

# LAPORAN PRAKTIKUM

## JOBSHEET I

### KONSEP DASAR PEMROGRAMAN

#### 1. Tujuan Praktikum

Setelah melakukan materi praktikum ini, mahasiswa mampu:

1. Mengimplementasikan pemilihan, perulangan, array, dan fungsi dalam kode program Java

#### 2. Praktikum

##### 2.1 Pemilihan

**Waktu percobaan: 50 menit**

Materi pada praktikum ini telah dijelaskan pada matakuliah Dasar Pemrograman, Sehingga didalam praktikum ini, tidak akan dilakukan langkah-langkah percobaan. Jawablah pertanyaan-pertanyaan yang ada berikut ini:

##### 2.2.1 Praktikum Pemilihan Pertanyaan

1. Buatlah program untuk menghitung nilai akhir dari mahasiswa dengan ketentuan 20% nilai tugas, 20% dari nilai kuis, 30% nilai UTS, dan 30% nilai UAS. Setiap nilai yang dimasukkan mempunyai batas nilai 0 - 100. Ketika pengguna memasukkan di luar rentang tersebut maka akan keluar output “nilai tidak valid”. Ketika nilai akhir sudah didapatkan selanjutnya lakukan konversi nilai dengan ketentuan sebagai berikut:

Nilai Angka	Nilai Mutu		
	Nilai Huruf	Nilai Setara	Kualifikasi
$80 < N \leq 100$	A	4	Sangat Baik
$73 < N \leq 80$	B+	3,5	Lebih dari Baik
$65 < N \leq 73$	B	3	Baik
$60 < N \leq 65$	C+	2,5	Lebih dari Cukup
$50 < N \leq 60$	C	2	Cukup
$39 < N \leq 50$	D	1	Kurang
$N \leq 39$	E	0	Gagal

Jika Nilai Huruf yang didapatkan adalah A, B+, B, C+, C maka LULUS, jika nilai huruf D dan E maka TIDAK LULUS.

- Input dari program berupa komponen nilai tugas, kuis, UTS, UAS
- Output dari program “nilai tidak valid” jika nilai yang dimasukkan diluar ketentuan
- Output dari program berupa hasil nilai akhir, nilai huruf, dan keterangan LULUS/TIDAK LULUS

### Contoh hasil Running program

```
Program Menghitung Nilai Akhir
=====
Masukkan Nilai Tugas: 85
Masukkan Nilai Kuis: 90
Masukkan Nilai UTS: 120
Masukkan Nilai UAS: 70
=====
nilai tidak valid
=====
Mamluatuls-MacBook-Air:Praktikum
```

```
Program Menghitung Nilai Akhir
=====
Masukkan Nilai Tugas: 90
Masukkan Nilai Kuis: 40
Masukkan Nilai UTS: 75
Masukkan Nilai UAS: 85
=====
nilai akhir : 74.0
Nilai Huruf :B+
=====
SELAMAT ANDA LULUS
```

1. Buka text editor. Buat file baru, beri nama **Pemilihan<NoAbsen>.java**
2. Mengimport **package Scanner** untuk membaca input dari pengguna.
3. Tuliskan struktur dasar bahasa pemrograman Java yang terdiri dari **fungsi main()**
4. Mendeklarasikan **fungsi nilaiAkhir** dengan parameter nilai bertipe data double.  
Fungsi mengembalikan true jika nilai berada di rentang 0-100, dan false jika tidak.

```
private static boolean nilaiAkhir(double nilai) {
    return nilai >= 0 && nilai <= 100;
}
```

5. Mendeklarasikan fungsi `konversiNilai` dengan parameter **nilai** bertipe data **double**.  
Fungsi ini mengembalikan nilai berdasarkan rentang nilai pada struktur pemilihan **if-else if-else**.

```
private static String konversiNilai(double nilai) {
    if (nilai > 80 && nilai <= 100) {
        return "A";
    } else if (nilai > 73 && nilai <= 80) {
        return "B+";
    } else if (nilai > 65 && nilai <= 73) {
        return "B";
    } else if (nilai > 60 && nilai <= 65) {
        return "C+";
    } else if (nilai > 50 && nilai <= 60) {
        return "C";
    } else if (nilai > 39 && nilai <= 50) {
        return "D";
    } else {
        return "E";
    }
}
```

6. Pada fungsi **main** Inisialisasi objek Scanner untuk input dari pengguna.

```
Run | Debug
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
```

7. Menampilkan judul program dengan menggunakan **System.out.println**, dan Meminta pengguna memasukkan nilai tugas, kuis, UTS, dan UAS dengan menggunakan **System.out.print**.

```
System.out.println(x:"Program Menghitung Nilai Akhir");
System.out.println(x:"=====");
System.out.print(s:"Masukkan Nilai Tugas : ");
double nilaiTugas = sc.nextDouble();
System.out.print(s:"Masukkan Nilai Kuis : ");
double nilaiKuis = sc.nextDouble();
System.out.print(s:"Masukkan Nilai UTS : ");
double nilaiUTS = sc.nextDouble();
System.out.print(s:"Masukkan Nilai UAS : ");
double nilaiUAS = sc.nextDouble();
System.out.println(x:"=====");
```

8. Memastikan bahwa nilai yang dimasukkan berada dalam rentang 0-100 dengan memeriksa menggunakan fungsi **nilaiAkhir**.

```
if (!nilaiAkhir(nilaiTugas) || !nilaiAkhir(nilaiKuis) || !nilaiAkhir(nilaiUTS) || !nilaiAkhir(nilaiUAS)) {
    System.out.println(x:"Nilai tidak valid. Harap masukkan nilai antara 0-100.");
    return;
}
```

9. Menghitung nilai akhir berdasarkan bobot yang diberikan pada masing-masing komponen penilaian.

```
double nilaiAkhir = 0.2 * nilaiTugas + 0.2 * nilaiKuis + 0.3 * nilaiUTS + 0.3 * nilaiUAS;
```



10. Menggunakan **fungsi konversiNilai** untuk mengonversi nilai akhir menjadi nilai huruf sesuai dengan kriteria dan akan Hasil konversi disimpan dalam variabel nilaiHuruf.

```
String nilaiHuruf = konversiNilai(nilaiAkhir);
```

11. Menentukan kelayakan kelulusan dengan menggunakan nilai huruf jika A, B+, B, C+, atau C, maka pesan "SELAMAT ANDA LULUS" akan ditampilkan. Jika tidak, maka pesan "MAAF ANDA TIDAK LULUS" akan ditampilkan.

```
System.out.println((nilaiHuruf.equals(anObject:"A") || nilaiHuruf.equals(anObject:"B+") || nilaiHuruf.equals(anObject:"B") || nilaiHuruf.equals(anObject:"C+") || nilaiHuruf.equals(anObject:"C") ? "SELAMAT ANDA LULUS" : "MAAF ANDA TIDAK LULUS");
sc.close();
```

12. Hasil running.

```
PS D:\Algoritma Praktikum\Praktikum 1> & 'C:\Program Files\Java\jdk-21\bin\java.exe' -cp 'C:\Users\LENOVO\AppData\Roaming\Code\User\workspaceStorage\1_63b1dc6b\bin' 'Pemilihan24'
Program Menghitung Nilai Akhir
=====
Masukkan Nilai Tugas : 85
Masukkan Nilai Kuis : 90
Masukkan Nilai UTS : 120
Masukkan Nilai UAS : 70
=====
Nilai tidak valid. Harap masukkan nilai antara 0-100.
PS D:\Algoritma Praktikum\Praktikum 1>
```

## 2.2 Perulangan

**Waktu percobaan: 50 menit**

Materi pada praktikum ini telah dijelaskan pada mata kuliah Dasar Pemrograman. Sehingga di dalam praktikum ini, tidak akan dilakukan langkah-langkah percobaan. Jawablah pertanyaan-pertanyaan yang ada berikut ini:

### 2.3.1 Praktikum Perulangan Pertanyaan

1. Buatlah program yang dapat menampilkan deretan bilangan dari angka 1 sampai n kecuali angka 6 dan 10, angka ganjil dicetak dengan asterik "\*", angka genap dicetak sesuai bilangan aslinya, dengan n = 2-digit terakhir NIM anda.

\*bila  $n < 10$  maka tambahkan 10 ( $n += 10$ ) Contoh:

Input NIM: 2341720102 maka  $n=12$

**OUTPUT: \* 2 \* 4 \* \* 8 \* \* 12**

Contoh 2:

Input NIM: 2341720113 maka  $n=13$

**OUTPUT: \* 2 \* 4 \* \* 8 \* \* 12**

Contoh hasil running program

```
Masukkan Nim :2341720102
=====
n : 12
* 2 * 4 * * 8 * * 12
```

1. Buka text editor. Buat file baru, beri nama **Perulangan<NoAbsen>.java**
2. Mengimport **package Scanner** untuk membaca input dari pengguna.
3. Tuliskan struktur dasar bahasa pemrograman Java yang terdiri dari **fungsi main()**

```
import java.util.Scanner;
public class Perulangan24{
    Run | Debug
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
```

4. Menampilkan pesan untuk meminta pengguna memasukkan NIM.

```
System.out.print(s:"Input NIM: ");
long nim = sc.nextLong();
```

5. Menghitung nilai **n** dengan mengambil dua digit terakhir dari NIM. Jika nilai **n** kurang dari 10, tambahkan 10 ke nilai **n**.

```
int n = (int) (nim % 100);
if (n < 10) {
    n += 10;
}
```

6. Menampilkan nilai **n** ke layar.

```
System.out.println(x:"=====");
System.out.println("Nilai n: " + n);
```

7. Memulai perulangan for dengan variabel **i** dimulai dari 1 hingga nilai **n**. Pengecualian nilai **i** jika sama dengan 6 atau 10 maka nilai **i** tidak akan ditampilkan. Saat nilai **i** genap, mencetak nilai **i**; jika ganjil, mencetak tanda bintang (\*).

```
for (int i = 1; i <= n; i++) {
    if (i != 6 && i != 10) {
        if (i % 2 == 0) {
            System.out.print(i + " ");
        } else {
            System.out.print(s:"* ");
        }
    }
}
```

8. Hasil Running.

```
PS D:\Algoritma Praktikum\Praktikum 1> & 'C:\Program Files\Java
ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\LENOVO\AppData
e048d2371a2fb3bd47a1fbf\redhat.java\jdt_ws\Praktikum 1_63b1dc6b\
Input NIM: 2341720102
=====
Nilai n: 12
* 2 * 4 * * 8 * * 12
PS D:\Algoritma Praktikum\Praktikum 1> |
```

## 2.3 Array

**Waktu percobaan: 50 menit**

Materi pada praktikum ini telah dijelaskan pada mata kuliah Dasar Pemrograman, sehingga di dalam praktikum ini, tidak akan dilakukan langkah-langkah percobaan. Jawablah pertanyaan-pertanyaan yang ada berikut ini:

### 2.4.1 Praktikum Array

#### Pertanyaan

1. Buatlah program untuk menghitung IP Semester dari matakuliah yang Anda tempuh semester lalu. Formula untuk menghitung IP semester sebagai berikut:

$$IP \text{ Semester} = \frac{\sum_i (Nilai \text{ Setara}_i * bobot \text{ SKS}_i)}{\sum SKS}$$

Nilai setara didapatkan dari tabel konversi berikut ini:

Nilai Angka	Nilai Mutu		
	Nilai Huruf	Nilai Setara	Kualifikasi
80 < N ≤ 100	A	4	Sangat Baik
73 < N ≤ 80	B+	3,5	Lebih dari Baik
65 < N ≤ 73	B	3	Baik
60 < N ≤ 65	C+	2,5	Lebih dari Cukup
50 < N ≤ 60	C	2	Cukup
39 < N ≤ 50	D	1	Kurang
N ≤ 39	E	0	Gagal

Input dari program berupa nama mata kuliah, bobot SKS, serta nilai angka dari mata kuliah tersebut.

Contoh Hasil Running Program

```

=====
Program Menghitung IP Semester
=====
masukkan nilai Angka untuk MK Pancasila: 75
masukkan nilai Angka untuk MK Konsep Teknologi Informasi: 85
masukkan nilai Angka untuk MK Critical Thinking dan Problem Solving: 70
masukkan nilai Angka untuk MK Matematika Dasar: 85
masukkan nilai Angka untuk MK Bahasa Inggris: 85
masukkan nilai Angka untuk MK Dasar Pemrograman: 62
masukkan nilai Angka untuk MK Praktikum Dasar Pemrograman: 62
masukkan nilai Angka untuk MK Keselamatan dan Kesehatan Kerja: 85
=====
hasil Konversi Nilai
=====
MK                                Nilai Angka    Nilai Huruf    Bobot Nilai
Pancasila                        75.00          B+             3.50
Konsep Teknologi Informasi        85.00          A              4.00
Critical Thinking dan Problem Solving 70.00          B              3.00
Matematika Dasar                  85.00          A              4.00
Bahasa Inggris                    85.00          A              4.00
Dasar Pemrograman                 62.00          C+             2.50
Praktikum Dasar Pemrograman       62.00          C+             2.50
Keselamatan dan Kesehatan Kerja    85.00          A              4.00
=====
IP : 3.42
    
```

1. Buka text editor. Buat file baru, beri nama **Array<NoAbsen>.java**
2. Mengimport **package Scanner** untuk membaca input dari pengguna.
3. Mengimport **package DecimalFormat** untuk memformat angka desimal.

```

import java.text.DecimalFormat;
import java.util.Scanner;
    
```

4. Mendeklarasikan **fungsi konversiNilaiKeSetara** dengan parameter nilai bertipe data double. Fungsi ini mengembalikan nilai berdasarkan rentang nilai menggunakan struktur pemilihan if-else if-else.

```

private static double konversiNilaiKeSetara(double nilai) {
    if (nilai > 80 && nilai <= 100) {
        return 4.0;
    } else if (nilai > 73 && nilai <= 80) {
        return 3.5;
    } else if (nilai > 65 && nilai <= 73) {
        return 3.0;
    } else if (nilai > 60 && nilai <= 65) {
        return 2.5;
    } else if (nilai > 50 && nilai <= 60) {
        return 2.0;
    } else if (nilai > 39 && nilai <= 50) {
        return 1.0;
    } else {
        return 0.0;
    }
}
    
```

5. Mendeklarasikan **fungsi konversiNilaiKeHuruf** yang menerima nilai setara dan mengembalikan nilai huruf sesuai dengan rentang tertentu. Implementasi logika konversi nilai setara ke nilai huruf.

```
private static String konversiNilaiKeHuruf(double nilai) {
    if (nilai == 4.0) {
        return "A";
    } else if (nilai == 3.5) {
        return "B+";
    } else if (nilai == 3.0) {
        return "B";
    } else if (nilai == 2.5) {
        return "C+";
    } else if (nilai == 2.0) {
        return "C";
    } else if (nilai == 1.0) {
        return "D";
    } else {
        return "E";
    }
}
```

6. Mendeklarasikan **fungsi konversiIPKeKualifikasi** yang menerima Indeks Prestasi (IP) dan mengembalikan kualifikasi berdasarkan rentang nilai.

```
private static String konversiIPKeKualifikasi(double ip) {
    if (ip >= 3.5) {
        return "Sangat Baik";
    } else if (ip >= 3.0) {
        return "Lebih dari Baik";
    } else if (ip >= 2.5) {
        return "Baik";
    } else if (ip >= 2.0) {
        return "Lebih dari Cukup";
    } else if (ip >= 1.0) {
        return "Cukup";
    } else {
        return "Gagal";
    }
}
```

7. Tuliskan struktur dasar bahasa pemrograman Java yang terdiri dari fungsi **main()**

```
public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    DecimalFormat df = new DecimalFormat(pattern:"#.##");
```

8. Mendeklarasikan array untuk nama mata kuliah dan bobot SKS.

```
String[] mataKuliah = {"Pancasila", "Konsep Teknologi Informasi", "Critical Thinking dan Problem  
"Matematika Dasar", "Bahasa Inggris", "Dasar Pemrograman", "Praktikum Dasar Pemrograman",  
"Keselamatan dan Kesehatan Kerja"};  
double[] bobotSKS = {3, 3, 3, 4, 3, 3, 2, 2};
```

9. Membuat array untuk nilai setara dengan panjang sesuai jumlah mata kuliah.  
Menginisialisasi variabel totalBobot.

```
double[] nilaiSetara = new double[mataKuliah.length];  
double totalBobot = 0;
```



10. Melakukan iterasi untuk setiap mata kuliah, meminta pengguna memasukkan nilai angka, kemudian mengonversi nilai angka menjadi nilai setara menggunakan fungsi **konversiNilaiKeSetara**, dan menghitung total bobot SKS.

```
for (int i = 0; i < mataKuliah.length; i++) {
    System.out.print("Masukkan nilai Angka untuk MK " + mataKuliah[i] + " : ");
    double nilai = scanner.nextDouble();

    nilaiSetara[i] = konversiNilaiKeSetara(nilai);
    totalBobot += bobotSKS[i];
}
```

11. Menampilkan header tabel serta hasil konversi nilai untuk setiap mata kuliah.

```
System.out.println(x:"=====");
System.out.println(x:"      Hasil Konversi Nilai");
System.out.println(x:"=====");
System.out.printf(format:"| %-45s | %-15s | %-15s | %-15s |\n", ...args:"Mata Kuliah", "Nilai Angka", "Nilai Setara", "Bobot SKS");
System.out.println(x:"=====");

for (int i = 0; i < mataKuliah.length; i++) {
    System.out.printf(format:"| %-45s | %-15s | %-15s | %-15s |\n", mataKuliah[i], df.format(nilaiSetara[i]), df.format(bobotSKS[i]), konversiNilaiKeHuruf(nilaiSetara[i]));
}
```

12. Menghitung total nilai untuk semua mata kuliah.

```
double totalNilai = 0;
for (int i = 0; i < mataKuliah.length; i++) {
    totalNilai += nilaiSetara[i] * bobotSKS[i];
}
```

13. Menghitung IP semester dengan membagi total nilai dengan total bobot SKS.

```
double ipSemester = totalNilai / totalBobot;
```

14. Menampilkan IP semester dengan dua angka desimal dan mengonversi menjadi kualifikasi dan menampilkannya.

```
double ipSemester = totalNilai / totalBobot;

System.out.println(x:"=====");
System.out.println("\nIP Semester: " + df.format(ipSemester));
System.out.println("Kualifikasi: " + konversiIPKeKualifikasi(ipSemester));
```

## 2.4 Fungsi

**Waktu percobaan: 50 menit**

Materi pada praktikum ini telah dijelaskan pada matakuliah Dasar Pemrograman, sehingga didalam praktikum ini, tidak akan dilakukan langkah-langkah percobaan. Jawablah pertanyaan-pertanyaan yang ada berikut ini:

### 2.5.1 Praktikum Fungsi

#### Pertanyaan

RoyalGarden adalah toko bunga yang memiliki banyak cabang. Setiap hari stock bunga dan bungabunga yang dijual selalu dicatat dengan rincian seperti berikut ini:

Baris = Cabang Toko, Kolom = Stock bunga pada hari x

	Aglonema	Keladi	Alocasia	Mawar
RoyalGarden 1	10	5	15	7
RoyalGarden 2	6	11	9	12
RoyalGarden 3	2	10	10	5
RoyalGarden 4	5	7	12	9

Rincian Harga Aglonema =75.000, Keladi = 50.000, Alocasia =60.000, Mawar =10.000.

1. Buatlah fungsi untuk menampilkan pendapatan setiap cabang jika semua bunga habis terjual.
2. Buatlah fungsi untuk mengetahui jumlah Stock setiap jenis bunga pada cabang royalgarden 4.

Jika terdapat informasi tambahan berupa pengurangan stock karena bunga tersebut mati.

Dengan rincian Aglonema -1, Keladi -2, Alocasia -0, Mawar -5.

1. Buka text editor. Buat file baru, beri nama **Fungsi<NoAbsen>.java**
2. Mengimport **package Scanner** untuk membaca input dari pengguna.
3. Tuliskan struktur dasar bahasa pemrograman Java yang terdiri dari fungsi **main()**

```
import java.util.Scanner;

public class Fungsi24 {

    Run | Debug
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
```

4. Inisialisasi dan deklarasikan array untuk menyimpan nama bunga, stok awal cabang, harga bunga, dan pengurangan stok cabang 4.

```
String[] namaBunga = { "Aglonema", "Keladi", "Alocasia", "Mawar" };
int[][] stockAwalCabang = {
    { 10, 6, 2, 5 },
    { 8, 4, 3, 7 },
    { 2, 10, 10, 5 },
    { 5, 7, 12, 9 }
};
int[] hargaBunga = { 75000, 50000, 60000, 10000 };
int[] penguranganStockCabang4 = { -1, -2, 0, -5 };
```

5. Membuat Array 2 dimensi untuk menyimpan total pendapatan setiap cabang.

```
int[][] totalPendapatanCabang = new int[4][5];
```

6. Melakukan iterasi untuk setiap cabang, serta memanggil fungsi **hitungPendapatan** untuk menghitung total pendapatan setiap bunga di cabang tersebut.

```
for (int i = 0; i < 4; i++) {
    totalPendapatanCabang[i] = hitungPendapatan(namaBunga, stockAwalCabang[i], hargaBunga);
}
```

7. Menampilkan total pendapatan setiap bunga di setiap cabang. Serta menghitung dan menampilkan total pendapatan cabang.

```
for (int i = 0; i < namaBunga.length; i++) {
    System.out.printf(format: "\nCabang %d:\n", i+1);
    for (int j = 0; j < 4; j++) {
        System.out.printf(format: "Pendapatan %s: Rp%d\n", namaBunga[j], totalPendapatanCabang[j][i]);
    }
    int totalPendapatan = totalPendapatanCabang[0][i] + totalPendapatanCabang[1][i] + totalPendapatanCabang[2][i] + totalPendapatanCabang[3][i];
    System.out.printf(format: "Total Pendapatan Cabang %d: Rp%d\n", i+1, totalPendapatan);
}
```

8. Memanggil fungsi **hitungStock** untuk menghitung total stok cabang 4 dan menampilkannya.

```
int[] totalStockCabang4 = hitungStock(stockAwalCabang[3], penguranganStockCabang4);
System.out.println(x: "\nTotal Stock Royal Garden 4:");
System.out.println(x: "=====");
System.out.printf(format: "| %-10s | %-10s | %-10s | \n", ...args: "Nama Bunga", "Stock Awal", "Total Stock");
System.out.println(x: "=====");
for (int i = 0; i < namaBunga.length; i++) {
    System.out.printf(format: "| %-10s | %-10d | %-10d | \n", namaBunga[i], stockAwalCabang[3][i], totalStockCabang4[i]);
}
System.out.println(x: "=====");
```

9. Deklarasikan fungsi **hitungPendapatan** untuk menghitung pendapatan setiap bunga di suatu cabang, termasuk perhitungan total terjual dan pendapatan untuk setiap bunga, dan mengembalikan array pendapatan.

```
static int[] hitungPendapatan(String[] namaBunga, int[] stockAwal, int[] hargaBunga) {
    int[] totalTerjual = new int[namaBunga.length];
    int[] pendapatan = new int[namaBunga.length + 1];
    int totalPendapatan = 0;
    for (int i = 0; i < namaBunga.length; i++) {
        totalTerjual[i] = stockAwal[i];
    }
    for (int i = 0; i < namaBunga.length; i++) {
        pendapatan[i] = totalTerjual[i] * hargaBunga[i];
        totalPendapatan += pendapatan[i];
    }

    pendapatan[4] = totalPendapatan;
    return pendapatan;
}
```

10. **Fungsi hitungStock** dideklarasikan untuk menghitung total stok setiap bunga setelah dikurangkan, serta mengembalikan array total stok.

```
static int[] hitungStock(int[] stockAwal, int[] penguranganStock) {
    int[] totalStock = new int[stockAwal.length];

    for (int i = 0; i < stockAwal.length; i++) {
        totalStock[i] = stockAwal[i] + penguranganStock[i];
    }

    return totalStock;
}
```

11. Hasil Running.

```
Cabang 1:
Pendapatan Aglonema: Rp750000
Pendapatan Aglonema: Rp600000
Pendapatan Aglonema: Rp150000
Pendapatan Aglonema: Rp375000
Total Pendapatan Cabang 1: Rp5105000
```

```
Cabang 2:
Pendapatan Keladi: Rp300000
Pendapatan Keladi: Rp200000
Pendapatan Keladi: Rp500000
Pendapatan Keladi: Rp350000
Total Pendapatan Cabang 2: Rp5105000
```

```
Cabang 3:
Pendapatan Alocasia: Rp120000
Pendapatan Alocasia: Rp180000
Pendapatan Alocasia: Rp600000
Pendapatan Alocasia: Rp720000
Total Pendapatan Cabang 3: Rp5105000
```

```
Cabang 4:
Pendapatan Mawar: Rp50000
Pendapatan Mawar: Rp70000
Pendapatan Mawar: Rp50000
Pendapatan Mawar: Rp90000
Total Pendapatan Cabang 4: Rp5105000
```

```
Total Stock Royal Garden 4:
=====
| Nama Bunga | Stock Awal | Total Stock |
=====
| Aglonema   | 5          | 4          |
| Keladi     | 7          | 5          |
| Alocasia   | 12         | 12         |
| Mawar      | 9          | 4          |
=====
```

### 3. Tugas

#### Waktu pengerjaan: 100 menit

- Susun program untuk membuat dua buah array berikut isinya sebagai berikut. Array pertama adalah array satu dimensi char KODE[10], berisi kode plat mobil. Array kedua, array dua dimensi char KOTA[10][12] berisi nama kota yang berpasangan dengan kode plat mobil.

Ilustrasi tampilan array tersebut adalah sebagai berikut:

A	B	A	N	T	E	N						
B	J	A	K	A	R	T	A					
D	B	A	N	D	U	N	G					
E	C	I	R	E	B	O	N					
F	B	O	G	O	R							
G	P	E	K	A	L	O	N	G	A	N		
H	S	E	M	A	R	A	N	G				
L	S	U	R	A	B	A	Y	A				
N	M	A	L	A	N	G						
T	T	E	G	A	L							

Ketika pengguna memberikan input kode plat nomor maka program akan mengeluarkan nama kota dari kode plat nomor tersebut.

1. Buka text editor. Buat file baru, beri nama **Tugas1<NoAbsen>.java**
2. Mengimport **package Scanner** untuk membaca input dari pengguna.
3. Tuliskan struktur dasar bahasa pemrograman Java yang terdiri dari **fungsi main()**

```
import java.util.Scanner;
public class Tugas124 {

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

4. Inisialisasi dan deklarasi array untuk menyimpan kode plat mobil dan array 2D untuk menyimpan kota-kota yang sesuai dengan setiap kode.

```
char[] kode = {'A', 'B', 'D', 'E', 'F', 'L', 'R', 'T', 'U', 'Z'};

String[][] kota = {
    {"J", "a", "k", "a", "r", "t", "a"},
    {"S", "u", "r", "a", "b", "a", "y", "a"},
    {"B", "a", "n", "d", "u", "n", "g"},
    {"Y", "o", "g", "y", "a", "k", "a", "r", "t", "a"},
    {"S", "e", "m", "a", "r", "a", "n", "g"},
    {"M", "e", "d", "a", "n"},
    {"M", "a", "k", "a", "s", "s", "a", "r"},
    {"D", "e", "n", "p", "a", "s", "a", "r"},
    {"M", "a", "n", "a", "d", "o"},
    {"P", "e", "k", "a", "n", "b", "a", "r", "u"}
};
```

5. Meminta pengguna memasukkan kode plat mobil.

```
System.out.print(s:"Masukkan kode plat mobil: ");
char inputKode = sc.next().charAt(index:0);
```

6. Melakukan iterasi untuk setiap kode plat. Jika kode plat cocok dengan input pengguna, menampilkan kota yang sesuai.

```
char[] kode = {'A', 'B', 'D', 'E', 'F', 'L', 'R', 'T', 'U', 'Z'};

String[][] kota = {
    {"J", "a", "k", "a", "r", "t", "a"},
    {"S", "u", "r", "a", "b", "a", "y", "a"},
    {"B", "a", "n", "d", "u", "n", "g"},
    {"Y", "o", "g", "y", "a", "k", "a", "r", "t", "a"},
    {"S", "e", "m", "a", "r", "a", "n", "g"},
    {"M", "e", "d", "a", "n"},
    {"M", "a", "k", "a", "s", "s", "a", "r"},
    {"D", "e", "n", "p", "a", "s", "a", "r"},
    {"M", "a", "n", "a", "d", "o"},
    {"P", "e", "k", "a", "n", "b", "a", "r", "u"}
};
```

7. Jika tidak ada kode plat yang cocok, menampilkan pesan bahwa kode plat tidak valid.

```
if (!kodePlat) {
    System.out.println(x:"Kode plat mobil tidak valid.");
}
```

8. Hasil Running.

```
PS D:\Algoritma Praktikum\Praktikum1> & 'C:\Program Files\Java\jdk-17.0.2\bin\java.exe' -cp 'C:\Users\LENOVO\AppData\Roaming\jdwt_ws\Praktikum1_b8e4b497\bin' 'Tugas124'
Masukkan kode plat mobil: L
Kota yang sesuai dengan kode L adalah: Medan
PS D:\Algoritma Praktikum\Praktikum1> d:; cd 'd:\Algoritma Praktikum\Praktikum1' & java -XX:+ShowCodeDetailsInExceptionMessages -e\6dc8b808861e2534737b19b73c6dc0f0\redhat.java\jdwt_ws\Praktikum1
Masukkan kode plat mobil: B
Kota yang sesuai dengan kode B adalah: Surabaya
PS D:\Algoritma Praktikum\Praktikum1>
```

2. Buat program untuk menghitung rumus kecepatan, jarak, dan waktu

Berikut adalah persamaan untuk menghitung rumus tersebut:

#### Rumus Kecepatan

$$v = \frac{s}{t}$$

#### Rumus Jarak

$$s = v \cdot t$$

#### Rumus Waktu

$$t = \frac{s}{v}$$

Keterangan:

$v$  = kecepatan

$s$  = jarak

$t$  = waktu

Program yang dibuat memiliki fungsi sebagai berikut:

- a. Menu (Untuk memilih rumus yang akan dihitung (kecepatan/jarak/waktu)

```
Run | Debug
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);

    while (true) {
        System.out.println(x:"Menu:");
        System.out.println(x:"1. Hitung Kecepatan");
        System.out.println(x:"2. Hitung Jarak");
        System.out.println(x:"3. Hitung Waktu");
        System.out.println(x:"4. Keluar");

        System.out.print(s:"Pilih menu (1-4): ");
        int pilihan = sc.nextInt();

        if (pilihan == 4) {
            System.out.println(x:"Terima kasih, program selesai.");
            break;
        }
    }
}
```

- b. Menghitung hasil perhitungan Kecepatan

```
public static void hitungKecepatan() {
    Scanner sc = new Scanner(System.in);

    System.out.print(s:"Masukkan jarak (s): ");
    double jarak = sc.nextDouble();

    System.out.print(s:"Masukkan waktu (t): ");
    double waktu = sc.nextDouble();

    double kecepatan = jarak / waktu;
    System.out.println("Kecepatan (v) = " + kecepatan + " m/s");
}
```

- c. Menghitung hasil perhitungan Jarak

```
public static void hitungJarak() {
    Scanner sc = new Scanner(System.in);

    System.out.print(s:"Masukkan kecepatan (v): ");
    double kecepatan = sc.nextDouble();

    System.out.print(s:"Masukkan waktu (t): ");
    double waktu = sc.nextDouble();

    double jarak = kecepatan * waktu;
    System.out.println("Jarak (s) = " + jarak + " meter");
}
```

- d. Menghitung hasil perhitungan Waktu

```
public static void hitungWaktu() {
    Scanner sc = new Scanner(System.in);

    System.out.print(s:"Masukkan kecepatan (v): ");
    double kecepatan = sc.nextDouble();

    System.out.print(s:"Masukkan jarak (s): ");
    double jarak = sc.nextDouble();

    double waktu = jarak / kecepatan;
    System.out.println("Waktu (t) = " + waktu + " detik");
}
```

Panggil fungsi-fungsi tersebut pada fungsi main!

```
switch (pilihan) {
    case 1:
        hitungKecepatan();
        break;
    case 2:
        hitungJarak();
        break;
    case 3:
        hitungWaktu();
        break;
    default:
        System.out.println(x:"Menu tidak valid. Silakan pilih lagi.");
}
```



### Hasil Running.

```

Menu:
1. Hitung Kecepatan
2. Hitung Jarak
3. Hitung Waktu
4. Keluar
Pilih menu (1-4): 1
Masukkan jarak (s): 20
Masukkan waktu (t): 5
Kecepatan (v) = 4.0 m/s
Menu:
1. Hitung Kecepatan
2. Hitung Jarak
3. Hitung Waktu
4. Keluar
Pilih menu (1-4): 2
Masukkan kecepatan (v): 4
Masukkan waktu (t): 5
Jarak (s) = 20.0 meter
Menu:
1. Hitung Kecepatan
2. Hitung Jarak
3. Hitung Waktu
4. Keluar
Pilih menu (1-4): 3
Masukkan kecepatan (v): 4
Masukkan jarak (s): 20
Waktu (t) = 5.0 detik
Menu:
1. Hitung Kecepatan
2. Hitung Jarak
3. Hitung Waktu
4. Keluar
Pilih menu (1-4): 4
Terima kasih, program selesai.
PS D:\Algoritma Praktikum\Praktikum1>
    
```