**LAPORAN PRAKTIKUM**

**PEMROGRAMAN II**

**MODUL**

****

**Enkapsulasi dan Collection**

**Oleh:**

**Muhammad Ryan Rizky Rahmadi NIM. 2210817310001**

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS TEKNIK**

**UNIVERSITAS LAMBUNG MANGKURAT**

**NOVEMBER 2023**

# LEMBAR PENGESAHAN

**LAPORAN PRAKTIKUM PEMROGRAMAN II**

**MODUL 3**

Laporan Praktikum Pemrograman II Modul 3 : Enkapsulasi dan Collection ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman II. Laporan Prakitkum ini dikerjakan oleh:

Nama Praktikan : Muhammad Ryan Rizky Rahmadi

NIM : 2210817310001

|  |  |
| --- | --- |
| Menyetujui,  Asisten Praktikum  Bachrul Uluum  NIM. 2010817210025 | Mengetahui,  Dosen Penanggung Jawab Praktikum  Andreyan Rizky Baskara, S.Kom., M.Kom.  NIP. 19930703 201903 1 011 |

# DAFTAR ISI

[LEMBAR PENGESAHAN 1](#_Toc149766938)

[DAFTAR ISI 2](#_Toc149766939)

[DAFTAR GAMBAR 3](#_Toc149766940)

[DAFTAR TABEL 4](#_Toc149766941)

[SOAL 1 5](#_Toc149766942)

[A. Source Code 5](#_Toc149766943)

[B. Output Program 7](#_Toc149766944)

[C. Pembahasan 7](#_Toc149766945)

[a. Dadu 7](#_Toc149766946)

[b. Main 7](#_Toc149766947)

[SOAL 2 8](#_Toc149766948)

[A. Source Code 10](#_Toc149766949)

[B. Output Program 12](#_Toc149766950)

[C. Pembahasan 12](#_Toc149766951)

[a. Negara 12](#_Toc149766952)

[b. Main 13](#_Toc149766953)

[SOAL 3 14](#_Toc149766954)

[A. Source Code 15](#_Toc149766955)

[B. Output Program 19](#_Toc149766956)

[C. Pembahasan 19](#_Toc149766957)

[b. Kopi 19](#_Toc149766958)

[c. Main 20](#_Toc149766959)

[LINK GIT HUB 20](#_Toc149766960)

# DAFTAR GAMBAR

[Gambar 1.1 Screenshoot Output Soal 1 7](#_Toc148472871)

[Gambar 2.2 Output Screenshoot Soal 2 12](#_Toc148472872)

[Gambar 3.3 Output Screenshoot Soal 3 19](#_Toc148472873)

# DAFTAR TABEL

[Tabel 1.1 Soal 1 5](#_Toc148472862)

[Tabel 2.1 Source Code Soal 1 Buah 6](#_Toc148472863)

[Tabel 3.1 Source Code Soal 1 Main 6](#_Toc148472864)

[Tabel 4.2 Soal 2 9](#_Toc148472865)

[Tabel 5.2 Source Code Soal 2 Kopi 11](#_Toc148472866)

[Tabel 6.2 Source Code Soal 2 Main 12](#_Toc148472867)

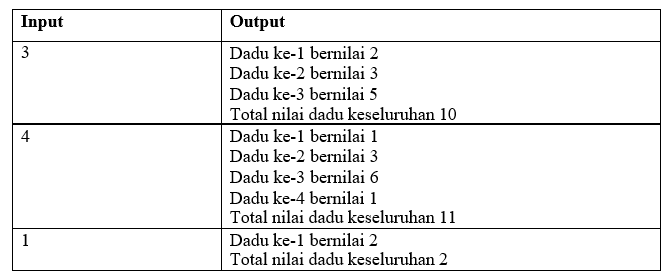
[Tabel 7.3 Soal 3 15](#_Toc148472868)

[Tabel 8.3 Source Code Soal 3 Pegawai 16](#_Toc148472869)

[Tabel 9.3 Source Code Soal 3 Main 19](#_Toc148472870)

# SOAL 1

Buatlah program yang mengimplementasikan enkapsulasi dan collection dengan ketentuan seperti berikut.

1. Buatlah kelas dengan nama Dadu.
2. Terdapat sebuah method acakNilai() yang akan memberikan nilai acak ketika objek dadu diinisiasi dengan nilai antara 1 – 6.
3. Gunakan collection dengan tipe LinkedList untuk menyimpan objek dadu yang diinputkan.
4. Program bersifat dinamis
5. Input baris pertama adalah banyaknya jumlah dadu.
6. Output adalah nilai tiap objek dadu yang telah diinisiasi
7. Output paling akhir adalah total jumlah nilai semua objek dadu.
8. Karena nilai dadu yang dihasilkan acak, maka nilai dadu output tidak harus sama dengan contoh table dibawah.

Tabel 1.1 Soal 1

## Source Code

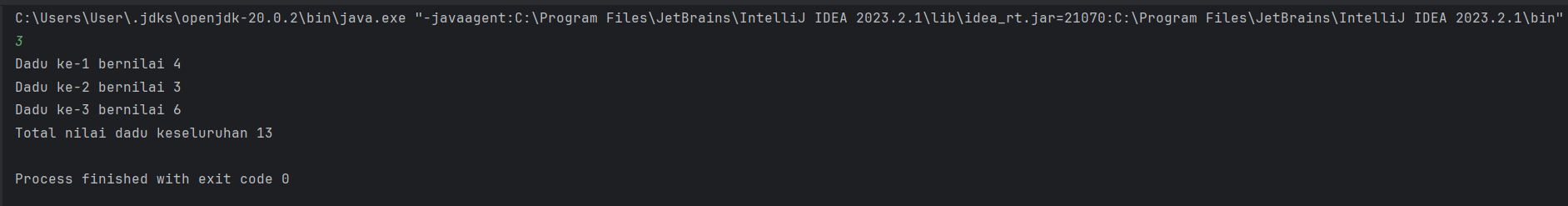
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  20  21  22  23  24  25  26  27  28 | package PRAK301\_soal1;  import java.util.LinkedList;  public class Dadu {  public int input;  public int min = 1;  public int max = 6;  int total;   public void setInput(int input){this.input = input;}   void acakNilai(){  LinkedList<Integer> ll = new LinkedList<Integer>();  ll.add(1);ll.add(2);ll.add(3);ll.add(4);ll.add(5);ll.add(6);  for (int i = 0; i < input; i++){  int random\_int = (int)Math.*floor*(Math.*random*()\*(max-min+1)+min);  System.*out*.println("Dadu ke-" + (i + 1) + " bernilai "+ random\_int);  total += random\_int;  }  System.*out*.println("Total nilai dadu keseluruhan " +total);  } } |

Tabel 2.1 Source Code Soal 1 Dadu

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | package PRAK301\_soal1;  import java.util.\*;  public class Main {  public static void main(String[] args) {  Scanner input = new Scanner(System.*in*);  Dadu dd = new Dadu();  int masukkan = input.nextInt();   dd.setInput(masukkan);  dd.acakNilai();  } } |

Tabel 3.1 Source Code Soal 1 Main

## Output Program



Gambar 1.1 Screenshoot Output Soal 1

## Pembahasan

### Dadu

Line 1 : *Package,* untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2: *Import,*  suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 5-9, dan 11: *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 5 dan 28 : {}, penanda awal dan akhir *block code*.

Line 6-9 : *int,* tipe data *integer* atau menerima bilangan bulat.

Line 14 : *LinkedList,*  suatu cara untuk menyimpan data dengan struktur.

Line 20 : *Math.floor,* sebuah metode mengembalikan nilai *double*  sama dengan bilangan bulat terbesar.

Line 25 : *System.out.println*  mencetak sesuatu dari program sebagai *output.*

### Main

Line 1 : *Package,* untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2: *Import,*  suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 5-6 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 5 dan 14 : {}, penanda awal dan akhir *block code*.

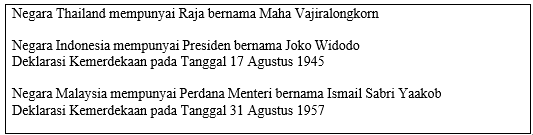
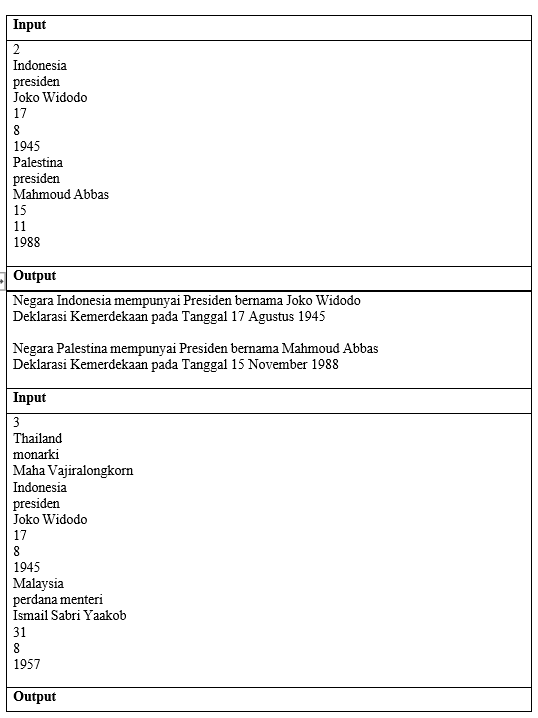
Line 7, : *Scanner* untuk menginput data.

Line 9 : *int,* tipe data *integer* atau menerima bilangan bulat.

# SOAL 2

Buatlah program yang mengimplementasikan enkapsulasi dan collection dengan ketentuan seperti berikut.

* 1. Buatlah kelas dengan nama Negara.
  2. Terdapat 5 attribute pada kelas Negara, yaitu nama, jenis kepemimpinan, nama pemimpin, tanggal kemerdekaan, bulan kemerdekaan, tahun kemerdekaan
  3. Inisiasi nilai attribute dilakukan pada constructor.
  4. Gunakan collection dengan tipe LinkedList untuk menyimpan objek Negara
  5. Gunakan collection dengan tipe HashMap untuk menyimpan daftar nama bulan. Nama bulan diambil dari hashmap berdasarkan angka bulan yang diinputkan.
  6. Program bersifat dinamis.
  7. Input baris pertama adalah banyaknya negara. Input baris berikutnya adalah data negara. Jika jenis kepemimpinan adalah monarki maka tidak perlu menginputkan tanggal kemerdekaan.
  8. Output adalah detail dari setiap objek negara yang telah diinputkan



Tabel 4.2 Soal 2

## Source Code

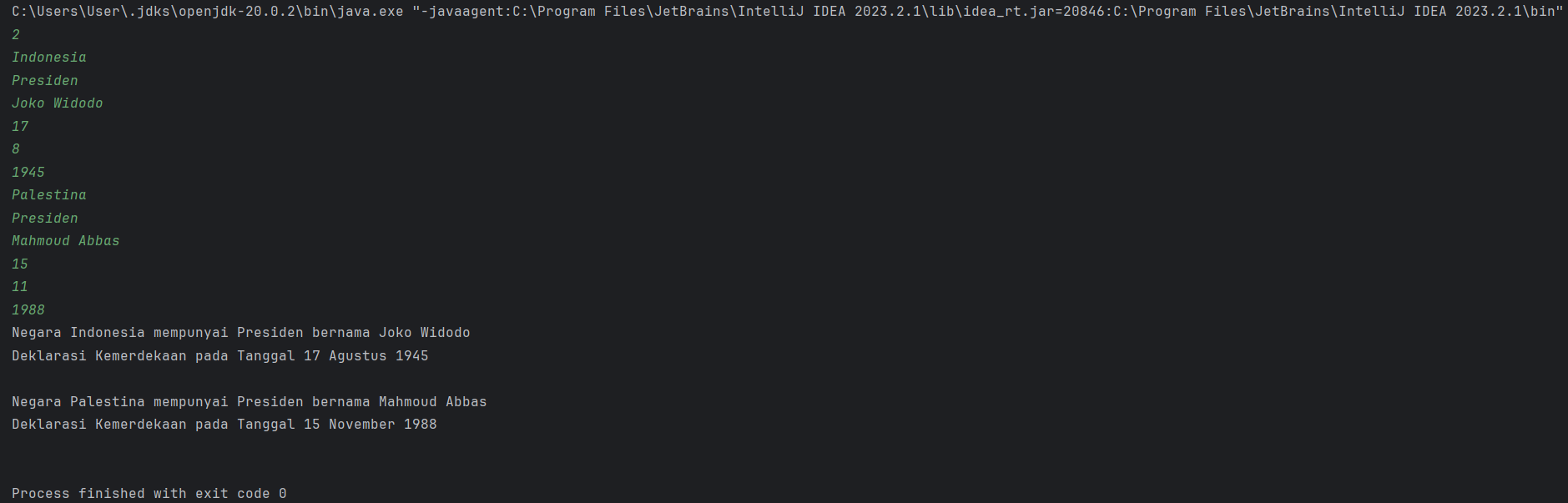
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70 | package PRAK302\_soal2;  import java.util.\*;  public class Negara {  private String nama, jenis, nama\_p;  private int tanggal, bulan, tahun;   // Konstruktor  public Negara(String nama, String jenis, String nama\_p, int tanggal, int bulan, int tahun){  this.nama = nama;  this.jenis = jenis;  this.nama\_p = nama\_p;  this.tanggal = tanggal;  this.bulan = bulan;  this.tahun = tahun;  }   // Encapsulation  public int getBulan(){  return bulan;  }   void getData(){  Scanner scn = new Scanner(System.*in*);  nama=scn.nextLine();  jenis=scn.nextLine();  nama\_p=scn.nextLine();  if (jenis.equalsIgnoreCase("presiden")||jenis.equalsIgnoreCase("perdana menteri")) {  tanggal = scn.nextInt();  bulan = scn.nextInt();  tahun = scn.nextInt();  }  }   void tampilkan(){  HashMap<Integer, String> month = new HashMap<Integer,String>();  month.put(1, "Januari");  month.put(2, "Februari");  month.put(3, "Maret");  month.put(4, "April");  month.put(5, "Mei");  month.put(6, "Juni");  month.put(7, "Juli");  month.put(8, "Agustus");  month.put(9, "September");  month.put(10, "Oktober");  month.put(11, "November");  month.put(12, "Desember");  if (jenis.equalsIgnoreCase("presiden")||jenis.equalsIgnoreCase("perdana menteri")) {  System.*out*.println("Negara "+nama+" mempunyai "+jenis+" bernama "+nama\_p);}  else if (jenis.equalsIgnoreCase("monarki")){  System.*out*.println("Negara "+nama+" mempunyai Raja"+" bernama "+nama\_p+"\n");  }  if (jenis.equalsIgnoreCase("presiden")||jenis.equalsIgnoreCase("perdana menteri")) {  System.*out*.println("Deklarasi Kemerdekaan pada Tanggal "+tanggal+" "+month.get(getBulan())+" "+tahun+"\n");}  } } |

Tabel 5.2 Source Code Soal 2 Negara

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31 | package PRAK302\_soal2;  import java.util.\*; import java.util.Scanner;  public class Main {  public static void main(String[] args) {  String nama = "", jenis="", nama\_p="";  int tanggal=0, bulan=0, tahun=0;  Scanner input = new Scanner(System.*in*);  Negara negara = new Negara(nama, jenis, nama\_p, tanggal, bulan, tahun);  LinkedList<Negara> llNegara = new LinkedList<Negara>();   int ulang;  ulang = input.nextInt();   for (int i=0; i<ulang; i++){  Negara c = new Negara(nama, jenis, nama\_p, tanggal, bulan, tahun);  c.getData();  llNegara.add(c);  }   for (int i = 0; i < llNegara.size(); i++){  Negara nIndex = llNegara.get(i);  nIndex.tampilkan();  }  } } |

Tabel 6.2 Source Code Soal 2 Main

## Output Program



Gambar 2.2 Output Screenshoot Soal 2

## Pembahasan

### Negara

Line 1 : *Package,* untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 2: *Import,*  suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 5, 10, dan 22 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 6-7 : *Private,* kunci pada java yang dapat diakses oleh kelas itu sendiri.

Line 5 dan 70 : {}, penanda awal dan akhir *block code*.

Line 10 : *String*, tipe data teks yang berisikan karakter lebih.

Line 11 : *int,* tipe data *integer* atau menerima bilangan bulat.

Line 41 : *HashMap*, adalah *class* yang berisi sekumpulan pasangan nilai atau  *value*  dan *key.*

Line 43-54 : *month.put,* metode menambahkan *key* bulan.

Line 55 dan 60 : *If Else* untuk memberikan kondisi *true* atau *false.*

### Main

Line 1 : *Package,* untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3-4: *Import,*  suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 6-7 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 6 dan 31 : {}, penanda awal dan akhir *block code*.

Line 8 : *String*, tipe data teks yang berisikan karakter lebih.

Line 9 : *int,* tipe data *integer* atau menerima bilangan bulat.

Line 10, : *Scanner* untuk menginput data.

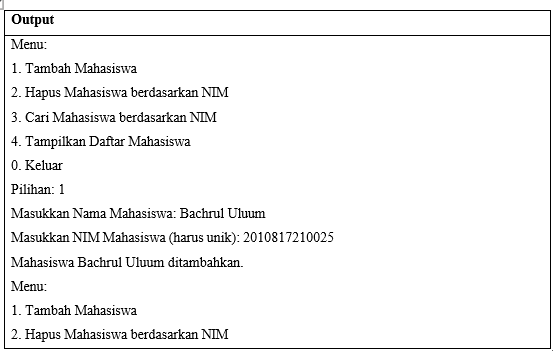
Line 13-14 : *LinkedList,*  suatu cara untuk menyimpan data dengan struktur.

Line 19 dan 26 : *for,*  untuk perulangan pada *statement* secara berulang – ulang selama terpenuhi.

# SOAL 3

Buatlah program yang mengimplementasikan enkapsulasi dan collection dengan ketentuan seperti berikut.

* 1. Buatlah kelas dengan nama Mahasiswa
  2. Terdapat 2 atribut pada kelas, yaitu nama dan nim
  3. Inisiasi nilai atribut dilakukan pada konstruktor
  4. Terdapat getter untuk 2 atribut tadi
  5. Gunakan collection dengan tipe ArrayList untuk menyimpan objek Mahasiswa
  6. Program bersifat dinamis dan interaktif
  7. Programa dapat melakukan operasi seperti berikut:
     + Tambah Mahasiswa, menambahkan objek baru ke ArrayList
     + Hapus Mahasiswa, menhapus data mahasiswa dari ArrayList berdasarkan NIM
     + Cari Mahasiswa, menampilkan data mahasiswa berdasarkan NIM yang di input oleh pengguna
     + Tampilkan seluruh data Mahasiswa, menampilkan seluruh data mahasiswa dari ArrayList. Tampilkan nama dan NIM mahasiswa
     + Keluar, program berhenti dan seluruh data pada ArrayList dihapus, ketika program dijalankan ulang, ArrayList masih kosong.







Tabel 7.3 Soal 3

## Source Code

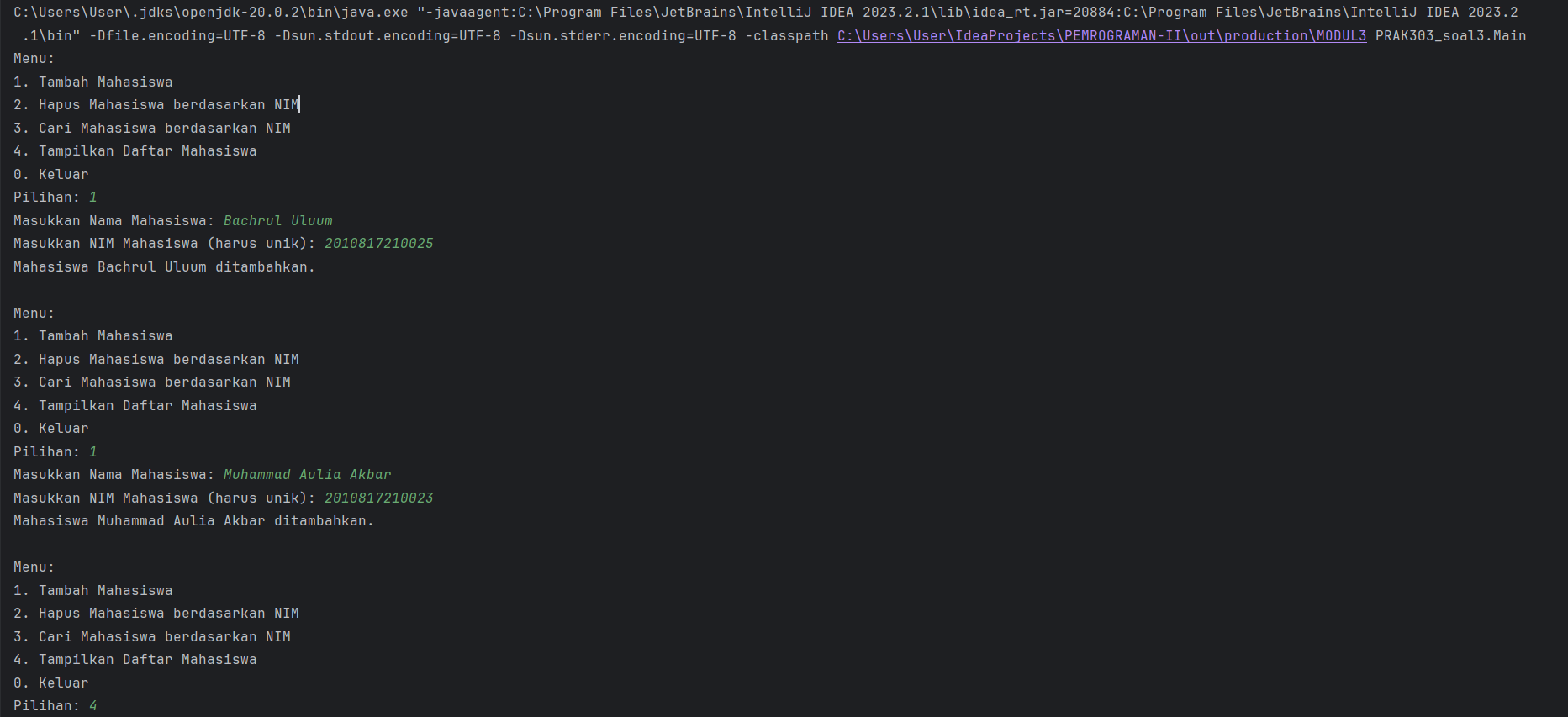
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19 | package PRAK303\_soal3;  class Mahasiswa {  private String nama;  private String nim;   public Mahasiswa(String nama, String nim) {  this.nama = nama;  this.nim = nim;  }   public String getNama() {  return nama;  }   public String getNim() {  return nim;  } } |

Tabel 8.3 Source Code Soal 3 Kopi

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  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  97  98  99  100  101  102  103  104  105  106  107 | package PRAK303\_soal3;  import java.util.ArrayList; import java.util.Scanner;  public class Main {  public static void main(String[] args) {  Scanner sc = new Scanner(System.*in*);  ArrayList<Mahasiswa> mahasiswaList = new ArrayList<>();   while (true) {  System.*out*.println("Menu:");  System.*out*.println("1. Tambah Mahasiswa");  System.*out*.println("2. Hapus Mahasiswa berdasarkan NIM");  System.*out*.println("3. Cari Mahasiswa berdasarkan NIM");  System.*out*.println("4. Tampilkan Daftar Mahasiswa");  System.*out*.println("0. Keluar");  System.*out*.print("Pilihan: ");  int pilihan = sc.nextInt();   if (pilihan == 0) {  mahasiswaList.clear();  System.*out*.println("Terima kasih!");  break;  } else if (pilihan == 1) {  System.*out*.print("Masukkan Nama Mahasiswa: ");  sc.nextLine();  String nama = sc.nextLine();  System.*out*.print("Masukkan NIM Mahasiswa (harus unik): ");  String nim = sc.nextLine();   boolean isNimExist = false;  for (Mahasiswa mahasiswa : mahasiswaList) {  if (mahasiswa.getNim().equals(nim)) {  isNimExist = true;  break;  }  }   if (isNimExist) {  System.*out*.println("NIM sudah digunakan.");  } else {  Mahasiswa mahasiswa = new Mahasiswa(nama, nim);  mahasiswaList.add(mahasiswa);  System.*out*.println("Mahasiswa " + nama + " ditambahkan.");  }  } else if (pilihan == 2) {  System.*out*.print("Masukkan NIM Mahasiswa yang akan dihapus: ");  String nim = sc.next();   boolean isNimExist = false;  for (Mahasiswa mahasiswa : mahasiswaList) {  if (mahasiswa.getNim().equals(nim)) {  isNimExist = true;  mahasiswaList.remove(mahasiswa);  System.*out*.println("Mahasiswa dengan NIM " + nim + " dihapus.");  break;  }  }   if (!isNimExist) {  System.*out*.println("NIM tidak ditemukan.");  }  } else if (pilihan == 3) {  System.*out*.print("Masukkan NIM Mahasiswa yang akan dicari: ");  String nim = sc.next();   boolean isNimExist = false;  for (Mahasiswa mahasiswa : mahasiswaList) {  if (mahasiswa.getNim().equals(nim)) {  isNimExist = true;  System.*out*.println("NIM: " + mahasiswa.getNim() + ", Nama: " + mahasiswa.getNama());  break;  }  }  if (!isNimExist) {  System.*out*.println("NIM tidak ditemukan.");  }  } else if (pilihan == 4) {  System.*out*.println("Daftar Mahasiswa:");  for (Mahasiswa mahasiswa : mahasiswaList) {  System.*out*.println("NIM: " + mahasiswa.getNim() + ", Nama: " + mahasiswa.getNama());  }  } else {  System.*out*.println("Pilihan tidak valid.");  }  System.*out*.println();  }  } } |

Tabel 9.3 Source Code Soal 3 Main

## Output Program



Gambar 3.3 Output Screenshoot Soal 3

## Pembahasan

### Kopi

Line 1 : *Package,* untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 4-5 : *Private,* kunci pada java yang dapat diakses oleh kelas itu sendiri.

Line 7,12, dan 16 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 5 dan 70 : {}, penanda awal dan akhir *block code*.

Line 4-5, 7, 12, dan 16 : *String*, tipe data teks yang berisikan karakter lebih.

Line 13 dan 17 : *return* untuk mengembalikan nilai.

### Main

Line 1 : *Package,* untuk pengelompokan *class* atau *interface* menjadi satu unit.

Line 3-4: *Import,*  suatu perintah untuk memasukkan suatu *method* atau perintah dalam Bahasa Java.

Line 6-7 : *Public*, kunci pada java yang dapat diakses kelas *class* lain.

Line 6 dan 107 : {}, penanda awal dan akhir *block code*.

Line 8, : *Scanner* untuk menginput data.

Line 9: *ArrayList,*  untuk menyimpan beberapa nilai dalam satu variable.

Line 12 : *While,* perulangan yang selalu dilakukan, jika perulangan tidak diketahui.

Line 13-23, 28, 31, 35, 48, 54, 58, 67, 74, 78, 86, 98, 102, dan 104 : *System.out.println*  mencetak sesuatu dari program sebagai *output.*

Line 26 dan 30 : *If Else* untuk memberikan kondisi *true* atau *false.*

# LINK GIT HUB