

LAPORAN PRATIKUM

“PEKAN 2”

Disusun Untuk Memenuhi Tugas Mata Kuliah Struktur Data

DOSEN PENGAMPU:

Wahyudi, Dr. S.T. M.T.



DISUSUN OLEH:

Karimah Irsyadiyah (2411533018)

UNIVERSITAS ANDALAS

T.A 2024/2025

Daftar Pustaka

BAB I PENDAHULUAN.....	3
1.1 Latar Belakang.....	3
1.2 Tujuan	3
1.3 Alat dan Bahan.....	3
BAB II PEMBAHASAN	4
2.1 Langkah-langkah Praktikum dan Pembahasan Program	4
BAB III PENUTUP	8
3.1 Kesimpulan	8

BAB I

PENDAHULUAN

1.1 Latar Belakang

Struktur data memegang peranan penting dalam pengembangan perangkat lunak karena berfungsi sebagai wadah penyimpanan dan pengelolaan data yang efisien. Pada bahasa pemrograman Java, terdapat berbagai struktur data dinamis yang tersedia melalui Java Collection Framework seperti ArrayList, LinkedList, Queue, dan Stack. Pemanfaatan struktur data yang sesuai dapat meningkatkan efisiensi dan fleksibilitas aplikasi yang dibangun.

Melalui praktikum pekan kedua ini, mahasiswa mempelajari penerapan struktur data dalam bentuk ArrayList untuk menyimpan data secara dinamis, serta pemanfaatan LinkedList, Queue, dan Stack melalui simulasi sistem perpustakaan. Praktikum ini diharapkan dapat meningkatkan pemahaman mahasiswa terhadap penggunaan struktur data dalam menyelesaikan permasalahan dunia nyata dengan pendekatan OOP (Object-Oriented Programming).

1.2 Tujuan

- Memahami dan menerapkan penggunaan struktur data ArrayList dalam Java.
- Mempelajari penerapan LinkedList, Queue, dan Stack dalam sistem manajemen buku perpustakaan.
- Meningkatkan keterampilan pemrograman Java melalui praktik langsung dengan skenario riil.

1.3 Alat dan Bahan

- Perangkat Keras: Laptop/PC
- IDE Java (NetBeans, IntelliJ, Eclipse, atau VS Code)
- JDK (Java Development Kit)
- Kelas Java: ArrayList1, ArrayListGetElement, ArrayListSisip, DaftarSiswa, Mahasiswa, MahasiswaMain

BAB II

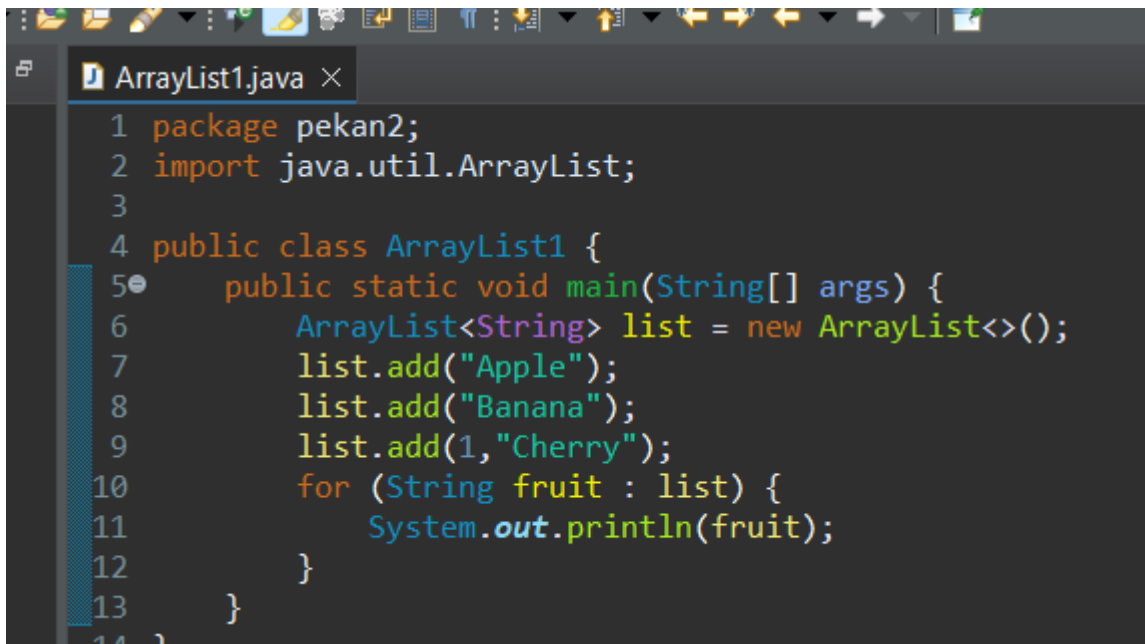
PEMBAHASAN

2.1 Langkah-langkah Praktikum dan Pembahasan Program

Pada praktikum ini, kita mempelajari enam buah program yang memanfaatkan ArrayList sebagai struktur data utama. Setiap file memiliki tujuan spesifik untuk menunjukkan cara kerja dan implementasi fitur ArrayList dalam konteks yang berbeda, dari yang paling sederhana hingga penerapan dengan objek buatan sendiri.

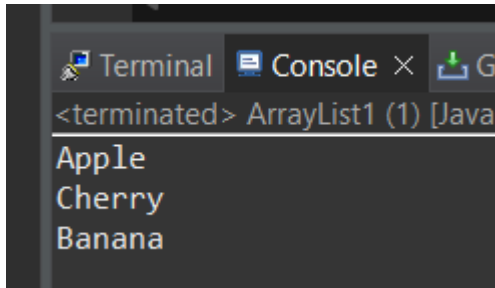
A. Kelas ArrayList1.java

1. Membuat objek ArrayList<String> untuk menyimpan nama buah.
2. Menambahkan elemen ke dalam ArrayList menggunakan metode .add().
3. Menyisipkan elemen di posisi tertentu (indeks ke-1) menggunakan .add(index, element).
4. Melakukan iterasi dan menampilkan isi ArrayList menggunakan perulangan for-each.



```
1 package pekan2;
2 import java.util.ArrayList;
3
4 public class ArrayList1 {
5     public static void main(String[] args) {
6         ArrayList<String> list = new ArrayList<>();
7         list.add("Apple");
8         list.add("Banana");
9         list.add(1, "Cherry");
10        for (String fruit : list) {
11            System.out.println(fruit);
12        }
13    }
14 }
```

Output yang dihasilkan:



B. Kelas Perpustakaan.java

1. Membuat kelas Buku sebagai representasi data buku (judul, pengarang, ISBN).
2. Membuat kelas Perpustakaan yang memiliki:
 - LinkedList<Buku> sebagai tempat penyimpanan seluruh koleksi buku.
 - Queue<Buku> sebagai antrian buku yang sedang dipinjam.
 - Stack<Buku> untuk menyimpan buku yang dikembalikan.
3. Menyediakan menu interaktif dalam main() untuk menjalankan fungsi:
 - Menambah buku ke koleksi.
 - Meminjam buku (memindahkan buku ke Queue).
 - Mengembalikan buku (memindahkan buku dari Queue ke Stack).
 - Keluar dari program.

Ilustrasi alur kerja:

- User memasukkan data buku → masuk ke LinkedList.
- Buku dipinjam berdasarkan judul → masuk ke Queue.
- Buku dikembalikan berdasarkan judul → masuk ke Stack.

```
package pekan2;

import java.util.*;

class Buku{
    String judul, pengarang, isbn;
    Buku (String judul, String pengarang, String isbn) {
        this.judul = judul;
        this.pengarang = pengarang;
        this.isbn = isbn;
    }
}

class Perpustakaan {
    LinkedList<Buku> koleksiBuku = new LinkedList<>();
    Queue<Buku> Peminjaman = new LinkedList<>();
    Stack<Buku> Pengembalian = new Stack<>();
    // use the linkedlist
    void tambahBuku(String judul, String pengarang, String isbn) {
        koleksiBuku.add (new Buku(judul, pengarang, isbn));
    }
    // use the queue
}
```

```

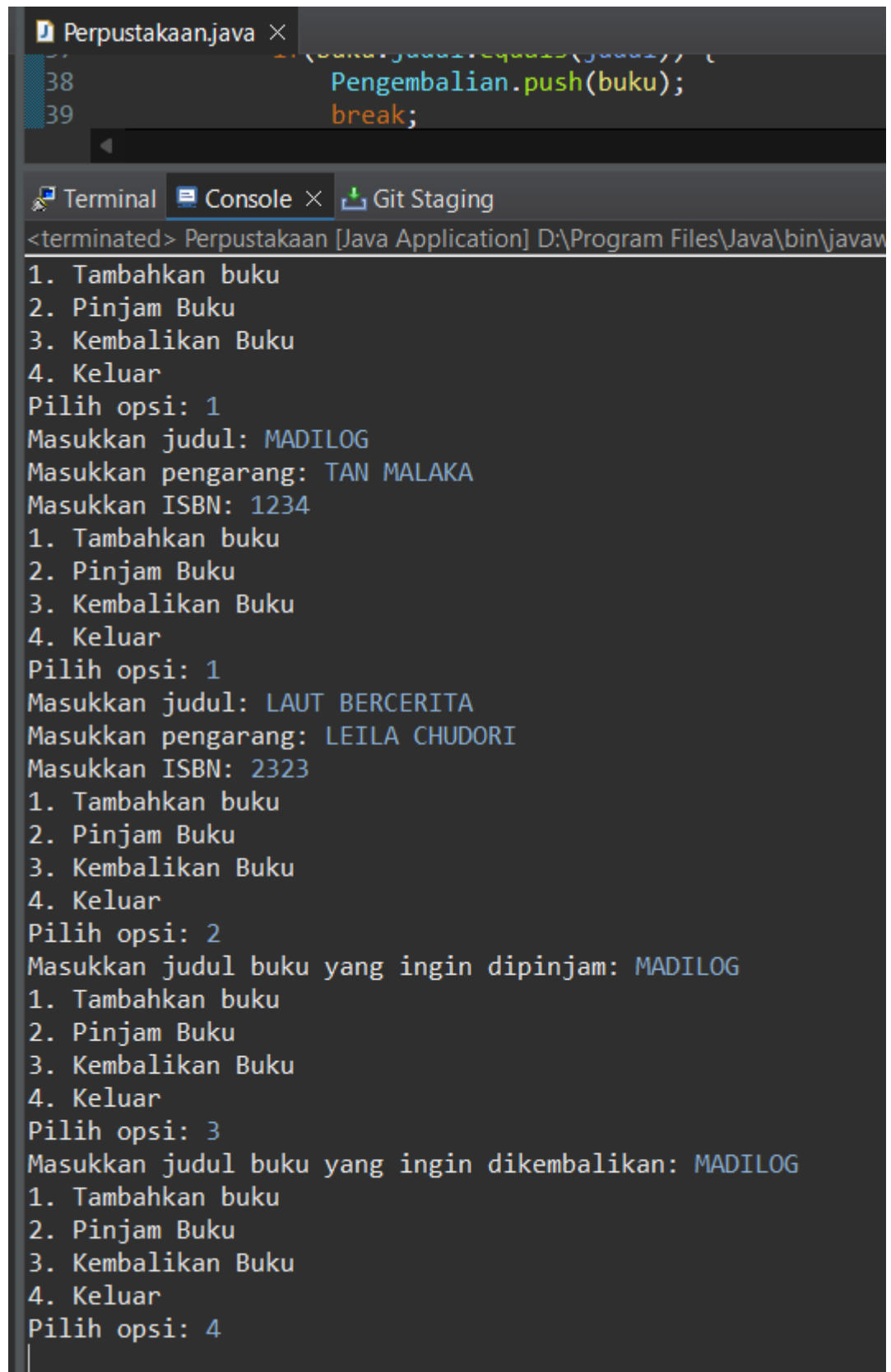
void pinjamBuku (String judul) {
    for (Buku buku : koleksiBuku) {
        if (buku.judul.equals(judul)) {
            Peminjaman.add(buku);
            break;
        }
    }
}

// use the stack
void kembalikanBuku (String judul) {
    for (Buku buku : Peminjaman) {
        if (buku.judul.equals(judul)) {
            Pengembalian.push(buku);
            break;
        }
    }
}

public static void main (String[] args) {
    Perpustakaan perpustakaan = new Perpustakaan();
    Scanner scanner = new Scanner (System.in);
    while (true) {
        System.out.println(" 1. Tambah Buku\n 2. Pinjam Buku\n " + "3.
Kembalikan Buku\n 4. keluar");
        System.out.print("Pilih opsi: ");
        int pilihan = scanner.nextInt();
        scanner.nextLine();
        if (pilihan == 1) {
            System.out.print("Masukkan judul: ");
            String judul = scanner.nextLine();
            System.out.print("Masukkan pengarang: ");
            String pengarang = scanner.nextLine();
            System.out.print("Masukkan ISBN: ");
            String isbn = scanner.nextLine();
            perpustakaan.tambahBuku(judul, pengarang, isbn);
        } else if (pilihan == 2) {
            System.out.print("Masukkan judul buku yang ingin
dipinjam: ");
            String judul = scanner.nextLine();
            perpustakaan.pinjamBuku(judul);
        } else if (pilihan == 3) {
            System.out.print("Masukkan judul buku yang ingin
dikembalikan: ");
            String judul = scanner.nextLine();
            perpustakaan.kembalikanBuku(judul);
        } else if (pilihan == 4) {
            break;
        }
    }
    scanner.close();
}
}

```

OUTPUT:



```
Perpustakaan.java ×
38         Pengembalian.push(buku);
39         break;
```

Terminal Console × Git Staging

<terminated> Perpustakaan [Java Application] D:\Program Files\Java\bin\javaw

```
1. Tambahkan buku
2. Pinjam Buku
3. Kembalikan Buku
4. Keluar
Pilih opsi: 1
Masukkan judul: MADILOG
Masukkan pengarang: TAN MALAKA
Masukkan ISBN: 1234
1. Tambahkan buku
2. Pinjam Buku
3. Kembalikan Buku
4. Keluar
Pilih opsi: 1
Masukkan judul: LAUT BERCERITA
Masukkan pengarang: LEILA CHUDORI
Masukkan ISBN: 2323
1. Tambahkan buku
2. Pinjam Buku
3. Kembalikan Buku
4. Keluar
Pilih opsi: 2
Masukkan judul buku yang ingin dipinjam: MADILOG
1. Tambahkan buku
2. Pinjam Buku
3. Kembalikan Buku
4. Keluar
Pilih opsi: 3
Masukkan judul buku yang ingin dikembalikan: MADILOG
1. Tambahkan buku
2. Pinjam Buku
3. Kembalikan Buku
4. Keluar
Pilih opsi: 4
|
```

BAB III

PENUTUP

3.1 Kesimpulan

Dari praktikum yang telah dilakukan pada Pekan 2 ini, dapat disimpulkan bahwa:

1. Struktur data seperti ArrayList, LinkedList, Queue, dan Stack sangat membantu dalam pengelolaan data secara efisien dan terstruktur pada pemrograman Java.
2. ArrayList cocok digunakan ketika diperlukan akses cepat ke elemen berdasarkan indeks, dan sangat fleksibel dalam penambahan maupun penghapusan elemen.
3. LinkedList, Queue, dan Stack masing-masing memiliki keunggulan dalam konteks tertentu seperti antrian (FIFO) dan tumpukan (LIFO), yang sangat relevan untuk kasus sistem peminjaman dan pengembalian buku pada aplikasi perpustakaan.
4. Penerapan langsung dalam bentuk program sederhana seperti sistem perpustakaan membantu memahami konsep dasar struktur data serta cara penggunaannya secara nyata dalam pengembangan aplikasi.