

# Descriptive Statistics

Program Statistik Sederhana Menggunakan Java

KELOMPOK 4



# Anggota kelompok :

Azilia Naning  
Rahmawati  
5002241127



Rizqi Al Rahmah  
5002241031

Apta Rizkika  
Rahmadhani  
5002241087



# Rumus



**Mean**

$$\Sigma X / n$$



**Median**

nilai data ke-  
(n+1 / 2).



**Modus**

Nilai yang  
paling sering  
muncul dalam  
kumpulan data.



**Varian**

$$\Sigma (X - \mu)^2 / n$$



**Standar  
Deviasi**

$$= \sqrt{n \Sigma (X - \mu)^2}$$



**Range**

$$X_{\text{maks}} - X_{\text{min}}$$



**Kuartil**

nilai data ke-  
 $k(n+1) / 4$



**Desil**

nilai data ke-  
 $k(n+1) / 10$



**Persentil**

nilai data ke-  
 $k(n+1) / 100$

## Metode yang Digunakan



### Array

Memudahkan penyimpanan dan pengolahan sejumlah data sekaligus.

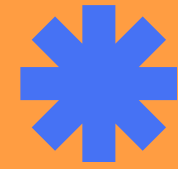
```
double [] data = new double[n];
```



### Method

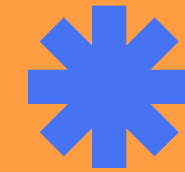
Setiap perhitungan (mean, median, modus, dll.) ditempatkan dalam method terpisah.

```
static double mean (int n, double[] data)
```



### If-Else

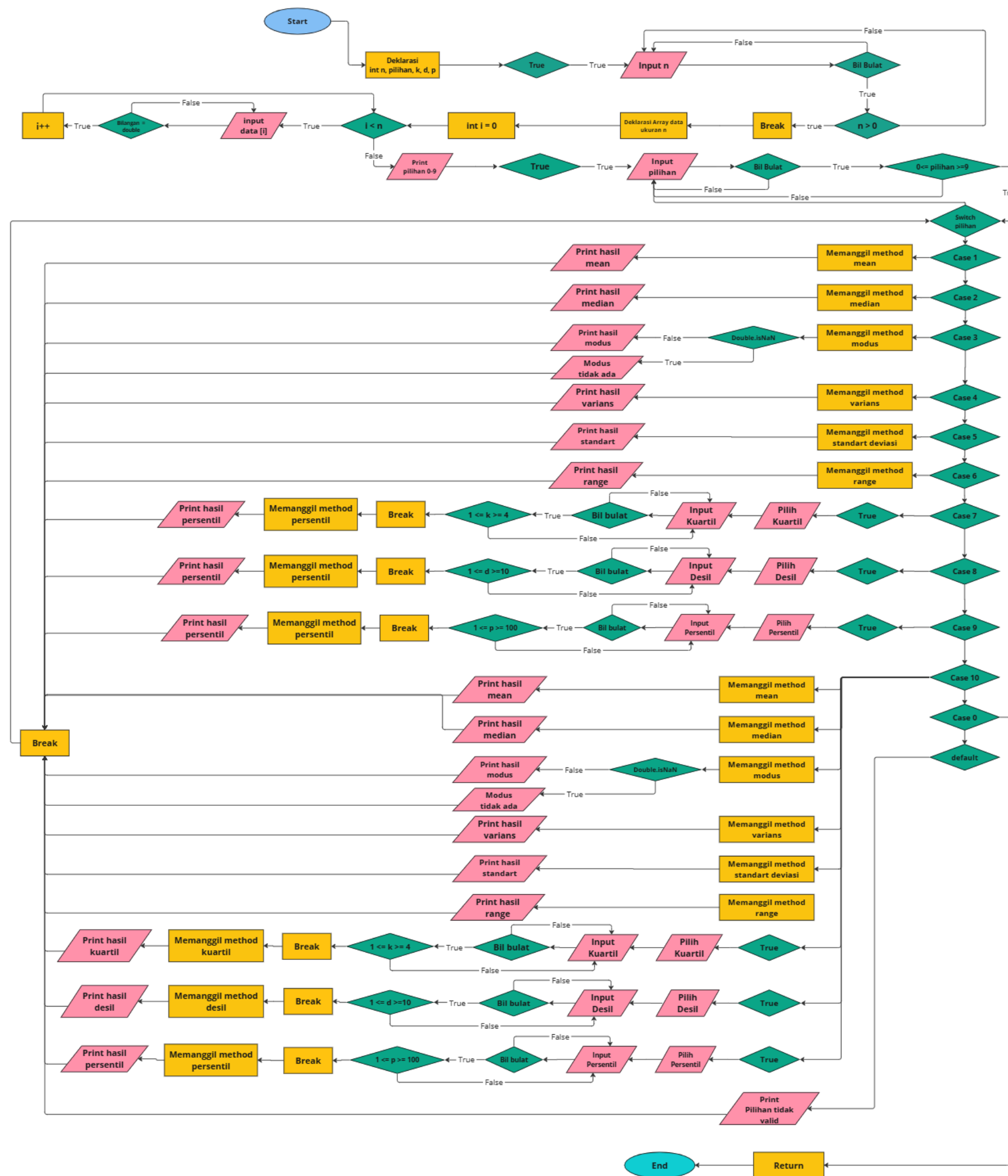
Digunakan untuk mengecek kondisi, misalnya jika data genap/ganjil saat menghitung median atau untuk memvalidasi input.



### Loop

Digunakan untuk mengulangi proses perhitungan nilai seperti mean, varians, dan mencari modus.

```
while (true) {}  
  
for(int i = 0; i < data.length; i++)
```



miro

# Flowchart main method

<https://intip.in/Flowchartmain>





## Main Method

```
public static void main(String[] args) {
    Scanner input = new Scanner (System.in);
    int n,pilihan,k,d,p;

    while (true) {
        System.out.print("Masukkan jumlah data: ");
        if (input.hasNextInt()) {
            n = input.nextInt();
            if (n > 0) {
                break;
            } else {
                System.out.println("Jumlah data harus lebih besar dari nol.");
            }
        } else {
            System.out.println("Input tidak valid. Harap masukkan
bilangan bulat positif.");
            input.next();
        }
    }

    double [] data = new double[n];
    for(int i = 0; i < data.length; i++) {
        System.out.print("Nilai ke-" + (i + 1) + " adalah: ");
        while (!input.hasNextDouble()) {
            System.out.println("Input tidak valid. Harap masukkan angka.");
            System.out.print("Nilai ke-" + (i + 1) + " adalah: ");
            input.next();
        }
        data[i] = input.nextDouble();
    }
}
```

```
System.out.println("\nPilih perhitungan yang ingin dilakukan:");
System.out.println("1. Hitung Mean\n2. Hitung Median\n3.
HitungModus\n4. Hitung Varians\n5. Hitung Standar Deviasi");
System.out.println("6. Hitung Range\n7. Hitung Kuartil\n8. Hitung
Desil\n9. Hitung Persentil\n0. Keluar");
```

```
while (true) {
    System.out.print("\nMasukkan pilihan (0-9): ");
    if (input.hasNextInt()) {
        pilihan = input.nextInt();
        if (pilihan >= 0 && pilihan <= 9) {
            switch (pilihan) {
                case 1:
                    System.out.println("Mean: " + mean(n, data));
                    break;
                case 2:
                    System.out.println("Median: " + median(n, data));
                    break;
                case 3:
                    double modus = modus(n, data);
                    if (Double.isNaN(modus)) {
                        System.out.println("Modus: Tidak ada (semua nilai
muncul hanya sekali)");
                    } else {
                        System.out.println("Modus: " + modus);
                    }
                    break;
                case 4:
                    System.out.println("Varians: " + varians(n, data));
                    break;
                case 5:
                    System.out.println("Standar deviasi: " + sdev(n, data));
                    break;
                case 6:
                    System.out.println("Range: " + range(n, data));
                    break;
            }
        }
    }
}
```







(lanjutan)

```
case 7:
    while (true) {
        System.out.print("Hitung kuartil ke (1, 2, 3, atau 4): ");
        if (input.hasNextInt()) {
            k = input.nextInt();
            if (k >= 1 && k <= 4) break;
            else System.out.println("Input tidak valid. Masukkan angka 1, 2, 4, atau 4.");
        } else {
            System.out.println("Input tidak valid. Harap masukkan bilangan bulat.");
            input.next();
        }
    }
    System.out.println("Kuartil ke-" + k + ": " + kuartil(k, data));
    break;
case 8:
    while (true) {
        System.out.print("Hitung desil ke (1 sampai 10): ");
        if (input.hasNextInt()) {
            d = input.nextInt();
            if (d >= 1 && d <= 10) break;
            else System.out.println("Input tidak valid. Desil harus antara 1 dan 10.");
        } else {
            System.out.println("Input tidak valid. Harap masukkan bilangan bulat.");
            input.next();
        }
    }
    System.out.println("Desil ke-" + d + ": " + desil(d, data));
    break;
```

```
case 9:
    while (true) {
        System.out.print("Hitung persentil ke (1 sampai 100): ");
        if (input.hasNextInt()) {
            p = input.nextInt();
            if (p >= 1 && p <= 100) break;
            else System.out.println("Input tidak valid. Persentil harus
antara 1 dan 100.");
        } else {
            System.out.println("Input tidak valid. Harap masukkan
bilangan bulat.");
            input.next();
        }
    }
    System.out.println("Persentil ke-" + p + ": " + persentil(p, data));
    break;
case 10:
    System.out.println("Mean: " + mean(n, data));
    System.out.println("Median: " + median(n, data));

    double[] allModus = modus(n, data);
    if (allModus.length == 0) {
        System.out.println("Modus: Tidak ada (semua nilai muncul
hanya sekali)");
    } else {
        System.out.print("Modus: ");
        for (double m : allModus) {
            System.out.print(m + " ");
        }
        System.out.println();
    }

    System.out.println("Varians: " + varians(n, data));
    System.out.println("Standar deviasi: " + sdev(n, data));
    System.out.println("Range: " + range(n, data));
```



```

while (true) {
    System.out.print("Hitung kuartil ke (1, 2, 3, atau 4): ");
    if (input.hasNextInt()) {
        k = input.nextInt();
        if (k >= 1 && k <= 4) break;
        else System.out.println("Input tidak valid. Masukkan angka 1, 2, 3, atau 4.");
    } else {
        System.out.println("Input tidak valid. Harap masukkan bilangan bulat.");
        input.next();
    }
}
System.out.println("Kuartil ke-" + k + ": " + kuartil(k, data));

while (true) {
    System.out.print("Hitung desil ke (1 sampai 10): ");
    if (input.hasNextInt()) {
        d = input.nextInt();
        if (d >= 1 && d <= 10) break;
        else System.out.println("Input tidak valid. Desil harus antara 1 dan 10.");
    } else {
        System.out.println("Input tidak valid. Harap masukkan bilangan bulat.");
        input.next();
    }
}
System.out.println("Desil ke-" + d + ": " + desil(d, data));

while (true) {
    System.out.print("Hitung persentil ke (1 sampai 100): ");
    if (input.hasNextInt()) {
        p = input.nextInt();
        if (p >= 1 && p <= 100) break;
        else System.out.println("Input tidak valid. Persentil harus antara 1 dan 100.");
    } else {
        System.out.println("Input tidak valid. Harap masukkan bilangan bulat.");
        input.next();
    }
}
System.out.println("Persentil ke-" + p + ": " + persentil(p, data));
break;

```

```

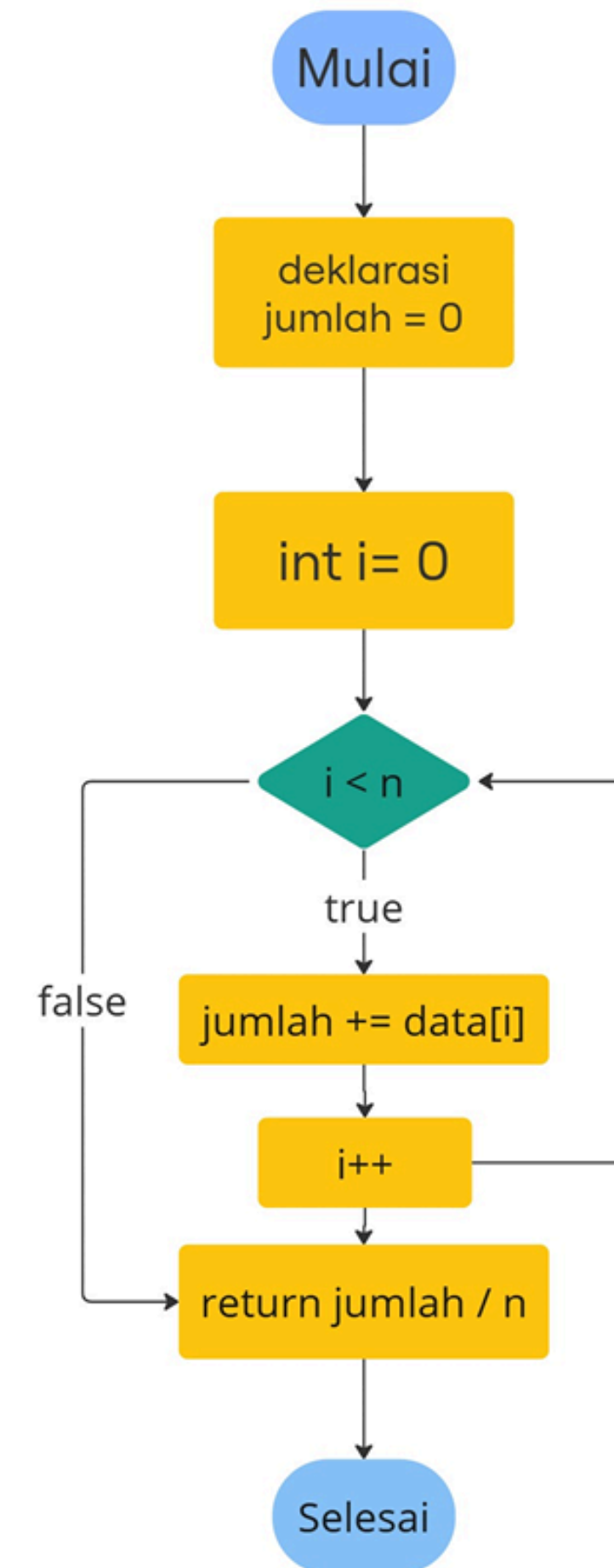
        case 0:
            System.out.println("Keluar dari program.");
            return;
        default:
            System.out.println("Pilihan tidak valid.");
            break;
    }
} else {
    System.out.println("Input tidak valid. Harap masukkan angka antara
0 dan 9.");
}
} else {
    System.out.println("Input tidak valid. Harap masukkan angka.");
    input.next();
}
}
}

```



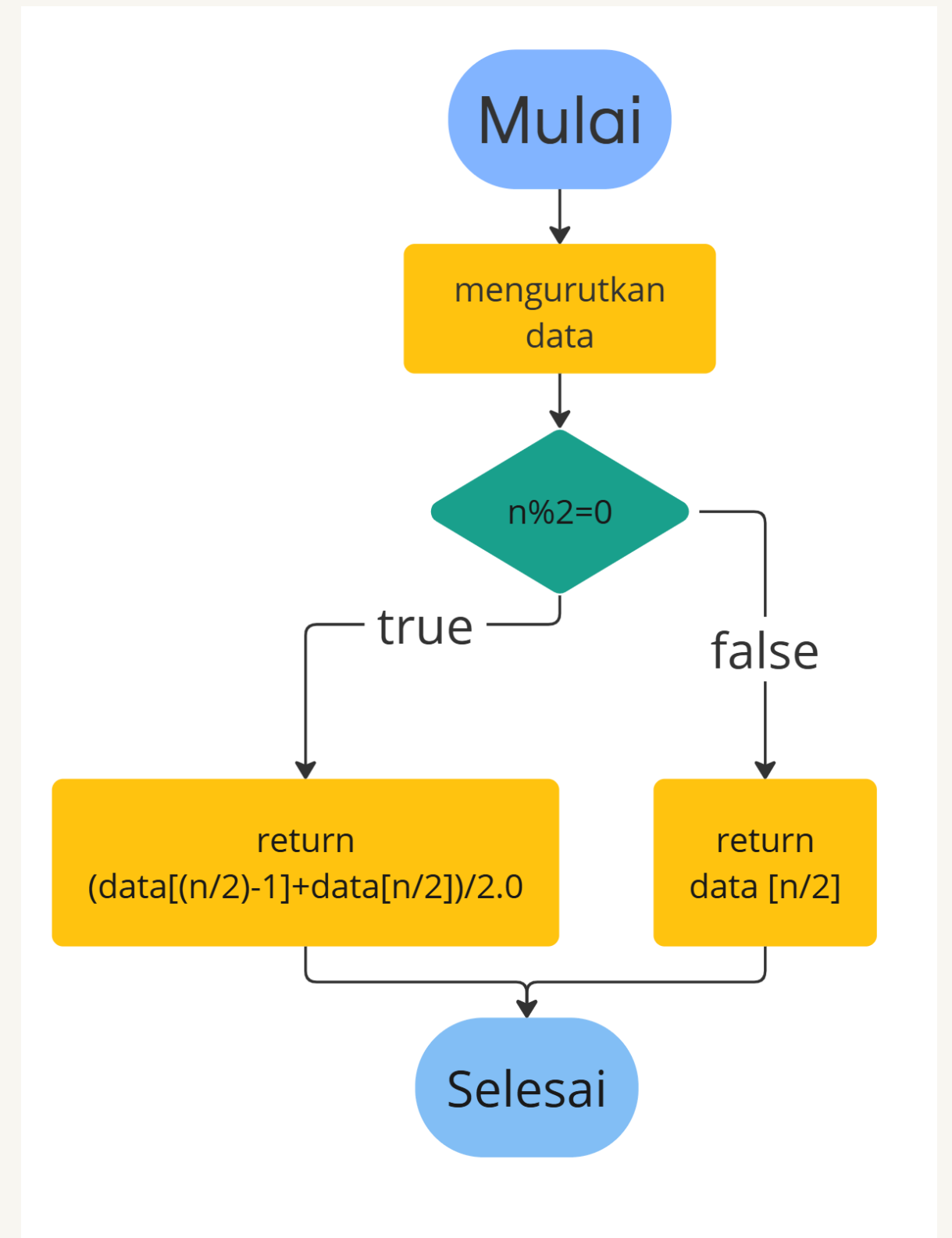
## Method Mean

```
static double mean (int n, double[] data){  
    double jumlah = 0;  
    for(int i=0;i<n;i++){  
        jumlah += data[i];  
    }  
    return jumlah / n;  
}
```



## Method Median

```
static double median (int n, double[] data){  
    java.util.Arrays.sort(data);  
    if (n%2==0){  
        return (data[(n/2)-1]+data[n/2])/2.0;  
    } else {  
        return data [n/2];  
    }  
}
```



## Method Modus

```
static double[] modus(int n, double[] data) {
    double[] hasilModus = new double[n];
    int jumlahModus = 0;
    int maxFrekuensi = 0;
```

```
for (int i = 0; i < n; i++) {  
    int frekuensi = 0;  
    for (int j = 0; j < n; j++) {  
        if (data[i] == data[j]) {  
            frekuensi++;  
        }  
    }  
}
```

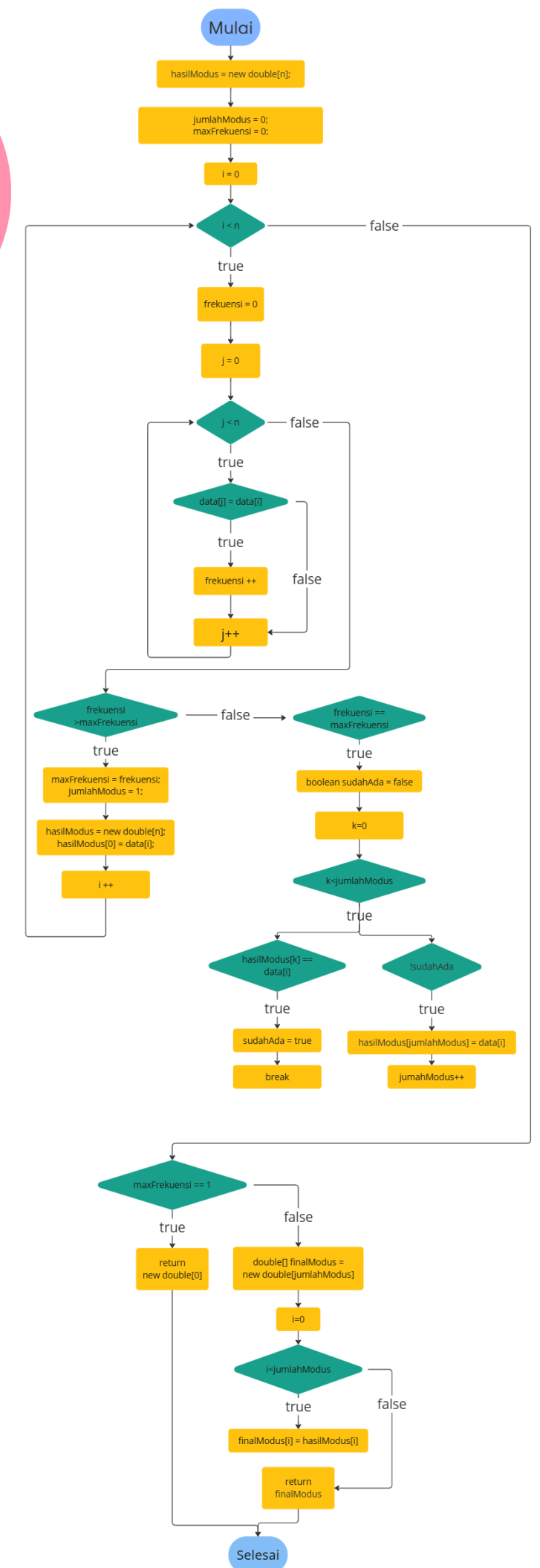
```
if (frekuensi > maxFrekuensi) {
    maxFrekuensi = frekuensi;
    hasilModus = new double[n];
    hasilModus[0] = data[i];
    jumlahModus = 1;
} else if (frekuensi == maxFrekuensi) {
    boolean sudahAda = false;
    for (int k = 0; k < jumlahModus; k++) {
        if (hasilModus[k] == data[i]) {
            sudahAda = true;
            break;
        }
    }
}
```

```
if (!sudahAda) {
    hasilModus[jumlahModus] = data[i];
    jumlahModus++;
}
```

```
if (maxFrekuensi == 1) {  
    return new double[0];  
}
```

```
double[] finalModus = new double[jumlahModus];
for (int i = 0; i < jumlahModus; i++) {
    finalModus[i] = hasilModus[i];
}
```

```
return finalModus;
```

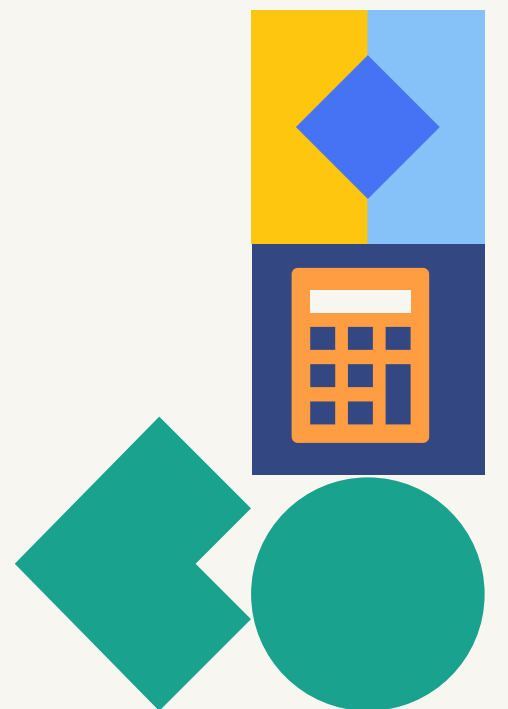
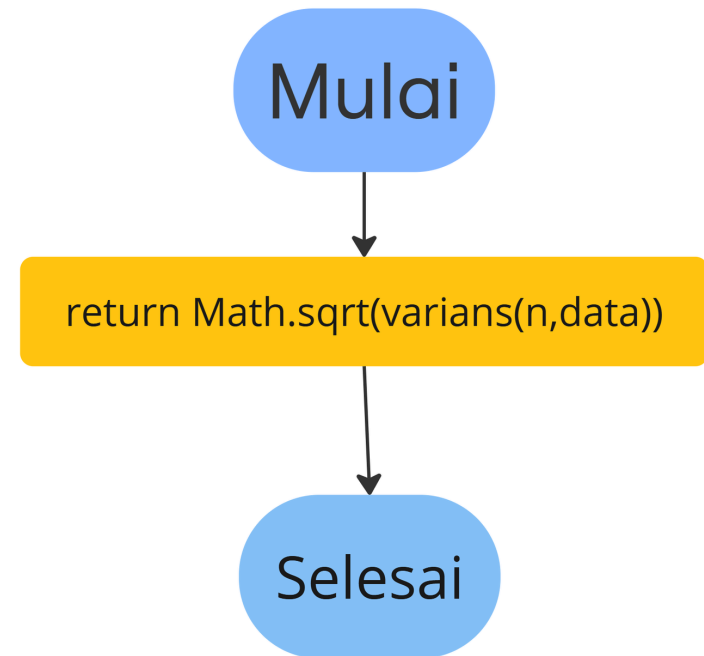
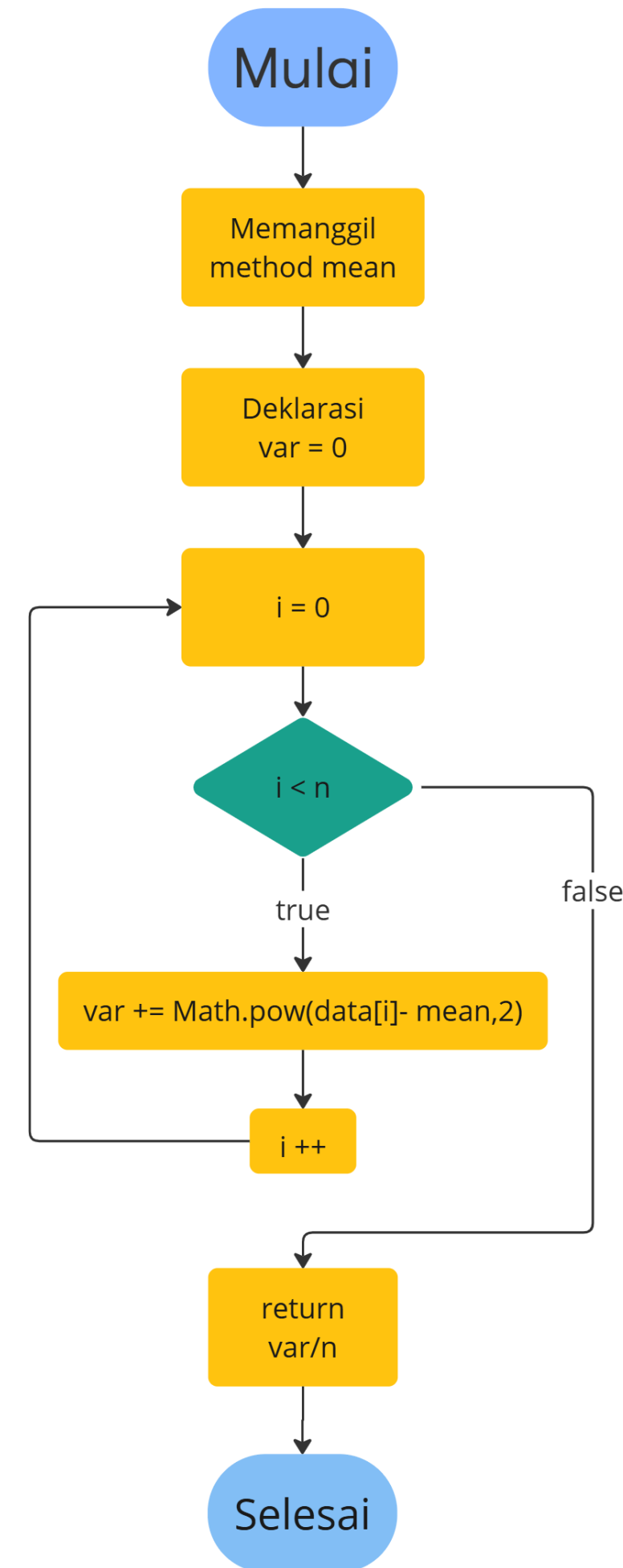


## Method Varians

```
static double varians (int n, double[] data){  
    double mean = mean(n,data);  
    double var = 0;  
    for (int i=0; i<n; i++){  
        var += Math.pow(data[i]- mean,2);  
    }  
    return var/n;  
}
```

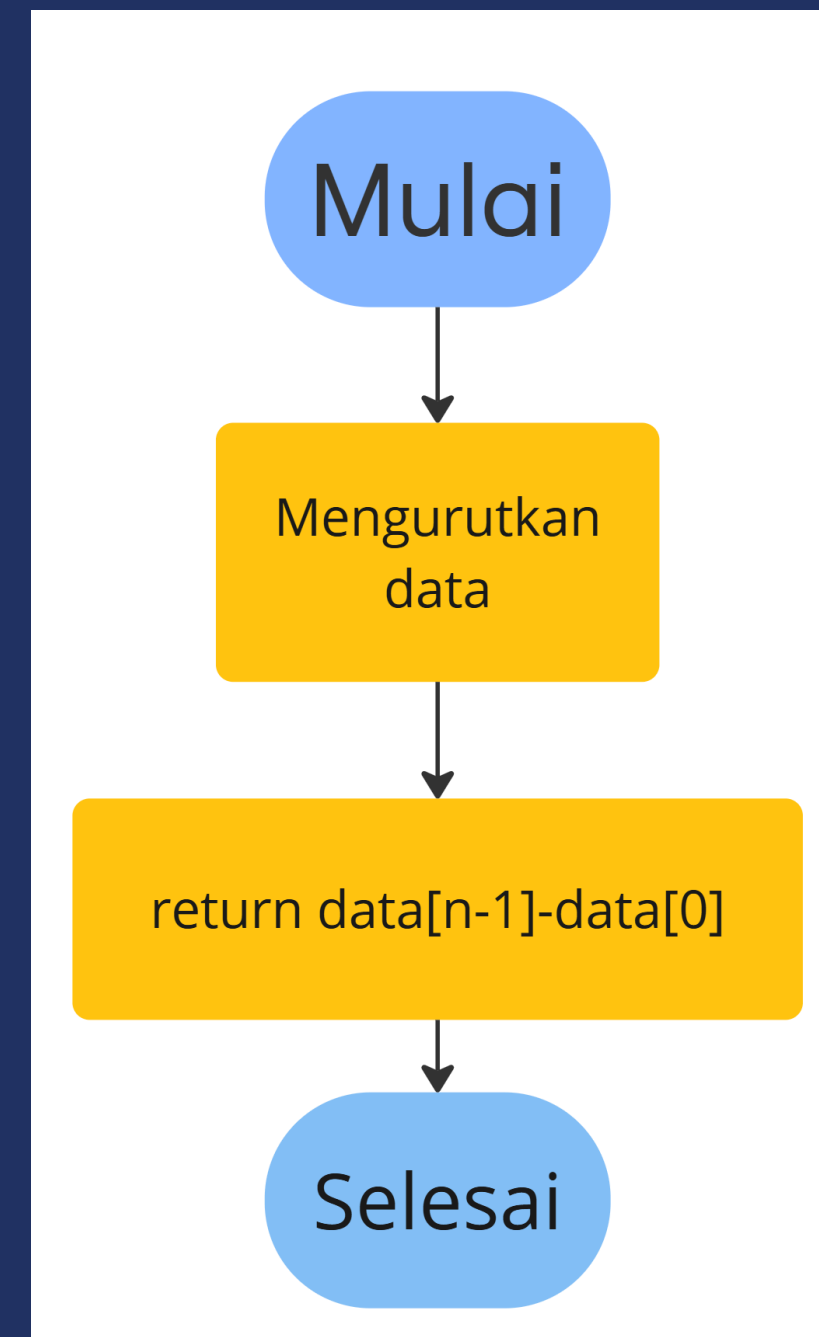
## Method Standar Deviasi

```
static double sdev (int n, double[] data){  
    return Math.sqrt(varians(n,data));  
}
```



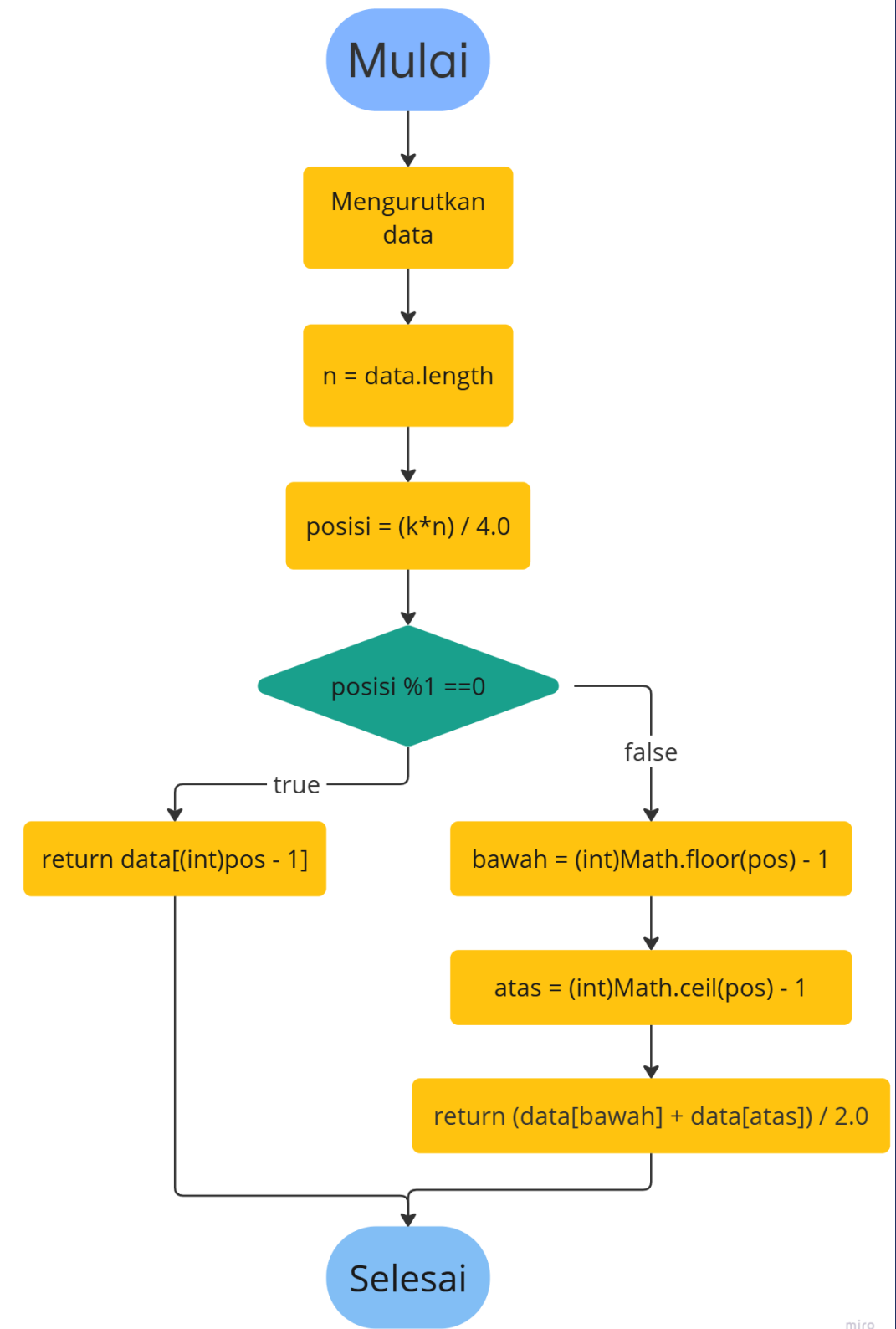
## Method Range

```
static double range (int n, double[] data){  
    java.util.Arrays.sort(data);  
    return data[n-1]-data[0];  
}
```

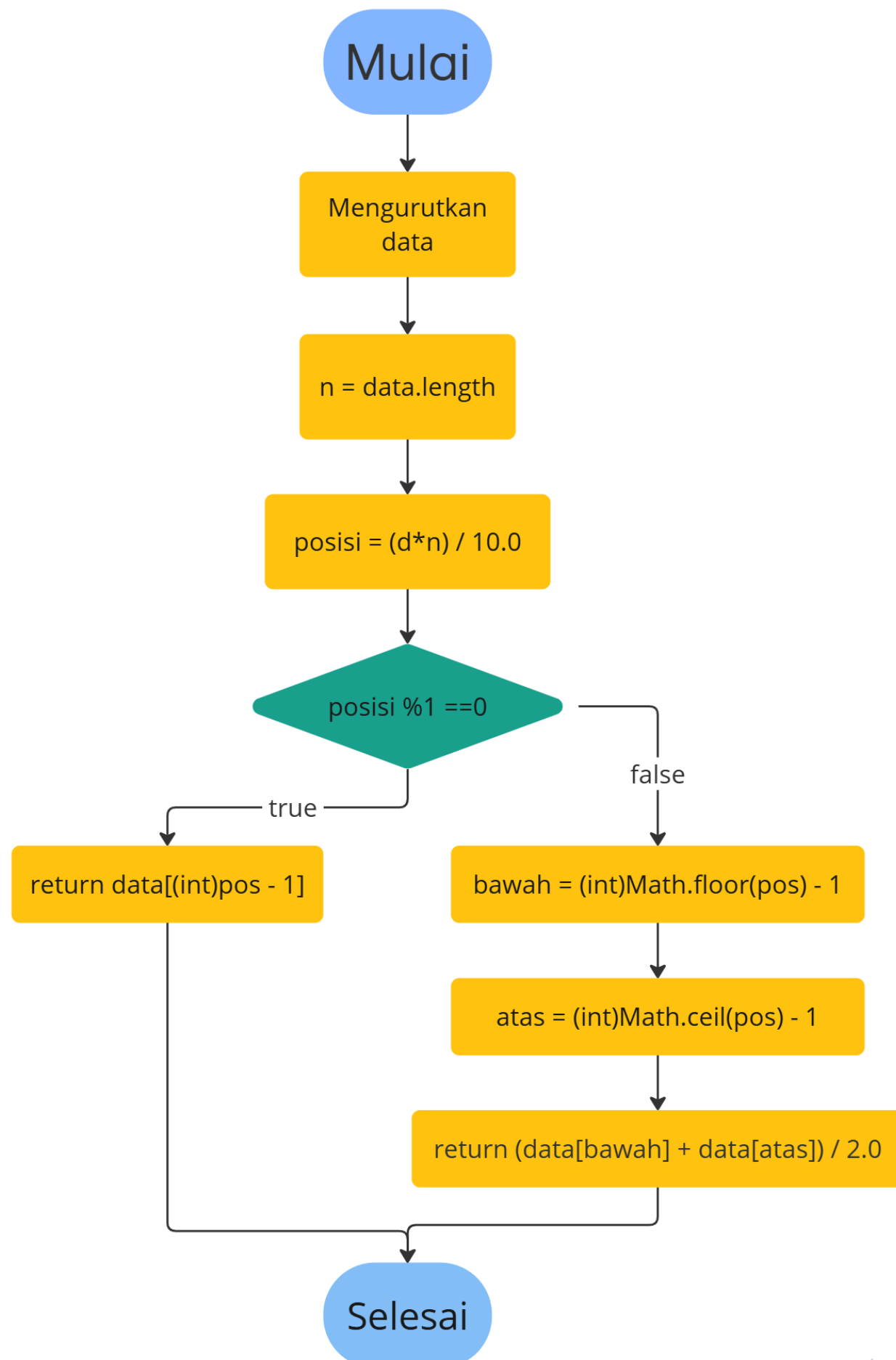


## Method Kuartil

```
static double kuartil (int k, double[] data){  
    java.util.Arrays.sort(data);  
    int n = data.length;  
    double pos = (k*n) / 4.0;  
  
    if (pos % 1 == 0) {  
        return data[(int)pos - 1];  
    } else {  
        int bawah = (int)Math.floor(pos) - 1;  
        int atas = (int)Math.ceil(pos) - 1;  
        return (data[bawah] + data[atas]) / 2.0;  
    }  
}
```







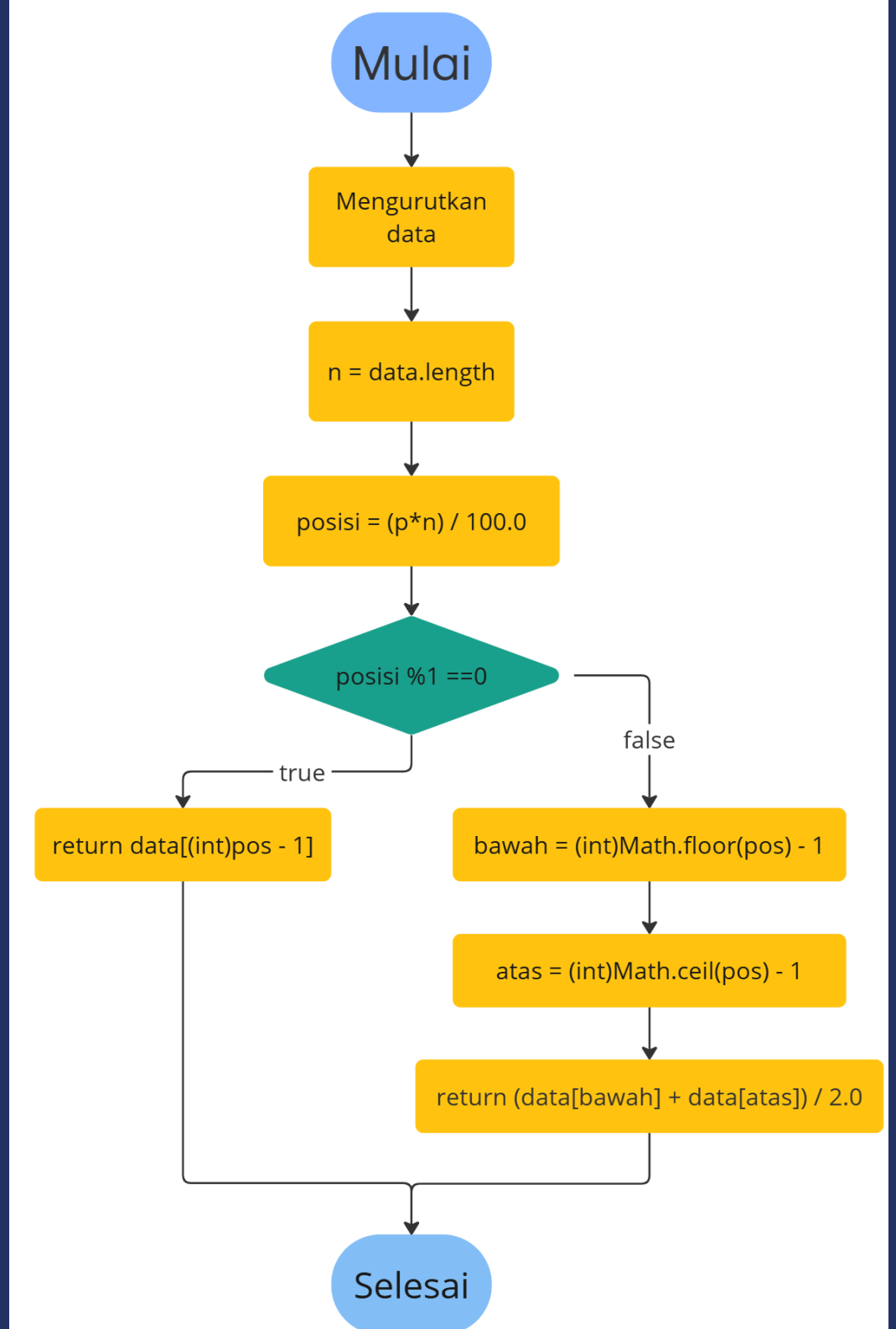
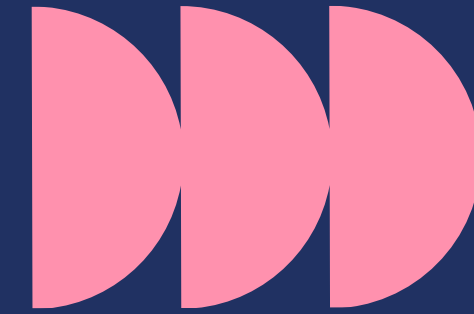
## Method Desil

```
static double desil(int d, double[] data) {
    java.util.Arrays.sort(data);
    int n = data.length;
    double pos = (d*n) / 10.0;

    if (pos % 1 == 0) {
        return data[(int)pos - 1];
    } else {
        int bawah = (int)Math.floor(pos) - 1;
        int atas = (int)Math.ceil(pos) - 1;
        return (data[bawah] + data[atas]) / 2.0;
    }
}
```

## Method Persentil

```
static double persentil(int p, double[] data) {  
    java.util.Arrays.sort(data);  
    int n = data.length;  
    double pos = (p*n) / 100.0;  
  
    if (pos % 1 == 0) {  
        return data[(int)pos - 1];  
    } else {  
        int lower = (int)Math.floor(pos) - 1;  
        int upper = (int)Math.ceil(pos) - 1;  
        return (data[lower] + data[upper]) / 2.0;  
    }  
}
```



# Output





Terima Kasih



funfact

ada  
**45**  
perulangan dan  
percabangan







<https://intip.in/MariBuka>

