

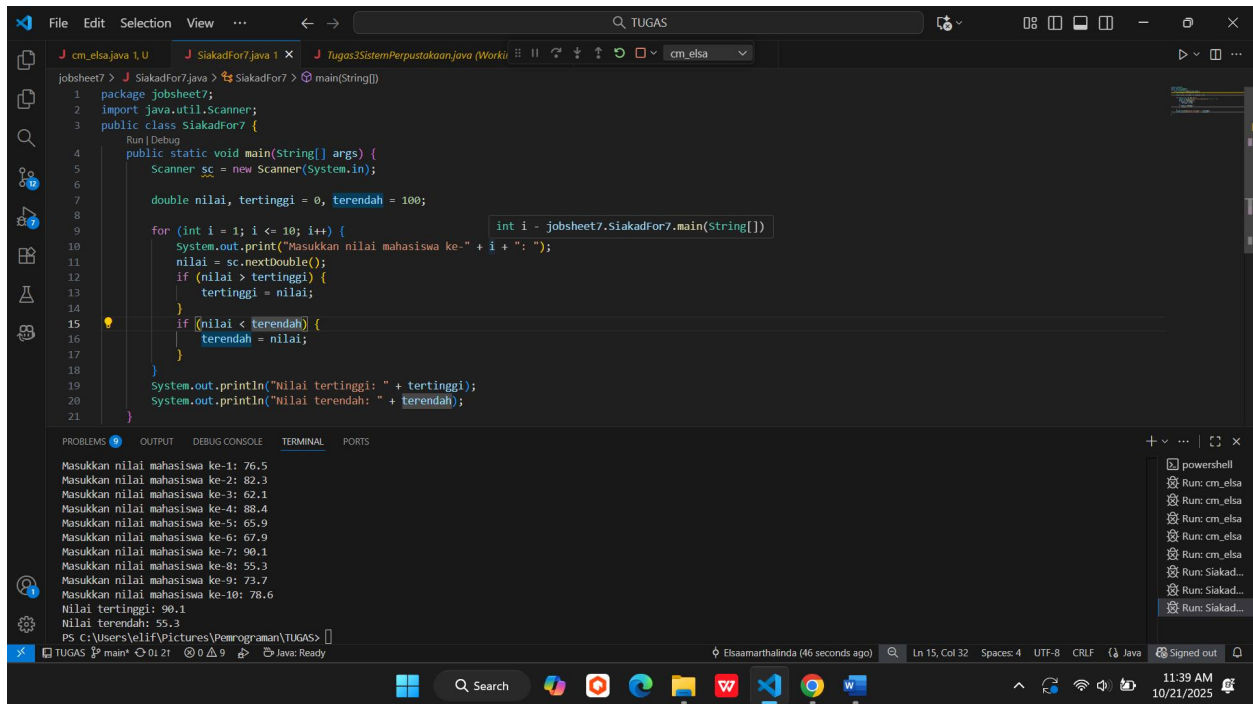
Nama : Elsa Marthalinda

NIM : 254107020204

Prodi : D-IV Teknik Informatika

Kelas : 1G

## PERCOBAAN 1



```
1 package jobsheet7;
2 import java.util.Scanner;
3 public class SiakadFor7 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         double nilai, tertinggi = 0, terendah = 100;
8
9         for (int i = 1; i <= 10; i++) {
10             System.out.print("Masukkan nilai mahasiswa ke-" + i + ": ");
11             nilai = sc.nextDouble();
12             if (nilai > tertinggi) {
13                 tertinggi = nilai;
14             }
15             if (nilai < terendah) {
16                 terendah = nilai;
17             }
18         }
19         System.out.println("Nilai tertinggi: " + tertinggi);
20         System.out.println("Nilai terendah: " + terendah);
21     }
22 }
```

Masukkan nilai mahasiswa ke-1: 76.5  
Masukkan nilai mahasiswa ke-2: 82.3  
Masukkan nilai mahasiswa ke-3: 62.1  
Masukkan nilai mahasiswa ke-4: 88.4  
Masukkan nilai mahasiswa ke-5: 65.9  
Masukkan nilai mahasiswa ke-6: 67.9  
Masukkan nilai mahasiswa ke-7: 90.1  
Masukkan nilai mahasiswa ke-8: 55.3  
Masukkan nilai mahasiswa ke-9: 73.7  
Masukkan nilai mahasiswa ke-10: 78.6  
Nilai tertinggi: 90.1  
Nilai terendah: 55.3

1. Sebutkan dan tunjukkan masing-masing komponen perulangan FOR pada kode program

Percobaan 1!

- Inisialisasi: **i = 1**
- Kondisi: **i <= 10, nilai > tertinggi, nilai < terendah**
- Update: **i++**

2. Mengapa **variabel** tertinggi diinisialisasi 0 dan **terendah** diinisialisasi 100? Apa yang terjadi jika variabel tertinggi diinisialisasi 100 dan terendah diinisialisasi 0?

- Inisialisasi tertinggi = 0 dan terendah = 100 digunakan agar perbandingan pertama **selalu memperbarui nilai** dengan input pertama.

- Jika dibalik (tertinggi = 100, terendah = 0), maka **program tidak akan menemukan nilai sebenarnya** karena logika perbandingan akan selalu salah di seluruh iterasi.

3. Jelaskan fungsi dan alur kerja dari potongan kode berikut!

```
if (nilai > tertinggi) {  
    tertinggi = nilai;  
}  
if (nilai < terendah) {  
    terendah = nilai;  
}
```

- Kedua if berfungsi sama yaitu dengan tujuan menemukan kondisi nilai tertinggi dan nilai terkecil.

```
if (nilai > tertinggi) {  
    tertinggi = nilai;  
}
```

- - Program bertanya: “Apakah nilai baru lebih besar dari **tertinggi**?”
  - Jika True: Program akan memperbarui nilai **tertinggi** dengan nilai tersebut.
  - Jika False: Program tidak akan memproses apa-apa dan tertinggi tetap.

```
if (nilai < terendah) {  
    terendah = nilai;  
}
```

- - Setelah pengecekan pertama, program lanjut mengecek ke bawah.
  - Program bertanya: “Apakah nilai baru lebih kecil dari **terendah**?”
  - Jika True: Program akan memperbarui nilai **terendah** dengan nilai tersebut
  - Jika False: Program tidak akan memproses apa-apa dan terendah tetap.

4. Modifikasi kode program sehingga terdapat perhitungan untuk menentukan berapa mahasiswa yang lulus dan yang tidak lulus berdasarkan batas kelulusan (nilai minimal 60). Tampilkan jumlah mahasiswa lulus dan tidak lulus setelah menampilkan nilai tertinggi dan terendah!

```
package jobsheet7;
import java.util.Scanner;
public class SiakadFor7 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        double nilai, tertinggi = 0, terendah = 100;
        int jumlahLulus = 0;
        int jumlahTidakLulus = 0;

        for (int i = 1; i <= 10; i++) {
            System.out.print("Masukkan nilai mahasiswa ke-" + i + ": ");
            nilai = sc.nextDouble();

            if (nilai > tertinggi) {
                tertinggi = nilai;
            }
        }

        System.out.println("Nilai tertinggi: " + tertinggi);
        System.out.println("Nilai terendah: " + terendah);
        System.out.println("Jumlah mahasiswa lulus: " + jumlahLulus);
        System.out.println("Jumlah mahasiswa tidak lulus: " + jumlahTidakLulus);
    }
}
```

Masukkan nilai mahasiswa ke-1: 70  
Masukkan nilai mahasiswa ke-2: 30  
Masukkan nilai mahasiswa ke-3: 40  
Masukkan nilai mahasiswa ke-4: 88.1  
Masukkan nilai mahasiswa ke-5: 50.3  
Masukkan nilai mahasiswa ke-6: 75  
Masukkan nilai mahasiswa ke-7: 69  
Masukkan nilai mahasiswa ke-8: 36  
Masukkan nilai mahasiswa ke-9: 90  
Masukkan nilai mahasiswa ke-10: 90  
Masukkan nilai mahasiswa ke-10: 26  
Nilai tertinggi: 90.0  
Nilai terendah: 26.0  
Jumlah mahasiswa lulus: 5  
Jumlah mahasiswa tidak lulus: 5

## 5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 1”



## PERCOBAAN 2

```
public class SiakadWhile7 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int jumlahMahasiswa = 20;
        int i = 1;

        while (i <= jumlahMahasiswa) {
            System.out.print("Masukkan nilai mahasiswa ke-" + i + ": ");
            double nilai = sc.nextDouble();

            if (nilai > 80 && nilai <= 100) {
                System.out.println("Nilai mahasiswa ke-" + i + " adalah A");
            } else if (nilai > 73 && nilai <= 80) {
                System.out.println("Nilai mahasiswa ke-" + i + " adalah B+");
            } else if (nilai > 65 && nilai <= 73) {
                System.out.println("Nilai mahasiswa ke-" + i + " adalah B");
            } else if (nilai > 60 && nilai <= 65) {
                System.out.println("Nilai mahasiswa ke-" + i + " adalah C+");
            } else if (nilai > 50 && nilai <= 60) {
                System.out.println("Nilai mahasiswa ke-" + i + " adalah C");
            } else if (nilai > 39 && nilai <= 50) {
                System.out.println("Nilai mahasiswa ke-" + i + " adalah D");
            } else {
                System.out.println("Nilai mahasiswa ke-" + i + " adalah E");
            }

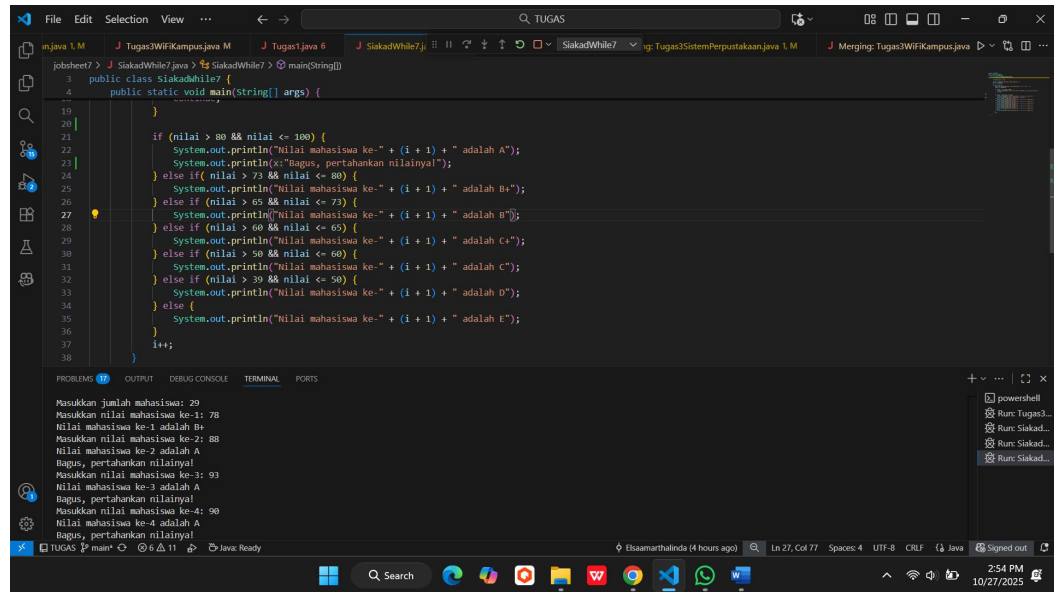
            i++;
        }
    }
}
```

Masukkan jumlah mahasiswa: 20  
Masukkan nilai mahasiswa ke-1: 85  
Nilai mahasiswa ke-1 adalah A  
Masukkan nilai mahasiswa ke-2: 63  
Nilai mahasiswa ke-2 adalah C+  
Masukkan nilai mahasiswa ke-3: 101  
Masukkan nilai mahasiswa ke-2: 63  
Masukkan nilai mahasiswa ke-2: 63  
Nilai mahasiswa ke-2 adalah C+  
Masukkan nilai mahasiswa ke-3: 101  
Nilai tidak valid. Masukkan lagi nilai yang valid!  
Masukkan nilai mahasiswa ke-3: 73  
Nilai mahasiswa ke-3 adalah E  
Masukkan nilai mahasiswa ke-4: -15  
Nilai tidak valid. Masukkan lagi nilai yang valid!  
Masukkan nilai mahasiswa ke-4: 70  
Nilai mahasiswa ke-4 adalah B  
Masukkan nilai mahasiswa ke-5: 55  
Nilai mahasiswa ke-5 adalah C

1. Pada potongan kode berikut, tentukan maksud dan kegunaan dari sintaks berikut:

```
if (nilai < 0 || nilai > 100) {  
    System.out.println(x:"Nilai tidak valid. Masukkan lagi nilai yang valid!");  
    continue;  
}
```

- **nilai < 0 || nilai > 100:** Pemilihan IF untuk mengecek valid atau tidaknya nilai yang dimasukkan, apakah nilai yang dimasukkan kurang dari 0 OR nilai lebih dari 100.  
Kegunaan: Validasi Input
  - **b. continue:** Digunakan untuk melewati (skip) iterasi saat ini (1 iterasi saja).
2. Mengapa sintaks **i++** dituliskan di akhir perulangan WHILE? Apa yang terjadi jika posisinya dituliskan di awal perulangan WHILE?
- Ditulis di akhir: berfungsi sebagai update (increment). Ia harus dijalankan setelah semua proses (input, validasi, penentuan nilai huruf) untuk iterasi saat ini selesai. Ini memastikan bahwa counter (i) bertambah hanya jika iterasi tersebut berhasil (yaitu, input nilai valid telah diterima).
  - Ditulis di awal: (**i++**) akan bertambah nilainya sebelum proses input selesai.
3. Apabila jumlah mahasiswa yang dimasukkan adalah 19, berapa kali perulangan WHILE akan berjalan?
- Perulangan **while** akan berjalan sebanyak 19 kali, karena i dimulai dari 0 dan kondisi perulangan adalah  $i < \text{jumlahMahasiswa}$  (yaitu  $i < 19$ ).
4. Modifikasi kode program sehingga apabila terdapat mahasiswa yang mendapat nilai A, program menampilkan pesan tambahan "Bagus, pertahankan nilainya"!



```
3 public class SiakadWhile7 {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6         int jumlahMahasiswa;
7         int i = 1;
8         while (true) {
9             System.out.println("Masukkan jumlah mahasiswa:");
10            jumlahMahasiswa = scanner.nextInt();
11            if (jumlahMahasiswa < 1) {
12                System.out.println("Masukkan jumlah mahasiswa yang valid!");
13                continue;
14            }
15            while (i <= jumlahMahasiswa) {
16                System.out.println("Masukkan nilai mahasiswa ke-" + i + ":");
17                int nilaiMahasiswa = scanner.nextInt();
18                if (nilaiMahasiswa < 0 || nilaiMahasiswa > 100) {
19                    System.out.println("Masukkan nilai mahasiswa yang valid!");
20                    continue;
21                }
22                if (nilaiMahasiswa >= 80 && nilaiMahasiswa <= 100) {
23                    System.out.println("Nilai mahasiswa ke-" + i + " adalah A");
24                } else if (nilaiMahasiswa >= 73 && nilaiMahasiswa <= 80) {
25                    System.out.println("Nilai mahasiswa ke-" + i + " adalah B+");
26                } else if (nilaiMahasiswa >= 65 && nilaiMahasiswa <= 73) {
27                    System.out.println("Nilai mahasiswa ke-" + i + " adalah B");
28                } else if (nilaiMahasiswa >= 60 && nilaiMahasiswa <= 65) {
29                    System.out.println("Nilai mahasiswa ke-" + i + " adalah C+");
30                } else if (nilaiMahasiswa >= 50 && nilaiMahasiswa <= 60) {
31                    System.out.println("Nilai mahasiswa ke-" + i + " adalah C");
32                } else if (nilaiMahasiswa >= 39 && nilaiMahasiswa <= 50) {
33                    System.out.println("Nilai mahasiswa ke-" + i + " adalah D");
34                } else {
35                    System.out.println("Nilai mahasiswa ke-" + i + " adalah E");
36                }
37                i++;
38            }
39            System.out.println("Masukkan nilai mahasiswa ke-" + i + ":");
40            i = 1;
41        }
42    }
43 }
```

Masukkan jumlah mahasiswa: 29  
Masukkan nilai mahasiswa ke-1: 78  
Nilai mahasiswa ke-1 adalah B+  
Masukkan nilai mahasiswa ke-2: 88  
Nilai mahasiswa ke-2 adalah A  
Bagus, pertahankan nilainya!  
Masukkan nilai mahasiswa ke-3: 93  
Nilai mahasiswa ke-3 adalah A  
Bagus, pertahankan nilainya!  
Masukkan nilai mahasiswa ke-4: 90  
Nilai mahasiswa ke-4 adalah A  
Bagus, pertahankan nilainya!

## 5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 2”



## PERCOBAAN 3

```
public class KafeDoWhile7 {
    public static void main(String[] args) {
        do {
            System.out.print("Masukkan nama pelanggan (ketik 'batal' untuk keluar): ");
            namaPelanggan = sc.nextLine();
            if (namaPelanggan.equalsIgnoreCase("batal")) {
                System.out.println("Transaksi dibatalkan.");
                break;
            }
            System.out.print("Jumlah kopi: ");
            kopi = sc.nextInt();
            System.out.print("Jumlah teh: ");
            teh = sc.nextInt();
            System.out.print("Jumlah roti: ");
            roti = sc.nextInt();
            totalHarga = (kopi * hargaKopi) + (teh * hargaTeh) + (roti * hargaRoti);
            System.out.println("Total yang harus dibayar: Rp " + totalHarga);
            sc.nextLine();
        } while (true);

        System.out.println("Semua transaksi selesai.");
    }
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Masukkan nama pelanggan (ketik 'batal' untuk keluar): Rena  
Jumlah kopi: 3  
Jumlah teh: 0  
Jumlah roti: 1  
Total yang harus dibayar: Rp 56000  
Masukkan nama pelanggan (ketik 'batal' untuk keluar): Yuni  
Jumlah kopi: 1  
Jumlah teh: 4  
Jumlah roti: 2  
Total yang harus dibayar: Rp 80000  
Masukkan nama pelanggan (ketik 'batal' untuk keluar): BATAL  
Transaksi dibatalkan.  
Semua transaksi selesai.

1. Pada penggunaan DO-WHILE ini, apabila nama pelanggan yang dimasukkan pertama kali adalah “batal”, maka berapa kali perulangan dilakukan?
  - Perulangan akan dilakukan 1 kali, karena pada perulangan **DO-WHILE**, blok kode di dalam do {} akan dieksekusi **minimal satu kali** sebelum kondisi while diperiksa.
2. Sebutkan kondisi berhenti yang digunakan pada perulangan DO-WHILE tersebut!
  - Kondisi **break**: Ketika pengguna memasukkan nama pelanggan "**batal**" (tidak peduli huruf besar/kecil). Kode break; akan menghentikan perulangan secara paksa.
3. Apa fungsi dari penggunaan nilai **true** pada kondisi DO-WHILE?
  - Fungsi dari penggunaan while (true) adalah untuk menciptakan perulangan tak terbatas (infinite loop) secara sengaja. Ini berarti kondisi while akan selalu bernilai benar (true) sehingga perulangan tidak akan pernah berhenti secara alami.
4. Mengapa perulangan DO-WHILE tersebut tetap berjalan meskipun tidak ada komponen inisialisasi dan update?

- Perulangan ini tetap berjalan karena fungsi utamanya bukan untuk menghitung iterasi, melainkan untuk Melakukan pemrosesan transaksi dan Menunggu kondisi eksternal (input "batal") untuk menghentikannya.

## TUGAS:

```
File Edit Selection View ... TUGAS
Tugas1.java 1, M X
Tugas1.java 1, U
SiakadWhite7.java 1
Tugas1 4
gas3SistemPerpustakaan.java 1, M
Merging: Tugas3WfKampus.java M

11 public class Tugas1 {
12     public static void main(String[] args) {
13         while (true) {
14             System.out.print("Masukkan jumlah tiket yang dibeli (ketik 0 untuk selesai): ");
15             jmlTiket = sc.nextInt();
16             if (jmlTiket == 0) break;
17             if (jmlTiket < 0) {
18                 System.out.println("Jumlah tiket tidak valid. Input ulang!");
19                 continue;
20             }
21             diskon = 0;
22             if (jmlTiket > 10) diskon = 0.15;
23             else if (jmlTiket > 4) diskon = 0.10;
24             total = jmlTiket * tiket * (1 - diskon);
25             totalPenjualan += total;
26             totalTiket += jmlTiket;
27         }
28     }
29 }
30
31
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Masukkan jumlah tiket yang dibeli (ketik 0 untuk selesai): 5
----- LAPORAN PENJUALAN -----
Total tiket terjual: 5
Total pendapatan : Rp 225000.0
Total harga: Rp 225000.0
Masukkan jumlah tiket yang dibeli (ketik 0 untuk selesai): -3
Jumlah tiket tidak valid. Input ulang!
Masukkan jumlah tiket yang dibeli (ketik 0 untuk selesai): 0
PS C:\Users\elif\Pictures\Programan\TUGAS>
TUGAS main 0 0 13 Java Ready
Ln 21, Col 14 Spaces: 4 UTF-8 CRLF Java Signed out
6:17 PM 10/27/2025
```

1.

2.

```
File Edit Selection View ... TUGAS
Tugas1.java Tugas2.java cm_elo.java 1.U SiakadWhile7.java 1 KafeDoWhile7.java 1 Merging: Tugas3SistemPerpustakaan.java 1.M Merging: Tugas3WiFiKampus.java M
jobsheet7
5 public class Tugas2 {
6     public static void main(String[] args) {
7
8     }
9     do {
10        System.out.print(s:"Masukkan jenis kendaraan (1 = Mobil, 2 = Motor, 0 = Keluar): ");
11        jenis = sc.nextInt();
12
13        if (jenis == 1 || jenis == 2) {
14            System.out.print(s:"Masukkan durasi: ");
15            durasi = sc.nextInt();
16            if (durasi > 5) {
17                total += 12500;
18            } else if (jenis == 1) {
19                total += durasi * 3000;
20            } else if (jenis == 2) {
21                total += durasi * 2000;
22            }
23        }
24        while (jenis != 0);
25        System.out.print("Total pembayaran parkir: Rp" + total);
26        sc.close();
27    }
28 }
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\elif\Pictures\Pemrograman\TUGAS> & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\elif\AppData\Roaming\Code\User\workspaceStorage\0da0eca37a02f6f40d7c77de6f577fc2\redhat.java\jdt_ws\TUGAS_8bf3a14\bin' 'jobsheet7.Tugas2'
Masukkan jenis kendaraan (1 = Mobil, 2 = Motor, 0 = Keluar): 1
Masukkan durasi: 3
Masukkan jenis kendaraan (1 = Mobil, 2 = Motor, 0 = Keluar): 0
Total pembayaran parkir: Rp9000
PS C:\Users\elif\Pictures\Pemrograman\TUGAS> ^C
PS C:\Users\elif\Pictures\Pemrograman\TUGAS>
PS C:\Users\elif\Pictures\Pemrograman\TUGAS> cd 'C:\Users\elif\Pictures\Pemrograman\TUGAS'; & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\elif\AppData\Roaming\Code\User\workspaceStorage\0da0eca37a02f6f40d7c77de6f577fc2\redhat.java\jdt_ws\TUGAS_8bf3a14\bin' 'jobsheet7.Tugas2'
Masukkan jenis kendaraan (1 = Mobil, 2 = Motor, 0 = Keluar): 2
Masukkan durasi: 6
Masukkan jenis kendaraan (1 = Mobil, 2 = Motor, 0 = Keluar): 0
Total pembayaran parkir: Rp12500
PS C:\Users\elif\Pictures\Pemrograman\TUGAS>
TUGAS main 0 13 Java Ready
Ln 21, Col 44 Spaces: 4 UTF-8 CRLF Java Signed out
8:10 PM 10/27/2025
```

## Tugas 2

Elsaamarthalinda committed now

f616b11