

# Laporan Modul 3: Dasar Pemrograman Java Lanjutan

**Mata Kuliah:** Praktikum Pemrograman Berorientasi Objek

**Nama:** Muhammad Alif Arrayyan

**NIM:** 2024573010085

**Kelas:** TI-2E

## Abstrak

Laporan ini berisi rangkuman hasil praktikum Modul 3: Dasar Pemrograman Java Lanjutan, yang berfokus pada tiga topik utama, yaitu Array, Method/Function, dan String/Manipulasi String. Tujuan utama penyusunan laporan ini adalah untuk mendokumentasikan pemahaman konsep, langkah implementasi kode, hasil eksekusi, serta analisis dari setiap kegiatan praktikum dan tugas yang dilakukan.

Melalui serangkaian kegiatan tersebut, mahasiswa mampu mengimplementasikan penggunaan array satu dan dua dimensi, membuat serta memanggil method dengan berbagai tipe, dan melakukan manipulasi string yang kompleks seperti validasi dan pemformatan data. Dengan demikian, laporan ini menjadi bukti pemahaman yang kuat terhadap konsep lanjutan dalam pemrograman Java.

## Praktikum 1.1 - Array Satu Dimensi

### Dasar Teori

Array adalah struktur data homogen yang menyimpan kumpulan nilai dengan tipe data yang sama. Array di Java memiliki ukuran yang tetap (*fixed size*) dan elemennya diakses melalui indeks yang dimulai dari 0.

### Langkah Praktikum

1. Membuat package `modul_3` dan class `ArraySatuDimensi.java` .
2. Mendeklarasikan dan menginisialisasi array integer `nilaiSiswa` .
3. Menambahkan perulangan *enhanced for* untuk mencari nilai tertinggi dan terendah dalam array.
4. Menampilkan hasil array, akses elemen spesifik, serta hasil pencarian nilai ekstrem.

### Screenshoot Hasil Praktikum

```
1 package modul_3;
2
3 public class ArraySatuDimensi { new*
4     public static void main(String[] args) { new*
5
6         // Langkah 1: Deklarasi dan inisialisasi array
7         int[] nilai = {85, 90, 78, 92, 88};
8         String[] nama = {"Alice", "Bob", "Charlie", "Diana", "Eva"};
9
10    | // Langkah 2: Menampilkan panjang array
11    System.out.println("Jumlah siswa: " + nilai.length);
12
13    | // Langkah 3: Mengakses elemen array
14    System.out.println("Nama siswa pertama: " + nama[0]);
15    System.out.println("Nilai siswa pertama: " + nilai[0]);
16
17    | // Langkah 4: Menampilkan semua data menggunakan loop
18    System.out.println("\nData semua siswa:");
19    for (int i = 0; i < nama.length; i++) {
20        System.out.println((i + 1) + ". " + nama[i] + " - Nilai: " + nilai[i]);
21    }
22
23    | // Langkah 5: MODIFIKASI - Mencari nilai tertinggi dan terendah
24    | // Inisialisasi variabel tertinggi dan terendah dengan nilai elemen pertama
25    int nilaiTertinggi = nilai[0];
26    int nilaiTerendah = nilai[0];
27
28    | // Loop untuk membandingkan setiap nilai dalam array
29    for (int i = 1; i < nilai.length; i++) {
30        | // Jika nilai saat ini lebih besar dari nilaiTertinggi, update nilaiTertinggi
31        if (nilai[i] > nilaiTertinggi) {
32            | nilaiTertinggi = nilai[i];
33        }
34        | // Jika nilai saat ini lebih kecil dari nilaiTerendah, update nilaiTerendah
35        if (nilai[i] < nilaiTerendah) {
36            | nilaiTerendah = nilai[i];
37        }
38    }
39
40    | // Menampilkan hasil nilai tertinggi dan terendah
41    System.out.println("\n--- Analisis Nilai ---");
42    System.out.println("Nilai Tertinggi: " + nilaiTertinggi);
43    System.out.println("Nilai Terendah: " + nilaiTerendah);
44    }
45 }
```

```
Jumlah siswa: 5
Nama siswa pertama: Alice
Nilai siswa pertama: 85

Data semua siswa:
1. Alice - Nilai: 85
2. Bob - Nilai: 90
3. Charlie - Nilai: 78
4. Diana - Nilai: 92
5. Eva - Nilai: 88

--- Analisis Nilai ---
Nilai Tertinggi: 92
Nilai Terendah: 78
```

## Penjelasan Latihan

Pada tugas ini, saya diminta untuk membuat array satu dimensi yang menyimpan nilai ujian dari 10 siswa. Tugas ini mencakup:

- Menampilkan nilai seluruh siswa yang disimpan dalam array.
- Menghitung rata-rata nilai ujian seluruh siswa.
- Mencari nilai tertinggi dan terendah dari array yang telah dibuat.

## Screenshoot Hasil Latihan

```

1 package modul_3.latihan;
2 import java.util.Arrays;
3
4 public class latihan1 { new *
5     public static void main(String[] args) { new *
6         // 1. Membuat array untuk menyimpan nilai ujian 10 siswa
7         int[] nilaiUjian = {85, 92, 78, 65, 88, 95, 74, 81, 98, 70};
8
9         // Inisialisasi variabel yang dibutuhkan
10        double totalNilai = 0;
11        int nilaiTertinggi = nilaiUjian[0]; // Asumsikan nilai pertama adalah yang tertinggi
12        int nilaiTerendah = nilaiUjian[0]; // Asumsikan nilai pertama adalah yang terendah
13
14        // Loop untuk menghitung total sekaligus mencari nilai tertinggi dan terendah
15        for (int nilai : nilaiUjian) {
16            // Menambahkan setiap nilai ke total
17            totalNilai += nilai;
18
19            // Memeriksa apakah nilai saat ini lebih tinggi dari nilai tertinggi yang tersimpan
20            if (nilai > nilaiTertinggi) {
21                nilaiTertinggi = nilai;
22            }
23
24            // Memeriksa apakah nilai saat ini lebih rendah dari nilai terendah yang tersimpan
25            if (nilai < nilaiTerendah) {
26                nilaiTerendah = nilai;
27            }
28        }
29
30        // 2. Menghitung rata-rata nilai
31        double rataRata = totalNilai / nilaiUjian.length;
32
33        // 3. Menampilkan semua hasil
34        System.out.println("Daftar Nilai Ujian: " + Arrays.toString(nilaiUjian));
35        System.out.println("=====");
36        System.out.printf("Rata-rata Nilai: %.2f\n", rataRata);
37        System.out.println("Nilai Tertinggi: " + nilaiTertinggi);
38        System.out.println("Nilai Terendah : " + nilaiTerendah);
39    }
40 }

```

```

Daftar Nilai Ujian: [85, 92, 78, 65, 88, 95, 74, 81, 98, 70]
=====
Rata-rata Nilai: 82.60
Nilai Tertinggi: 98
Nilai Terendah : 65

```

## Analisa dan Pembahasan

Array `nilaiSiswa` berhasil dideklarasikan dan diakses. Modifikasi kode untuk mencari nilai tertinggi dan terendah dilakukan dengan inisialisasi awal `nilaiTertinggi` dan `nilaiTerendah` menggunakan elemen pertama array (`nilaiSiswa[0]`). Perulangan `for-each` digunakan untuk membandingkan setiap elemen dengan nilai ekstrem yang tersimpan, memperbarui variabel jika ditemukan nilai baru.

## Praktikum 1.2 - Array Dua Dimensi

### Dasar Teori

Array dua dimensi (2D) atau matriks adalah array dari array. Array 2D digunakan untuk merepresentasikan data dalam bentuk baris dan kolom. Akses elemen memerlukan dua indeks: `namaArray[baris][kolom]` .

### Langkah Praktikum

1. Membuat class `ArrayDuaDimensi.java` .
2. Mendeklarasikan array 2D `nilai` yang merepresentasikan nilai 3 siswa untuk 4 mata pelajaran.
3. Menjalankan program dan menganalisis struktur data.

### Screenshoot Hasil Praktikum

### ArrayDuaDimensi.java

```
1 package modul_3;
2
3 import java.util.Scanner;
4
5 public class ArrayDuaDimensi { new *
6     public static void main(String[] args) { new *
7         Scanner input = new Scanner(System.in);
8
9         // Langkah 1: Deklarasi array 2D untuk nilai siswa (3 siswa, 4 mata pelajaran)
10        int[][] nilaiSiswa = new int[3][4];
11        String[] namaSiswa = {"Alice", "Bob", "Charlie"};
12        String[] mataPelajaran = {"Matematika", "Bahasa Indonesia", "IPA", "IPS"};
13
14        // Langkah 2: Input nilai
15        System.out.println("== INPUT NILAI SISWA ==");
16        for (int i = 0; i < 3; i++) {
17            System.out.println("Input nilai untuk " + namaSiswa[i] + ":");
18            for (int j = 0; j < 4; j++) {
19                System.out.print(mataPelajaran[j] + ": ");
20                nilaiSiswa[i][j] = input.nextInt();
21            }
22            System.out.println();
23        }
24
25        // Langkah 3: Menampilkan data dalam bentuk tabel
26        System.out.println("\n== TABEL NILAI SISWA ==");
27        System.out.printf("%-12s", "Nama");
28        for (String mapel : mataPelajaran) {
29            // Memotong nama mapel jika terlalu panjang agar tabel rapi
30            System.out.printf("%-8s", mapel.substring(0, Math.min(7, mapel.length())));
31        }
32        System.out.println("Rata-rata");
33
34        for (int i = 0; i < 3; i++) {
35            System.out.printf("%-12s", namaSiswa[i]);
36            int total = 0;
37            for (int j = 0; j < 4; j++) {
38                System.out.printf("%-8d", nilaiSiswa[i][j]);
39                // KESALAHAN: Baris ini seharusnya menjumlahkan (total += ...), bukan menimpa nilai total
40                total = nilaiSiswa[i][j];
41            }
42            double rataRata = total / 4.0;
43            System.out.printf("%.2f\n", rataRata);
44        }
45    }
46 }
```

```
C:\amazon-corretto-21.0.8.9.1-windows-x64-jdk\jdk21.0.8
== INPUT NILAI SISWA ==
Input nilai untuk Alice:
Matematika: 89
Bahasa Indonesia: 92
IPA: 86
IPS: 90

Input nilai untuk Bob:
Matematika: 87
Bahasa Indonesia: 93
IPA: 89
IPS: 91

Input nilai untuk Charlie:
Matematika: 90
Bahasa Indonesia: 87
IPA: 94
IPS: 89

== TABEL NILAI SISWA ==
Nama      Matemat Bahasa  IPA     IPS      Rata-rata
Alice     89       92       86     90      22.50
Bob       87       93       89     91      22.75
Charlie   90       87       94     89      22.25
```

## Penjelasan Latihan

Pada tugas ini, saya diminta untuk membuat array dua dimensi untuk menyimpan nilai ujian siswa pada beberapa mata pelajaran.

## Screenshot Hasil Latihan

```
1 package modul_3.latihan;
2
3 import java.util.Scanner;
4 import java.util.Arrays;
5
6 public class latihan2 { new +
7     public static void main(String[] args) { new +
8         Scanner input = new Scanner(System.in);
9
10        // Langkah 1: Deklarasi array
11        String[] namaSiswa = {"Alice", "Bob", "Charlie"};
12        String[] mataPelajaran = {"Matematika", "Bahasa Ind", "IPA", "IPS"};
13        int[][] nilaiSiswa = new int[namaSiswa.length][mataPelajaran.length];
14
15        // Langkah 2: Input nilai
16        System.out.println("==> INPUT NILAI SISWA ==>");
17        for (int i = 0; i < namaSiswa.length; i++) {
18            System.out.println("Input nilai untuk " + namaSiswa[i] + ":");
19            for (int j = 0; j < mataPelajaran.length; j++) {
20                System.out.print(mataPelajaran[j] + ": ");
21                nilaiSiswa[i][j] = input.nextInt();
22            }
23            System.out.println();
24        }
25
26        // Langkah 3: Menampilkan data, menghitung rata-rata per siswa, dan mencari nilai tertinggi per siswa
27        System.out.println("\n==> TABEL NILAI SISWA ==>");
28
29        // Header Tabel
30        System.out.printf("%-12s", "Nama");
31        for (String mapel : mataPelajaran) {
32            System.out.printf("%-12s", mapel);
33        }
34        System.out.println("Rata-rata");
35
36        // Array untuk menyimpan info nilai tertinggi setiap siswa
37        String[] infoNilaiTertinggi = new String[namaSiswa.length];
38
39        // Isi Tabel
40        for (int i = 0; i < namaSiswa.length; i++) {
41            System.out.printf("%-12s", namaSiswa[i]);
42
43            int total = 0;
44            int nilaiTertinggiSiswa = nilaiSiswa[i][0]; // Asumsikan nilai pertama tertinggi
45            String mapelTertinggi = mataPelajaran[0]; // Asumsikan mapel pertama tertinggi
46
47            for (int j = 0; j < mataPelajaran.length; j++) {
48                System.out.printf("%-12d", nilaiSiswa[i][j]);
```

```
49
50     // [PERBAIKAN] Menggunakan += untuk menjumlahkan nilai
51     total += nilaiSiswa[i][j];
52
53     // [FITUR 1] Mencari nilai dan mapel tertinggi untuk siswa ini
54     if (nilaiSiswa[i][j] > nilaiTertinggiSiswa) {
55         nilaiTertinggiSiswa = nilaiSiswa[i][j];
56         mapelTertinggi = mataPelajaran[j];
57     }
58 }
59
60 double rataRata = (double) total / mataPelajaran.length;
61 System.out.printf("%.2f\n", rataRata);
62
63 // Menyimpan hasil fitur 1 ke dalam array
64 infoNilaiTertinggi[i] = String.format("Nilai tertinggi %s: %d (pada mata pelajaran %s)", namaSiswa[i], nilaiTertinggiSiswa, mapelTertinggi);
65 }
66
67 // Menampilkan hasil Fitur 1
68 System.out.println("\n--- NILAI TERTINGGI PER SISWA ---");
69 for (String info : infoNilaiTertinggi) {
70     System.out.println(info);
71 }
72
73 // [FITUR 2] Menghitung dan menampilkan rata-rata nilai untuk setiap mata pelajaran
74 System.out.println("\n--- RATA-RATA NILAI PER MATA PELAJARAN ---");
75 for (int j = 0; j < mataPelajaran.length; j++) { // Loop luar iterasi per MATA PELAJARAN (kolom)
76     double totalPerMapel = 0;
77     for (int i = 0; i < namaSiswa.length; i++) { // Loop dalam iterasi per SISWA (baris)
78         totalPerMapel += nilaiSiswa[i][j];
79     }
80     double rataRataMapel = totalPerMapel / namaSiswa.length;
81     System.out.printf("%-12s: %.2f\n", mataPelajaran[j], rataRataMapel);
82 }
83 }
84 }
```

```
==> INPUT NILAI SISWA ==>
Input nilai untuk Alice:
Matematika: 89
Bahasa Ind: 87
IPA: 98
IPS: 82

Input nilai untuk Bob:
Matematika: 82
Bahasa Ind: 85
IPA: 91
IPS: 87

Input nilai untuk Charlie:
Matematika: 93
Bahasa Ind: 82
IPA: 85
IPS: 91

==> TABEL NILAI SISWA ==


| Nama    | Matematika | Bahasa Ind | IPA | IPS | Rata-rata |
|---------|------------|------------|-----|-----|-----------|
| Alice   | 89         | 87         | 98  | 82  | 89.00     |
| Bob     | 82         | 85         | 91  | 87  | 86.25     |
| Charlie | 93         | 82         | 85  | 91  | 87.75     |



--- NILAI TERTINGGI PER SISWA ---
Nilai tertinggi Alice: 98 (pada mata pelajaran IPA)
Nilai tertinggi Bob: 91 (pada mata pelajaran IPA)
Nilai tertinggi Charlie: 93 (pada mata pelajaran Matematika)

--- RATA-RATA NILAI PER MATA PELAJARAN ---
Matematika : 88.00
Bahasa Ind : 84.67
IPA : 91.33
IPS : 86.67
```

## Analisa dan Pembahasan

Array 2D berhasil diimplementasikan, di mana indeks pertama merepresentasikan siswa (baris) dan indeks kedua merepresentasikan mata pelajaran (kolom).

## Praktikum 2.1 - Method Dasar

### Dasar Teori

Method (atau Function) adalah blok kode yang menjalankan tugas tertentu. Praktikum ini mengeksplorasi pembuatan method dengan berbagai jenis: `void` (tidak mengembalikan nilai) dan `return method` (mengembalikan nilai).

## Langkah Praktikum

1. Membuat class `MethodDasar.java` .
2. Membuat method `tampilkanHeader()` dan `tampilkanHasil()` .
3. Membuat method matematika ( `tambah` , `kurang` , `kali` , `bagi` ) dengan return value.
4. Membuat method `validasiAngka()` untuk memastikan input valid.
5. Memanggil semua method di `main` .

## Screenshot Hasil Praktikum

```
1 package modul_3;
2
3 public class MethodDasar { new *
4
5     // ... (method tampilanHeader, tampilanHasil, tambah, kurang, kali, bagi, validasiAngka tetap sama) ...
6
7     public static void tampilanHeader() { 1 usage new *
8         System.out.println("=====");
9         System.out.println("===== PROGRAM KALKULATOR SEDERHANA =====");
10        System.out.println("=====");
11    }
12
13    public static void tampilanHasil(String operasi, double a, double b, double hasil) { 4 usages new *
14        System.out.printf("%.2f %s %.2f\n", a, operasi, b, hasil);
15    }
16
17    public static double tambah(double a, double b) { 1 usage new *
18        return a + b;
19    }
20
21    public static double kurang(double a, double b) { 1 usage new *
22        return a - b;
23    }
24
25    public static double kali(double a, double b) { 1 usage new *
26        return a * b;
27    }
28
29    public static double bagi(double a, double b) { 1 usage new *
30        if (b != 0) {
31            return a / b;
32        } else {
33            System.out.println("Error: Pembagian dengan nol!");
34            return 0;
35        }
36    }
37
38    public static boolean validasiAngka(double angka) { 2 usages new *
39        return !Double.isNaN(angka) && Double.isFinite(angka);
40    }
41
```

```
42 public static void main(String[] args) { new *
43     tampilanHeader();
44
45     // Di sini nilai x dan y diubah
46     double x = 25.0;
47     double y = 0.0;
48
49     if (validasiAngka(x) && validasiAngka(y)) {
50         double hasilTambah = tambah(x, y);
51         double hasilKurang = kurang(x, y);
52         double hasilKali = kali(x, y);
53         double hasilBagi = bagi(x, y);
54
55         tampilanHasil(operasi: "+", x, y, hasilTambah);
56         tampilanHasil(operasi: "-", x, y, hasilKurang);
57         tampilanHasil(operasi: "*", x, y, hasilKali);
58         tampilanHasil(operasi: "/", x, y, hasilBagi);
59     }
60 }
61 }
```

```
=====
===== PROGRAM KALKULATOR SEDERHANA =====
=====
Error: Pembagian dengan nol!
25.00 + 0.00 = 25.00
25.00 - 0.00 = 25.00
25.00 * 0.00 = 0.00
25.00 / 0.00 = 0.00
```

## Penjelasan Latihan

Tugas ini mencakup operasi matematika dasar, validasi pembagian dengan nol, dan penggunaan method dengan parameter dan return type.

## Screenshoot Hasil Latihan

```
1 package modul_3.latihan;
2
3 public class latihan3 { new *
4
5     // ... (method-method yang sudah ada) ...
6     public static void tampilkanHeader() { 1 usage new *
7         System.out.println("=====");
8         System.out.println("===== PROGRAM KALKULATOR LANJUTAN =====");
9         System.out.println("=====");
10    }
11
12    public static void tampilkanHasil(String operasi, double a, double b, double hasil) { 5 usages new *
13        System.out.printf("%.2f %s %.2f = %.2f\n", a, operasi, b, hasil);
14    }
15
16    public static double tambah(double a, double b) { 1 usage new *
17        return a + b;
18    }
19
20    public static double kurang(double a, double b) { 1 usage new *
21        return a - b;
22    }
23
24    public static double kali(double a, double b) { 1 usage new *
25        return a * b;
26    }
27
28    // Method bagi disederhanakan, validasi dipindah ke method sendiri
29    public static double bagi(double a, double b) { 1 usage new *
30        return a / b;
31    }
32
33    public static boolean validasiAngka(double angka) { 2 usages new *
34        return !Double.isNaN(angka) && Double.isFinite(angka);
35    }
36
37    // =====
38    // PENAMBAHAN DAN PERUBAHAN DI BAWAHINI
39    // =====
40
41    /**
42     * [BARU] Method untuk menghitung pangkat.
43     * Menggunakan Math.pow() dari library Java.
44     */
45    public static double pangkat(double basis, double eksponen) { 1 usage new *
46        return Math.pow(basis, eksponen);
47    }
48
```

```
49     /**
50      * [BARU] Method untuk menghitung akar kuadrat.
51      * Menggunakan Math.sqrt() dan menambahkan validasi untuk angka negatif.
52      */
53     public static double akarKuadrat(double angka) { 1 usage new *
54         if (angka < 0) {
55             System.out.println("Error: Akar kuadrat dari angka negatif tidak valid!");
56             return 0;
57         }
58         return Math.sqrt(angka);
59     }
60
61     /**
62      * [BARU] Method khusus untuk validasi pembagian dengan nol.
63      * Mengembalikan boolean (true jika valid, false jika tidak).
64      */
65     public static boolean validasiPembagian(double pembagi) { 1 usage new *
66         if (pembagi == 0) {
67             System.out.println("Error: Pembagian dengan nol tidak diizinkan!");
68             return false;
69         }
70         return true;
71     }
72
73
74 ▶ public static void main(String[] args) { new *
75     tampilkanHeader();
76
77     // Menggunakan nilai baru untuk menunjukkan fitur pangkat dan akar
78     double x = 9.0;
79     double y = 2.0;
80
81     System.out.println("\nOperasi untuk x = " + x + " dan y = " + y);
82
83     if (validasiAngka(x) && validasiAngka(y)) {
84         // Operasi dasar
85         tampilkanHasil( operasi: "+", x, y, tambah(x, y));
86         tampilkanHasil( operasi: "-", x, y, kurang(x, y));
87         tampilkanHasil( operasi: "*", x, y, kali(x, y));
88
89         // [REFACTOR] Panggil validasi sebelum melakukan pembagian
90         if (validasiPembagian(y)) {
91             double hasilBagi = bagi(x, y);
92             tampilkanHasil( operasi: "/", x, y, hasilBagi);
93         }
94     }
95 }
```

```

94
95         // [BARU] Panggil method pangkat
96         double hasilPangkat = pangkat(x, y);
97         tampilkanHasil( operasi: "^", x, y, hasilPangkat);
98
99         // [BARU] Panggil method akar kuadrat
100        double hasilAkar = akarKuadrat(x);
101        // Tampilkan hasil akar secara terpisah karena formatnya berbeda
102        System.out.printf("Akar kuadrat dari %.2f = %.2f\n", x, hasilAkar)
103    }
104}
105}

=====
===== PROGRAM KALKULATOR LANJUTAN =====
=====

Operasi untuk x = 9.0 dan y = 2.0
9.00 + 2.00 = 11.00
9.00 - 2.00 = 7.00
9.00 * 2.00 = 18.00
9.00 / 2.00 = 4.50
9.00 ^ 2.00 = 81.00
Akar kuadrat dari 9.00 = 3.00

```

## Analisa dan Pembahasan

Implementasi berjalan baik. Pemisahan logika ke dalam method membuat kode lebih bersih dan mudah dipahami, menunjukkan prinsip modularitas.

## Praktikum 2.2 - Method Dengan Array

### Dasar Teori

Method dapat menerima dan mengembalikan array sebagai parameter.

### Langkah Praktikum

1. Membuat class `ArrayMethod.java` .
2. Membuat method `inputArray` , `tampilkanArray` , `cariMaksimum` , `cariMinimum` , `hitungRataRata` , dan `urutkanArray` .
3. Menggabungkan semua method di `main` untuk alur analisis array lengkap.

### Screenshoot Hasil Praktikum

```
1 package modul_3;
2
3 import java.util.Scanner;
4
5 public class ArrayMethod {
6
7     /**
8      * Method untuk meminta input dari pengguna dan mengembalikan sebuah array integer.
9      */
10 @ public static int[] inputArray(int ukuran) { 1 usage new+
11     Scanner input = new Scanner(System.in);
12     int[] array = new int[ukuran];
13
14     System.out.println("Masukkan " + ukuran + " angka:");
15     for (int i = 0; i < ukuran; i++) {
16         System.out.print("Angka ke-" + (i + 1) + ": ");
17         array[i] = input.nextInt();
18     }
19     return array;
20 }
21
22 /**
23  * Method untuk menampilkan isi dari sebuah array integer.
24  */
25 @ public static void tampilkanArray(int[] array) { 2 usages new+
26     System.out.print("[");
27     // Loop untuk mencetak semua elemen kecuali yang terakhir, dengan koma
28     for (int i = 0; i < array.length - 1; i++) {
29         System.out.print(array[i] + ", ");
30     }
31     // Cetak elemen terakhir tanpa koma
32     if (array.length > 0) {
33         System.out.print(array[array.length - 1]);
34     }
35     System.out.println("]");
36 }
37
38 /**
39  * Method untuk mencari nilai maksimum dari sebuah array.
40  */
41 @ public static int cariMaksimum(int[] array) { 1 usage new+
42     int maks = array[0]; // Asumsikan elemen pertama adalah yang terbesar
43     for (int i = 1; i < array.length; i++) {
44         if (array[i] > maks) {
45             maks = array[i];
46         }
47     }

```

```
48     |     return maks;
49     |
50
51     /**
52      * Method untuk mencari nilai minimum dari sebuah array.
53      */
54     @public static int cariMinimum(int[] array) { 1 usage new*
55         int min = array[0]; // Asumsikan elemen pertama adalah yang terkecil
56         for (int i = 1; i < array.length; i++) {
57             if (array[i] < min) {
58                 min = array[i];
59             }
60         }
61         return min;
62     }
63
64     /**
65      * Method untuk menghitung rata-rata dari sebuah array.
66      */
67     @public static double hitungRataRata(int[] array) { 1 usage new*
68         int total = 0;
69         for (int nilai : array) {
70             total += nilai;
71         }
72         // Cast ke double agar hasil pembagian tidak berupa integer
73         return (double) total / array.length;
74     }
75
76     /**
77      * Method untuk mengurutkan array menggunakan algoritma Bubble Sort.
78      * Method ini membuat salinan array agar array asli tidak berubah.
79      */
80     @public static int[] urutkanArray(int[] array) { 1 usage new*
81         int[] arrayBaru = array.clone(); // Membuat salinan array
82         int n = arrayBaru.length;
83         // Algoritma Bubble Sort
84         for (int i = 0; i < n - 1; i++) {
85             for (int j = 0; j < n - i - 1; j++) {
86                 if (arrayBaru[j] > arrayBaru[j + 1]) {
87                     // Tukar elemen
88                     int temp = arrayBaru[j];
89                     arrayBaru[j] = arrayBaru[j + 1];
90                     arrayBaru[j + 1] = temp;
91                 }
92             }
93         }
94     }
```

```
93     }
94     return arrayBaru;
95 }
96
97 public static void main(String[] args) { new *
98     System.out.println("==> PROGRAM ANALISIS ARRAY ==");
99
100    // Meminta pengguna memasukkan 5 angka
101    int[] data = inputArray( ukuran: 5);
102
103    System.out.println("\n==> HASIL ANALISIS ==");
104    System.out.print("Data Asli      : ");
105    tampilkanArray(data);
106
107    System.out.println("Nilai Maksimum : " + cariMaksimum(data));
108    System.out.println("Nilai Minimum  : " + cariMinimum(data));
109    System.out.printf("Rata-rata      : %.2f\n", hitungRataRata(data));
110
111    // Mengurutkan array
112    int[] dataUrut = urutkanArray(data);
113    System.out.print("Array setelah diurutkan: ");
114    tampilkanArray(dataUrut);
115 }
116 }
```

```
C:\amazon-corretto-21.0.8.9.1-windows-x64-j
==> PROGRAM ANALISIS ARRAY ==
Masukkan 5 angka:
Angka ke-1: 7
Angka ke-2: 4
Angka ke-3: 6
Angka ke-4: 2
Angka ke-5: 9

==> HASIL ANALISIS ==
Data Asli      : [7, 4, 6, 2, 9]
Nilai Maksimum : 9
Nilai Minimum  : 2
Rata-rata      : 5.60
Array setelah diurutkan: [2, 4, 6, 7, 9]
```

## Penjelasan Latihan

Program menghitung rata-rata nilai dalam array, mencari nilai maksimum dan minimum, serta mengurutkan array.

## Screenshoot Hasil Latihan

```
1 package modul_3.latihan;
2
3 import java.util.Scanner;
4
5 public class latihan4 { new *
6
7 // ... (method-method yang sudah ada: inputArray, tampilkanArray, cariMaksimum, cariMinimum, hitungRataRata, urutkanArray)
8 @ public static int[] inputArray(int ukuran) { 1 usage new *
9     Scanner input = new Scanner(System.in);
10    int[] array = new int[ukuran];
11    System.out.println("Masukkan " + ukuran + " angka:");
12    for (int i = 0; i < ukuran; i++) {
13        System.out.print("Angka ke-" + (i + 1) + ": ");
14        array[i] = input.nextInt();
15    }
16    return array;
17 }
18 @ public static void tampilkanArray(int[] array) { 2 usages new *
19     System.out.print("[");
20    for (int i = 0; i < array.length - 1; i++) {
21        System.out.print(array[i] + ", ");
22    }
23    if (array.length > 0) {
24        System.out.print(array[array.length - 1]);
25    }
26    System.out.println("]");
27 }
28 @ public static int cariMaksimum(int[] array) { 1 usage new *
29     int maks = array[0];
30    for (int i = 1; i < array.length; i++) {
31        if (array[i] > maks) {
32            maks = array[i];
33        }
34    }
35    return maks;
36 }
37 @ public static int cariMinimum(int[] array) { 1 usage new *
38     int min = array[0];
39    for (int i = 1; i < array.length; i++) {
40        if (array[i] < min) {
41            min = array[i];
42        }
43    }
44    return min;
45 }
46 @ public static double hitungRataRata(int[] array) { 1 usage new *
47     int total = 0;
48     for (int nilai : array) {
```

```

49         total += nilai;
50     }
51     return (double) total / array.length;
52 }
53 @
54     public static int[] urutkanArray(int[] array) { 1 usage new*
55         int[] arrayBaru = array.clone();
56         int n = arrayBaru.length;
57         for (int i = 0; i < n - 1; i++) {
58             for (int j = 0; j < n - i - 1; j++) {
59                 if (arrayBaru[j] > arrayBaru[j + 1]) {
60                     int temp = arrayBaru[j];
61                     arrayBaru[j] = arrayBaru[j + 1];
62                     arrayBaru[j + 1] = temp;
63                 }
64             }
65         }
66         return arrayBaru;
67     }
68     // =====
69     // PENAMBAHAN METHOD BARU DI BAWAH INI
70     // =====
71
72     /**
73      * [BARU] Method untuk mencari nilai tertentu dalam array (Linear Search).
74      * Mengembalikan indeks dari elemen jika ditemukan, atau -1 jika tidak.
75      */
76     @
77     public static int cariNilai(int[] array, int nilaiYangDicari) { 1 usage new*
78         for (int i = 0; i < array.length; i++) {
79             if (array[i] == nilaiYangDicari) {
80                 return i; // Mengembalikan indeks jika nilai ditemukan
81             }
82         }
83         return -1; // Mengembalikan -1 jika nilai tidak ditemukan setelah loop selesai
84     }
85     /**
86      * [BARU] Method untuk menghitung median dari array yang SUDAH diurutkan.
87      */
88     @
89     public static double hitungMedian(int[] arrayUrut) { 1 usage new*
90         int n = arrayUrut.length;
91         if (n % 2 != 0) {
92             // Jika jumlah elemen ganjil, median adalah nilai tengah
93             return arrayUrut[n / 2];
94         } else {

```

```
94     // Jika jumlah elemen genap, median adalah rata-rata dari dua nilai tengah
95     int tengah1 = arrayUrut[(n / 2) - 1];
96     int tengah2 = arrayUrut[n / 2];
97     return (double) (tengah1 + tengah2) / 2.0;
98 }
99 }
100
101 public static void main(String[] args) { new *
102     Scanner input = new Scanner(System.in);
103     System.out.println("== PROGRAM ANALISIS ARRAY ==");
104
105     int[] data = inputArray( ukuran: 5);
106
107     System.out.println("\n== HASIL ANALISIS ==");
108     System.out.print("Data Asli      : ");
109     tampilkanArray(data);
110
111     System.out.println("Nilai Maksimum : " + cariMaksimum(data));
112     System.out.println("Nilai Minimum   : " + cariMinimum(data));
113     System.out.printf("Rata-rata       : %.2f\n", hitungRataRata(data));
114
115     // Mengurutkan array
116     int[] dataUrut = urutkanArray(data);
117     System.out.print("Array setelah diurutkan: ");
118     tampilkanArray(dataUrut);
119
120     // [BARU] Menghitung dan menampilkan median dari data yang sudah diurutkan
121     System.out.printf("Median        : %.2f\n", hitungMedian(dataUrut));
122
123     // [BARU] Proses pencarian nilai
124     System.out.println("\n== PENCARIAN NILAI ==");
125     System.out.print("Masukkan nilai yang ingin Anda cari: ");
126     int nilaiCari = input.nextInt();
127
128     int indeksHasil = cariNilai(data, nilaiCari);
129
130     if (indeksHasil != -1) {
131         System.out.println("Nilai " + nilaiCari + " ditemukan pada indeks ke-" + indeksHasil);
132     } else {
133         System.out.println("Nilai " + nilaiCari + " tidak ditemukan dalam array.");
134     }
135 }
136 }
```

```
== PROGRAM ANALISIS ARRAY ==
Masukkan 5 angka:
Angka ke-1: 98
Angka ke-2: 91
Angka ke-3: 82
Angka ke-4: 95
Angka ke-5: 86

== HASIL ANALISIS ==
Data Asli      : [98, 91, 82, 95, 86]
Nilai Maksimum : 98
Nilai Minimum  : 82
Rata-rata      : 90.40
Array setelah diurutkan: [82, 86, 91, 95, 98]
Median         : 91.00

== PENCARIAN NILAI ==
Masukkan nilai yang ingin Anda cari: 91
Nilai 91 ditemukan pada indeks ke-1
```

## Analisa dan Pembahasan

Pemisahan logika pemrosesan array ke dalam method yang berbeda terbukti efektif. Method `urutkanArray` menggunakan `array.clone()` agar array asli tidak berubah.

## Praktikum 3.1 - Manipulasi String Dasar

### Dasar Teori

String di Java bersifat *immutable* (tidak dapat diubah setelah dibuat). Java menyediakan berbagai method built-in seperti `length()` , `charAt()` , `toUpperCase()` , `toLowerCase()` , dan `contains()` .

### Langkah Praktikum

1. Membuat class `StringDasar.java` .
2. Membuat method `analysisString` , `hitungVokal` , `reverseString` , dan `isPalindrome` .
3. Menggunakan `Scanner` untuk input dinamis.

### Screenshoot Hasil Praktikum

```
1 package modul_3;
2
3 import java.util.Scanner;
4
5 public class StringDasar {
6
7     /**
8      * Method untuk menampilkan analisis dasar dari sebuah string.
9      * Method ini juga memanggil method hitungJumlahKata dan hitungVokal.
10     */
11    @public static void analisisString(String teks) { 1 usage new *
12        System.out.println("\n--- ANALISIS DASAR ---");
13        System.out.println("Teks asli: \"" + teks + "\"");
14        System.out.println("Panjang string: " + teks.length());
15        System.out.println("Karakter pertama: " + teks.charAt(0));
16        System.out.println("Karakter terakhir: " + teks.charAt(teks.length() - 1));
17        System.out.println("UPPERCASE: " + teks.toUpperCase());
18        System.out.println("lowercase: " + teks.toLowerCase());
19
20        // Menghitung jumlah kata
21        int jumlahKata = hitungJumlahKata(teks);
22        System.out.println("Jumlah kata: " + jumlahKata);
23
24        // Menghitung jumlah vokal
25        int jumlahVokal = hitungVokal(teks);
26        System.out.println("Jumlah vokal: " + jumlahVokal);
27    }
28
29    /**
30     * Method untuk menghitung jumlah kata dalam sebuah string.
31     */
32    @public static int hitungJumlahKata(String teks) { 1 usage new *
33        // trim() untuk hapus spasi di awal/akhir, split() untuk memecah berdasarkan spasi
34        String[] kata = teks.trim().split(regex: "\\\s+");
35        return kata.length;
36    }
37
38    /**
39     * Method untuk menghitung jumlah huruf vokal dalam sebuah string.
40     */
41    @public static int hitungVokal(String teks) { 1 usage new *
42        String vokal = "aeiouAEIOU";
43        int counter = 0;
44        for (int i = 0; i < teks.length(); i++) {
45            // Cek apakah karakter saat ini ada di dalam string vokal
46            if (vokal.contains(String.valueOf(teks.charAt(i)))) {
47                counter++;
48            }
49        }
50    }
51}
```

```
49     }
50     return counter;
51 }
52
53 /**
54 * Method untuk membalik urutan karakter dalam sebuah string.
55 */
56 @
57 public static String reverseString(String teks) { 1 usage new*
58     StringBuilder reversed = new StringBuilder();
59     for (int i = teks.length() - 1; i >= 0; i--) {
60         reversed.append(teks.charAt(i));
61     }
62     return reversed.toString();
63 }
64 /**
65 * Method untuk mengecek apakah sebuah string adalah palindrom.
66 * Palindrom: dibaca sama dari depan maupun belakang (setelah mengabaikan spasi dan simbol).
67 */
68 @
69 public static boolean isPalindrome(String teks) { 1 usage new*
70     // 1. Bersihkan string dari spasi, simbol, dan ubah ke huruf kecil
71     String clean = teks.replaceAll( regex: "[^a-zA-Z0-9]", replacement: "" ).toLowerCase();
72     // 2. Buat versi terbalik dari string yang sudah bersih
73     String reversed = new StringBuilder(clean).reverse().toString();
74     // 3. Bandingkan keduanya
75     return clean.equals(reversed);
76 }
77 ▶ public static void main(String[] args) { new*
78     Scanner input = new Scanner(System.in);
79     System.out.println("== PROGRAM MANIPULASI STRING ==");
80     System.out.print("Masukkan sebuah kalimat: ");
81     String kalimat = input.nextLine();
82
83     // Analisis dasar
84     analisisString(kalimat);
85
86     // String terbalik
87     System.out.println("\nString terbalik: \" " + reverseString(kalimat) + " \"");
88
89     // Cek Palindrom
90     if (isPalindrome(kalimat)) {
91         System.out.println("String ini adalah palindrom.");
92     } else {
93         System.out.println("String ini bukan palindrom.");
```

```
94     }
95
96     // Pencarian substring
97     System.out.println("\nMasukkan kata yang ingin dicari:");
98     System.out.print("> ");
99     String cari = input.nextLine();
100
101    // Cek apakah kata 'cari' ada di dalam 'kalimat' (abaikan huruf besar/kecil)
102    if (kalimat.toLowerCase().contains(cari.toLowerCase())) {
103        // Cari posisi (indeks) pertama kali kata itu muncul
104        int posisi = kalimat.toLowerCase().indexOf(cari.toLowerCase());
105        System.out.println("Kata \"" + cari + "\" ditemukan dalam kalimat.");
106        System.out.println("Posisi pertama: " + posisi);
107    } else {
108        System.out.println("Kata \"" + cari + "\" tidak ditemukan dalam kalimat.");
109    }
110 }
111 }
```

```
== PROGRAM MANIPULASI STRING ==
Masukkan sebuah kalimat: Alif pergi Kuliah

--- ANALISIS DASAR ---
Teks asli: "Alif pergi Kuliah"
Panjang string: 17
Karakter pertama: A
Karakter terakhir: h
UPPERCASE: ALIF PERGI KULIAH
lowercase: alif pergi kuliah
Jumlah kata: 3
Jumlah vokal: 7

String terbalik: "hailuK igrep filA"
String ini bukan palindrom.
```

## Penjelasan Latihan

Tugas ini mencakup menghitung jumlah vokal, mengubah case string, dan menampilkan jumlah kata.

## Screenshot Hasil Latihan

```
1 package modul_3.latihan;
2
3 import java.util.Scanner;
4
5 public class latihan5 { new *
6
7 @ public static void analisisString(String teks) { 1 usage new *
8     System.out.println("\n--- ANALISIS DASAR ---");
9     System.out.println("Teks asli: '" + teks + "'");
10    System.out.println("Panjang string: " + teks.length());
11    System.out.println("Karakter pertama: " + teks.charAt(0));
12    System.out.println("Karakter terakhir: " + teks.charAt(teks.length() - 1));
13    System.out.println("UPPERCASE: " + teks.toUpperCase());
14    System.out.println("lowercase: " + teks.toLowerCase());
15
16    // Menghitung jumlah kata
17    System.out.println("Jumlah kata: " + hitungJumlahKata(teks));
18
19    // Menghitung jumlah vokal
20    System.out.println("Jumlah vokal: " + hitungVokal(teks));
21
22    // [BARU] Menghitung jumlah konsonan
23    System.out.println("Jumlah konsonan: " + hitungKonsonan(teks));
24 }
25
26 @ public static int hitungJumlahKata(String teks) { 1 usage new *
27     String[] kata = teks.trim().split( regex: "\\\s+");
28     return kata.length;
29 }
30
31 @ public static int hitungVokal(String teks) { 2 usages new *
32     String vokal = "aeiouAEIOU";
33     int counter = 0;
34     for (int i = 0; i < teks.length(); i++) {
35         if (vokal.contains(String.valueOf(teks.charAt(i)))) {
36             counter++;
37         }
38     }
39     return counter;
40 }
41
42 // =====
43 // PENAMBAHAN METHOD BARU DI BAWAH INI
44 // =====
45
46 /**
47 * [BARU] Method untuk menghitung jumlah huruf konsonan.
48 */
49 @ public static int hitungKonsonan(String teks) { 1 usage new *
```

```
50     int jumlahHuruf = 0;
51     // Pertama, hitung semua karakter yang merupakan huruf
52     for (char ch : teks.toCharArray()) {
53         if (Character.isLetter(ch)) {
54             jumlahHuruf++;
55         }
56     }
57     // Konsonan = Total Huruf - Total Vokal
58     return jumlahHuruf - hitungVokal(teks);
59 }
60
61 /**
62 * [BARU] Method untuk mengubah string menjadi format Judul (Title Case).
63 * Setiap kata diawali huruf besar, sisanya huruf kecil.
64 */
65 @*
66 public static String toTitleCase(String teks) { 1 usage new*
67     if (teks == null || teks.isEmpty()) {
68         return "";
69     }
70     // Pecah kalimat menjadi kata-kata
71     String[] kataArray = teks.split( regex: "\\s+" );
72     StringBuilder hasil = new StringBuilder();
73
74     for (String kata : kataArray) {
75         if (kata.length() > 0) {
76             // Ambil huruf pertama, ubah jadi kapital
77             char hurufPertama = Character.toUpperCase(kata.charAt(0));
78             // Ambil sisa kata, ubah jadi huruf kecil
79             String sisaKata = kata.substring( beginIndex: 1 ).toLowerCase();
80             // Gabungkan kembali dan tambahkan spasi
81             hasil.append(hurufPertama).append(sisaKata).append(" ");
82         }
83     }
84     // Hilangkan spasi ekstra di akhir
85     return hasil.toString().trim();
86 }
87
88 /**
89 * ... (method reverseString dan isPalindrome tetap sama) ...
90 */
91 public static String reverseString(String teks) { 1 usage new*
92     StringBuilder reversed = new StringBuilder();
93     for (int i = teks.length() - 1; i >= 0; i--) {
94         reversed.append(teks.charAt(i));
95     }
96     return reversed.toString();
97 }
```

```
96 @~    public static boolean isPalindrome(String teks) { 1 usage new +
97      String clean = teks.replaceAll( regex: "[^a-zA-Z0-9]", replacement: "" ).toLowerCase();
98      String reversed = new StringBuilder(clean).reverse().toString();
99      return clean.equals(reversed);
100 }
101
102 ▶~  public static void main(String[] args) { new +
103     Scanner input = new Scanner(System.in);
104     System.out.println("== PROGRAM MANIPULASI STRING ==");
105     System.out.print("Masukkan sebuah kalimat: ");
106     String kalimat = input.nextLine();
107
108     // Analisis dasar (sekarang juga menampilkan jumlah konsonan)
109     analisisString(kalimat);
110
111     // [BARU] Menampilkan hasil konversi ke Title Case
112     System.out.println("\nFormat Judul (Title Case): \" + toTitleCase(kalimat) + "\"");
113
114     // String terbalik
115     System.out.println("String terbalik: \" + reverseString(kalimat) + "\"");
116
117     // Cek Palindrom
118     if (isPalindrome(kalimat)) {
119         System.out.println("String ini adalah palindrom.");
120     } else {
121         System.out.println("String ini bukan palindrom.");
122     }
123
124     // Pencarian substring
125     System.out.println("\nMasukkan kata yang ingin dicari:");
126     System.out.print("> ");
127     String cari = input.nextLine();
128
129     if (kalimat.toLowerCase().contains(cari.toLowerCase())) {
130         int posisi = kalimat.toLowerCase().indexOf(cari.toLowerCase());
131         System.out.println("Kata \" + cari + "\" ditemukan dalam kalimat.");
132         System.out.println("Posisi pertama: " + posisi);
133     } else {
134         System.out.println("Kata \" + cari + "\" tidak ditemukan dalam kalimat.");
135     }
136 }
137 }
```

```
== PROGRAM MANIPULASI STRING ==
Masukkan sebuah kalimat: Alif ingin pergi ke kampus

--- ANALISIS DASAR ---
Teks asli: "Alif ingin pergi ke kampus"
Panjang string: 26
Karakter pertama: A
Karakter terakhir: s
UPPERCASE: ALIF INGIN PERGI KE KAMPUS
lowercase: alif ingin pergi ke kampus
Jumlah kata: 5
Jumlah vokal: 9
Jumlah konsonan: 13

Format Judul (Title Case): "Alif Ingin Pergi Ke Kampus"
String terbalik: "supmak ek igrep nigni filA"
String ini bukan palindrom.

Masukkan kata yang ingin dicari:
> Alif
Kata "Alif" ditemukan dalam kalimat.
Posisi pertama: 0
```

## Analisa dan Pembahasan

String manipulation berjalan efisien dengan method bawaan Java. Penggunaan *regular expression* dan *StringBuilder* membantu efisiensi.

## Praktikum 3.2 - Pengolahan Data Text

### Dasar Teori

Pemrosesan data teks melibatkan *tokenizing*, *formatting*, dan *validation*.

### Langkah Praktikum

1. Membuat class `TextProcessing.java` .
2. Membuat method `prosesDataMahasiswa` untuk memecah data nama.
3. Membuat method `formatTitleCase` , `formatKataInitial` , dan `buatUsername` .

### Screenshot Hasil Praktikum

```
1 package modul_3;
2
3 import java.util.Scanner;
4
5 // Rename usages
6 public class TextProcessing { new *
7 /**
8 * Method utama untuk memproses dan menganalisis data mahasiswa.
9 * Mencari nama terpanjang, terpendek, dan mengurutkan nama.
10 */
11 @
12     public static void prosesDataMahasiswa(String data) { 1 usage new *
13         // Memisahkan data string menjadi array nama
14         String[] dataMahasiswa = data.split( regex: "\n");
15         System.out.println("\n===== ANALISIS DATA =====");
16         System.out.println("Jumlah mahasiswa: " + dataMahasiswa.length);
17
18         // Mencari nama terpanjang dan terpendek
19         String namaTerpanjang = dataMahasiswa[0];
20         String namaTerpendek = dataMahasiswa[0];
21         for (int i = 1; i < dataMahasiswa.length; i++) {
22             if (dataMahasiswa[i].length() > namaTerpanjang.length()) {
23                 namaTerpanjang = dataMahasiswa[i];
24             }
25             if (dataMahasiswa[i].length() < namaTerpendek.length()) {
26                 namaTerpendek = dataMahasiswa[i];
27             }
28         }
29         System.out.println("Nama terpanjang: " + namaTerpanjang + " (" + namaTerpanjang.length() + " karakter)");
30         System.out.println("Nama terpendek : " + namaTerpendek + " (" + namaTerpendek.length() + " karakter)");
31
32         // Mengurutkan nama mahasiswa secara alfabetis (Bubble Sort)
33         for (int i = 0; i < dataMahasiswa.length - 1; i++) {
34             for (int j = 0; j < dataMahasiswa.length - i - 1; j++) {
35                 // compareTo() digunakan untuk membandingkan string
36                 if (dataMahasiswa[j].compareTo(dataMahasiswa[j + 1]) > 0) {
37                     String temp = dataMahasiswa[j];
38                     dataMahasiswa[j] = dataMahasiswa[j + 1];
39                     dataMahasiswa[j + 1] = temp;
40                 }
41             }
42
43             System.out.println("\nUrutan nama mahasiswa (sesuai abjad):");
44             for (int i = 0; i < dataMahasiswa.length; i++) {
45                 System.out.println((i + 1) + ". " + dataMahasiswa[i]);
46             }
47         }
48     }
49 }
```

```
48
49     /**
50      * Method untuk menampilkan hasil format nama dan username.
51      */
52 @public static void formatNama(String[] nama) { 1 usage new*
53     System.out.println("\n===== FORMAT NAMA & USERNAME =====");
54     for (String namaLengkap : nama) {
55         String namaFormatted = formatTitleCase(namaLengkap);
56         String username = buatUsername(namaFormatted);
57         System.out.println("Nama Lengkap: " + namaFormatted);
58         System.out.println("Username : " + username);
59         System.out.println("---");
60     }
61 }
62
63 /**
64  * Method untuk mengubah format teks menjadi Title Case (Setiap Kata Diawali Huruf Besar).
65  */
66 @public static String formatTitleCase(String teks) { 1 usage new*
67     if (teks == null || teks.isEmpty()) {
68         return "";
69     }
70     String[] kata = teks.split( regex: "\\s+" );
71     StringBuilder hasil = new StringBuilder();
72     for (String k : kata) {
73         if (k.length() > 0) {
74             hasil.append(Character.toUpperCase(k.charAt(0)))
75                 .append(k.substring( beginIndex: 1 ).toLowerCase())
76                 .append(" ");
77         }
78     }
79     return hasil.toString().trim();
80 }
81
82 /**
83  * Method untuk membuat username dari nama lengkap.
84  * Format: namadepan + hurufpertamanamakedua (jika ada).
85  */
86 @public static String buatUsername(String nama) { 1 usage new*
87     String[] kata = nama.toLowerCase().split( regex: "\\s+" );
88     if (kata.length == 0) {
89         return "";
90     }
91     StringBuilder username = new StringBuilder(kata[0]);
92     if (kata.length > 1) {
```

```
93     |         username.append(kata[1].charAt(0));
94     |     }
95     |     return username.toString();
96   }
97
98 /**
99  * Method untuk membuat password sederhana dari nama.
100 * Format: nama dibalik tanpa huruf vokal dan spasi.
101 */
102 @
103 public static String buatPassword(String nama) { no usages new*
104     String reversedName = new StringBuilder(nama).reverse().toString();
105     // Regex (?i) membuatnya case-insensitive
106     return reversedName.replaceAll( regex: "(?i)[aeiou\\s]", replacement: "" );
107 }
108 ▶ public static void main(String[] args) { new*
109     Scanner input = new Scanner(System.in);
110     System.out.println("== PROGRAM PENGOLAHAN DATA TEKS ==");
111     System.out.println("Masukkan nama-nama mahasiswa (pisahkan dengan titik koma ','):");
112     System.out.print("> ");
113     String dataInput = input.nextLine();
114
115     // Proses data utama
116     prosesDataMahasiswa(dataInput);
117
118     // Format nama
119     String[] database = dataInput.split( regex: "," );
120     formatNama(database);
121 }
122 }
```

```
== PROGRAM PENGOLAHAN DATA TEKS ==
Masukkan nama-nama mahasiswa (pisahkan dengan titik koma ','):
> Alif;Aziz;Dayat;Hawari;Alfi

===== ANALISIS DATA =====
Jumlah mahasiswa: 5
Nama terpanjang: Hawari (6 karakter)
Nama terpendek : Alif (4 karakter)

Urutan nama mahasiswa (sesuai abjad):
1. Alfi
2. Alif
3. Aziz
4. Dayat
5. Hawari

===== FORMAT NAMA & USERNAME =====
Nama Lengkap: Alif
Username    : alif
---
Nama Lengkap: Aziz
Username    : aziz
---
Nama Lengkap: Dayat
Username    : dayat
---
Nama Lengkap: Hawari
Username    : hawari
---
Nama Lengkap: Alfi
Username    : alfi
---
```

## Penjelasan Latihan

Program memvalidasi format nama dan membuat alamat email berdasarkan nama lengkap.

## Screenshoot Hasil Latihan

```
1 package modul_3.latihan;
2
3 import java.util.Scanner;
4
5 public class latihan6 { new *
6
7 @
8     public static void prosesDataMahasiswa(String data) { 1 usage new *
9         String[] dataMahasiswa = data.split( regex: ".");
10        System.out.println("\n===== ANALISIS DATA =====");
11        System.out.println("Jumlah mahasiswa: " + dataMahasiswa.length);
12
13        // ... (kode di dalam method ini tetap sama) ...
14        String namaTerpanjang = dataMahasiswa[0];
15        String namaTerpendek = dataMahasiswa[0];
16        for (int i = 1; i < dataMahasiswa.length; i++) {
17            if (dataMahasiswa[i].length() > namaTerpanjang.length()) {
18                namaTerpanjang = dataMahasiswa[i];
19            }
20            if (dataMahasiswa[i].length() < namaTerpendek.length()) {
21                namaTerpendek = dataMahasiswa[i];
22            }
23        }
24        System.out.println("Nama terpanjang: " + namaTerpanjang + " (" + namaTerpanjang.length() + " karakter)");
25        System.out.println("Nama terpendek : " + namaTerpendek + " (" + namaTerpendek.length() + " karakter)");
26
27        for (int i = 0; i < dataMahasiswa.length - 1; i++) {
28            for (int j = 0; j < dataMahasiswa.length - i - 1; j++) {
29                if (dataMahasiswa[j].compareTo(dataMahasiswa[j + 1]) > 0) {
30                    String temp = dataMahasiswa[j];
31                    dataMahasiswa[j] = dataMahasiswa[j + 1];
32                    dataMahasiswa[j + 1] = temp;
33                }
34            }
35            System.out.println("\nUrutan nama mahasiswa (sesuai abjad):");
36            for (int i = 0; i < dataMahasiswa.length; i++) {
37                System.out.println((i + 1) + ". " + dataMahasiswa[i]);
38            }
39        }
40
41 @
42     public static void formatNama(String[] nama) { 1 usage new *
43         System.out.println("\n===== FORMAT NAMA, USERNAME & EMAIL =====");
44         for (String namaLengkap : nama) {
45             String namaFormatted = formatTitleCase(namaLengkap.trim()); // trim() untuk hapus spasi berlebih
46             String username = buatUsername(namaFormatted);
47             // [BARU] Panggil method untuk membuat email
48             String email = buatEmail(namaFormatted);
49
50             System.out.println("Nama Lengkap: " + namaFormatted);
```

```
50     System.out.println("Username : " + username);
51     System.out.println("Email : " + email); // Tampilkan email
52     System.out.println("---");
53 }
54 }
55
56 @public static String formatTitleCase(String teks) { 1 usage new *
57     // ... (kode di dalam method ini tetap sama) ...
58     if (teks == null || teks.isEmpty()) {
59         return "";
60     }
61     String[] kata = teks.split( regex: "\\\s+");
62     StringBuilder hasil = new StringBuilder();
63     for (String k : kata) {
64         if (k.length() > 0) {
65             hasil.append(Character.toUpperCase(k.charAt(0)))
66                 .append(k.substring( beginIndex: 1).toLowerCase())
67                 .append(" ");
68         }
69     }
70     return hasil.toString().trim();
71 }
72
73 @public static String buatUsername(String nama) { 2 usages new *
74     // ... (kode di dalam method ini tetap sama) ...
75     String[] kata = nama.toLowerCase().split( regex: "\\\s+");
76     if (kata.length == 0) {
77         return "";
78     }
79     StringBuilder username = new StringBuilder(kata[0]);
80     if (kata.length > 1) {
81         username.append(kata[1].charAt(0));
82     }
83     return username.toString();
84 }
85
86 @public static String buatPassword(String nama) { no usages new *
87     // ... (kode di dalam method ini tetap sama, meskipun tidak dipanggil di main) ...
88     String reversedName = new StringBuilder(nama).reverse().toString();
89     return reversedName.replaceAll( regex: "(?i)[aeiou\\s]", replacement: "");
90 }
91
92 // =====
93 // PENAMBAHAN METHOD BARU DI BAWAH INI
94 // =====
95
96 /**
```

```
97     * [BARU] Method untuk menghasilkan alamat email dari nama.  
98     * Menggunakan username yang sudah dibuat dan menambahkan domain.  
99     */  
100    @  
101    public static String buatEmail(String nama) { 1 usage new *  
102        String username = buatUsername(nama);  
103        return username + "@mahasiswa.campus.id";  
104    }  
105      
106    /**
107     * [BARU] Method untuk memvalidasi format nama.  
108     * Nama dianggap valid jika hanya mengandung huruf dan spasi.  
109     */  
110    @  
111    public static boolean isNamaValid(String nama) { 1 usage new *  
112        // Regex: ^[a-zA-Z\s]+$  
113        // Artinya: String harus terdiri dari awal (^) sampai akhir ($)  
114        // hanya oleh karakter huruf (a-z, A-Z) dan spasi (\s).  
115        return nama.trim().matches( regex: "^[a-zA-Z\\s]+$");  
116    }  
117      
118    public static void main(String[] args) { new *  
119        Scanner input = new Scanner(System.in);  
120        System.out.println("==> PROGRAM PENGOLAHAN DATA TEKS ==");  
121        System.out.println("Masukkan nama-nama mahasiswa (pisahkan dengan titik koma ',');");  
122        System.out.print("> ");  
123        String dataInput = input.nextLine();  
124          
125        // Memisahkan data input menjadi array  
126        String[] database = dataInput.split( regex: "," );  
127          
128        // [BARU] Proses validasi input sebelum diproses lebih lanjut  
129        boolean semuaNamaValid = true;  
130        for (String nama : database) {  
131            if (!isNamaValid(nama)) {  
132                System.out.println("\n[ERROR] Input tidak valid!");  
133                System.out.println("Nama '" + nama.trim() + "' mengandung karakter selain huruf dan spasi.");  
134                semuaNamaValid = false;  
135                break; // Hentikan pengecekan jika satu nama sudah tidak valid  
136            }  
137        }  
138        // [BARU] Hanya jalankan proses jika semua nama valid  
139        if (semuaNamaValid) {  
140            // Proses data utama  
141            prosesDataMahasiswa(dataInput);  
142              
143            // Format nama  
144            formatNama(database);
```

```
== PROGRAM PENGOLAHAN DATA TEKS ==
Masukkan nama-nama mahasiswa (pisahkan dengan titik koma ','):
> Muhammad Alif Arrayyan;Muhammad Aziz;Muhammad Hawari;Riski Al Fatah; Abadan Syakira;Rahmad Hidayat

===== ANALISIS DATA =====
Jumlah mahasiswa: 6
Nama terpanjang: Muhammad Alif Arrayyan (22 karakter)
Nama terpendek : Muhammad Aziz (13 karakter)

Urutan nama mahasiswa (sesuai abjad):
1. Abadan Syakira
2. Muhammad Alif Arrayyan
3. Muhammad Aziz
4. Muhammad Hawari
5. Rahmad Hidayat
6. Riski Al Fatah

===== FORMAT NAMA, USERNAME & EMAIL =====
Nama Lengkap: Muhammad Alif Arrayyan
Username      : muhammada
Email        : muhammada@mahasiswa.campus.id
---
Nama Lengkap: Muhammad Aziz
Username      : muhammada
Email        : muhammada@mahasiswa.campus.id
---
Nama Lengkap: Muhammad Hawari
Username      : muhammadh
Email        : muhammadh@mahasiswa.campus.id
---
Nama Lengkap: Riski Al Fatah
Username      : riskia
Email        : riskia@mahasiswa.campus.id
---
Nama Lengkap: Abadan Syakira
Username      : abadans
Email        : abadans@mahasiswa.campus.id
---
Nama Lengkap: Rahmad Hidayat
Username      : rahmadh
Email        : rahmadh@mahasiswa.campus.id
---
```

## Analisa dan Pembahasan

Penggunaan `split(",")` efektif untuk memecah data teks menjadi array yang dapat diproses. Pengurutan nama menggunakan `compareToIgnoreCase()` untuk hasil alfabetis yang konsisten.

## Kesimpulan

Melalui modul ini, pemahaman mendalam mengenai tiga konsep fundamental telah tercapai. Struktur data Array terbukti sangat efektif untuk mengelola kumpulan data yang sejenis. Sementara itu, penerapan Method secara signifikan meningkatkan modularitas kode, menjadikannya lebih terorganisir dan mudah dikelola. Terakhir, kapabilitas manipulasi teks yang kompleks dapat diatasi melalui penggunaan String.

Secara keseluruhan, penguasaan materi ini membekali kemampuan untuk merancang alur program yang jauh lebih terstruktur dan efisien.

## Referensi

1. [Java Documentation: Arrays](#)
2. [Java Documentation: Methods](#)
3. [Java Documentation: The String Class](#)