

# Laporan Modul 3 : Dasar Pemrograman Java Lanjutan

**Mata Kuliah:** Praktikum Pemrograman Berorientasi Objek

**Nama:** Muhammad Hawari **NIM:** 2024573010123

**Kelas:** TI 2E

## 1. Abstrak

Laporan ini membahas penerapan konsep dasar pemrograman Java lanjutan yang meliputi struktur data Array, penggunaan Method (fungsi), dan manipulasi String. Tujuan dari laporan ini adalah untuk memahami cara mengelola data jamaik menggunakan array satu dan dua dimensi, memodularisasi kode program menggunakan method agar lebih terstruktur dan *reusable*, serta melakukan pengolahan data teks.

Dalam praktikum ini, saya mempelajari cara mendeklarasikan dan memanipulasi array, membuat method statis dengan parameter dan nilai kembalian (*return value*), serta memanfaatkan *library* bawaan Java untuk memanipulasi String seperti pemisahan teks, pencarian karakter, dan konversi format. Selain itu, dilakukan pula penyelesaian tugas-tugas praktikum untuk melatih logika pemrograman dalam studi kasus nyata seperti pengolahan nilai siswa dan data teks mahasiswa.

Secara keseluruhannya, laporan ini bertujuan untuk melatih kemampuan dalam menulis kode Java yang efisien, rapi, dan sesuai dengan standar pemrograman modern.

## 2. Teori Dasar

### Array

Array adalah struktur data yang dapat menyimpan beberapa nilai dengan tipe data yang sama dalam satu variabel. Array menggunakan indeks untuk mengakses elemen-elemennya, dimulai dari indeks 0.

#### Karakteristik Array:

- Semua elemen memiliki tipe data yang sama.
- Ukuran array bersifat tetap (*fixed size*).
- Indeks dimulai dari 0.
- Elemen dapat diakses secara langsung menggunakan indeks.

### Method / Function

Method adalah blok kode yang dapat dipanggil untuk menjalankan tugas tertentu. Method memungkinkan kita untuk menghindari pengulangan kode, membuat program lebih terstruktur, dan memudahkan *debugging* serta *maintenance*.

#### Jenis Method:

- **Static Method:** Dapat dipanggil tanpa membuat object.
- **Non-static Method:** Perlu membuat object terlebih dahulu.
- **Void Method:** Tidak mengembalikan nilai.
- **Return Method:** Mengembalikan nilai.

## String

String adalah tipe data yang menyimpan rangkaian karakter. Di Java, String adalah object yang memiliki banyak method *built-in* untuk manipulasi teks. String bersifat *immutable* (tidak dapat diubah setelah dibuat).

## 3. Praktikum

### A. Array

#### Praktikum 1.1: Array Satu Dimensi

Memahami cara mendeklarasikan, menginisialisasi, dan mengakses array satu dimensi.

#### Kode Program ( `ArraySatuDimensi.java` ):

```
public class ArraySatuDimensi {  
    public static void main(String[] args) {  
        // Langkah 1: Deklarasi dan inisialisasi array  
        int[] nilai = {85, 90, 78, 92, 88};  
        String[] nama = {"Alice", "Bob", "Charlie", "Diana", "Eva"};  
  
        // Langkah 2: Menampilkan panjang array  
        System.out.println("Jumlah siswa: " + nilai.length);  
  
        // Langkah 3: Mengakses elemen array  
        System.out.println("Nama siswa pertama: " + nama[0]);  
        System.out.println("Nilai siswa pertama: " + nilai[0]);  
  
        // Langkah 4: Menampilkan semua data menggunakan loop  
        System.out.println("\nData semua siswa:");  
        for (int i = 0; i < nama.length; i++) {  
            System.out.println((i + 1) + ". " + nama[i] + " - Nilai: " + nilai[i]);  
        }  
    }  
}
```

#### Output/Analisis

Program di atas mendemonstrasikan pembuatan dan penggunaan array satu dimensi. Dua array, nilai ( tipe int ) dan nama ( tipe String ), diinisialisasi secara langsung. Program kemudian menampilkan jumlah total elemen (panjang

array), mengakses elemen pada indeks ke-0, dan akhirnya melakukan iterasi ( looping ) untuk menampilkan seluruh data siswa yang tersimpan di dalam kedua array tersebut.

### Tugas Praktikum 1.1

Modifikasi kode program diatas untuk menambahkan fitur pencarian nilai tertinggi dan terendah.

#### Tugas Tambahan:

- Buat array untuk menyimpan nilai ujian 10 siswa.
- Hitung dan tampilkan rata-rata nilai.
- Cari dan tampilkan nilai tertinggi dan terendah. (Catatan: Tugas ini akan diimplementasikan dalam file latihan1.java)

#### Implementasi Tugas 1.1 (latihan1.java):

```
package modul_3.latihan;

import java.util.Scanner;

public class latihan1 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        // 1. Buat array untuk 10 siswa
        int[] nilaiSiswa = new int[10];
        int totalNilai = 0;
        int nilaiTertinggi = Integer.MIN_VALUE;
        int nilaiTerendah = Integer.MAX_VALUE;

        System.out.println("Masukkan nilai ujian untuk 10 siswa:");

        // Loop untuk input nilai dan langsung menghitung total, min, max
        for (int i = 0; i < nilaiSiswa.length; i++) {
            System.out.print("Nilai siswa ke-" + (i + 1) + ": ");
            int nilai = input.nextInt();
            nilaiSiswa[i] = nilai;

            // 2. Hitung total nilai
            totalNilai += nilai;

            // 3. Cari nilai tertinggi
            if (nilai > nilaiTertinggi) {
                nilaiTertinggi = nilai;
            }

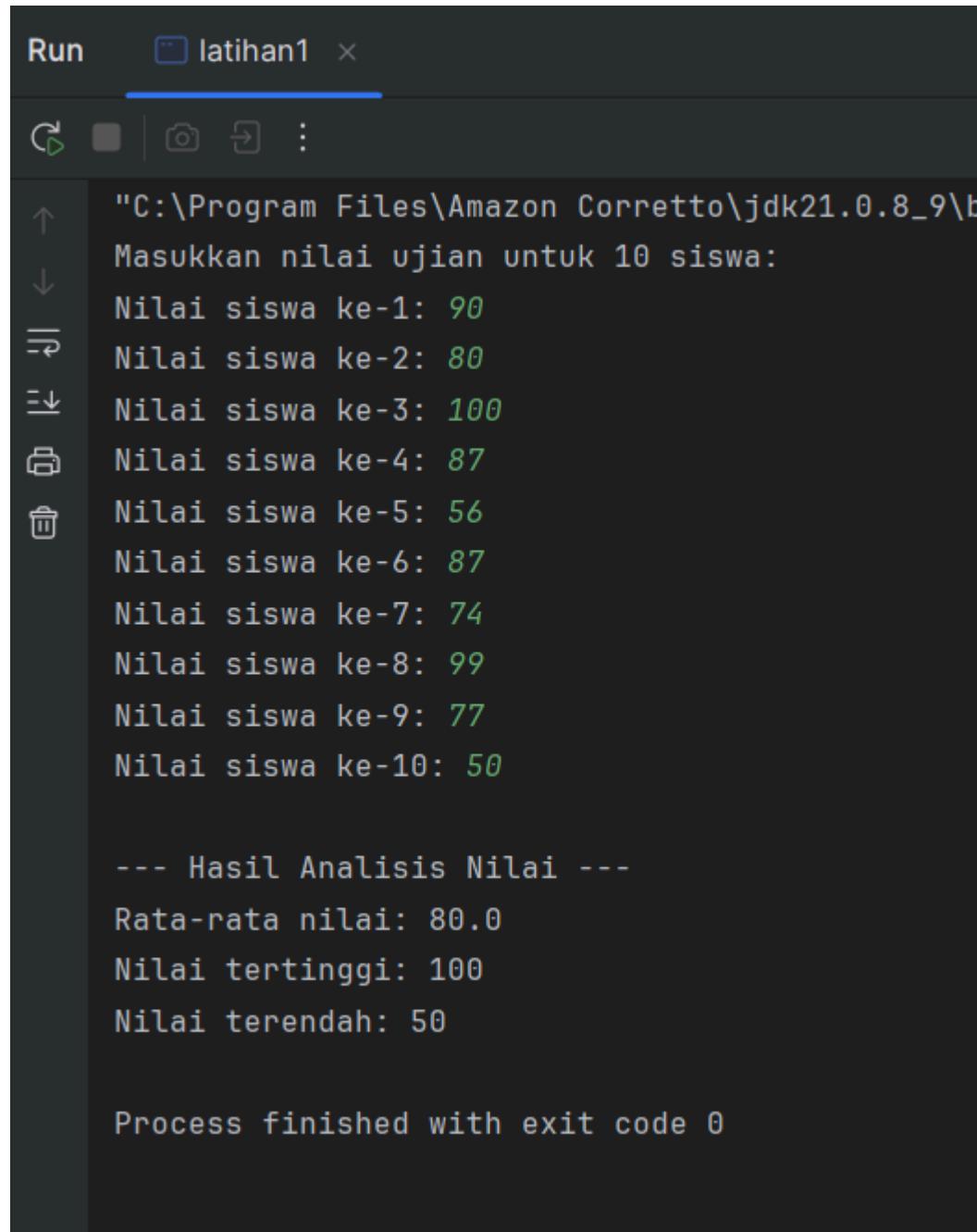
            // 4. Cari nilai terendah
            if (nilai < nilaiTerendah) {
                nilaiTerendah = nilai;
            }
        }

        // 5. Hitung rata-rata
        double rataRata = (double) totalNilai / nilaiSiswa.length;
    }
}
```

```
// 6. Tampilkan hasil
System.out.println("\n--- Hasil Analisis Nilai ---");
System.out.println("Rata-rata nilai: " + rataRata);
System.out.println("Nilai tertinggi: " + nilaiTertinggi);
System.out.println("Nilai terendah: " + nilaiTerendah);

input.close();
}
}
```

Output Tugas 1.1:



The screenshot shows a terminal window titled "latihan1". The window contains the following text:

```
"C:\Program Files\Amazon Corretto\jdk21.0.8_9\b
Masukkan nilai ujian untuk 10 siswa:
Nilai siswa ke-1: 90
Nilai siswa ke-2: 80
Nilai siswa ke-3: 100
Nilai siswa ke-4: 87
Nilai siswa ke-5: 56
Nilai siswa ke-6: 87
Nilai siswa ke-7: 74
Nilai siswa ke-8: 99
Nilai siswa ke-9: 77
Nilai siswa ke-10: 50

--- Hasil Analisis Nilai ---
Rata-rata nilai: 80.0
Nilai tertinggi: 100
Nilai terendah: 50

Process finished with exit code 0
```

Praktikum 1.2: Array Dua Dimensi

## Tujuan

Memahami konsep dan penggunaan array dua dimensi (matriks) untuk studi kasus nilai siswa per mata pelajaran.

Kode Program ( ArrayDuaDimensi.java ):

```
public class MethodDasar {  
    // Langkah 1: Method void tanpa parameter  
    public static void tampilanHeader() {  
        System.out.println("=====");  
        System.out.println("      PROGRAM KALKULATOR SEDERHANA      ");  
        System.out.println("=====");  
    }  
  
    // Langkah 2: Method void dengan parameter  
    public static void tampilanHasil(String operasi, double a, double b, double hasil) {  
        System.out.printf("%.2f %s %.2f = %.2f%n", a, operasi, b, hasil);  
    }  
  
    // Langkah 3: Method dengan return value  
    public static double tambah(double a, double b) { return a + b; }  
    public static double kurang(double a, double b) { return a - b; }  
    public static double kali(double a, double b) { return a * b; }  
  
    public static double bagi(double a, double b) {  
        if (b != 0) {  
            return a / b;  
        } else {  
            System.out.println("Error: Pembagian dengan nol!");  
            return 0;  
        }  
    }  
  
    // Langkah 4: Method dengan validasi input  
    public static boolean validasiAngka(double angka) {  
        return !Double.isNaN(angka) && !Double.isInfinite(angka);  
    }  
  
    public static void main(String[] args) {  
        // Langkah 5: Menggunakan semua method  
        tampilanHeader();  
        double x = 15.5;  
        double y = 4.2;  
  
        if (validasiAngka(x) && validasiAngka(y)) {  
            tampilanHasil("+", x, y, tambah(x, y));  
            tampilanHasil("-", x, y, kurang(x, y));  
            tampilanHasil("*", x, y, kali(x, y));  
            tampilanHasil("/", x, y, bagi(x, y));  
        }  
    }  
}
```

## Output/Analisis

Program ini menggunakan array dua dimensi (nilaiSiswa) untuk menyimpan data nilai yang terstruktur seperti matriks (baris dan kolom). Baris mewakili siswa (diindeks i), dan kolom mewakili mata pelajaran (diindeks j). Program ini

mengambil input dari pengguna untuk mengisi array 2D tersebut, kemudian menampilkannya kembali dalam format tabel yang rapi, sekaligus menghitung rata-rata nilai untuk setiap siswa.

## Tugas Praktikum 1.2

Dari kode praktikum diatas:

- Tambahkan fitur untuk menampilkan mata pelajaran dengan nilai tertinggi untuk setiap siswa.
- Hitung rata-rata nilai untuk setiap mata pelajaran. (Catatan: Tugas ini akan diimplementasikan dalam file latihan2.java)

### Implementasi Tugas 1.2 (latihan2.java):

```
package modul_3.latihan;

import java.util.Scanner;

public class latihan2 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        // Langkah 1: Deklarasi array 2D
        int[][] nilaiSiswa = new int[3][4];
        String[] namaSiswa = {"Alice", "Bob", "Charlie"};
        String[] mataPelajaran = {"Matematika", "B. Indo", "IPA", "IPS"};

        // Langkah 2: Input nilai
        System.out.println("===== INPUT NILAI SISWA =====");
        for (int i = 0; i < 3; i++) {
            System.out.println("Input nilai untuk " + namaSiswa[i] + ":");
            for (int j = 0; j < 4; j++) {
                System.out.print(mataPelajaran[j] + ": ");
                nilaiSiswa[i][j] = input.nextInt();
            }
            System.out.println();
        }

        // Langkah 3: Menampilkan data dalam bentuk tabel
        System.out.println("===== TABEL NILAI SISWA =====");
        System.out.printf("%-12s", "Nama");
        for (String mapel : mataPelajaran) {
            System.out.printf("%-10s", mapel);
        }
        System.out.println("Rata-rata | Nilai Tertinggi");
        System.out.println("-----");

        for (int i = 0; i < 3; i++) {
            System.out.printf("%-12s", namaSiswa[i]);
            int total = 0;
            int nilaiTertinggiSiswa = -1;
            String mapelTertinggi = "";

            for (int j = 0; j < 4; j++) {
                System.out.printf("%-10d", nilaiSiswa[i][j]);
                total += nilaiSiswa[i][j];
                if (nilaiSiswa[i][j] > nilaiTertinggiSiswa) {
                    nilaiTertinggiSiswa = nilaiSiswa[i][j];
                    mapelTertinggi = mapel;
                }
            }
            System.out.println();
            System.out.println("Rata-rata: " + (total / 4));
            System.out.println("Nilai Tertinggi: " + nilaiTertinggiSiswa + " (" + mapelTertinggi + ")");
        }
    }
}
```

```

// Fitur 1: Mencari nilai tertinggi per siswa
if (nilaiSiswa[i][j] > nilaiTertinggiSiswa) {
    nilaiTertinggiSiswa = nilaiSiswa[i][j];
    mapelTertinggi = mataPelajaran[j];
}
}
double rataRata = (double) total / 4.0;
System.out.printf("%-11.2f | %s (%d)%n", rataRata, mapelTertinggi, nilaiTertinggiSiswa);
}
System.out.println("-----");

// Fitur 2: Hitung rata-rata per mata pelajaran
System.out.println("Rata-rata per Mata Pelajaran:");
for (int j = 0; j < 4; j++) { // Loop per kolom (mata pelajaran)
    int totalMapel = 0;
    for (int i = 0; i < 3; i++) { // Loop per baris (siswa)
        totalMapel += nilaiSiswa[i][j];
    }
    double rataRataMapel = (double) totalMapel / 3.0;
    System.out.printf("- %-10s: %.2f%n", mataPelajaran[j], rataRataMapel);
}
}
}

```

### Output Tugas 1.2:

```
Run latihan2 x
[Run]

Input nilai untuk Bob:
Matematika: 78
B. Indo: 90
IPA: 75
IPS: 90

Input nilai untuk Charlie:
Matematika: 69
B. Indo: 72
IPA: 70
IPS: 78

===== TABEL NILAI SISWA =====


| Nama    | Matematika | B. Indo | IPA | IPS | Rata-rata | Nilai Tertinggi |
|---------|------------|---------|-----|-----|-----------|-----------------|
| Alice   | 88         | 92      | 89  | 87  | 89.00     | B. Indo (92)    |
| Bob     | 78         | 90      | 75  | 90  | 83.25     | B. Indo (90)    |
| Charlie | 69         | 72      | 70  | 78  | 72.25     | IPS (78)        |


Rata-rata per Mata Pelajaran:
- Matematika: 78.33
- B. Indo : 84.67
- IPA : 78.00
- IPS : 85.00

Process finished with exit code 0
```

### ### Method / Function

Teori Dasar

Method adalah blok kode yang dapat dipanggil untuk menjalankan tugas tertentu. Method memungkinkan kita untuk:

- Menghindari pengulangan kode
  - Membuat program lebih terstruktur
  - Memudahkan debugging dan maintenance

Jenis Method:

- Static Method : Dapat dipanggil tanpa membuat object
  - Non-static Method : Perlu membuat object terlebih dahulu

- Void Method : Tidak mengembalikan nilai
- Return Method : Mengembalikan nilai

Syntax Method:

```
[access modifier] [static] returnType namaMethod(parameter) {
    // body method
    return value; // jika ada return type
}
```

## Praktikum 2.1: Method Dasar

### Tujuan

Memahami cara membuat dan memanggil method dengan berbagai jenis parameter dan return type.

Langkah - Langkah :

1. Buat file baru dengan nama MethodDasar.java. Ketik kode berikut:

```
public class MethodDasar {

    // Langkah 1: Method void tanpa parameter
    public static void tampilanHeader() {
        System.out.println("=====");
        System.out.println("      PROGRAM KALKULATOR SEDERHANA      ");
        System.out.println("=====");
    }

    // Langkah 2: Method void dengan parameter
    public static void tampilanHasil(String operasi, double a, double b, double hasil) {
        System.out.printf("%.2f %s %.2f = %.2f%n", a, operasi, b, hasil);
    }

    // Langkah 3: Method dengan return value
    public static double tambah(double a, double b) {
        return a + b;
    }

    public static double kurang(double a, double b) {
        return a - b;
    }

    public static double kali(double a, double b) {
        return a * b;
    }

    public static double bagi(double a, double b) {
        if (b != 0) {
            return a / b;
        } else {
            System.out.println("Error: Pembagian dengan nol!");
            return 0;
        }
    }
}
```

```

}

// Langkah 4: Method dengan validasi input
public static boolean validasiAngka(double angka) {
    return !Double.isNaN(angka) && Double.isInfinite(angka);
}

public static void main(String[] args) {
    // Langkah 5: Menggunakan semua method
    tampilanHeader();

    double x = 15.5;
    double y = 4.2;

    if (validasiAngka(x) && validasiAngka(y)) {
        double hasilTambah = tambah(x, y);
        double hasilKurang = kurang(x, y);
        double hasilKali = kali(x, y);
        double hasilBagi = bagi(x, y);

        tampilanHasil("+", x, y, hasilTambah);
        tampilanHasil("-", x, y, hasilKurang);
        tampilanHasil("*", x, y, hasilKali);
        tampilanHasil("/", x, y, hasilBagi);
    }
}
}
}

```

## Output/Analisis

Program ini mendemonstrasikan berbagai jenis method untuk membuat kalkulator sederhana. `tampilanHeader` adalah method void tanpa parameter. `tampilanHasil` adalah method void dengan parameter. `tambah`, `kurang`, `kali`, dan `bagi` adalah method yang mengembalikan nilai (double). Method `main` bertindak sebagai "driver" yang memanggil method-method lain untuk menjalankan fungsionalitas program.

## Tugas Praktikum 2.1

Dari kode praktikum diatas:

- Tambahkan method untuk operasi pangkat dan akar kuadrat.
- Buat method untuk validasi operasi pembagian dengan nol. (Catatan: Tugas ini akan diimplementasikan dalam file `latihan3.java`)

### Implementasi Tugas 2.1 (`latihan3.java`):

```

package modul_3.latihan;

public class latihan3 {

    // Method dari praktikum
    public static void tampilanHeader() {
        System.out.println("=====");
    }
}

```

```
System.out.println("      PROGRAM KALKULATOR CANGGIH      ");
System.out.println("=====");
}

// Method dari praktikum
public static void tampilanHasil(String operasi, double a, double b, double hasil) {
    System.out.printf("%.2f %s %.2f = %.2f%n", a, operasi, b, hasil);
}

// Method baru untuk operasi dengan 1 angka (Akar)
public static void tampilanHasilSatuAngka(String operasi, double a, double hasil) {
    System.out.printf("%s(%s) = %.2f%n", operasi, a, hasil);
}

// Method dari praktikum
public static double tambah(double a, double b) {
    return a + b;
}

// Method dari praktikum
public static double kurang(double a, double b) {
    return a - b;
}

// Method dari praktikum
public static double kali(double a, double b) {
    return a * b;
}

// Method dari praktikum (validasi pembagian nol sudah ada)
public static double bagi(double a, double b) {
    if (b != 0) {
        return a / b;
    } else {
        System.out.println("Error: Pembagian dengan nol!");
        return 0;
    }
}

// TUGAS 1: Method Pangkat
public static double pangkat(double a, double b) {
    return Math.pow(a, b);
}

// TUGAS 2: Method Akar Kuadrat
public static double akarKuadrat(double a) {
    if (a < 0) {
        System.out.println("Error: Tidak bisa menghitung akar angka negatif!");
        return 0;
    }
    return Math.sqrt(a);
}

// Method dari praktikum
public static boolean validasiAngka(double angka) {
    return !Double.isNaN(angka) && Double.isInfinite(angka);
}

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

```

        double x = 16.0; // Diubah agar mudah diakar
        double y = 2.0; // Pangkat 2

        if (validasiAngka(x) && validasiAngka(y)) {
            // Operasi dari praktikum
            tampilkanHasil("+", x, y, tambah(x, y));
            tampilkanHasil("-", x, y, kurang(x, y));
            tampilkanHasil("*", x, y, kali(x, y));
            tampilkanHasil("/", x, y, bagi(x, y));

            System.out.println("--- Operasi Tambahan ---");
            // Memanggil method baru
            tampilkanHasil("^", x, y, pangkat(x, y)); // x pangkat y
            tampilkanHasilSatuAngka("sqrt", x, akarKuadrat(x)); // akar x
        }
    }
}

```

#### Output Tugas 2.1:

"C:\Program Files\Amazon Corretto\jdk21.0.8\_9\bin\java.e  
=====
PROGRAM KALKULATOR CANGGIH  
=====
Process finished with exit code 0

## Praktikum 2.2: Method Dengan Array

### Tujuan

Memahami cara menggunakan method untuk memproses array.

Langkah - Langkah :

1. Buat file baru dengan nama ArrayMethod.java.
2. Ketik kode berikut:

```
import java.util.Scanner;
import java.util.Arrays;

public class ArrayMethod {

    // Method untuk input array
    public static int[] inputArray(int ukuran) {
        int[] array = new int[ukuran];
        Scanner input = new Scanner(System.in);
        System.out.println("Masukkan " + ukuran + " angka:");
        for (int i = 0; i < ukuran; i++) {
            System.out.print("Angka ke-" + (i + 1) + ": ");
            array[i] = input.nextInt();
        }
        return array;
    }

    // Method untuk menampilkan array
    public static void tampilkanArray(int[] array) {
        System.out.print("Array: [");
        for (int i = 0; i < array.length; i++) {
            System.out.print(array[i]);
            if (i < array.length - 1) {
                System.out.print(", ");
            }
        }
        System.out.println("]");
    }

    // Method untuk mencari nilai maksimum
    public static int cariMaksimum(int[] array) {
        int maks = array[0];
        for (int i = 1; i < array.length; i++) {
            if (array[i] > maks) {
                maks = array[i];
            }
        }
        return maks;
    }

    // Method untuk mencari nilai minimum
    public static int cariMinimum(int[] array) {
        int min = array[0];
        for (int i = 1; i < array.length; i++) {
            if (array[i] < min) {
                min = array[i];
            }
        }
        return min;
    }

    // Method untuk menghitung rata-rata
    public static double hitungRataRata(int[] array) {
        int total = 0;
        for (int nilai : array) {
            total += nilai;
        }
        return (double) total / array.length;
    }
}
```

```

// Method untuk mengurutkan array (Bubble Sort)
public static int[] urutkanArray(int[] array) {
    int[] arrayBaru = Arrays.copyOf(array, array.length); // Buat salinan array
    int n = arrayBaru.length;
    for (int i = 0; i < n - 1; i++) {
        for (int j = 0; j < n - i - 1; j++) {
            if (arrayBaru[j] > arrayBaru[j + 1]) {
                // tukar posisi
                int temp = arrayBaru[j];
                arrayBaru[j] = arrayBaru[j + 1];
                arrayBaru[j + 1] = temp;
            }
        }
    }
    return arrayBaru;
}

public static void main(String[] args) {
    System.out.println("===== PROGRAM ANALISIS ARRAY =====");
    int[] data = inputArray(5);

    System.out.println("\n===== HASIL ANALISIS =====");
    tampilkanArray(data);
    System.out.println("Nilai Maksimum: " + cariMaksimum(data));
    System.out.println("Nilai Minimum: " + cariMinimum(data));
    System.out.printf("Rata-rata: %.2f\n", hitungRataRata(data));

    int[] dataUrut = urutkanArray(data);
    System.out.print("Array setelah diurutkan: ");
    tampilkanArray(dataUrut);
}
}

```

## Output/Analisis

Program ini memecah operasi-operasi kompleks pada array menjadi method-method terpisah. Ada method untuk input, menampilkan, mencari nilai maks/min, menghitung rata-rata, dan mengurutkan array. Ini adalah contoh yang baik dari code modularity, di mana main method menjadi bersih dan mudah dibaca, sementara logika yang rumit dienkapsulasi di dalam method-nya masing-masing.

## Tugas Praktikum 2.2

Dari kode praktikum diatas:

- Tambahkan method untuk mencari nilai tertentu dalam array.
- Buat method untuk menghitung median dari array yang sudah diurutkan. (Catatan: Tugas ini akan diimplementasikan dalam file latihan4.java)

## Implementasi Tugas 2.2 (latihan4.java):

```

package modul_3.latihan;

import java.util.Scanner;
import java.util.Arrays;

```

```
public class latihan4 {

    // Method dari praktikum
    public static int[] inputArray(int ukuran) {
        int[] array = new int[ukuran];
        Scanner input = new Scanner(System.in);
        System.out.println("Masukkan " + ukuran + " angka:");
        for (int i = 0; i < ukuran; i++) {
            System.out.print("Angka ke-" + (i + 1) + ": ");
            array[i] = input.nextInt();
        }
        return array;
    }

    // Method dari praktikum
    public static void tampilkanArray(int[] array) {
        System.out.print(Arrays.toString(array)); // Cara lebih singkat
    }

    // Method dari praktikum
    public static int cariMaksimum(int[] array) {
        int maks = array[0];
        for (int i = 1; i < array.length; i++) {
            if (array[i] > maks) maks = array[i];
        }
        return maks;
    }

    // Method dari praktikum
    public static int cariMinimum(int[] array) {
        int min = array[0];
        for (int i = 1; i < array.length; i++) {
            if (array[i] < min) min = array[i];
        }
        return min;
    }

    // Method dari praktikum
    public static double hitungRataRata(int[] array) {
        int total = 0;
        for (int nilai : array) total += nilai;
        return (double) total / array.length;
    }

    // Method dari praktikum
    public static int[] urutkanArray(int[] array) {
        int[] arrayBaru = Arrays.copyOf(array, array.length);
        Arrays.sort(arrayBaru); // Menggunakan method sort bawaan Java
        return arrayBaru;
    }

    // TUGAS 1: Method untuk mencari nilai tertentu
    public static int cariNilai(int[] array, int nilaiDicari) {
        for (int i = 0; i < array.length; i++) {
            if (array[i] == nilaiDicari) {
                return i; // Mengembalikan indeks jika ditemukan
            }
        }
        return -1; // Mengembalikan -1 jika tidak ditemukan
    }
}
```

```

// TUGAS 2: Method untuk menghitung median (menerima array terurut)
public static double hitungMedian(int[] arrayUrut) {
    int n = arrayUrut.length;
    if (n % 2 != 0) {
        // Jika ganjil, ambil nilai tengah
        return arrayUrut[n / 2];
    } else {
        // Jika genap, ambil rata-rata dua nilai tengah
        int tengah1 = arrayUrut[(n / 2) - 1];
        int tengah2 = arrayUrut[n / 2];
        return (double) (tengah1 + tengah2) / 2.0;
    }
}

public static void main(String[] args) {
    Scanner input = new Scanner(System.in);
    System.out.println("===== PROGRAM ANALISIS ARRAY =====");
    int[] data = inputArray(5);

    System.out.println("\n===== HASIL ANALISIS =====");
    System.out.print("Array Asli: ");
    tampilkanArray(data);
    System.out.println("\nNilai Maksimum: " + cariMaksimum(data));
    System.out.println("Nilai Minimum: " + cariMinimum(data));
    System.out.printf("Rata-rata: %.2f\n", hitungRataRata(data));

    int[] dataUrut = urutkanArray(data);
    System.out.print("Array Urut: ");
    tampilkanArray(dataUrut);

    // Menjalankan method tugas
    System.out.println("\n\n===== FITUR TAMBAHAN =====");

    // Tugas 1: Pencarian
    System.out.print("Masukkan nilai yang ingin dicari: ");
    int cari = input.nextInt();
    int indeks = cariNilai(data, cari);
    if (indeks != -1) {
        System.out.println("Nilai " + cari + " ditemukan di indeks " + indeks);
    } else {
        System.out.println("Nilai " + cari + " tidak ditemukan");
    }

    // Tugas 2: Median
    double median = hitungMedian(dataUrut); // Menggunakan array yang sudah diurutkan
    System.out.println("Median dari array terurut: " + median);

    input.close();
}
}

```

## Output Tugas 2.2:

```
Run latihan4 ×
C | : 
↑ "C:\Program Files\Amazon Corretto\jdk21.0.8_9\bin\java.exe" "-javaagent:.
↓ ===== PROGRAM ANALISIS ARRAY =====
≡ Masukkan 5 angka:
≡ Angka ke-1: 9
≡ Angka ke-2: 2
≡ Angka ke-3: 5
≡ Angka ke-4: 6
≡ Angka ke-5: 7

===== HASIL ANALISIS =====
Array Asli: [9, 2, 5, 6, 7]
Nilai Maksimum: 9
Nilai Minimum: 2
Rata-rata: 5.80
Array Urut: [2, 5, 6, 7, 9]

===== FITUR TAMBAHAN =====
Masukkan nilai yang ingin dicari: 5
Nilai 5 ditemukan di indeks 2
Median dari array terurut: 6.0

Process finished with exit code 0
```

## 4. String dan Manipulasi String

### Teori Dasar

String adalah tipe data yang menyimpan rangkaian karakter. Di Java, String adalah object yang memiliki banyak method built-in untuk manipulasi teks.

Karakteristik String di Java:

- String bersifat immutable (tidak dapat diubah)

- Setiap operasi String menghasilkan String baru
- Java menyediakan banyak method untuk manipulasi String

Method String yang Penting:

- `length()` : mendapatkan panjang string
- `charAt()` : mendapatkan karakter pada indeks tertentu
- `substring()` : mengambil bagian string
- `toLowerCase()` / `toUpperCase()` : mengubah case
- `contains()` : mengecek apakah string mengandung substring
- `replace()` : mengganti karakter/substring

## Praktikum 3.1: Manipulasi String Dasar

### Tujuan

Memahami berbagai method manipulasi string dan penggunaannya.

Langkah - Langkah :

1. Buat file baru dengan nama `StringDasar.java`.
2. Ketik kode berikut:

```
import java.util.Scanner;

public class StringDasar {

    public static void analisisString(String teks) {
        System.out.println("\n===== ANALISIS STRING =====");
        System.out.println("Teks asli: " + teks);
        System.out.println("Panjang string: " + teks.length());
        System.out.println("Karakter pertama: " + teks.charAt(0));
        System.out.println("Karakter terakhir: " + teks.charAt(teks.length() - 1));
        System.out.println("Uppercase: " + teks.toUpperCase());
        System.out.println("Lowercase: " + teks.toLowerCase());

        // Menghitung jumlah kata
        String[] kata = teks.split("\\s+");
        System.out.println("Jumlah kata: " + kata.length);

        // Menghitung jumlah vokal
        System.out.println("Jumlah vokal: " + hitungVokal(teks));
    }

    public static int hitungVokal(String teks) {
        String vokal = "aeiouAEIOU";
        int count = 0;
        for (int i = 0; i < teks.length(); i++) {
            if (vokal.contains(String.valueOf(teks.charAt(i)))) {
                count++;
            }
        }
        return count;
    }
}
```

```

public static String reverseString(String teks) {
    StringBuilder reversed = new StringBuilder();
    for (int i = teks.length() - 1; i >= 0; i--) {
        reversed.append(teks.charAt(i));
    }
    return reversed.toString();
}

public static boolean isPalindrome(String teks) {
    String clean = teks.replaceAll("[^a-zA-Z0-9]", "").toLowerCase();
    String reversed = reverseString(clean);
    return clean.equals(reversed);
}

public static void main(String[] args) {
    Scanner input = new Scanner(System.in);
    System.out.println("===== PROGRAM MANIPULASI STRING =====");
    System.out.print("Masukkan sebuah kalimat: ");
    String kalimat = input.nextLine();

    // Analisis dasar
    analisisString(kalimat);

    // String terbalik
    System.out.println("\nString terbalik: \" " + reverseString(kalimat) + " \"");

    // Cek palindrome
    if (isPalindrome(kalimat)) {
        System.out.println("String ini adalah palindrome.");
    } else {
        System.out.println("String ini bukan palindrome.");
    }

    // Pencarian substring
    System.out.print("\nMasukkan kata yang ingin dicari: ");
    String cari = input.nextLine();

    if (kalimat.toLowerCase().contains(cari.toLowerCase())) {
        System.out.println("Kata '" + cari + "' ditemukan dalam kalimat.");
        int posisi = kalimat.toLowerCase().indexOf(cari.toLowerCase());
        System.out.println("Posisi pertama: " + posisi);
    } else {
        System.out.println("Kata '" + cari + "' tidak ditemukan dalam kalimat.");
    }
}
}

```

## Output/Analisis

Program ini menunjukkan berbagai method bawaan dari String dan StringBuilder. Method analisisString menggunakan length(), charAt(), toUpperCase(), toLowerCase(), dan split(). Program ini juga mendefinisikan method helper sendiri seperti hitungVokal, reverseString, dan isPalindrome untuk melakukan operasi yang lebih spesifik.

## Tugas Praktikum 3.1

Dari kode praktikum diatas:

- Tambahkan fitur untuk menghitung jumlah konsonan.
- Buat method untuk mengubah string menjadi pascal case (setiap kata diawali huruf besar). (Catatan: Tugas ini akan diimplementasikan dalam file latihan5.java)

### Implementasi Tugas 3.1 (latihan5.java) :

```
package modul_3.latihan;

import java.util.Scanner;

public class latihan5 {

    public static void analisisString(String teks) {
        System.out.println("\n===== ANALISIS STRING =====");
        System.out.println("Teks asli: " + teks);
        System.out.println("Panjang string: " + teks.length());

        // Menghitung jumlah kata
        String[] kata = teks.split("\s+");
        System.out.println("Jumlah kata: " + kata.length);

        // Menghitung jumlah vokal
        System.out.println("Jumlah vokal: " + hitungVokal(teks));

        // TUGAS 1: Menghitung jumlah konsonan
        System.out.println("Jumlah konsonan: " + hitungKonsonan(teks));
    }

    public static int hitungVokal(String teks) {
        String vokal = "aeiouAEIOU";
        int count = 0;
        for (int i = 0; i < teks.length(); i++) {
            if (vokal.contains(String.valueOf(teks.charAt(i)))) {
                count++;
            }
        }
        return count;
    }

    // TUGAS 1: Method hitung konsonan
    public static int hitungKonsonan(String teks) {
        String vokal = "aeiouAEIOU";
        int count = 0;
        for (int i = 0; i < teks.length(); i++) {
            char ch = teks.charAt(i);
            // Cek apakah karakter adalah huruf DAN bukan vokal
            if (Character.isLetter(ch) && !vokal.contains(String.valueOf(ch))) {
                count++;
            }
        }
        return count;
    }

    // Method dari praktikum
    public static String reverseString(String teks) {
```

```

        return new StringBuilder(teks).reverse().toString();
    }

    // Method dari praktikum
    public static boolean isPalindrome(String teks) {
        String clean = teks.replaceAll("[^a-zA-Z0-9]", "").toLowerCase();
        String reversed = new StringBuilder(clean).reverse().toString();
        return clean.equals(reversed);
    }

    // TUGAS 2: Method Pascal Case (Title Case)
    public static String toTitleCase(String teks) {
        String[] kata = teks.split("\s+");
        StringBuilder result = new StringBuilder();
        for (String k : kata) {
            if (k.length() > 0) {
                result.append(Character.toUpperCase(k.charAt(0))) // Huruf pertama kapital
                    .append(k.substring(1).toLowerCase()) // Sisanya huruf kecil
                    .append(" ");
            }
        }
        return result.toString().trim(); // Menghilangkan spasi di akhir
    }

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.println("===== PROGRAM MANIPULASI STRING =====");
        System.out.print("Masukkan sebuah kalimat: ");
        String kalimat = input.nextLine();

        // Analisis dasar (termasuk vokal dan konsonan)
        analisisString(kalimat);

        // Cek palindrome
        System.out.println("\nCek Palindrome:");
        if (isPalindrome(kalimat)) {
            System.out.println("-> String ini ADALAH palindrome.");
        } else {
            System.out.println("-> String ini BUKAN palindrome.");
        }

        // TUGAS 2: Panggil method Title Case
        System.out.println("\nFormat Title Case:");
        System.out.println("-> " + toTitleCase(kalimat));

        input.close();
    }
}

```

### Output Tugas 3.1:

```
Run latihan5 ×
C | : 
"C:\Program Files\Amazon Corretto\jdk21.0.8_9\bin\java.exe" "-javaagent:C:\U
===== PROGRAM MANIPULASI STRING =====
Masukkan sebuah kalimat: assalamualaikum
=====
===== ANALISIS STRING =====
Teks asli: assalamualaikum
Panjang string: 15
Jumlah kata: 1
Jumlah vokal: 8
Jumlah konsonan: 7

Cek Palindrome:
-> String ini BUKAN palindrome.

Format Title Case:
-> Assalamualaikum

Process finished with exit code 0
```

## Praktikum 3.2: Pengolahan Data Text

### Tujuan

Memahami penggunaan string untuk pengolahan data dalam format teks.

Langkah - Langkah :

1. Buat file baru dengan nama `TextProcessing.java`.
2. Ketik kode berikut:

```
import java.util.Scanner;
import java.util.Arrays;

public class TextProcessing {

    public static String[] prosesDataMahasiswa(String data) {
        System.out.println("Data yang dimasukkan: " + data);
        // Memisahkan data berdasarkan koma
        String[] namaMahasiswa = data.split(",");
        for (int i = 0; i < namaMahasiswa.length; i++) {
            // Menghilangkan spasi
            namaMahasiswa[i] = namaMahasiswa[i].trim();
        }
        return namaMahasiswa;
    }

    // Analisis data
    public static void analisisData(String[] namaMahasiswa) {
        System.out.println("\n===== ANALISIS DATA =====");
        System.out.println("Jumlah mahasiswa: " + namaMahasiswa.length);

        // Cari nama terpanjang dan terpendek
        String namaTerpanjang = namaMahasiswa[0];
        String namaTerpendek = namaMahasiswa[0];

        for (String nama : namaMahasiswa) {
            if (nama.length() > namaTerpanjang.length()) {
                namaTerpanjang = nama;
            }
            if (nama.length() < namaTerpendek.length()) {
                namaTerpendek = nama;
            }
        }
        System.out.println("Nama terpanjang: " + namaTerpanjang + " (" + namaTerpanjang.length() + " karakter)");
        System.out.println("Nama terpendek: " + namaTerpendek + " (" + namaTerpendek.length() + " karakter)");

        // Mengurutkan nama secara alfabetis
        String[] namaUrut = Arrays.copyOf(namaMahasiswa, namaMahasiswa.length);
        Arrays.sort(namaUrut, String.CASE_INSENSITIVE_ORDER);

        System.out.println("\nNama mahasiswa (urutan alfabetis):");
        for (int i = 0; i < namaUrut.length; i++) {
            System.out.println((i + 1) + ". " + namaUrut[i]);
        }
    }

    // Format Nama
    public static void formatNama(String nama) {
        System.out.println("\n===== FORMAT NAMA =====");
        String namaFormatted = formatTitleCase(nama);
        System.out.println("Nama lengkap: " + namaFormatted);
        System.out.println("Inisial: " + buatInisial(nama));
        System.out.println("Username: " + buatUsername(namaFormatted));
    }

    public static String formatTitleCase(String teks) {
        String[] kata = teks.split("\\s+");
        StringBuilder result = new StringBuilder();
        for (String k : kata) {
            if (k.length() > 0) {

```

```

        result.append(Character.toUpperCase(k.charAt(0)))
            .append(k.substring(1).toLowerCase())
            .append(" ");
    }
}
return result.toString().trim();
}

public static String buatInisial(String nama) {
    String[] kata = nama.split("\s+");
    StringBuilder inisial = new StringBuilder();
    for (String k : kata) {
        if (k.length() > 0) {
            inisial.append(Character.toUpperCase(k.charAt(0)));
        }
    }
    return inisial.toString();
}

public static String buatUsername(String nama) {
    return nama.toLowerCase().replaceAll("\s+", ".").replaceAll("[^a-zA-Z0-9.]", "");
}

public static void main(String[] args) {
    System.out.println("===== PROGRAM PENGOLAHAN DATA TEKS =====");
    System.out.print("Masukkan nama-nama mahasiswa (pisahkan dengan koma): ");
    Scanner input = new Scanner(System.in);
    String dataNama = input.nextLine();

    // Proses data
    String[] arrayNama = prosesDataMahasiswa(dataNama);

    // Analisis Data
    analisisData(arrayNama);

    // Format Nama (ambil nama pertama sebagai contoh)
    if (arrayNama.length > 0) {
        formatNama(arrayNama[0]);
    }
}
}
}

```

## Output/Analisis

Program ini mensimulasikan pengolahan data teks yang umum. Data mentah (string panjang dengan pemisah koma) diubah menjadi array string menggunakan `split()`. Spasi yang tidak perlu dihilangkan dengan `trim()`. Program kemudian melakukan analisis (mencari nama terpanjang/terpendek) dan melakukan sorting. Terakhir, program ini menunjukkan cara memformat satu nama menjadi berbagai format (Title Case, inisial, username) menggunakan kombinasi `split()`, `substring()`, `toUpperCase()`, dan `replaceAll()`.

## Tugas Praktikum 3.2

Dari kode praktikum diatas:

- Tambahkan fitur untuk menghasilkan email address berdasarkan nama (contoh: username@domain.com).
- Buat method untuk memvalidasi format nama (tidak boleh mengandung angka atau karakter khusus). (Catatan: Tugas ini akan diimplementasikan dalam file latihan6.java)

**Implementasi Tugas 3.2** (latihan6.java) :

```

package modul_3.latihan;

import java.util.Scanner;
import java.util.Arrays;
import java.util.regex.Pattern; // Import untuk validasi

public class latihan6 {

    // Method dari praktikum
    public static String[] prosesDataMahasiswa(String data) {
        String[] namaMahasiswa = data.split(",");
        for (int i = 0; i < namaMahasiswa.length; i++) {
            namaMahasiswa[i] = namaMahasiswa[i].trim();
        }
        return namaMahasiswa;
    }

    // Method dari praktikum
    public static void analisisData(String[] namaMahasiswa) {
        System.out.println("\n===== ANALISIS DATA =====");
        System.out.println("Jumlah mahasiswa: " + namaMahasiswa.length);
        // ... (Analisis nama terpanjang/terpendek, dll. bisa ditambahkan di sini)
    }

    // Method dari praktikum (dimodifikasi untuk TUGAS 1)
    public static void formatNama(String nama) {
        System.out.println("\n===== FORMAT NAMA =====");
        String namaFormatted = formatTitleCase(nama);
        String username = buatUsername(namaFormatted);

        System.out.println("Nama lengkap: " + namaFormatted);
        System.out.println("Inisial: " + buatInisial(nama));
        System.out.println("Username: " + username);

        // TUGAS 1: Menghasilkan email address
        System.out.println("Email: " + username + "@latihan.com");
    }

    // Method dari praktikum
    public static String formatTitleCase(String teks) {
        String[] kata = teks.split("\\s+");
        StringBuilder result = new StringBuilder();
        for (String k : kata) {
            if (k.length() > 0) {
                result.append(Character.toUpperCase(k.charAt(0)))
                    .append(k.substring(1).toLowerCase())
                    .append(" ");
            }
        }
        return result.toString().trim();
    }

    // Method dari praktikum
}

```

```

public static String buatInisial(String nama) {
    String[] kata = nama.split("\s+");
    StringBuilder inisial = new StringBuilder();
    for (String k : kata) {
        if (k.length() > 0) {
            inisial.append(Character.toUpperCase(k.charAt(0)));
        }
    }
    return inisial.toString();
}

// Method dari praktikum
public static String buatUsername(String nama) {
    return nama.toLowerCase().replaceAll("\s+", ".").replaceAll("[^a-zA-Z]", "");
}

// TUGAS 2: Method validasi nama (hanya boleh huruf dan spasi)
public static boolean validasiNama(String nama) {
    // Regex ini berarti: "harus terdiri dari 1 atau lebih huruf (a-z, A-Z) dan spasi"
    return Pattern.matches("^[a-zA-Z\\s]+$", nama);
}

public static void main(String[] args) {
    System.out.println("===== PROGRAM PENGOLAHAN DATA TEKS =====");
    System.out.print("Masukkan nama-nama mahasiswa (pisahkan dengan koma): ");
    Scanner input = new Scanner(System.in);
    String dataNama = input.nextLine();

    // Proses data
    String[] arrayNama = prosesDataMahasiswa(dataNama);
    analisisData(arrayNama);

    // Validasi dan Format setiap nama
    for (String nama : arrayNama) {
        System.out.println("\n-----");
        // TUGAS 2: Memanggil validasi
        if (validasiNama(nama)) {
            System.out.println("Validasi '" + nama + "' : OK");
            formatNama(nama);
        } else {
            System.out.println("Validasi '" + nama + "' : GAGAL (Mengandung angka/simbol)");
        }
    }
}
}

```

### Output Tugas 3.2:

```
Run latihan6 ×
C | : 
"C:\Program Files\Amazon Corretto\jdk21.0.8_9\bin\java.exe" "-javaagent:C:\Us
===== PROGRAM PENGOLAHAN DATA TEKS =====
Masukkan nama-nama mahasiswa (pisahkan dengan koma): Hawari, Fauzan, Aziz, Al
=====
===== ANALISIS DATA =====
Jumlah mahasiswa: 7
-----
Validasi 'Hawari': OK

===== FORMAT NAMA =====
Nama lengkap: Hawari
Inisial: H
Username: hawari
Email: hawari@latihan.com

-----
Validasi 'Fauzan': OK

===== FORMAT NAMA =====
Nama lengkap: Fauzan
Inisial: F
Username: fauzan
Email: fauzan@latihan.com

-----
Validasi 'Aziz': OK

===== FORMAT NAMA =====
Nama lengkap: Aziz
Aktikum-pbo-2024573010123 > src > modul_3 > latihan > C latihan6
```

## 5. Kesimpulan

Praktikum Modul 3 ini memberikan pemahaman mendalam tentang tiga pilar penting dalam pemrograman Java: Array, Method, dan String. Melalui implementasi kode dan penyelesaian tugas-tugas, telah dipelajari cara:

Menggunakan Array: Mengelola kumpulan data dengan tipe yang sama secara efisien, baik dalam satu dimensi untuk data linear maupun dua dimensi untuk data tabular/matriks.

Menggunakan Method: Menstrukturkan program agar lebih rapi, modular, dan reusable. Pemecahan masalah menjadi fungsi-fungsi kecil (seperti kalkulator, analisis array) membuat kode lebih mudah dibaca dan di-debug.

Menggunakan String: Memanipulasi data teks, yang merupakan kebutuhan umum dalam aplikasi. Operasi seperti parsing, pemformatan, validasi, dan pencarian telah dipraktikkan secara langsung.

Secara keseluruhan, konsep-konsep ini adalah fondasi penting yang akan sering digunakan dalam pengembangan aplikasi Java yang lebih besar dan kompleks.