

Nama : M. Bintang Royyan
NIM : 242410102040
Kelas : A

Soal

Kelas A

Sistem Manajemen Perpustakaan Mini

Sebuah perpustakaan kecil ingin membuat sistem manajemen sederhana menggunakan aplikasi console. Dalam sistem ini, terdapat berbagai jenis buku seperti Buku Fiksi, Buku Non-Fiksi, dan Majalah. Setiap buku memiliki informasi dasar seperti judul, penulis, dan tahun terbit.

Petugas ingin dapat melakukan hal-hal berikut:

- Menambahkan buku ke dalam daftar perpustakaan.
- Mengubah data buku
- Melihat informasi setiap buku secara umum.
- Menyimpan data buku dalam daftar menggunakan
- Meminjam buku (maksimal 3 buku).
- Mengembalikan buku.
- Melihat daftar buku yang sedang dipinjam.

Syarat : Program menerapkan 4 pilar OOP yaitu inheritance, enkapsulasi, polymorphism, abstraction dan interface.

A. interface

```
1 reference
interface iBuku
{
    6 references
    void DisplayInfo();
}
```

interface digunakan karena kita akan menerapkan multiple inheritance dan akan membuat list objek yang berbeda dengan behaviour yang sama

B. class Buku

```

16 references
abstract class Buku : iBuku
{
    7 references
    public string Judul { get; private set; }
    5 references
    public string Penulis { get; private set; }
    5 references
    public int Tahun_terbit { get; private set; }
    4 references
    public bool Dipinjam { get; set; }

    3 references
    public Buku(string judul, string penulis, int tahun_terbit)
    {
        Judul = judul;
        Penulis = penulis;
        Tahun_terbit = tahun_terbit;
        Dipinjam = false;
    }

    6 references
    public abstract void DisplayInfo();
    1 reference
    public void Edit(string judul, string penulis, int tahun_terbit)
    {
        Judul = judul;
        Penulis = penulis;
        Tahun_terbit = tahun_terbit;
    }
}

```

class Buku digunakan sebagai cetakan untuk semua jenis buku yang menyimpan informasi tentang Judul, Penulis, Tahun_terbit, dan Dipinjam. Dan terdapat enkapsulasi yang dimana pada property Judul, Penulis, dan Tahun_terbit memuat “private set” agar property tersebut tidak dapat di ubah di luar dari class tersebut dan harus melalui method Edit() untuk melakukan pengeditan.

C. subclass BukuFiksi, BukuNonFiksi, dan Majalah

```

2 references
class BukuFiksi : Buku
{
    1 reference
    public BukuFiksi(string judul, string penulis, int tahun_terbit) : base(judul, penulis, tahun_terbit) { }
    4 references
    public override void DisplayInfo()
    {
        Console.WriteLine($"[Buku Fiksi] {Judul} Ditulis oleh {Penulis},{Tahun_terbit}");
    }
}

2 references
class BukuNonFiksi : Buku
{
    1 reference
    public BukuNonFiksi(string judul, string penulis, int tahun_terbit) : base(judul, penulis, tahun_terbit) { }
    4 references
    public override void DisplayInfo()
    {
        Console.WriteLine($"[Buku Non - Fiksi] {Judul} Ditulis oleh {Penulis},{Tahun_terbit}");
    }
}

2 references
class Majalah : Buku
{
    1 reference
    public Majalah(string judul, string penulis, int tahun_terbit) : base(judul, penulis, tahun_terbit) { }
    4 references
    public override void DisplayInfo()
    {
        Console.WriteLine($"[Majalah] {Judul} Ditulis oleh {Penulis},{Tahun_terbit}");
    }
}

```

Turunan dari class Buku ini menggunakan override agar tampilannya sesuai dengan jenis buku masing masing.

D. class Pengguna

```
class Pengguna
{
    7 references
    public List<Buku> Buku_dipinjam { get; } = new List<Buku>();
    1 reference
    public bool Pinjam(Buku buku)
    {
        if (Buku_dipinjam.Count >= 3)
        {
            Console.WriteLine("Hanya dapat meminjam 3 buku!");
            return false;
        }
        if (buku.Dipinjam)
        {
            Console.WriteLine("Buku sedang dipinjam!");
            return false;
        }
        buku.Dipinjam = true;
        Buku_dipinjam.Add(buku);
        Console.WriteLine($"Berhasil meminjam buku: {buku.GetType().Name} - {buku.Judul}");
        return true;
    }
    1 reference
    public void Kembali(Buku buku)
    {
        if (Buku_dipinjam.Contains(buku))
        {
            buku.Dipinjam = false;
            Buku_dipinjam.Remove(buku);
            Console.WriteLine($"Buku {buku.Judul} berhasil dikembalikan!");
        }
        else
        {
            Console.WriteLine("Buku tidak sedang dipinjam");
        }
    }
}

2 references
public void LiatDipinjam()
{
    Console.WriteLine("Buku yang sedang dipinjam:");
    foreach (var book in Buku_dipinjam)
    {
        book.DisplayInfo();
    }
}
```

class Pengguna digunakan untuk menyimpan list buku yang sedang, akan, dan telah dipinjam yang menggunakan List<Buku> untuk menciptakan list. Pada method Pinjam() akan memberikan operator kondisi, jika telah meminjam lebih dari 3 buku, maka sistem akan mengeluarkan output "Hanya dapat meminjam 3 buku!" hal ini dilakukan agar pengguna tidak dapat meminjam lebih dari 3 buku. Lalu, jika pengguna meminjam kurang dari 3 buku, sistem akan menyimpan informasi tersebut ke dalam list Buku_dipinjam. Pada method Kembali() digunakan oleh pengguna untuk mengembalikan buku, terdapat operator kondisi jika, dalam list Buku_dipinjam terdapat objek buku, maka buku tersebut akan dihapus dari

list tersebut dan mengganti boolean Dipinjam menjadi false. Jika pada list Buku_dipinjam tidak ada objek buku, maka sistem akan mengeluarkan output “Buku tidak sedang dipinjam”. Method LiatDipinjam() berfungsi untuk melihat semua buku yang sedang dipinjam oleh pengguna.

E. class Perpus

```
class Perpus
{
    7 references
    public List<Buku> Buku { get; } = new List<Buku>();

    1 reference
    public void TambahBuku(Buku buku)
    {
        Buku.Add(buku);
        Console.WriteLine("Buku berhasil ditambahkan!");
    }

    1 reference
    public void EditBuku(int index, string judul, string penulis, int tahun_terbit)
    {
        if (index >= 0 && index < Buku.Count)
        {
            Buku[index].Edit(judul, penulis, tahun_terbit);
            Console.WriteLine("Buku berhasil diubah!");
        }
        else
        {
            Console.WriteLine("Indeks tidak valid.");
        }
    }

    3 references
    public void LiatSemua()
    {
        for (int i = 0; i < Buku.Count; i++)
        {
            Console.WriteLine($"{i + 1}. ");
            Buku[i].DisplayInfo();
        }
    }

    1 reference
    public Buku? GetBookByIndex(int index)
    {
        return (index >= 0 && index < Buku.Count) ? Buku[index] : null;
    }
}
```

class Perpus digunakan untuk menyimpan semua list buku, menambah buku, mengedit buku, dan menampilkan semua buku. Method TambahBuku() digunakan untuk menambah buku ke dalam objek buku. Method EditBuku() digunakan untuk mengedit buku seperti mengedit Judul, Penulis, dan Tahun terbitnya. Method LiatSemua() digunakan untuk menunjukkan semua buku.

F. class Program

```
0 references
class Program
{
    0 references
    static void Main()
    {
        Perpus perpustakaan = new Perpus();
        Pengguna pengguna = new Pengguna();

        while (true)
        {
            Console.WriteLine("\n--- Menu ---");
            Console.WriteLine("1. Tambah Buku");
            Console.WriteLine("2. Ubah Buku");
            Console.WriteLine("3. Lihat Semua Buku");
            Console.WriteLine("4. Pinjam Buku");
            Console.WriteLine("5. Kembalikan Buku");
            Console.WriteLine("6. Lihat Buku yang Dipinjam");
            Console.WriteLine("0. Keluar");
            Console.Write("Pilihan: ");
            string choice = Console.ReadLine();

            switch (choice)
            {
                case "1":
                    Console.Write("Jenis buku (1=Fiksi, 2=Non-Fiksi, 3=Majalah): ");
                    string type = Console.ReadLine();
                    Console.Write("Judul: ");
                    string judul = Console.ReadLine();
                    Console.Write("Penulis: ");
                    string penulis = Console.ReadLine();
                    Console.Write("Tahun: ");
                    int tahun_terbit = int.Parse(Console.ReadLine());

                    Buku buku = type switch
                    {
                        "1" => new BukuFiksi(judul, penulis, tahun_terbit),
                        "2" => new BukuNonFiksi(judul, penulis, tahun_terbit),
                        "3" => new Majalah(judul, penulis, tahun_terbit),
                        _ => null
                    };
                }
            }
        }
    }
}
```

```

        if (buku != null)
            perpus.TambahBuku(buku);
        else
            Console.WriteLine("Jenis tidak dikenali.");
        break;

    case "2":
        perpus.LiatSemua();
        Console.Write("Pilih nomor buku yang ingin diubah: ");
        int editIndex = int.Parse(Console.ReadLine()) - 1;
        Console.Write("Judul baru: ");
        string newJudul = Console.ReadLine();
        Console.Write("Penulis baru: ");
        string newPenulis = Console.ReadLine();
        Console.Write("Tahun baru: ");
        int newTahun_terbit = int.Parse(Console.ReadLine());
        perpus.EditBuku(editIndex, newJudul, newPenulis, newTahun_terbit);
        break;

    case "3":
        perpus.LiatSemua();
        break;

    case "4":
        perpus.LiatSemua();
        Console.Write("Pilih nomor buku yang ingin dipinjam: ");
        int borrowIndex = int.Parse(Console.ReadLine()) - 1;
        var borrowBook = perpus.GetBookByIndex(borrowIndex);
        if (borrowBook != null)
            pengguna.Pinjam(borrowBook);
        else
            Console.WriteLine("Buku tidak ditemukan.");
        break;

    case "5":
        pengguna.LiatDipinjam();
        Console.Write("Pilih nomor buku untuk dikembalikan: ");
        int returnIndex = int.Parse(Console.ReadLine()) - 1;
        if (returnIndex >= 0 && returnIndex < pengguna.Buku_dipinjam.Count)
            pengguna.Kembali(pengguna.Buku_dipinjam[returnIndex]);
        else
            Console.WriteLine("Indeks tidak valid.");
        break;

    case "6":
        pengguna.LiatDipinjam();
        break;

    case "0":
        return;

    default:
        Console.WriteLine("Pilihan tidak valid.");
        break;
    }
}
}
}
}

```

Class Program digunakan sebagai Menu untuk pengguna yang menangani input dari pengguna dan sebagai jembatan untuk menjalankan class Perpus dan class Pengguna. Pilih 1 untuk menggunakan fitur menambah buku, pilih 2 untuk menggunakan fitur mengubah buku, pilih 3 untuk melihat semua buku, pilih 4 untuk meminjam buku, pilih 5 untuk

mengembalikan buku yang telah dipinjam, dan pilih 6 untuk melihat buku apa saja yang dipinjam.

Output

Menu

```
--- Menu ---
1. Tambah Buku
2. Ubah Buku
3. Lihat Semua Buku
4. Pinjam Buku
5. Kembalikan Buku
6. Lihat Buku yang Dipinjam
0. Keluar
Pilihan: |
```

Tambah Buku

```
--- Menu ---
1. Tambah Buku
2. Ubah Buku
3. Lihat Semua Buku
4. Pinjam Buku
5. Kembalikan Buku
6. Lihat Buku yang Dipinjam
0. Keluar
Pilihan: 1
Jenis buku (1=Fiksi, 2=Non-Fiksi, 3=Majalah): 2
Judul: Tutorial Velocity Kece
Penulis: Windah Batubara
Tahun: 1937
Buku berhasil ditambahkan!
```

Ubah Buku

```
--- Menu ---
1. Tambah Buku
2. Ubah Buku
3. Lihat Semua Buku
4. Pinjam Buku
5. Kembalikan Buku
6. Lihat Buku yang Dipinjam
0. Keluar
Pilihan: 2
1. [Buku Non - Fiksi] Tutorial Velocity Kece Ditulis oleh Windah Batubara,1937
Pilih nomor buku yang ingin diubah: 1
Judul baru: Tutorial Velocity mengkane
Penulis baru: Windah Basudara
Tahun baru: 2077
Buku berhasil diubah!
```

Lihat Semua Buku

```
--- Menu ---
1. Tambah Buku
2. Ubah Buku
3. Lihat Semua Buku
4. Pinjam Buku
5. Kembalikan Buku
6. Lihat Buku yang Dipinjam
0. Keluar
Pilihan: 3
1. [Buku Non - Fiksi] Tutorial Velocity mengkane Ditulis oleh Windah Basudara,2077
2. [Majalah] Playboy Ditulis oleh Ilham Absen,1978
3. [Buku Fiksi] Harry Potter Ditulis oleh J.K. Rowling,1717
4. [Buku Non - Fiksi] Ternak Lele Ampuh Ditulis oleh Prof. Zikri T.Le,2026
```

Pinjam Buku

```
--- Menu ---
1. Tambah Buku
2. Ubah Buku
3. Lihat Semua Buku
4. Pinjam Buku
5. Kembalikan Buku
6. Lihat Buku yang Dipinjam
0. Keluar
Pilihan: 4
1. [Buku Non - Fiksi] Tutorial Velocity mengkane Ditulis oleh Windah Basudara,2077
2. [Majalah] Playboy Ditulis oleh Ilham Absen,1978
3. [Buku Fiksi] Harry Potter Ditulis oleh J.K. Rowling,1717
4. [Buku Non - Fiksi] Ternak Lele Ampuh Ditulis oleh Prof. Zikri T.Le,2026
Pilih nomor buku yang ingin dipinjam: 4
Berhasil meminjam buku: BukuNonFiksi - Ternak Lele Ampuh
```

Kembalikan Buku

```
--- Menu ---
1. Tambah Buku
2. Ubah Buku
3. Lihat Semua Buku
4. Pinjam Buku
5. Kembalikan Buku
6. Lihat Buku yang Dipinjam
0. Keluar
Pilihan: 5
Buku yang sedang dipinjam:
[Buku Non - Fiksi] Ternak Lele Ampuh Ditulis oleh Prof. Zikri T.Le,2026
[Buku Non - Fiksi] Tutorial Velocity mengkane Ditulis oleh Windah Basudara,2077
Pilih nomor buku untuk dikembalikan: 1
Buku Ternak Lele Ampuh berhasil dikembalikan!
```


Lihat Buku yang Dipinjam

```
--- Menu ---
1. Tambah Buku
2. Ubah Buku
3. Lihat Semua Buku
4. Pinjam Buku
5. Kembalikan Buku
6. Lihat Buku yang Dipinjam
0. Keluar
Pilihan: 6
Buku yang sedang dipinjam:
[Buku Non - Fiksi] Tutorial Velocity mengkane Ditulis oleh Windah Basudara,2077
```

Error Management

Indeks Menu Salah

```
--- Menu ---
1. Tambah Buku
2. Ubah Buku
3. Lihat Semua Buku
4. Pinjam Buku
5. Kembalikan Buku
6. Lihat Buku yang Dipinjam
0. Keluar
Pilihan: 7
Pilihan tidak valid.
```

Pinjam Lebih dari 3 Buku

```
--- Menu ---
1. Tambah Buku
2. Ubah Buku
3. Lihat Semua Buku
4. Pinjam Buku
5. Kembalikan Buku
6. Lihat Buku yang Dipinjam
0. Keluar
Pilihan: 4
1. [Buku Non - Fiksi] Tutorial Velocity mengkane Ditulis oleh Windah Basudara,2077
2. [Majalah] Playboy Ditulis oleh Ilham Absen,1978
3. [Buku Fiksi] Harry Potter Ditulis oleh J.K. Rowling,1717
4. [Buku Non - Fiksi] Ternak Lele Ampuh Ditulis oleh Prof. Zikri T.Le,2026
Pilih nomor buku yang ingin dipinjam: 4
Hanya dapat meminjam 3 buku!
```

Class Diagram

