN FOR untuk menampilkan semua data
103            for (int i = 0; i < jumlahData; i++)
104            {
105                Console.WriteLine($"{i + 1}. {namaTransaksi[i]} - {jenisTransaksi[i]} Rp{jumlahTransaksi[i]}");
106            }
107            Console.WriteLine();
108        }
109
110    // -----
111    // TD FUNGSI: HITUNG TOTAL SALDO
112    // -----
113    // reference
114    static void HitungSaldo()
115    {
116        int totalPemasukan = 0;
117        int totalPengeluaran = 0;
118
119        // PERULANGAN FOR untuk menghitung total
120        for (int i = 0; i < jumlahData; i++)
121        {
122            // PERCABANGAN IF-ELSE
123            if (jenisTransaksi[i].ToLower() == "pemasukan")
124                totalPemasukan += jumlahTransaksi[i];
125            else if (jenisTransaksi[i].ToLower() == "pengeluaran")
126                totalPengeluaran += jumlahTransaksi[i];
127
128            int saldoAkhir = totalPemasukan - totalPengeluaran;
129            Console.WriteLine("==== LAPORAN KEUANGAN PERUSAHAAN ====");
130            Console.WriteLine($"Total Pemasukan : Rp{totalPemasukan}");
131            Console.WriteLine($"Total Pengeluaran : Rp{totalPengeluaran}");
132            Console.WriteLine($"Saldo Akhir : Rp{saldoAkhir}\n");
133        }
134
135    // -----
136    // ED FUNGSI: HAPUS SEMUA DATA TRANSAKSI
137    // -----
138    // reference
139    static void HapusData()
140    {
141        Console.Write("Apakah yakin ingin menghapus semua data? (y/n): ");
142        string konfirmasi = Console.ReadLine();
143
144        // PERCABANGAN IF-ELSE
145        if (konfirmasi.ToLower() == "y")
146        {
147            int i = 0;
148
149            // PERULANGAN WHILE untuk menghapus isi array
150            while (i < jumlahData)
151            {
152                namaTransaksi[i] = "";
153                jumlahTransaksi[i] = 0;
154                jenisTransaksi[i] = "";
155                i++;
156            }
157
158            jumlahData = 0; // Reset jumlah data
159            Console.WriteLine("Semua data transaksi berhasil dihapus!\n");
160        }
161        else
162        {
163            Console.WriteLine("Penghapusan dibatalkan.\n");
164        }
165    }

```