

# LAPORAN PRATIKUM

## “PEKAN 1”

*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**

<b>BAB I PENDAHULUAN.....</b>	<b>3</b>
1.1    Latar Belakang.....	3
1.2    Tujuan .....	3
1.3    Alat dan Bahan.....	3
<b>BAB II PEMBAHASAN .....</b>	<b>4</b>
2.1    Langkah-langkah Praktikum dan Pembahasan Program .....	4
<b>BAB III PENUTUP .....</b>	<b>15</b>
3.1    Kesimpulan .....	15

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Bahasa Java menyediakan berbagai macam struktur data yang memudahkan dalam pengolahan informasi, salah satunya adalah `ArrayList`. Struktur data ini tergolong dalam koleksi dinamis yang mampu menampung data dengan jumlah yang tidak tetap dan memungkinkan berbagai operasi manipulasi data secara efisien.

Pada praktiknya, penggunaan array konvensional memiliki banyak keterbatasan, terutama dalam hal fleksibilitas ukuran dan kemudahan operasi seperti penambahan, penghapusan, dan pencarian data. Oleh karena itu, `ArrayList` hadir sebagai solusi dari permasalahan tersebut, dengan menyediakan metode yang sudah tersedia dalam kelas pustaka Java untuk melakukan berbagai operasi tersebut secara mudah dan efisien.

Praktikum ini bertujuan untuk memberikan pemahaman mendalam mengenai cara kerja dan penerapan `ArrayList` dalam berbagai konteks, mulai dari menyimpan tipe data primitif hingga objek buatan sendiri. Melalui enam program yang dianalisis, mahasiswa diharapkan mampu memahami konsep dasar dan lanjutan penggunaan `ArrayList`, serta mampu mengimplementasikannya dalam pengembangan aplikasi sederhana berbasis Java.

### **1.2 Tujuan**

Praktikum ini bertujuan untuk memahami dan mengimplementasikan struktur data dinamis menggunakan `ArrayList` dalam bahasa pemrograman Java. Fokus utama pada operasi dasar seperti penambahan, penghapusan, pencarian, dan pengambilan data dalam `ArrayList`, baik dengan tipe data primitif maupun objek.

### **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

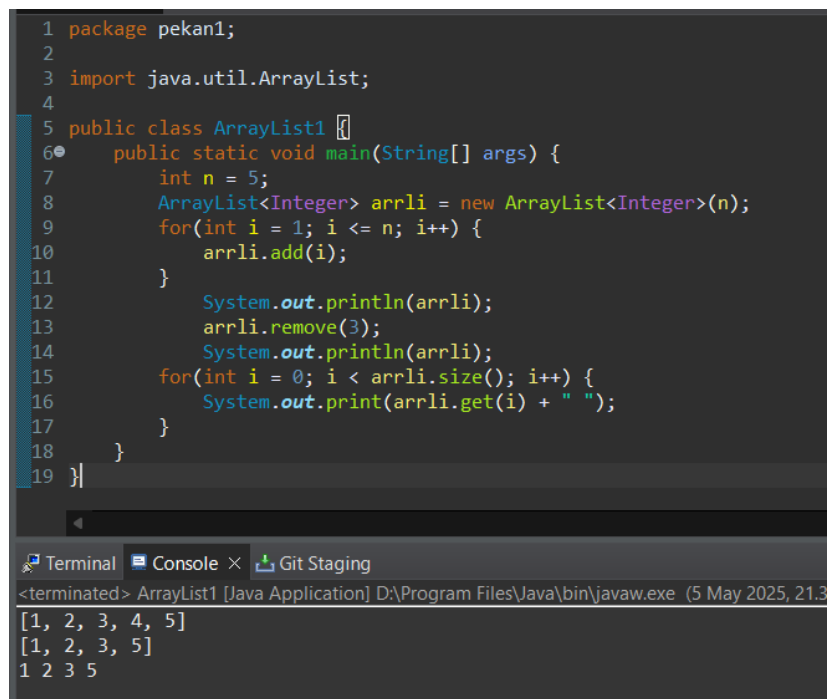
Pada praktikum ini, kita mempelajari enam buah program yang memanfaatkan ArrayList sebagai struktur data utama. Setiap file memiliki tujuan spesifik untuk menunjukkan cara kerja dan implementasi fitur ArrayList dalam konteks yang berbeda, dari yang paling sederhana hingga penerapan dengan objek buatan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

##### 1. ArrayList1.java

Program ini mendemonstrasikan cara dasar untuk:

- Membuat ArrayList<Integer>
- Menambahkan elemen
- Menghapus elemen berdasarkan indeks
- Menampilkan isi list dengan perulangan for

Penjelasan kode:



```
1 package pekan1;
2
3 import java.util.ArrayList;
4
5 public class ArrayList1 {
6     public static void main(String[] args) {
7         int n = 5;
8         ArrayList<Integer> arrli = new ArrayList<Integer>(n);
9         for(int i = 1; i <= n; i++) {
10             arrli.add(i);
11         }
12         System.out.println(arrli);
13         arrli.remove(3);
14         System.out.println(arrli);
15         for(int i = 0; i < arrli.size(); i++) {
16             System.out.print(arrli.get(i) + " ");
17         }
18     }
19 }
```

Terminal Console × Git Staging

<terminated> ArrayList1 [Java Application] D:\Program Files\Java\bin\javaw.exe (5 May 2025, 21:3

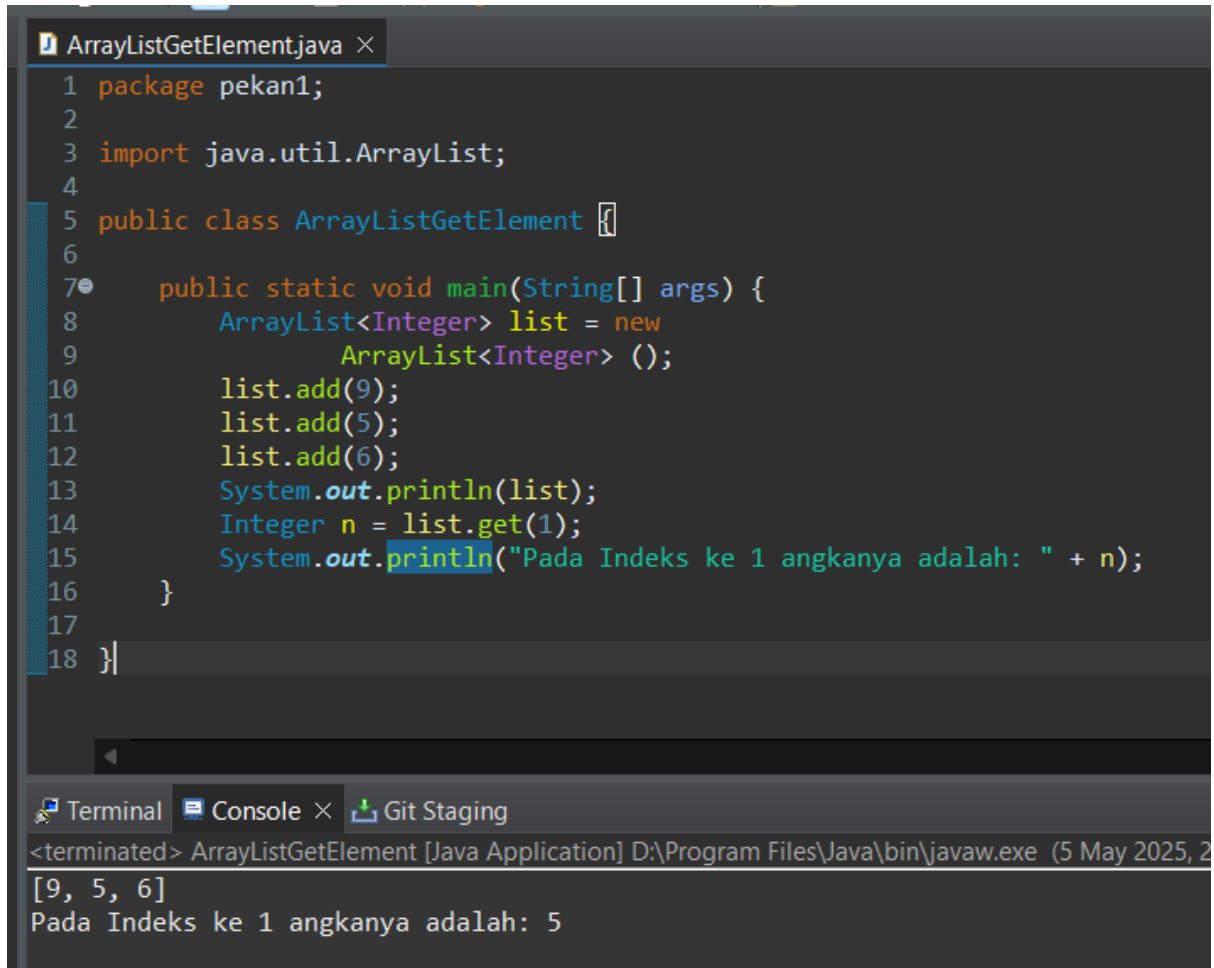
[1, 2, 3, 4, 5]  
[1, 2, 3, 5]  
1 2 3 5

##### 2. ArrayListGetElement.java

Program ini berfokus pada penggunaan method `get()` dari `ArrayList`.

**Penjelasan:**

- Menambahkan tiga elemen: 9, 5, dan 6.
- Mengambil elemen pada indeks ke-1.



```
1 package pekan1;
2
3 import java.util.ArrayList;
4
5 public class ArrayListGetElement {
6
7     public static void main(String[] args) {
8         ArrayList<Integer> list = new
9             ArrayList<Integer> ();
10        list.add(9);
11        list.add(5);
12        list.add(6);
13        System.out.println(list);
14        Integer n = list.get(1);
15        System.out.println("Pada Indeks ke 1 angkanya adalah: " + n);
16    }
17
18 }
```

Terminal Console × Git Staging

<terminated> ArrayListGetElement [Java Application] D:\Program Files\Java\bin\javaw.exe (5 May 2025, 2

[9, 5, 6]  
Pada Indeks ke 1 angkanya adalah: 5

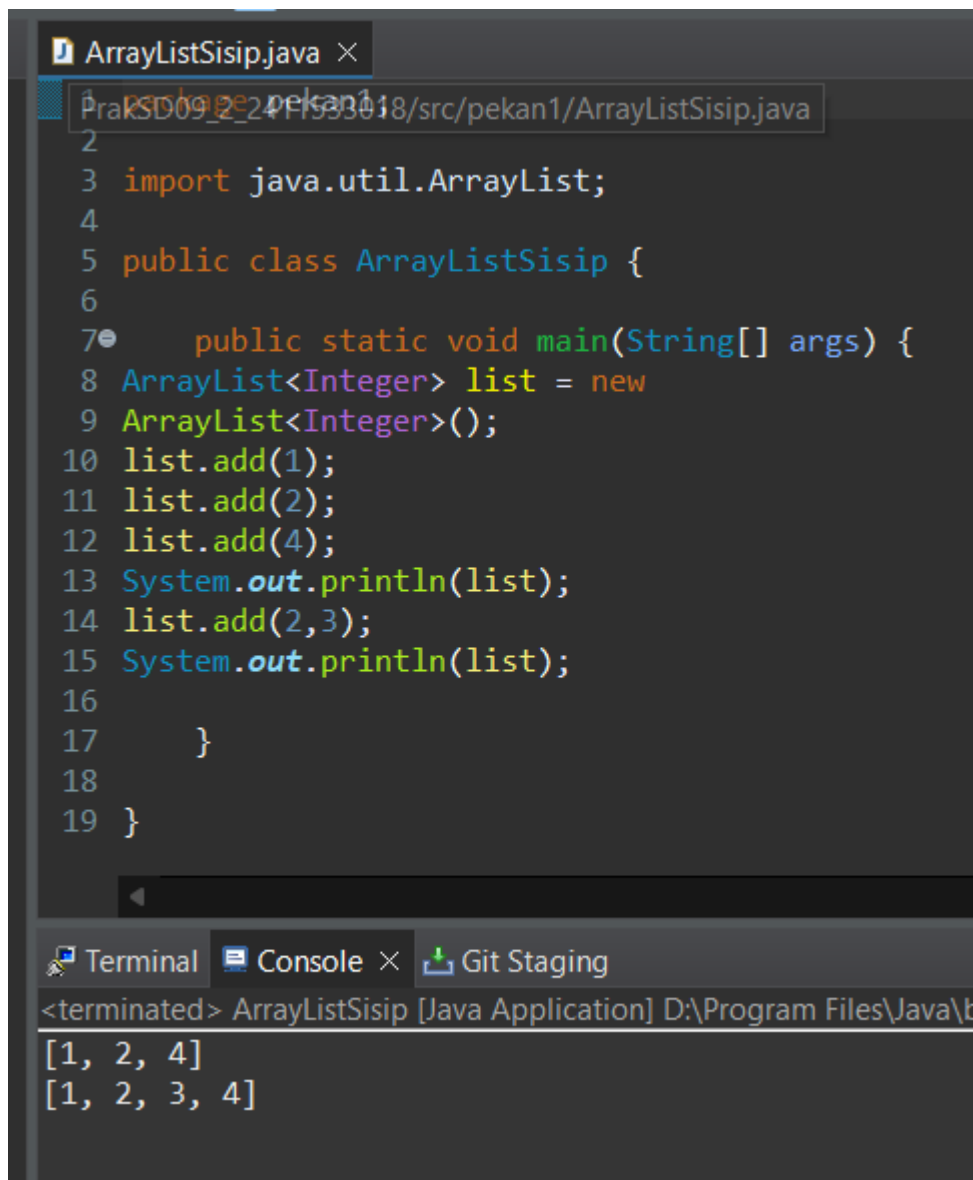
**Output:**

Elemen pada indeks 1 adalah: 5

Program ini menunjukkan bahwa akses ke elemen list dapat dilakukan dengan indeks layaknya array, namun dengan fleksibilitas ukuran yang dinamis.

### 3. ArrayListSisip.java

Program ini memperkenalkan konsep penyisipan elemen pada indeks tertentu menggunakan method `add(index, element)`.



```
ArrayListSisip.java x
PrakSD09_2_241533018/src/pekan1/ArrayListSisip.java
2
3 import java.util.ArrayList;
4
5 public class ArrayListSisip {
6
7     public static void main(String[] args) {
8         ArrayList<Integer> list = new
9         ArrayList<Integer>();
10        list.add(1);
11        list.add(2);
12        list.add(4);
13        System.out.println(list);
14        list.add(2,3);
15        System.out.println(list);
16
17    }
18
19 }
```

Terminal Console x Git Staging

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

[1, 2, 4]  
[1, 2, 3, 4]

**Hasil akhir list:** [1, 2, 3, 4]

Program ini menggambarkan bahwa ArrayList secara otomatis menggeser elemen setelah indeks penyisipan untuk memberi ruang bagi elemen baru.

#### 4. DaftarSiswa.java

Program ini merupakan implementasi aplikasi berbasis teks sederhana menggunakan ArrayList<String> untuk menyimpan nama-nama siswa. Disediakan beberapa menu interaktif:

- Tambah nama
- Tampilkan semua nama

- Hapus nama
- Cari nama

### Fitur penting:

- Perulangan while dengan switch-case untuk menu
- Penggunaan method add(), remove(), dan contains()

### Catatan teknis:

- Setelah scanner.nextInt(), perlu ada scanner.nextLine() untuk membuang newline sebelum membaca input string selanjutnya.
- Pencarian nama bersifat case-sensitive.

### Analisis:

Program ini mensimulasikan sistem input data berbasis pengguna, cocok untuk memperkenalkan logika interaktif dan kontrol alur dasar.

```
package pekan1;
import java.util.ArrayList;
import java.util.Scanner;
public class DaftarSiswa {
private static ArrayList<String> daftarNamaSiswa = new ArrayList<>();

    public static void main(String[] args) {
        Scanner scanner = new Scanner (System.in);
        int pilihan;
        do {
            System.out.println("\nMenu");
            System.out.println("1. Tambah Nama Siswa");
            System.out.println("2. Tampilkan Daftar Nama Siswa");
            System.out.println("3. Hapus Nama Siswa");
            System.out.println("4. Cari Nama Siswa");
            System.out.println("5. Keluar");
            pilihan = scanner.nextInt();
            switch (pilihan) {
                case 1:
                    tambahNamaSiswa(scanner);
                    break;
                case 2:
                    tampilkanDaftarNamaSiswa();
```

```

                break;

            case 3:

                hapusNamaSiswa(scanner);

                break;

            case 4:

                cariNamaSiswa(scanner);

                break;

            case 5:

                System.out.println("Keluar dari program");

                break;

            default:

                System.out.println("Pilihan tidak valid");

        }

    } while (pilihan !=5);

}

private static void tambahNamaSiswa (Scanner scanner) {

    System.out.println("Masukkan nama siswa: ");

    String nama = scanner.nextLine();

    daftarNamaSiswa.add(nama);

    System.out.println("Nama siswa berhasil ditambahkan");

}

private static void tampilkanDaftarNamaSiswa() {

    if (daftarNamaSiswa.isEmpty()) {

        System.out.println("Tidak ada siswa dalam daftar");

    } else {

        System.out.println("Daftar Nama Siswa");

        for (String nama : daftarNamaSiswa) {

            System.out.println(nama);

        }

    }

}

private static void hapusNamaSiswa (Scanner scanner) {

    System.out.println("Masukkan nama siswa yang akan dihapus");

    String nama = scanner.nextLine();

    if (daftarNamaSiswa.remove(nama)) {

        System.out.println("Nama siswa berhasil dihapus.");

    } else {

        System.out.println("Nama siswa tidak ditemukan.");

    }

}

private static void cariNamaSiswa (Scanner scanner) {

```



```
System.out.println("Masukkan nama siswa yang dicari: ");  
  
String nama = scanner.nextLine();  
  
if (daftarNamaSiswa.contains (nama)) {  
    System.out.println("Nama siswa ditemukan: " + nama);  
} else {  
    System.out.println("Nama siswa tidak ditemukan");  
}  
}  
}
```

```
DaftarSiswa.java ×
1 package pekan1;
2

Terminal Console × Git Staging
<terminated> DaftarSiswa [Java Application] D:\Program Files\Java\bin\javaw.exe (5 May 2025, 21.41.03 – 21.41.15)

Menu
1. Tambah Nama Siswa
2. Tampilkan Daftar Nama Siswa
3. Hapus Nama Siswa
4. Cari Nama Siswa
5. Keluar
2
Daftar Nama Siswa

Menu
1. Tambah Nama Siswa
2. Tampilkan Daftar Nama Siswa
3. Hapus Nama Siswa
4. Cari Nama Siswa
5. Keluar
3
Masukkan nama siswa yang akan dihapus
Nama siswa berhasil dihapus.

Menu
1. Tambah Nama Siswa
2. Tampilkan Daftar Nama Siswa
3. Hapus Nama Siswa
4. Cari Nama Siswa
5. Keluar
4
Masukkan nama siswa yang dicari:
Nama siswa tidak ditemukan

Menu
1. Tambah Nama Siswa
2. Tampilkan Daftar Nama Siswa
3. Hapus Nama Siswa
4. Cari Nama Siswa
5. Keluar
5
Keluar dari program
```

## 5. Mahasiswa.java

File ini mendefinisikan kelas **Mahasiswa** yang akan digunakan sebagai tipe data dalam ArrayList.

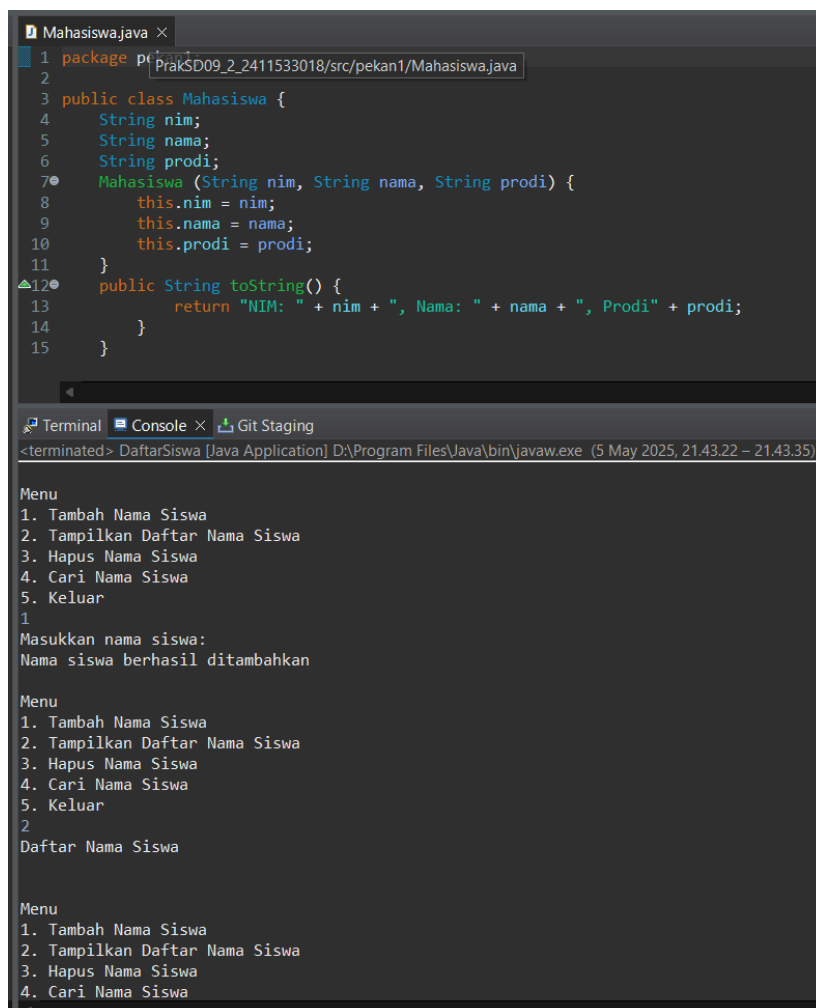
### Atribut:

- nim (String)

- nama (String)
- prodi (String)

### Method penting:

- Konstruktor
- Getter dan setter
- toString(): menampilkan format string dari objek



```

1 package p...
2
3 public class Mahasiswa {
4     String nim;
5     String nama;
6     String prodi;
7     Mahasiswa (String nim, String nama, String prodi) {
8         this.nim = nim;
9         this.nama = nama;
10        this.prodi = prodi;
11    }
12    public String toString() {
13        return "NIM: " + nim + ", Nama: " + nama + ", Prodi" + prodi;
14    }
15 }

```

```

Menu
1. Tambah Nama Siswa
2. Tampilkan Daftar Nama Siswa
3. Hapus Nama Siswa
4. Cari Nama Siswa
5. Keluar
1
Masukkan nama siswa:
Nama siswa berhasil ditambahkan

Menu
1. Tambah Nama Siswa
2. Tampilkan Daftar Nama Siswa
3. Hapus Nama Siswa
4. Cari Nama Siswa
5. Keluar
2
Daftar Nama Siswa

Menu
1. Tambah Nama Siswa
2. Tampilkan Daftar Nama Siswa
3. Hapus Nama Siswa
4. Cari Nama Siswa

```

Objek Mahasiswa dapat disimpan dalam `ArrayList<Mahasiswa>` dan diproses secara fleksibel seperti tipe data primitif.

## 6. MahasiswaMain.java

Program utama yang memanfaatkan kelas Mahasiswa untuk membuat aplikasi mini sistem data mahasiswa.

### Fitur menu:

1. Tambah data mahasiswa
2. Tampilkan seluruh data
3. Hapus mahasiswa berdasarkan NIM
4. Cari mahasiswa berdasarkan NIM

### Logika utama:

- Menggunakan ArrayList<Mahasiswa>
- Loop while(true) untuk menu
- Pencarian dengan perulangan dan if (m.getNim().equals(nim))

```
package pekan1;

import java.util.ArrayList;
import java.util.Scanner;
import java.util.*;

public class MahasiswaMain {

    public static void main (String[] args) {

        ArrayList<Mahasiswa> mahasiswaList = new ArrayList<>();

        Scanner scanner = new Scanner (System.in);

        int choice;

        do {

            System.out.println("Menu: ");

            System.out.println("1. Tambah Mahasiswa");

            System.out.println("2. Tampilkan Semua Mahasiswa");

            System.out.println("3. Hapus Mahasiswa Berdasarkan NIM");

            System.out.println("4. Cari Mahasiswa Berdasarkan NIM");

            System.out.println("5. Keluar");

            System.out.println("Pilih Menu: ");

            choice = scanner.nextInt();

            scanner.nextLine(); //Consume newLine

            switch (choice) {

                case 1:

                    System.out.println("Masukan NIM: ");

                    String nim = scanner.nextLine();

                    System.out.println("Masukan Nama: ");

                    String nama = scanner.nextLine();
```

```

        System.out.println("Masukan Prodi: ");
        String prodi = scanner.nextLine();
        mahasiswaList.add(new Mahasiswa(nim, nama, prodi));
        break;
    case 2:
        System.out.println("Data Mahasiswa: ");
        for (Mahasiswa mhs : mahasiswaList) {
            System.out.println(mhs);
        }
        break;
    case 3:
        System.out.println("Masukan NIM yang akan di hapus: ");
        String nimHapus = scanner.nextLine();
        mahasiswaList.removeIf(mhs -> mhs.nim.equals(nimHapus));
        break;
    case 4:
        System.out.println("Masukan NIM yang di cari: ");
        String nimCari = scanner.nextLine();
        for (Mahasiswa mhs : mahasiswaList) {
            if (mhs.nim.equals(nimCari)) {
                System.out.println(mhs);
                break;
            } else {
                System.out.println("nim tidak ada");
            }
        }
        break;
    case 5:
        System.out.println("Keluar dari program.");
        break;
    default:
        System.out.println("Pilihan tidak valid.");
    }
} while (choice != 5);

scanner.close();
}
}

```

```
MahasiswaMain.java x
51 }
52 }

Terminal Console x Git Staging
<terminated> MahasiswaMain [Java Application] D:\Program Files\Java\bin\javaw.exe (5 May 2025, 21:45:44 – 21:46:49) [pid: 28636]
Menu:
1. Tambah Mahasiswa
2. Tampilkan Semua Mahasiswa
3. Hapus Mahasiswa Berdasarkan NIM
4. Cari Mahasiswa Berdasarkan NIM
5. Keluar
Pilih Menu:
1
Masukan NIM:
2411533018
Masukan Nama:
Karimah Irsyadiyah
Masukan Prodi:
Informatika
Menu:
1. Tambah Mahasiswa
2. Tampilkan Semua Mahasiswa
3. Hapus Mahasiswa Berdasarkan NIM
4. Cari Mahasiswa Berdasarkan NIM
5. Keluar
Pilih Menu:
2
Data Mahasiswa:
NIM: 2411533018, Nama: Karimah Irsyadiyah, ProdiInformatika
Menu:
1. Tambah Mahasiswa
2. Tampilkan Semua Mahasiswa
3. Hapus Mahasiswa Berdasarkan NIM
4. Cari Mahasiswa Berdasarkan NIM
5. Keluar
Pilih Menu:
3
Masukan NIM yang akan di hapus:
2411533018
Menu:
1. Tambah Mahasiswa
2. Tampilkan Semua Mahasiswa
3. Hapus Mahasiswa Berdasarkan NIM
4. Cari Mahasiswa Berdasarkan NIM
5. Keluar
```

## **BAB III**

### **PENUTUP**

#### **3.1 Kesimpulan**

Berdasarkan hasil praktikum dan pembahasan enam program yang telah dilakukan, dapat disimpulkan bahwa:

- 1) ArrayList merupakan salah satu struktur data dinamis di Java yang sangat berguna dalam pengolahan data yang ukurannya tidak tetap dan memerlukan manipulasi seperti penambahan, penghapusan, dan pencarian.
- 2) Dalam program dasar seperti ArrayList1.java, ArrayListGetElement.java, dan ArrayListSisip.java, mahasiswa belajar tentang operasi dasar seperti menambah, menyisipkan, mengakses, dan menghapus elemen dari ArrayList.
- 3) Program interaktif seperti DaftarSiswa.java menunjukkan bagaimana ArrayList dapat diterapkan dalam sistem berbasis teks yang melibatkan masukan pengguna, logika menu, dan pencarian elemen.
- 4) Implementasi kelas Mahasiswa dan penggunaan ArrayList<Mahasiswa> dalam MahasiswaMain.java memperlihatkan bahwa ArrayList tidak hanya terbatas pada tipe data primitif, tetapi juga sangat efektif dalam mengelola kumpulan objek.
- 5) Kesalahan sintaks kecil maupun logika (seperti kesalahan pada loop atau pengelolaan Scanner) dapat memengaruhi jalannya program. Oleh karena itu, penting bagi mahasiswa untuk memahami baik logika dasar maupun struktur kontrol dalam pemrograman Java.