

LAPORAN PRAKTIKUM STRUKTUR DATA
MEMBAHAS SOAL UAS



OLEH :

JOVANTRI IMMANUEL GULO

NIM 2411532014

MATA KULIAH STRUKTUR DATA

DOSEN PENGAMPU :

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

FAKULTAS TEKNOLOGI INFORMASI

DEPARTEMEN INFORMATIKA

UNIVERSITAS ANDALAS

PADANG, 29 APRIL 2025

A. Pendahuluan

Pada pengelolaan data, banyak cara yang dapat kita gunakan untuk mengelola dan menyimpan data secara efisien dan mudah. Namun, dalam pemrograman, terutama dalam penggunaan library Java, terdapat library yang dapat kita gunakan untuk menyimpan atau mengelola data, yaitu Array List, Linked List, dan Queue. Ketiga library ini marak digunakan dalam penyimpanan dan pengelolaan struktur data dalam pemrograman berbasis Java.

Penggunaannya dapat beragam cara, dapat digunakan secara langsung dengan cara melakukan import dari path library nya, ataupun juga dapat digunakan dengan melakukan pengimplementasian dari masing-masing library tersebut. Dalam praktikum ini, kita akan mengimplementasikan setiap penggunaan masing-masing library tersebut, terlibat dalam fungsinya untuk melakukan penyimpanan dan pengelolaan berbagai macam struktur data dan fungsi-fungsinya yang mudah untuk digunakan.

B. Tujuan

Tujuan dari dilakukannya praktikum ini adalah

1. Mampu menggunakan Array List dalam pengelolaan data.
2. Mampu membuat kelas untuk program data perpustakaan.

C. Langkah – langkah Pengerjaan

Berikut adalah langkah-langkah dalam pengerjaan praktikum kali ini:

1. Buat kelas baru dengan nama **ArrayList1**.

```
> ArrayList1.java
```

2. Kemudian, lakukan import untuk library array.

```
3 import java.util.ArrayList;
```

3. Kemudian, tulis kelas main nya.

```
public static void main(String[] args) {
```

4. Kemudian, deklarasi array list dengan nama **list**.

```
ArrayList<String> list = new ArrayList<>();
```

5. Lalu, gunakan **add** pada array list yang sudah dideklarasikan dengan nama **list** tadi, tambahkan variable dengan **list.add**

```
list.add("Apple");  
list.add("Banana");  
list.add(1, "Cherry");
```

6. Kita bisa menambahkan list juga dengan menambahkannya sesuai dengan index yang kita inginkan.

7. Terakhir, kita bisa menampilkan semua di dalam list dengan menggunakan **for**

```
for(String fruit : list) {  
    System.out.println(fruit);  
}
```

8. Berikut adalah kode penuh untuk file ArrayList1

```

1 package pekan2;
2
3 import java.util.ArrayList;
4
5 public class ArrayList1 {
6
7     public static void main(String[] args) {
8         ArrayList<String> list = new
ArrayList<>();
9         list.add("Apple");
10        list.add("Banana");
11        list.add(1, "Cherry");
12        for(String fruit : list) {
13            System.out.println(fruit);
14        }
15    }
16 }

```

```

ArrayList1.java x Perpustakaan.java TugasParkiranMobil.java
1 package pekan2;
2
3 import java.util.ArrayList;
4
5 public class ArrayList1 {
6
7     public static void main(String[] args) {
8         ArrayList<String> list = new ArrayList<>();
9         list.add("Apple");
10        list.add("Banana");
11        list.add(1, "Cherry");
12        for(String fruit : list) {
13            System.out.println(fruit);
14        }
15    }
16 }

```

9. Selanjutnya, kita buat class baru dengan nama **Perpustakaan**

> **Perpustakaan.java**

10. Kita lakukan import pada **java.util** dengan package **LinkedList**, **Queue**, **Scanner**, dan **Stack**.

```

3 import java.util.LinkedList;
4 import java.util.Queue;
5 import java.util.Scanner;
6 import java.util.Stack;

```

11. Kemudian kita buat class **Buku** dan kita lakukan deklarasi untuk **judul**, **pengarang**, dan **isbn**. Lalu juga, kita deklarasi kelas buku dengan menggunakan **this.judul**, **this.pengarang**, dan **this.isbn**.

```

8 class Buku {
9     String judul, pengarang, isbn;
10    Buku(String judul, String pengarang, String
isbn) {
11        this.judul = judul;
12        this.pengarang = pengarang;
13        this.isbn = isbn;
14    }
15 }

```

12. Lalu, kita buat class baru lagi dengan nama class **Perpustakaan**

```

16
17 class Perpustakaan {
18     LinkedList<Buku> k

```

13. Deklarasikan **LinkedList** untuk **KoleksiBuku**, **Queue** untuk **Peminjaman**, dan **Stack** untuk **Pengembalian**. Jadi, semua hal yang sudah dideklarasikan ini, akan digunakan untuk penyimpanan data nantinya.

```
18      LinkedList<Buku> koleksiBuku = new LinkedList<>();
19      Queue<Buku> Peminjaman = new LinkedList<>();
20      Stack<Buku> Pengembalian = new Stack<>();
```

14. Lalu, kita buat method void baru, yaitu **tambahBuku** dengan menggunakan judul, pengarang, dan isbn, di mana method ini menggunakan **koleksiBuku.add** yang merupakan fungsi inti dari method tersebut.

```
22      void tambahBuku(String judul, String
23      pengarang, String isbn) {
24          koleksiBuku.add(new Buku(judul,
25          pengarang, isbn));
26      }
```

15. Kemudian, kita membuat method void yang baru juga yaitu method void dengan nama **pinjamBuku** yang di mana memiliki fungsi utama yaitu **Peminjaman.add** untuk menambahkan data buku yang dipinjam pada koleksi **Peminjaman**.

```
26      void pinjamBuku(String judul) {
27          for(Buku buku : koleksiBuku) {
28              if(buku.judul.equals(judul)) {
29                  Peminjaman.add(buku);
30                  break;
31              }
32          }
33      }
```

16. Selanjutnya, kita juga perlu untuk membuat method void dengan nama **kembalikanBuku**, yang di mana punya fungsi utama yaitu **Pengembalian.push** yang berfungsi untuk mengembalikan buku ke dalam koleksi **Stack** pada **Pengembalian**.

```
35      void kembalikanBuku(String judul) {
36          for(Buku buku : Peminjaman) {
37              if(buku.judul.equals(judul)) {
38                  Pengembalian.push(buku);
39                  break;
40              }
41          }
42      }
```

17. Kemudian, kita masuk ke dalam fungsi main atau fungsi utamanya, pertama deklarasikan terlebih dahulu.

```
43      public static void main(String[] args) {
44          Perpustakaan perpustakaan = new
45          Perpustakaan();
```

18. Lalu, kita deklarasikan kelas baru yang telah kita deklarasikan di atas tadi, yaitu kelas **Perpustakaan**, sekaligus juga dengan **Scanner**.

```
45      Scanner scanner = new Scanner(System.
46      in);
```

19. Kemudian, kita menggunakan sebuah perulangan yang baru, yaitu **while(true)**. Pada perulangan ini, kita menggunakan semua fungsi yang terkait dengan **Perpustakaan** dan juga kita akan membuat menu terlebih dahulu.

```

47         while(true) {
48             System.out.println
49             ("1. Tambahkan buku\n2. Pinjam Buku\n
50             3. Kembalikan Buku\n4. Keluar");
51             System.out.print(
52             "Pilih opsi: ");
53             int pilihan = scanner.nextInt
54             ();
55             scanner.nextLine();

```

20. Lalu, kita membuat kondisi bila si user memilih beberapa pilihan yang berdasarkan kepada opsi yang tersedia, kita membuat kondisi untuk pilihan 1 terlebih dahulu.

```

52         if(pilihan == 1) {
53             System.out.print(
54             "Masukkan judul: ");
55             String judul =
56             scanner.nextLine();
57             System.out.print(
58             "Masukkan pengarang: ");
59             String pengarang =
60             scanner.nextLine();
61             System.out.print(
62             "Masukkan ISBN: ");
63             String isbn = scanner
64             .nextLine();
65             perpustakaan.
66             tambahBuku(judul, pengarang, isbn);

```

21. Lalu, ini kondisi untuk pilihan 2, kita menggunakan else if untuk mempermudah peralihan kondisi. Dalam tiap kondisi ini, selalu ada method yang dijalankan.

```

60         } else if (pilihan == 2) {
61             System.out.print(
62             "Masukkan judul buku yang ingin dipinjam: ");
63             String judul =
64             scanner.nextLine();
65             perpustakaan.
66             pinjamBuku(judul);

```

22. Kemudian, ini untuk kondisi pilihan ke-3.

```

64         } else if (pilihan == 3) {
65             System.out.print(
66             "Masukkan judul buku yang ingin dikembalikan: ");
67             String judul =
68             scanner.nextLine();
69             perpustakaan.
70             kembalikanBuku(judul);

```

23. Lalu, untuk pilihan ke-4, kondisinya adalah **break** atau program akan berhenti.

```

68         } else if (pilihan == 4) {
69             break;
70         }

```

24. Kemudian, untuk yang terakhir, kita perlu menutup scanner dengan **scanner.close()**

```

71         }
72         scanner.close();
73     }

```

D. Kesimpulan

Dari praktikum yang telah dilakukan, maka dapat diambil kesimpulan bahwa penggunaan implementasi dari Linked List, Queue, dan Array List sangat membantu para developer untuk mengembangkan program yang berkaitan dengan pengelolaan dan penyimpanan struktur data. Selain itu, dengan menggabungkan beberapa method yang membantu, semakin melengkapi setiap proyek yang dibuat.