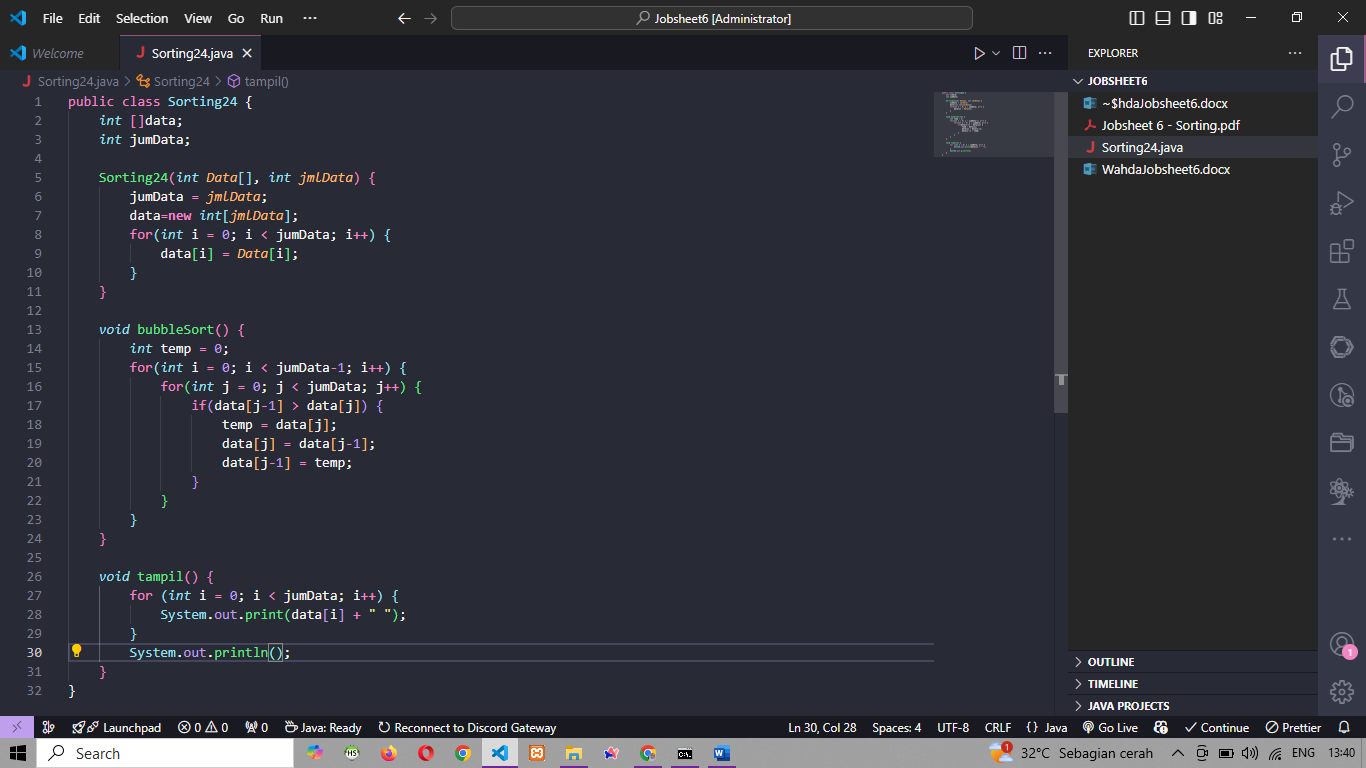
Nama: Wahda Adella Putri Febriana

Kelas: 1B / 24

NIM: 2440700156

**SORTING – BUBBLE SORT**

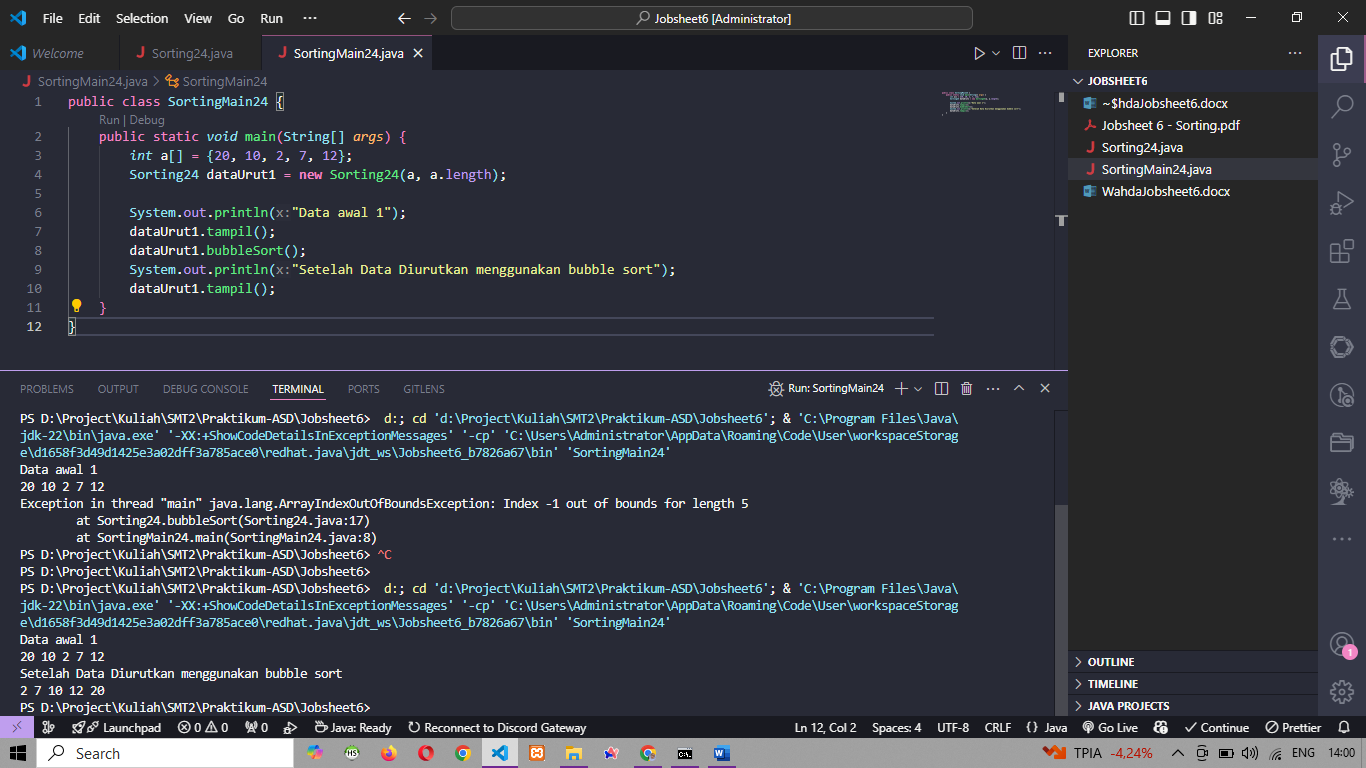
1. Buat file sorting24, beserta attr, konstruktor dan method nya



1. Buat SortingMain24.java dan panggil class sorting

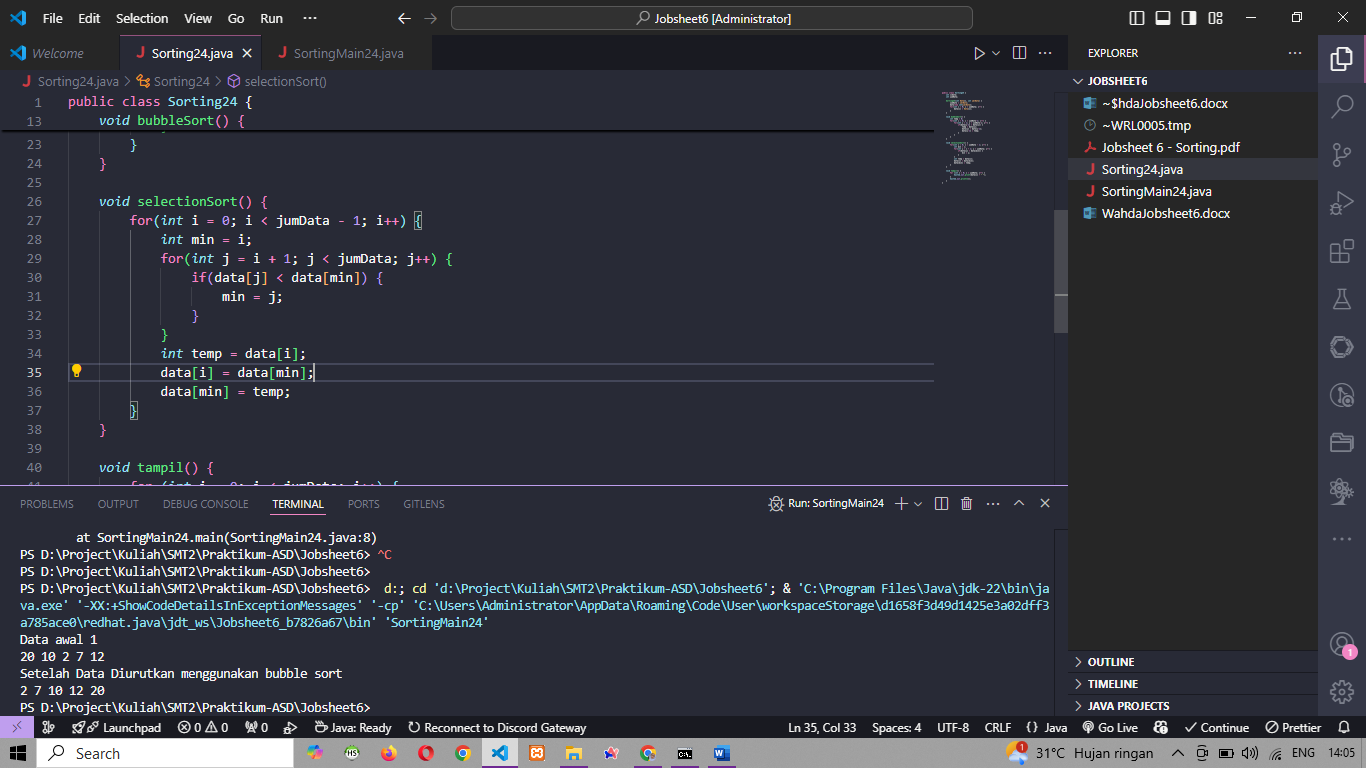


1. Hasil run

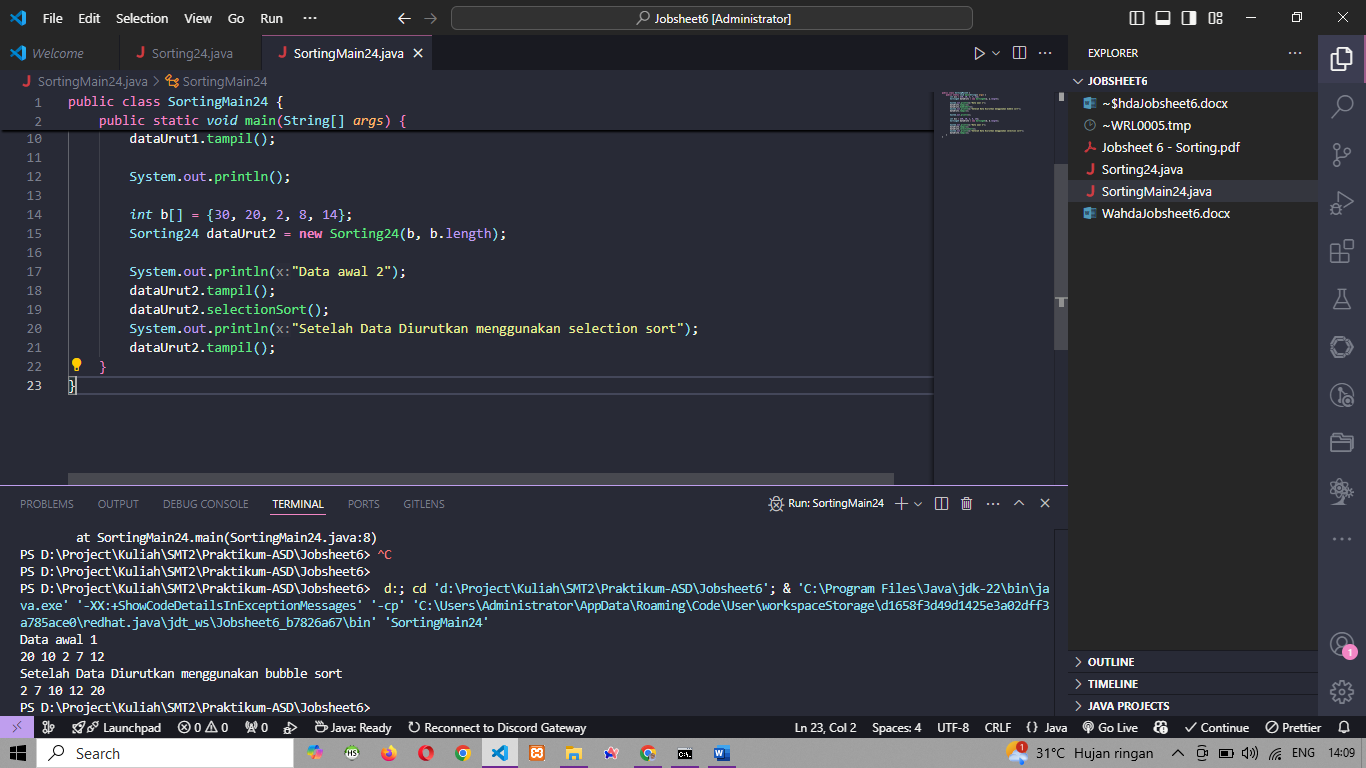


**SORTING – BUBBLE SORT**

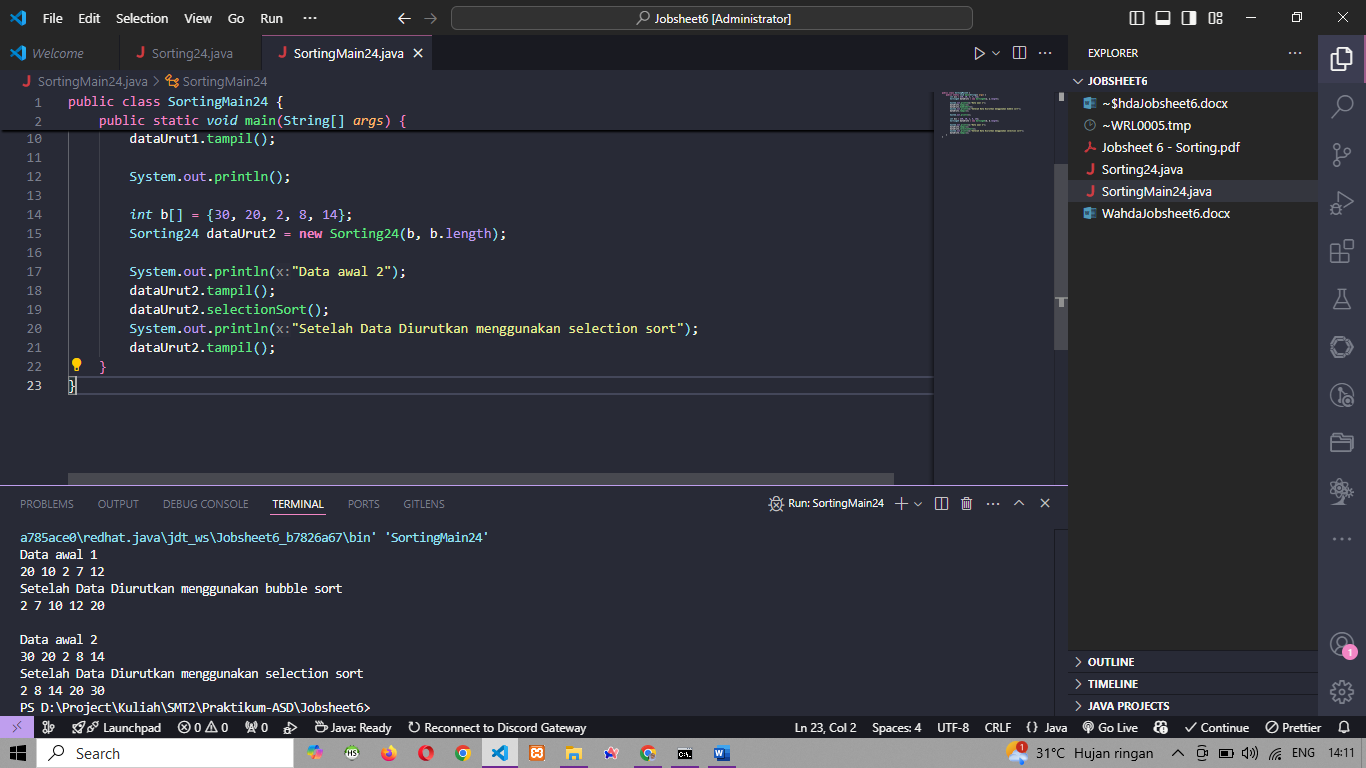
1. Tambahkan method baru Bernama selectionSort



1. Tambahkan kodingan ini di sortingmain24.java



1. Hasil setelah di run

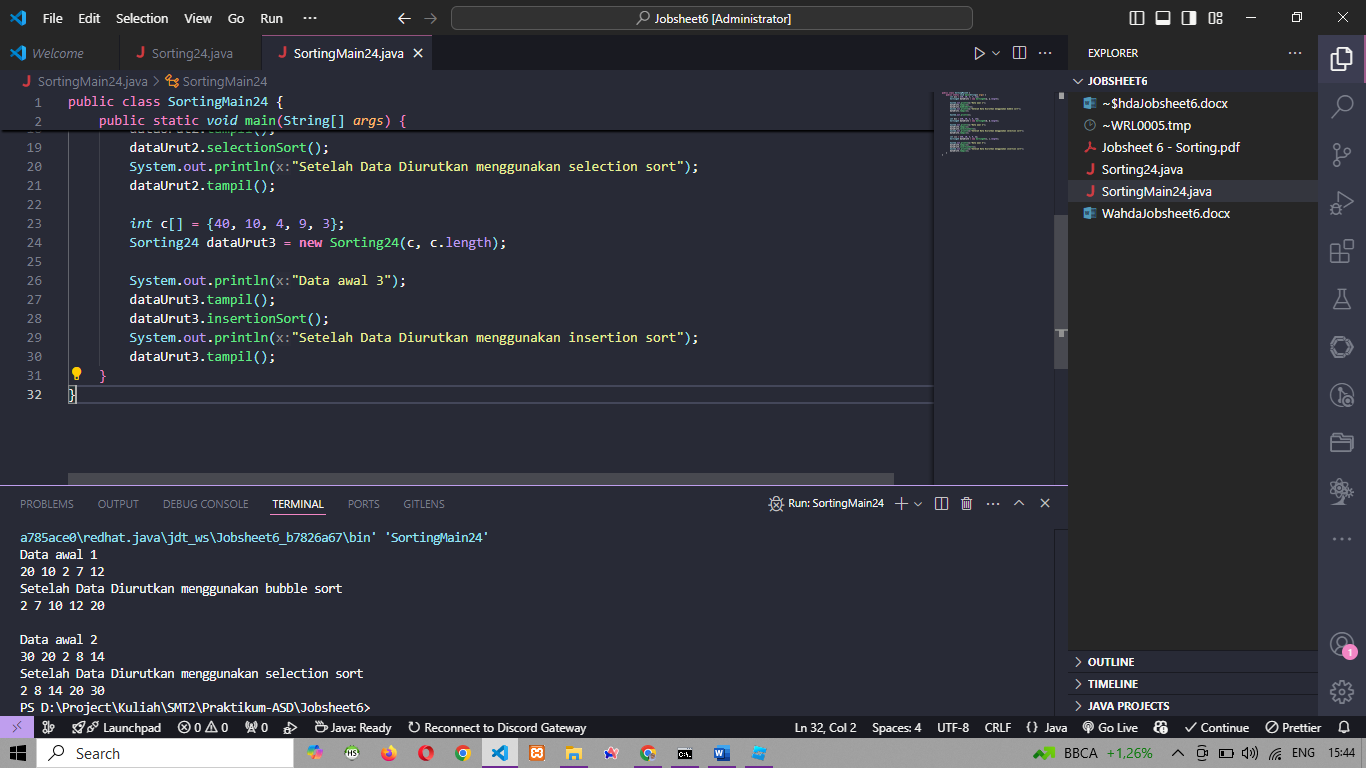


**SORTING – INSERTION SORT**

1. Tambahkan function insertionSort di sorting24.java



1. Panggil functionnya di sortingmain24.java



1. Hasil Run



**PERTANYAAN**

1. Jika data[j-1] lebih besar dari data[j], maka temp akan menyimpan data[j] lalu data[j] value nya akan direplace dengan data[j-1], lalu data[j-1] akan di replace value nya dengan temp
2. Pada bagian ini

int min = i;

for(int j = i + 1; j < jumData; j++) {

if(data[j] < data[min]) {

min = j;

}

}

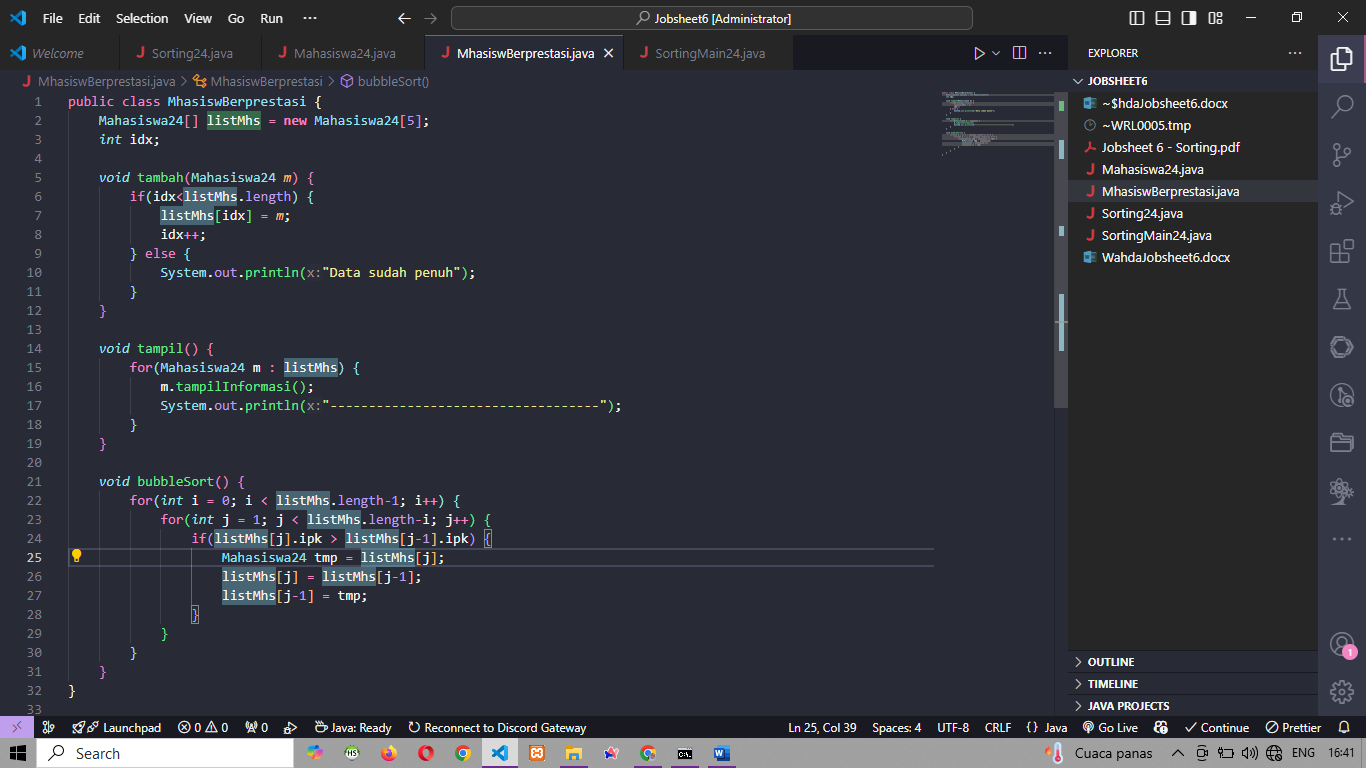
1. Jika j masi lebih besar sama dengan 0 dan data[j] lebih besar dari temp maka akan masuk kedalam if
2. Mereplace value data[j+1] dengan data[j]

**PERCOBAAN 2**

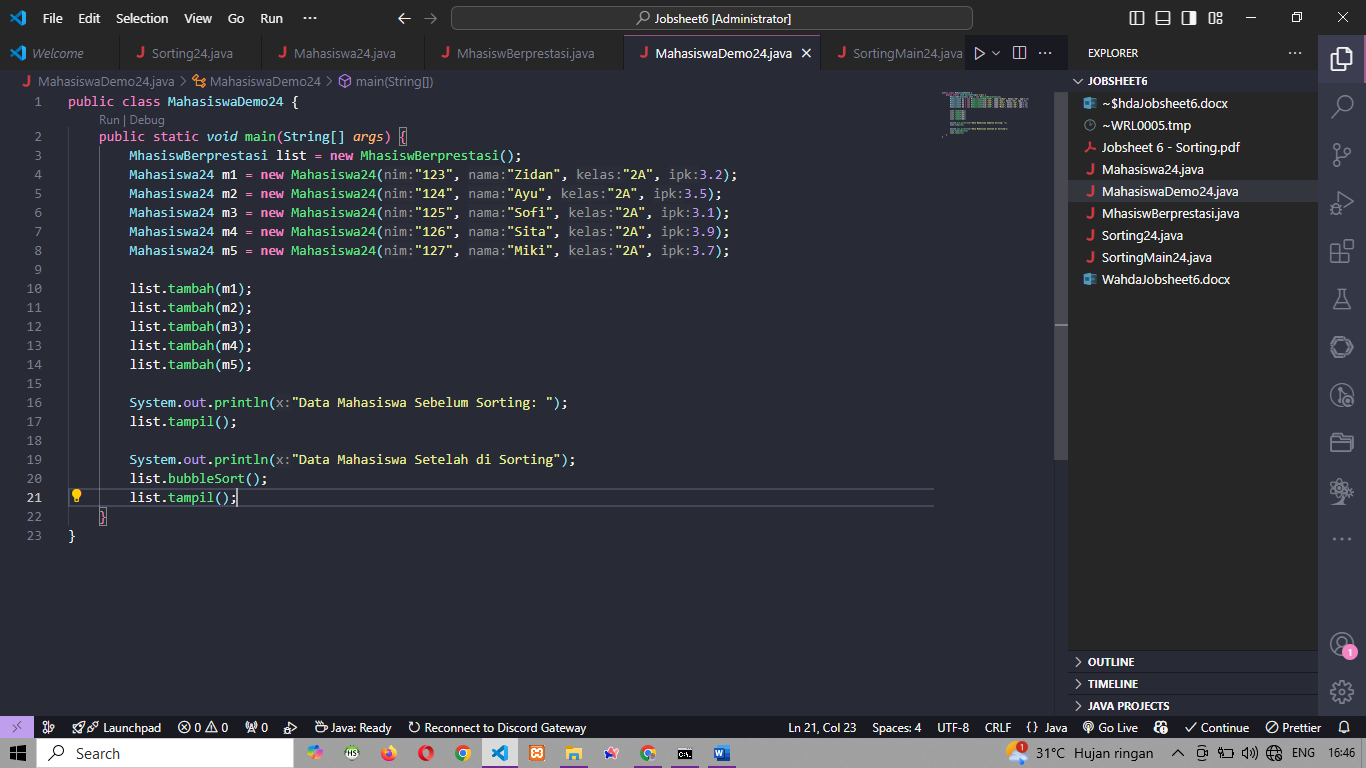
1. Buatkan file Mahasiswa24.java dan isi code nya



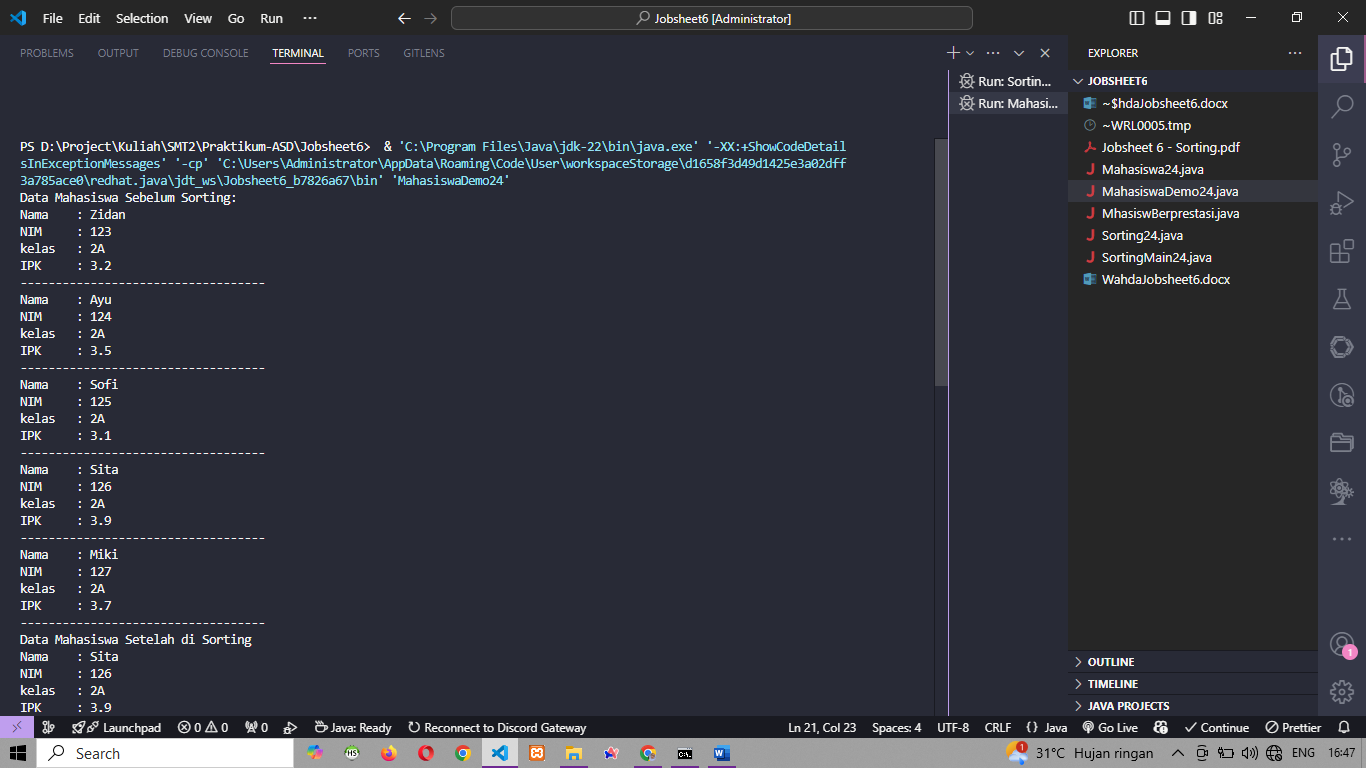
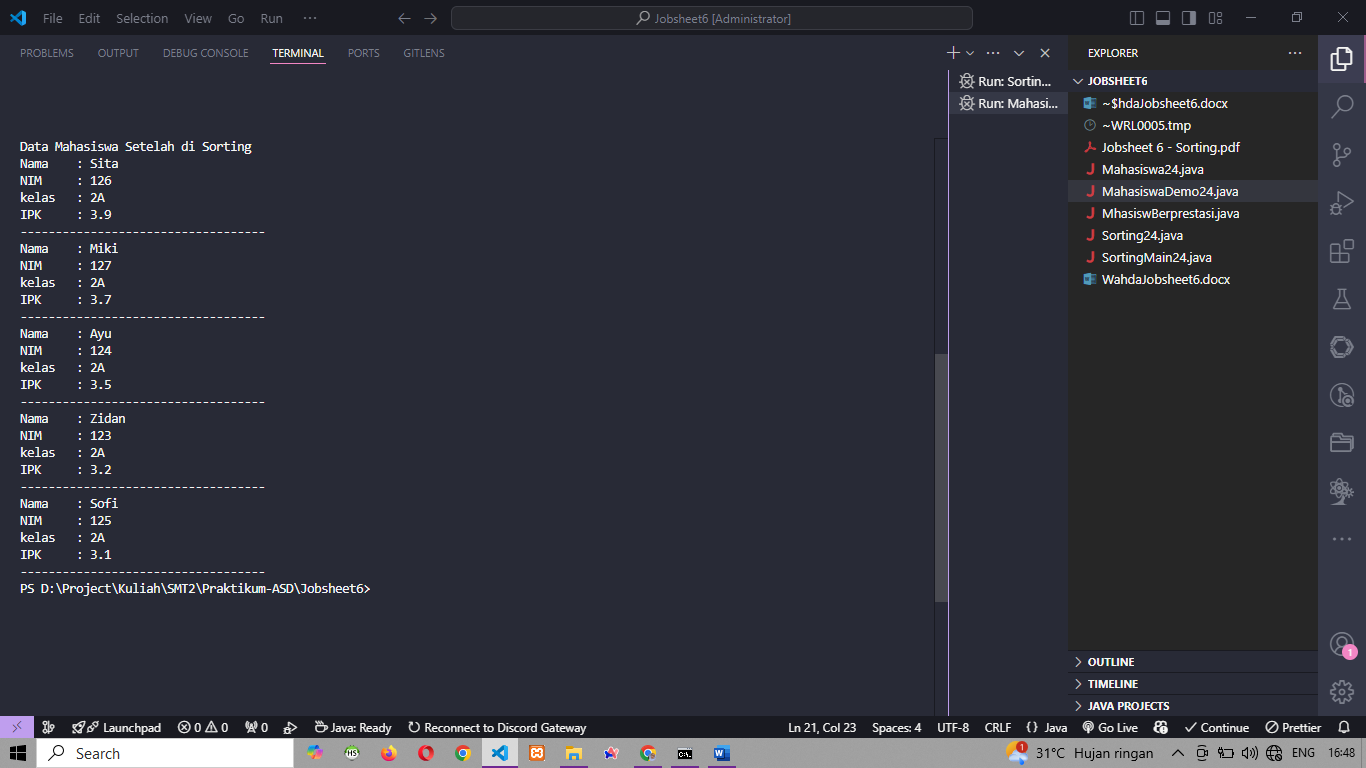
1. Buat file MahasiswaBerprestasi24.java dan dan isi filenya



1. Buat file MahasiswaDemo24.java dan code nya

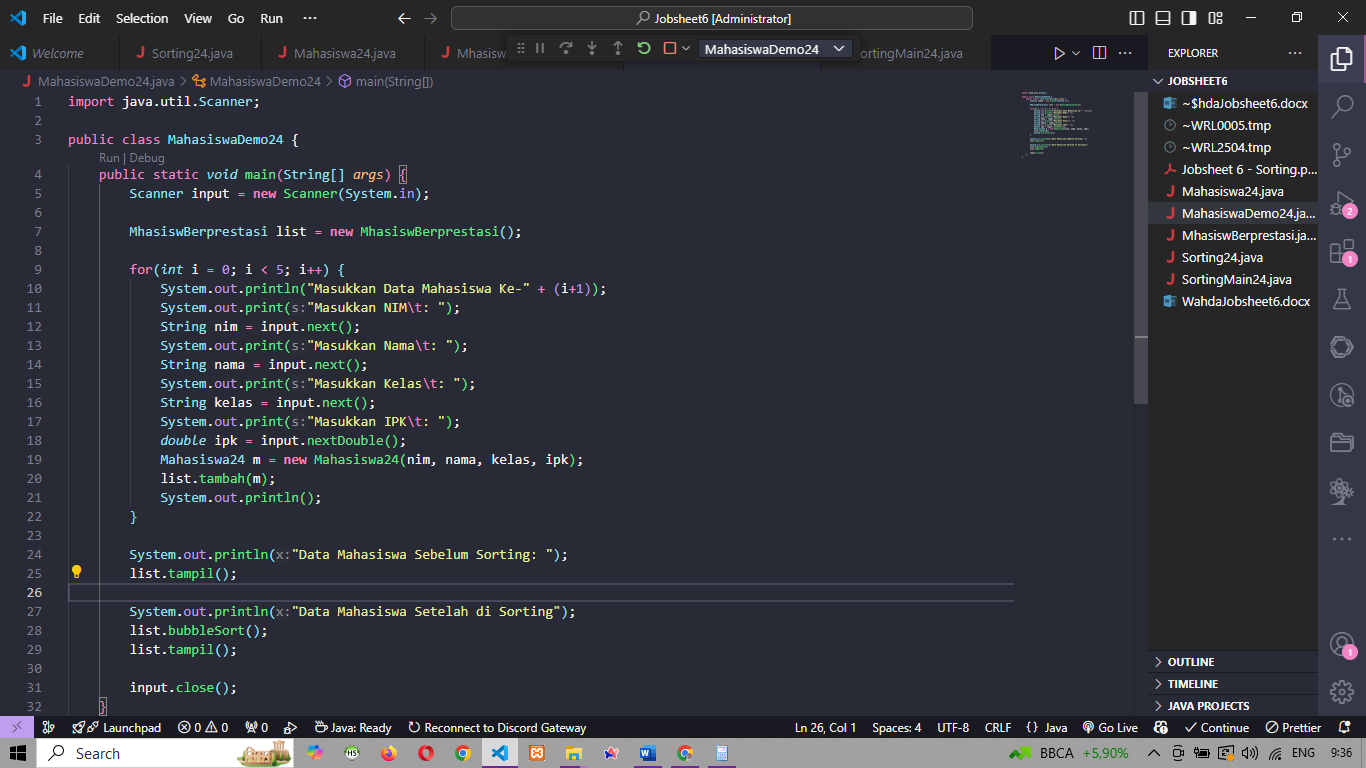


1. Hasil run

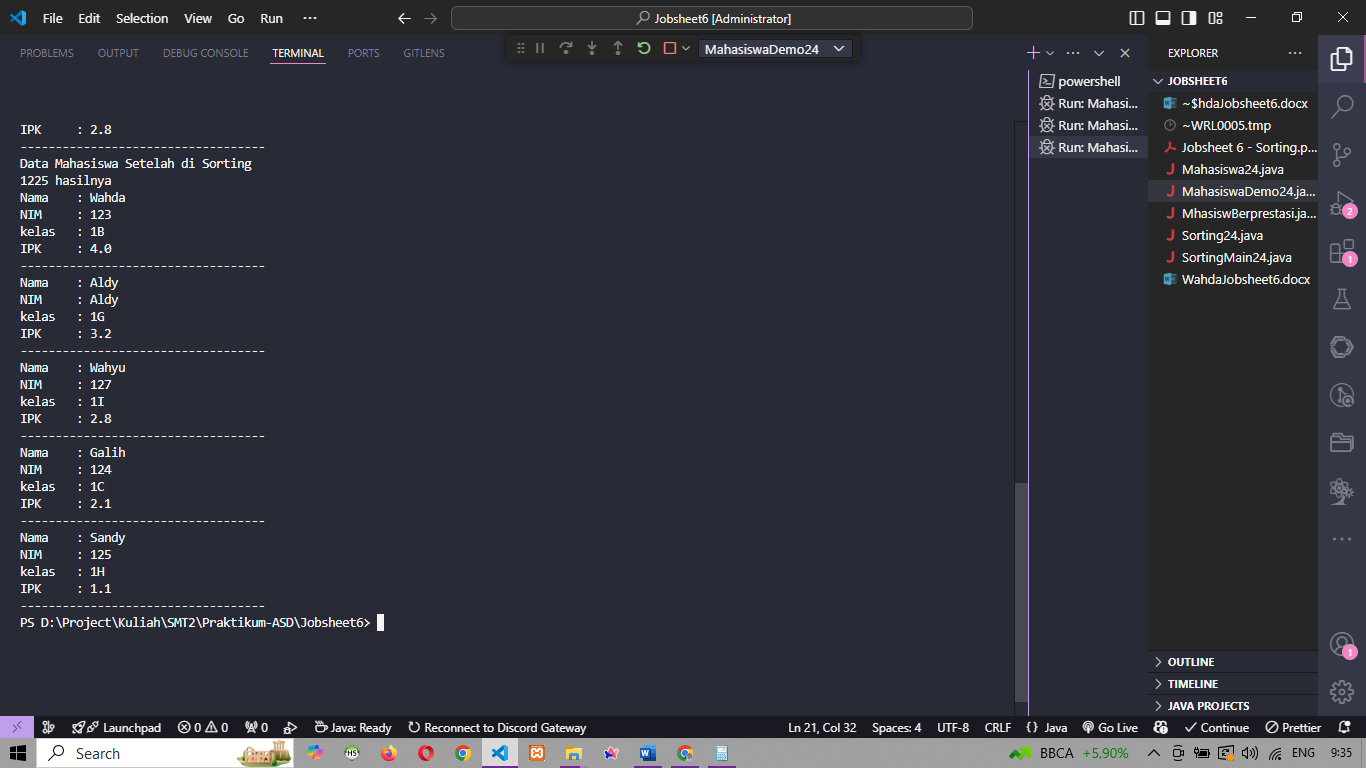
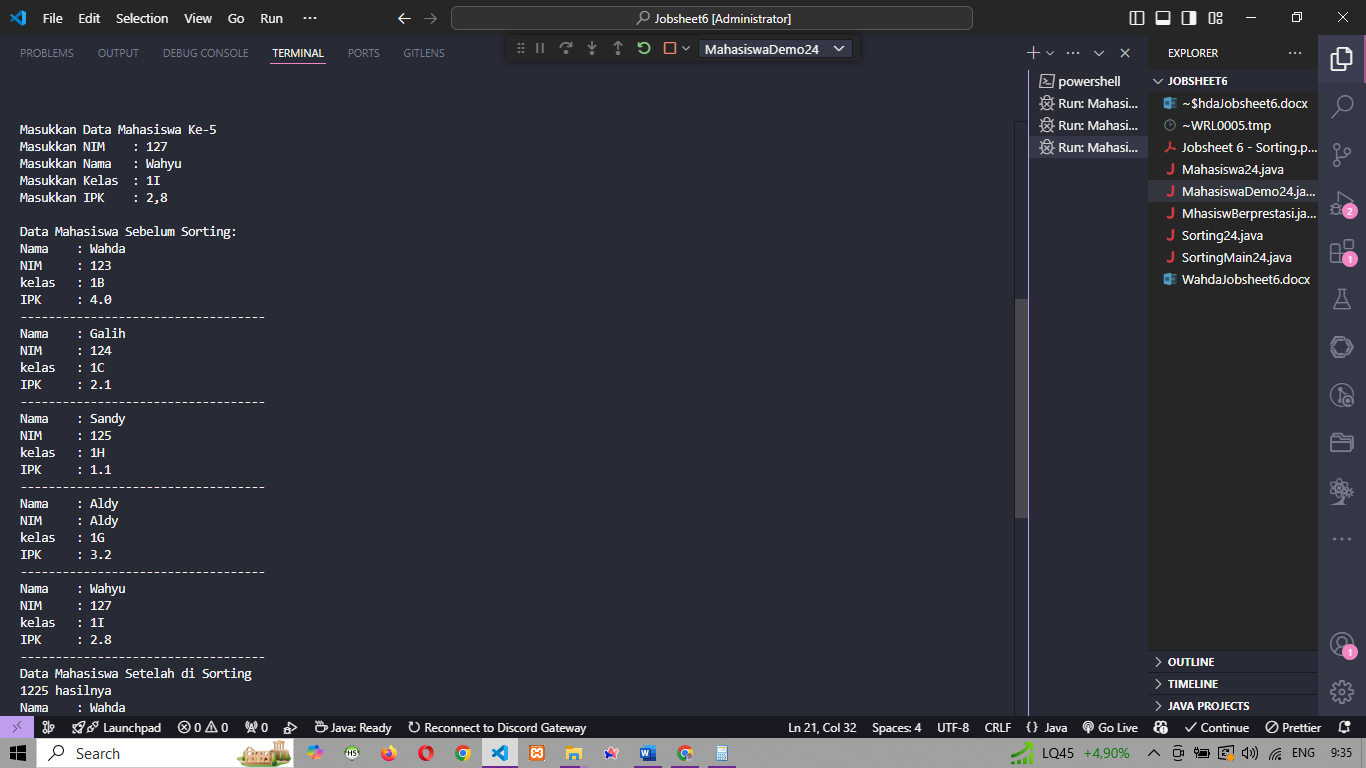
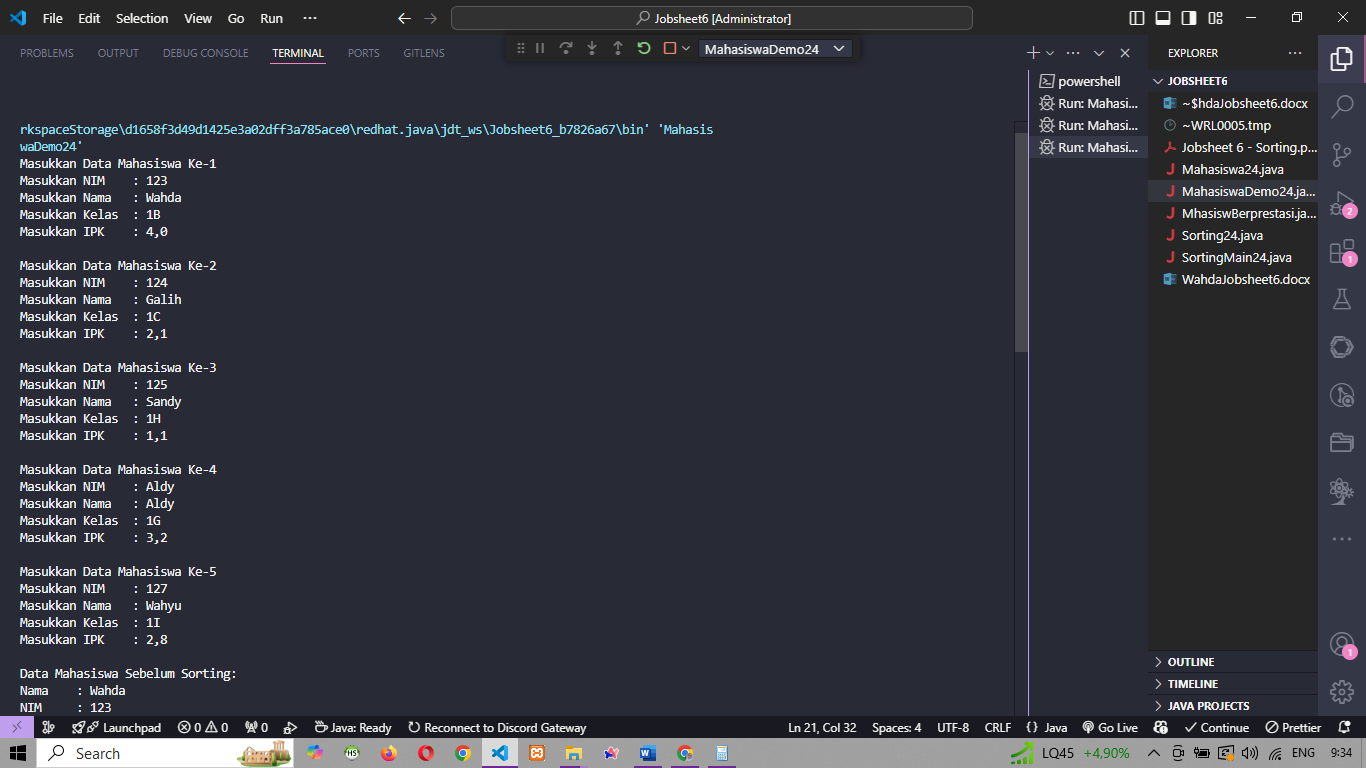
 

**PERTANYAAN**

1. Karena bubble sort menggunakan perulangan n-1
2. Untuk menghindari perulnagan yang tidak perlu
3. I nya akan dijalankan sebanyak n-1 = 50 -1 = 49x. Total tahap bubble sort yng ditempuh = n(n-1)/2 = 50(50-1)/2 = 50(49)/2 = 2450/2 = 1225x
4. Modifikasi Code

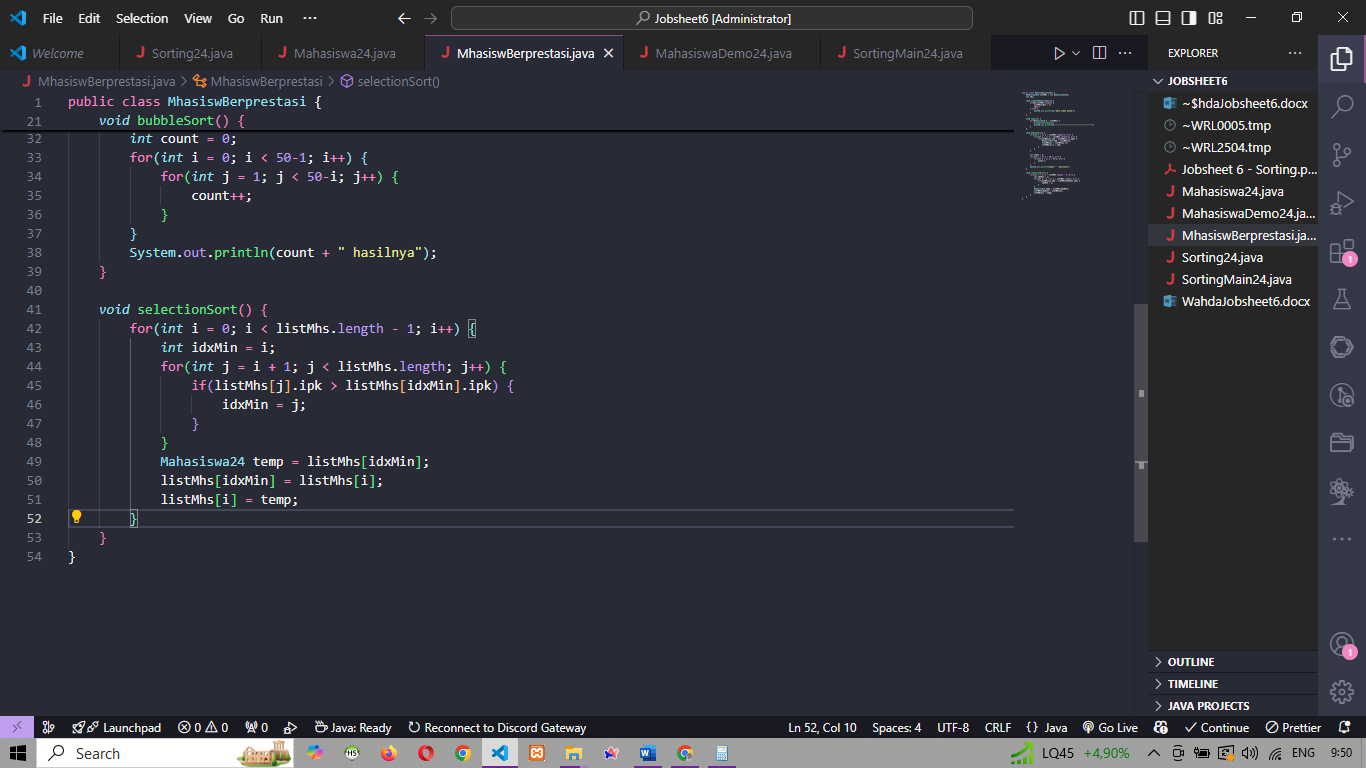


Hasil Run

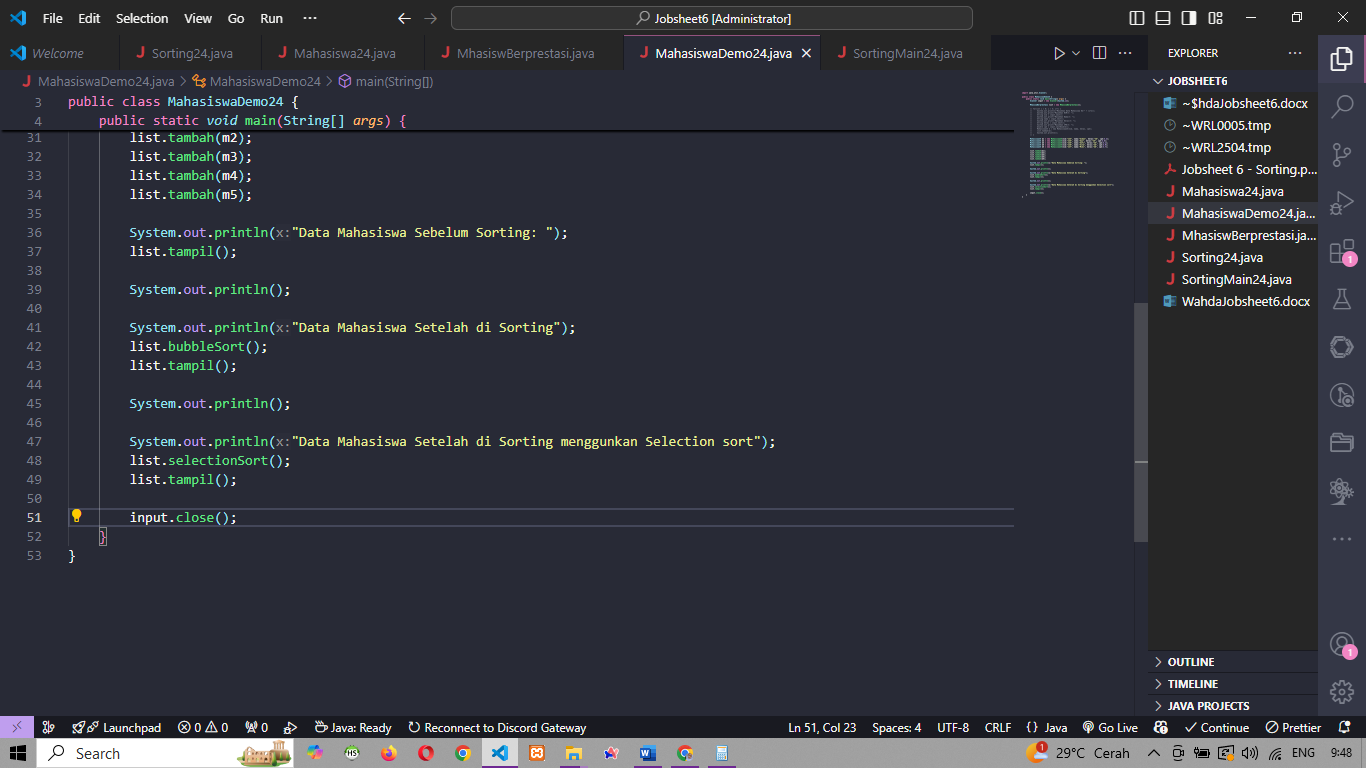


**PERCOBAAN 3**

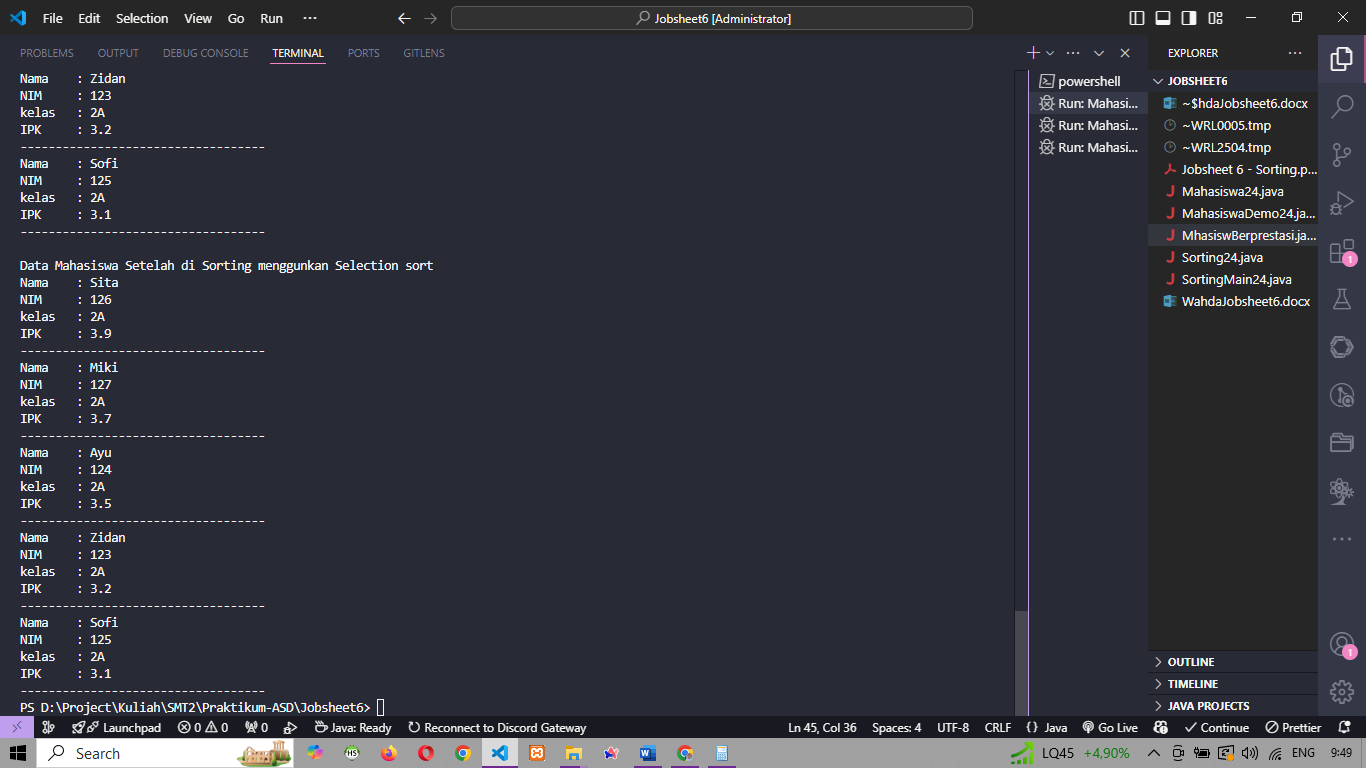
1. Tambahkan method selectionsort di mahasiswaberprestasi24.java



1. File mainSum



1. Hasil Run

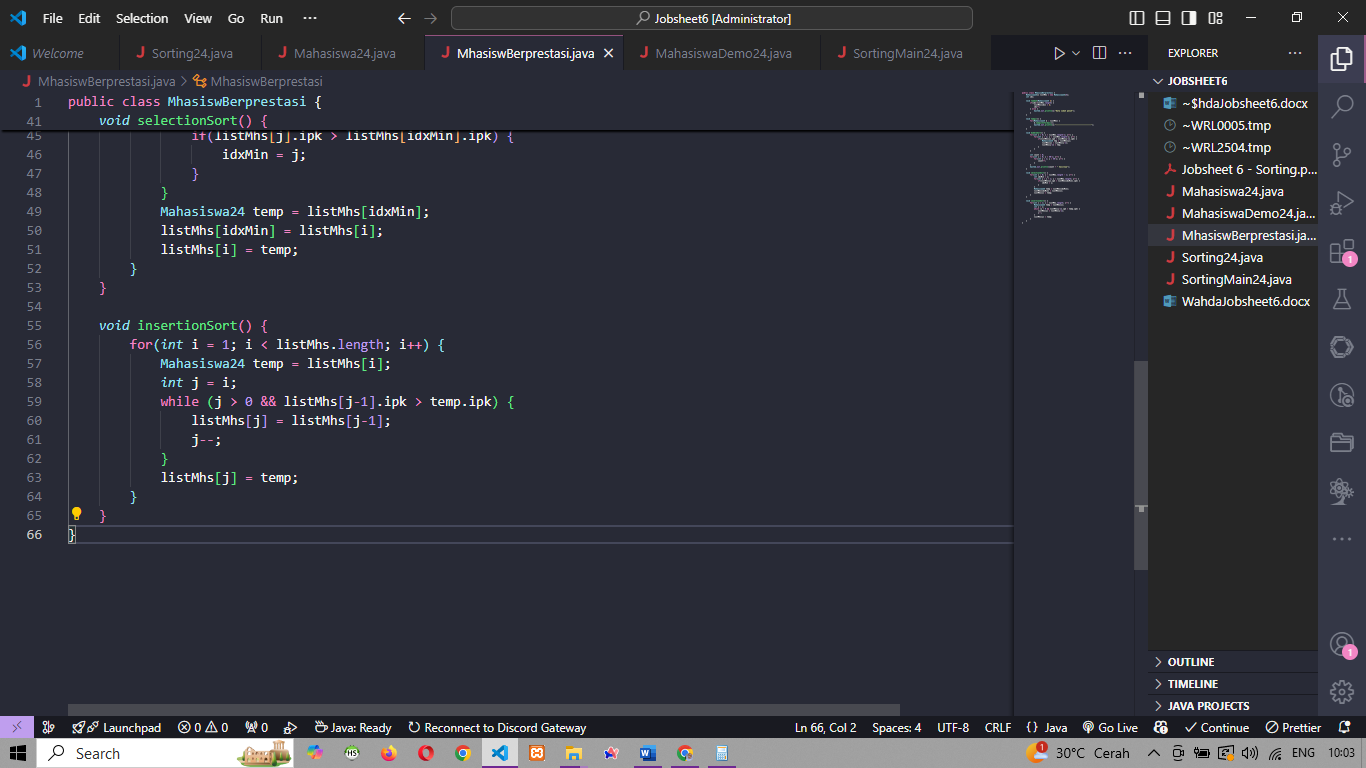


**PERTANYAAN**

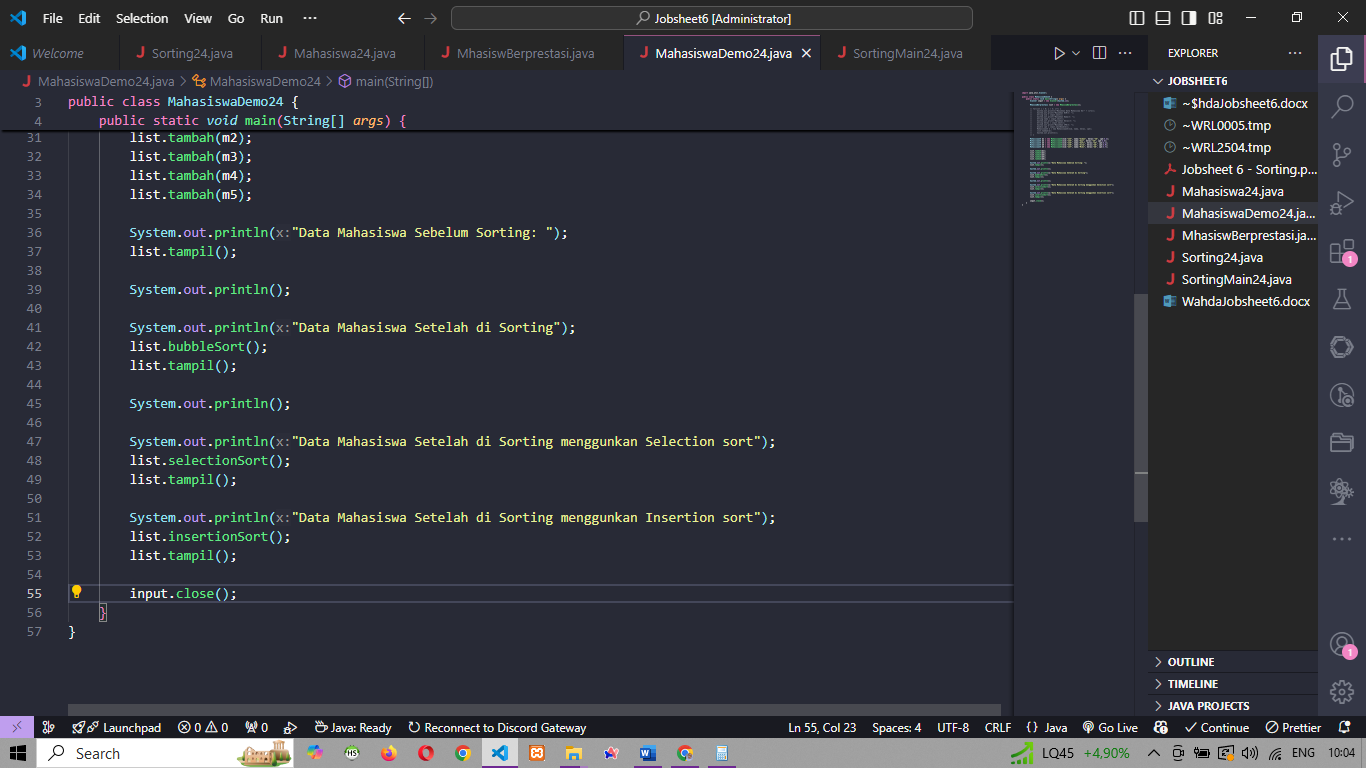
1. idxmin di gunakn untuk menyimpn index yang akan dibandingkan dengan index index yang lainnya. Lalu melakukan perulangan yang dimulai dari sebelah kanan idxmin dengan cara idxmin+ 1 sampai data.length. lalu di check apakah data index j ipk nya lebih kecil dari ipk data index idxmin. Kalau iya idxmin value nya di Ganti dengan j. Ini digunakan untuk mencari data dengan value lebih kecil ditaruh di sebelah kanan.

**PERCOBAAN 4**

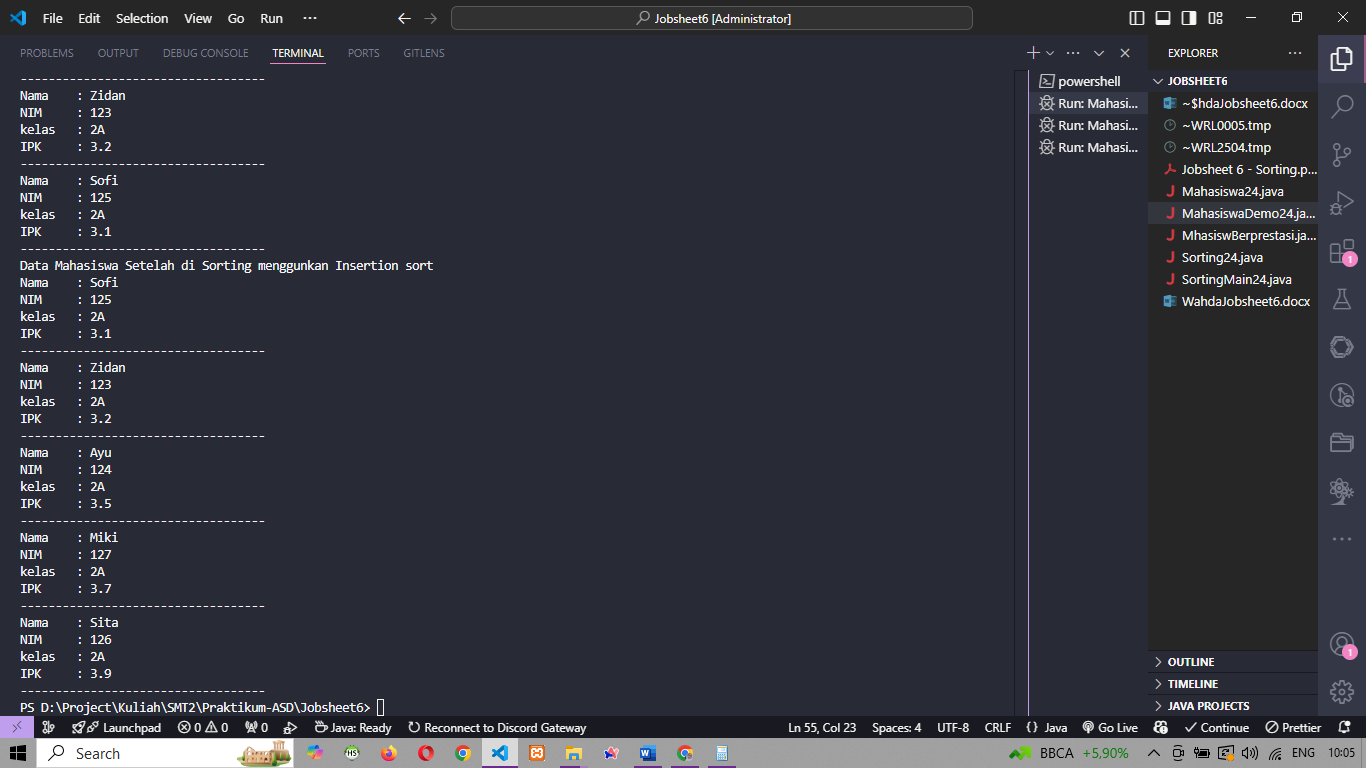
1. Tambahkan InsertionSort kedalam mahasiswaBerprestasi24.java



1. Tambahkan pemanggilan method di mahasiswademo24.java

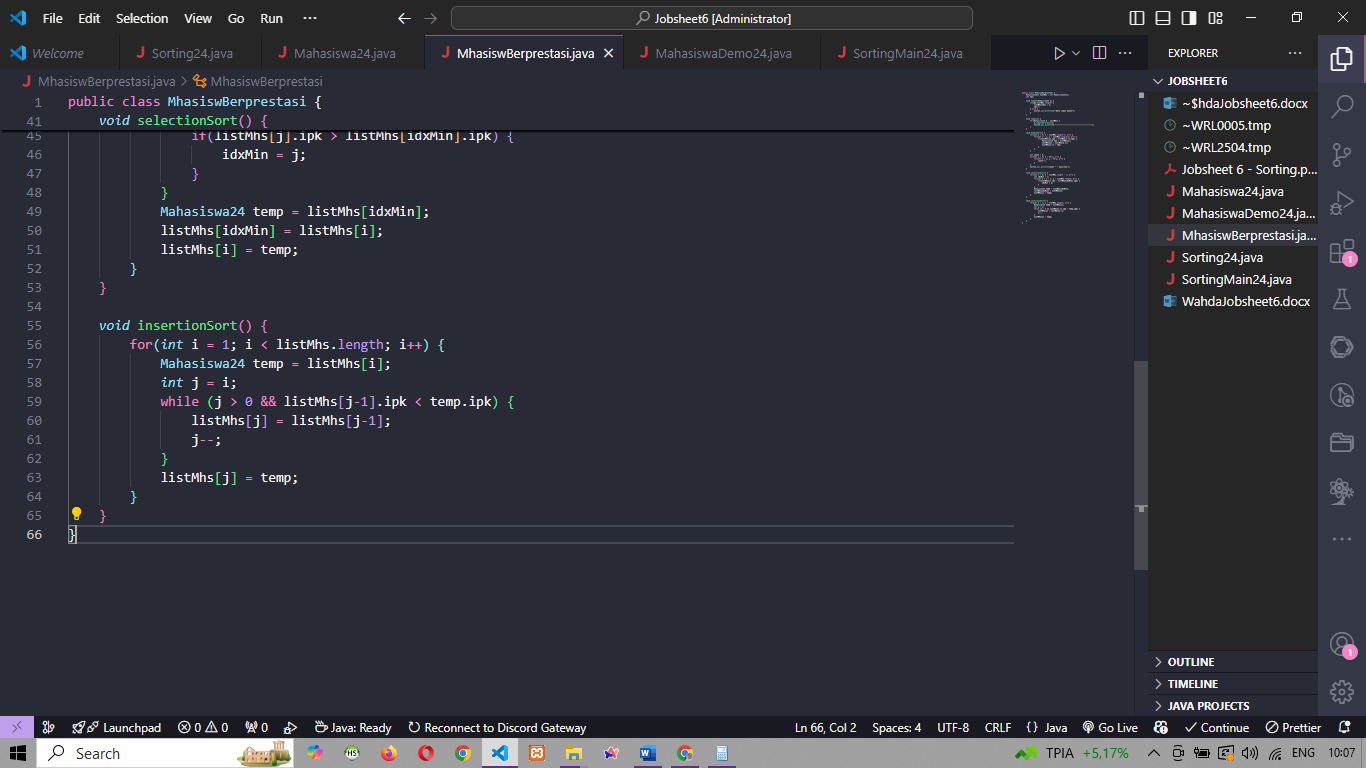


1. Hasil Run

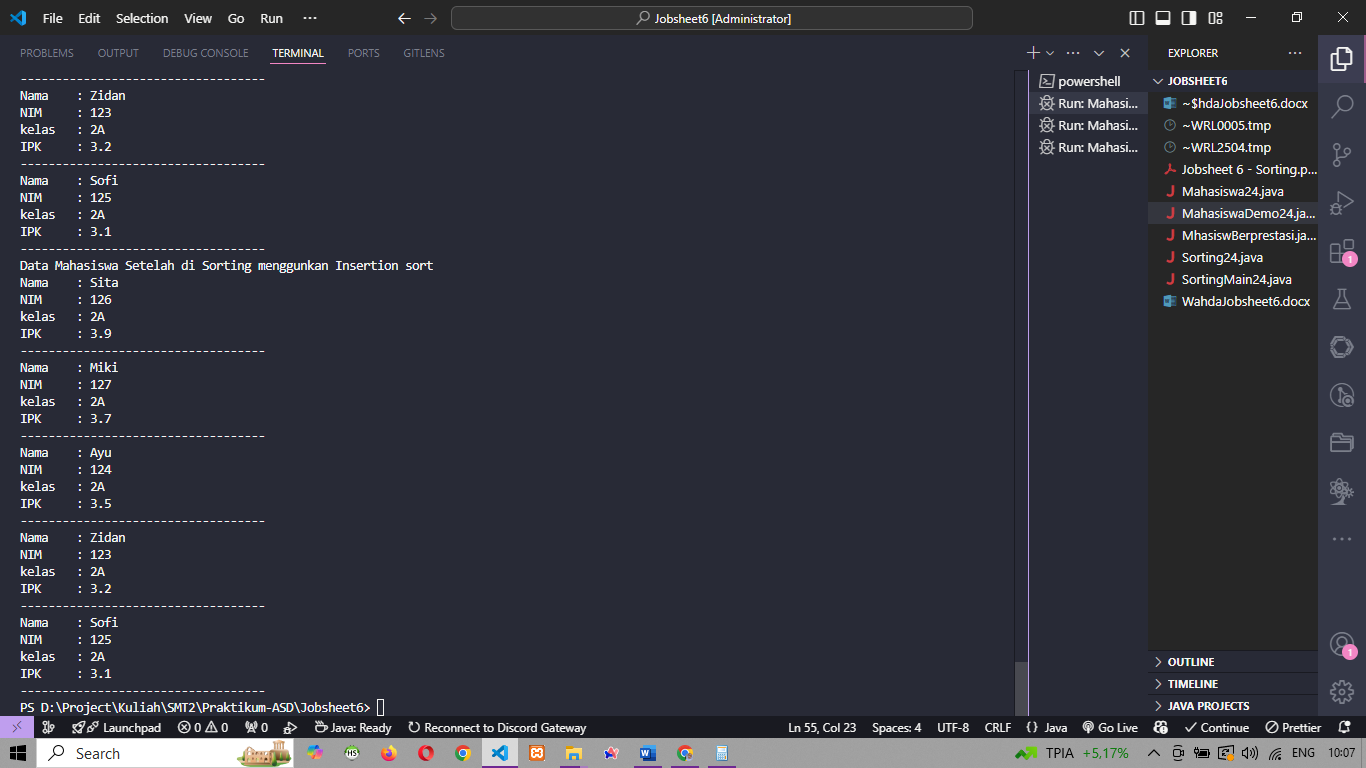


**PERTANYAAN**

1. Modifikasi kode agar bisa descending

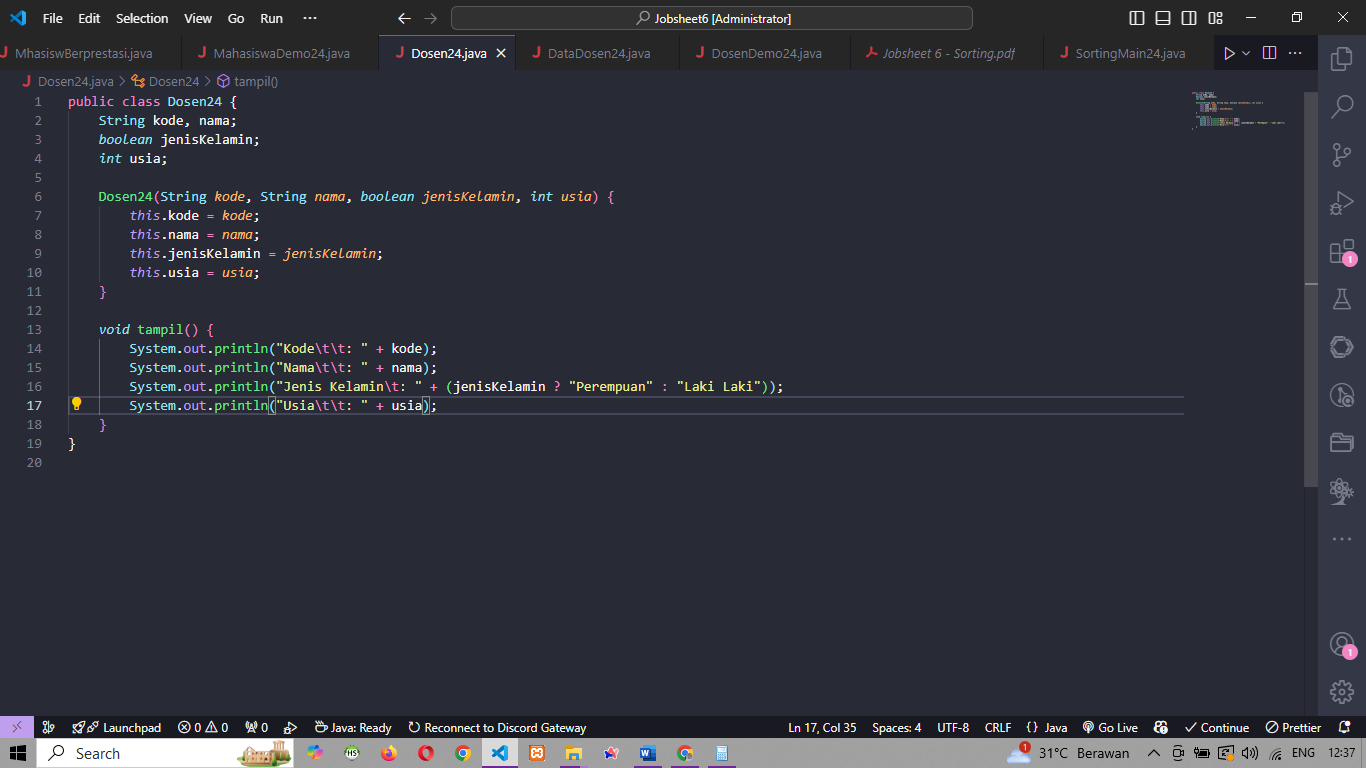


Hasil run

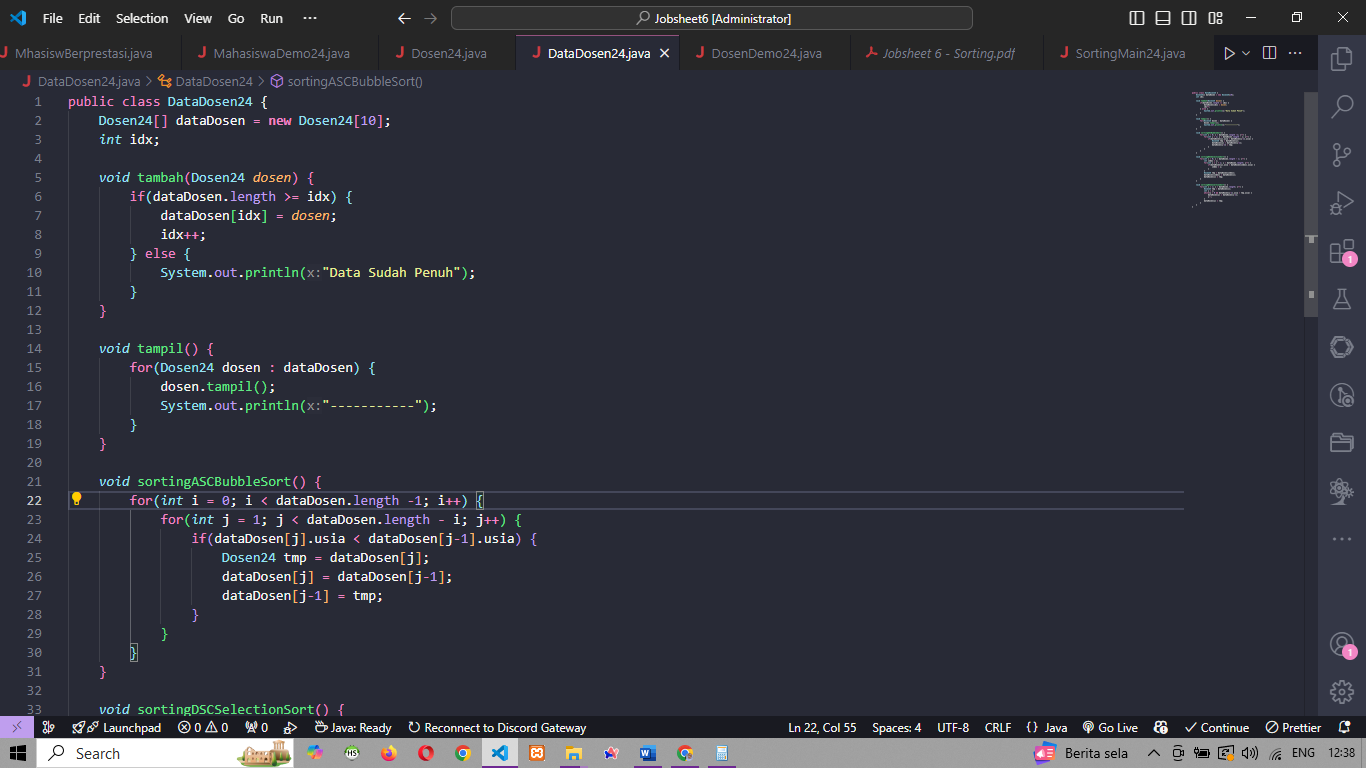


**TUGAS**

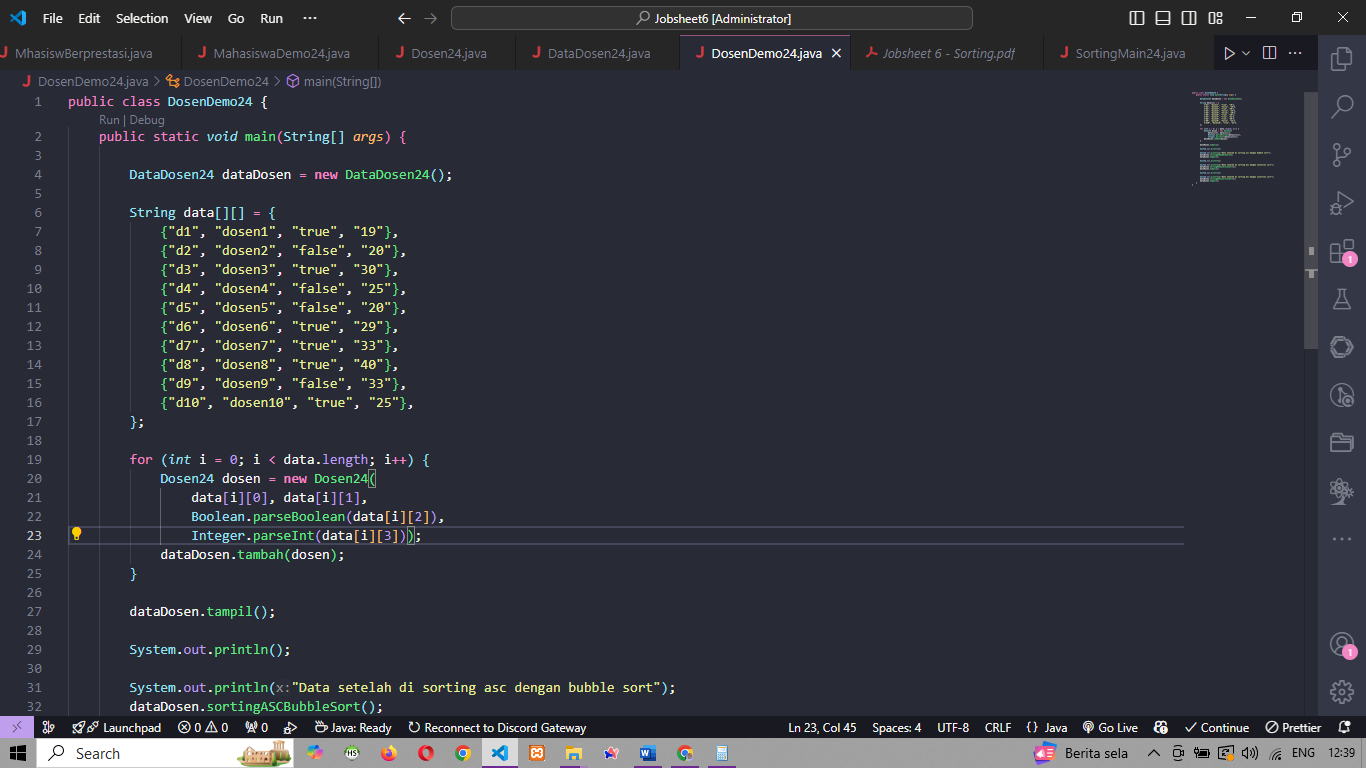
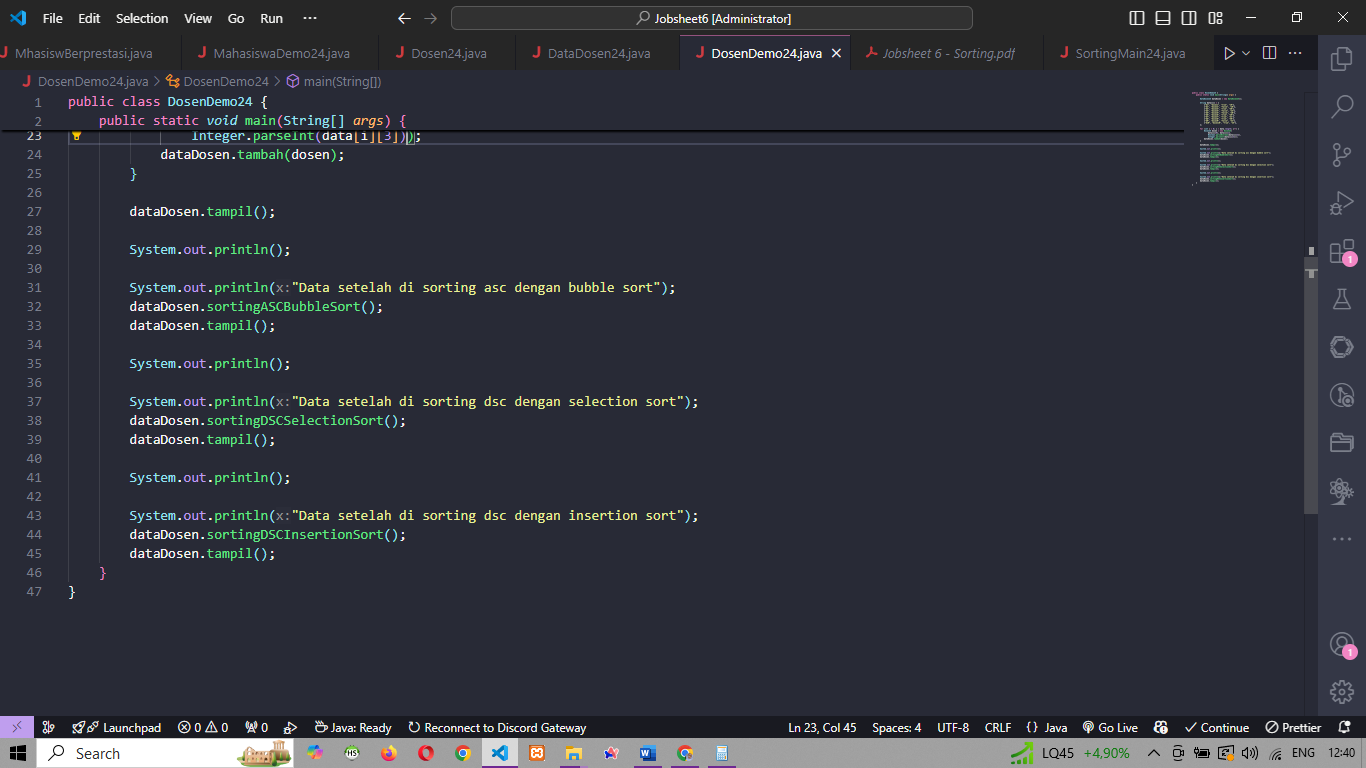
Kode Dosen24.java



Kode DataDosen24.java

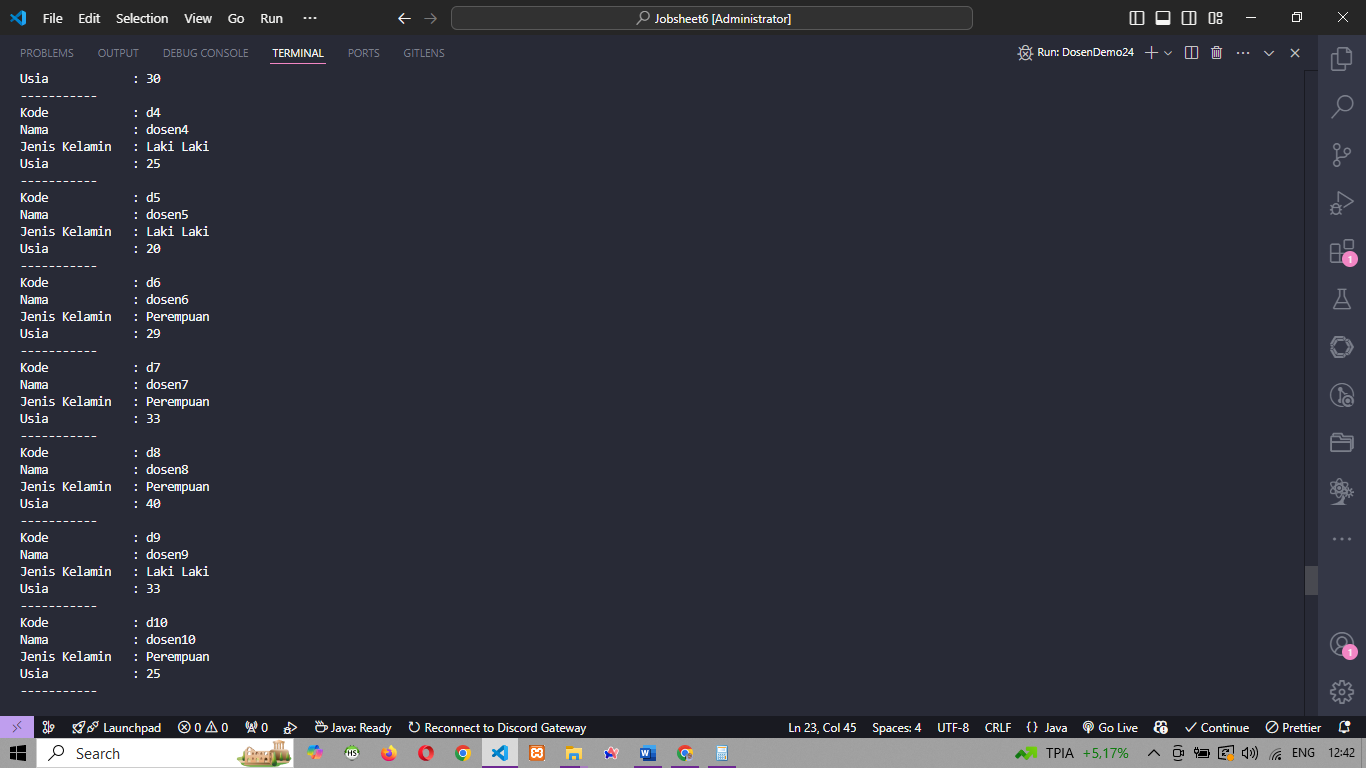
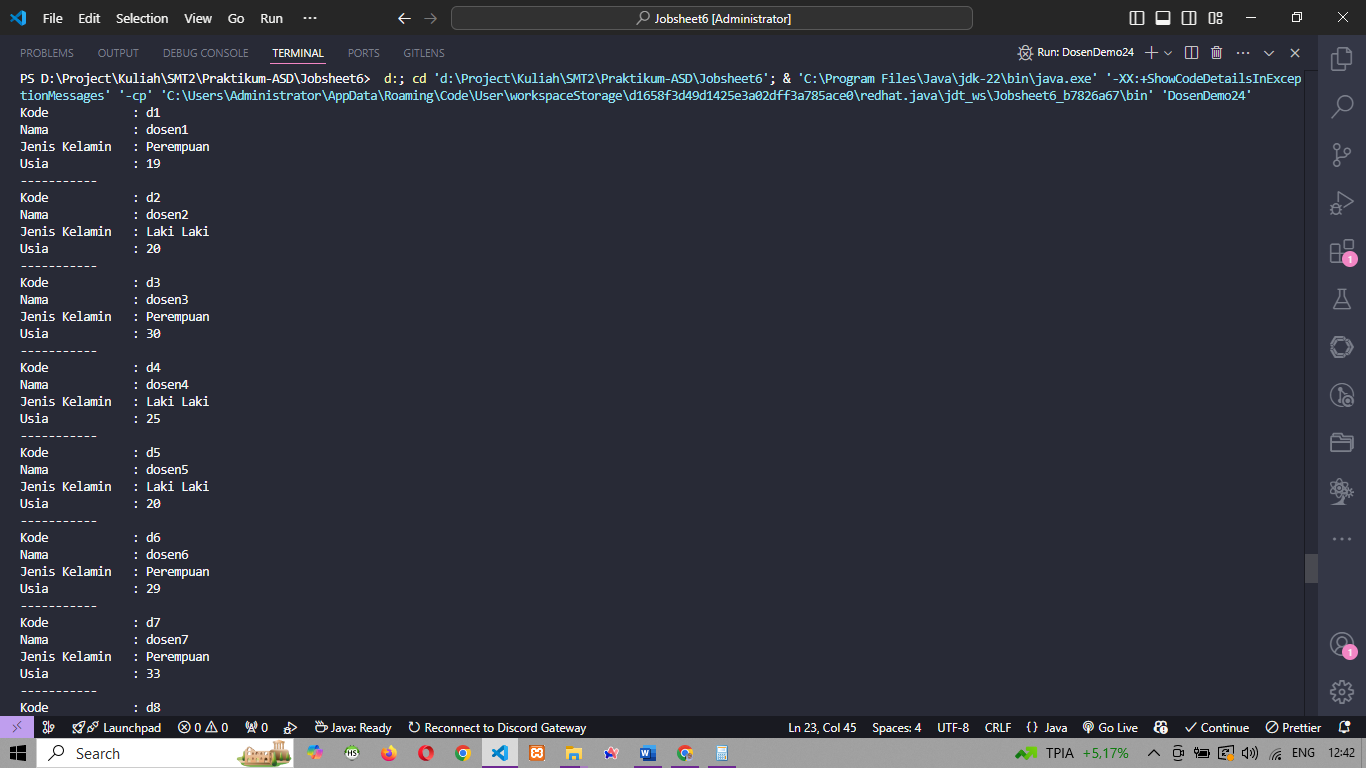


Kode DosenDemo24.java

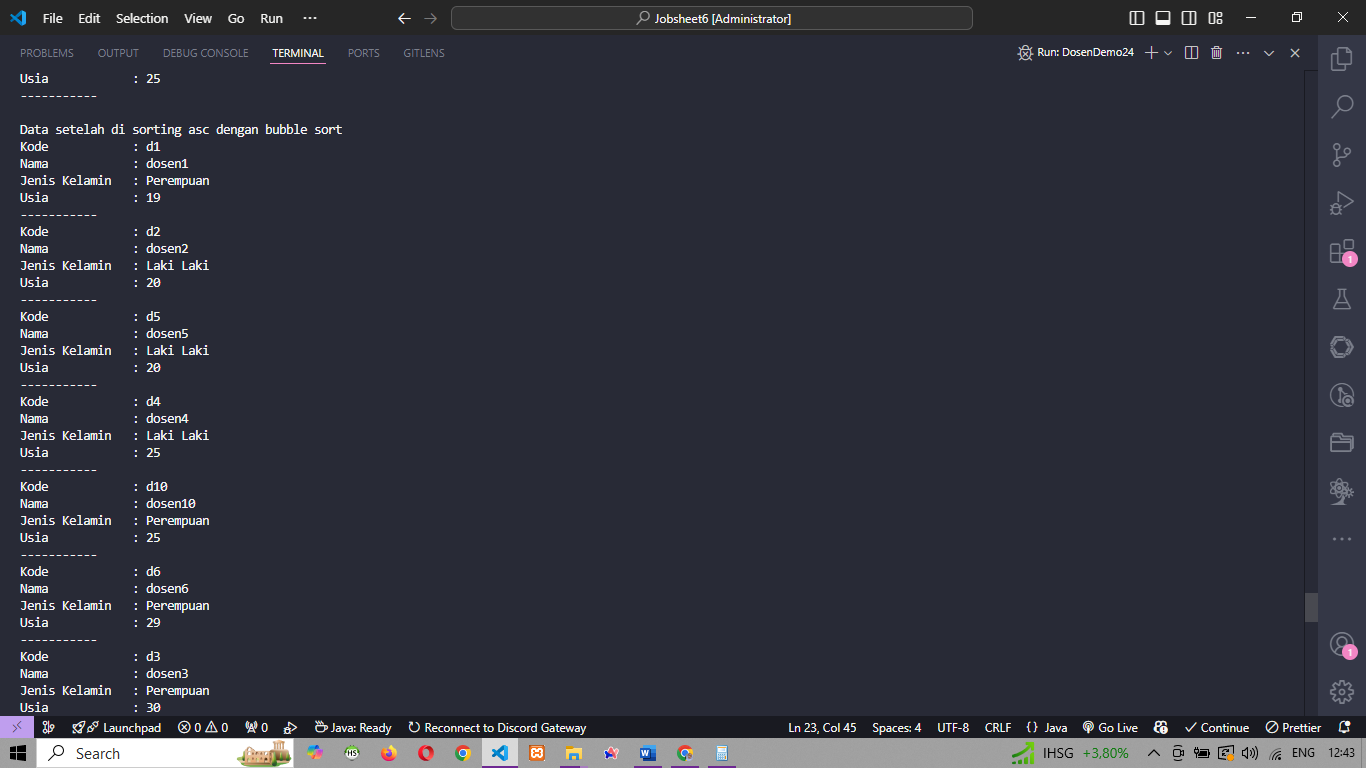
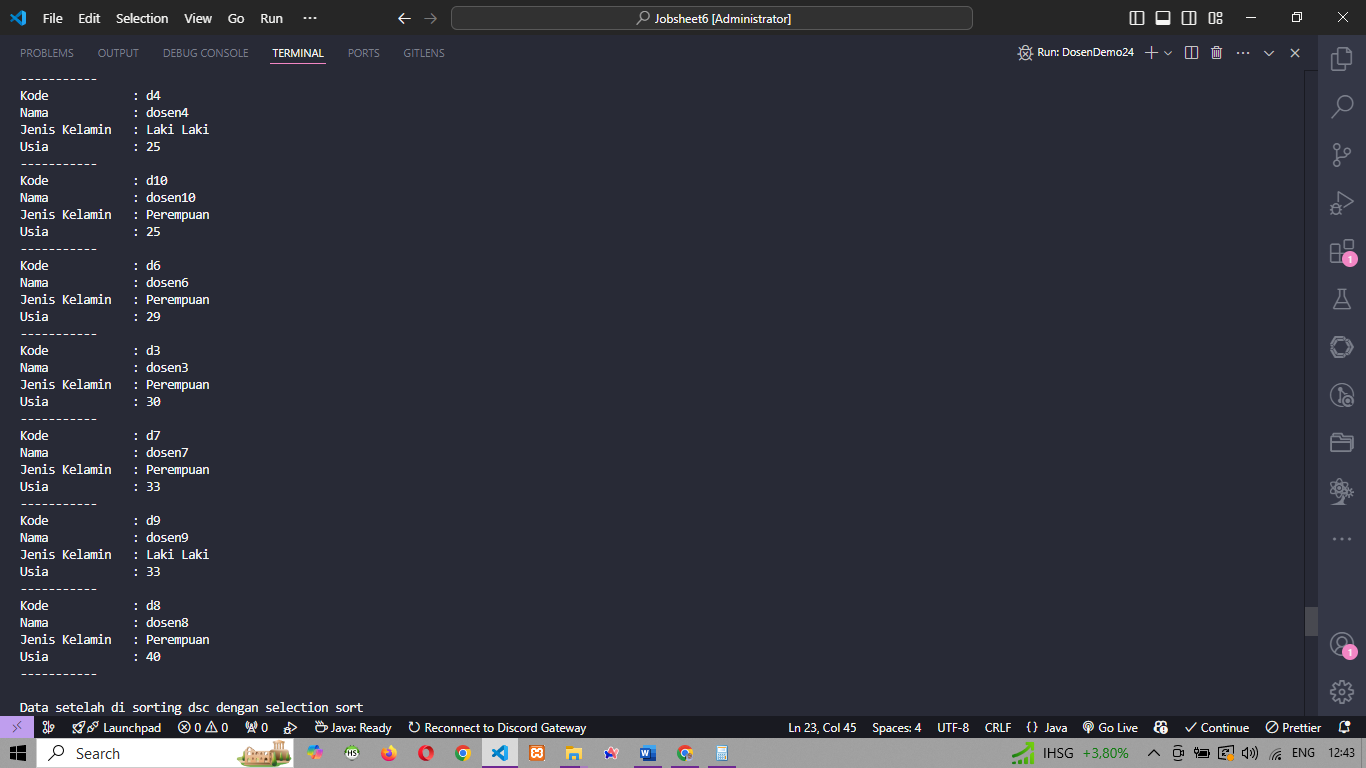
 

**Hasil Run**

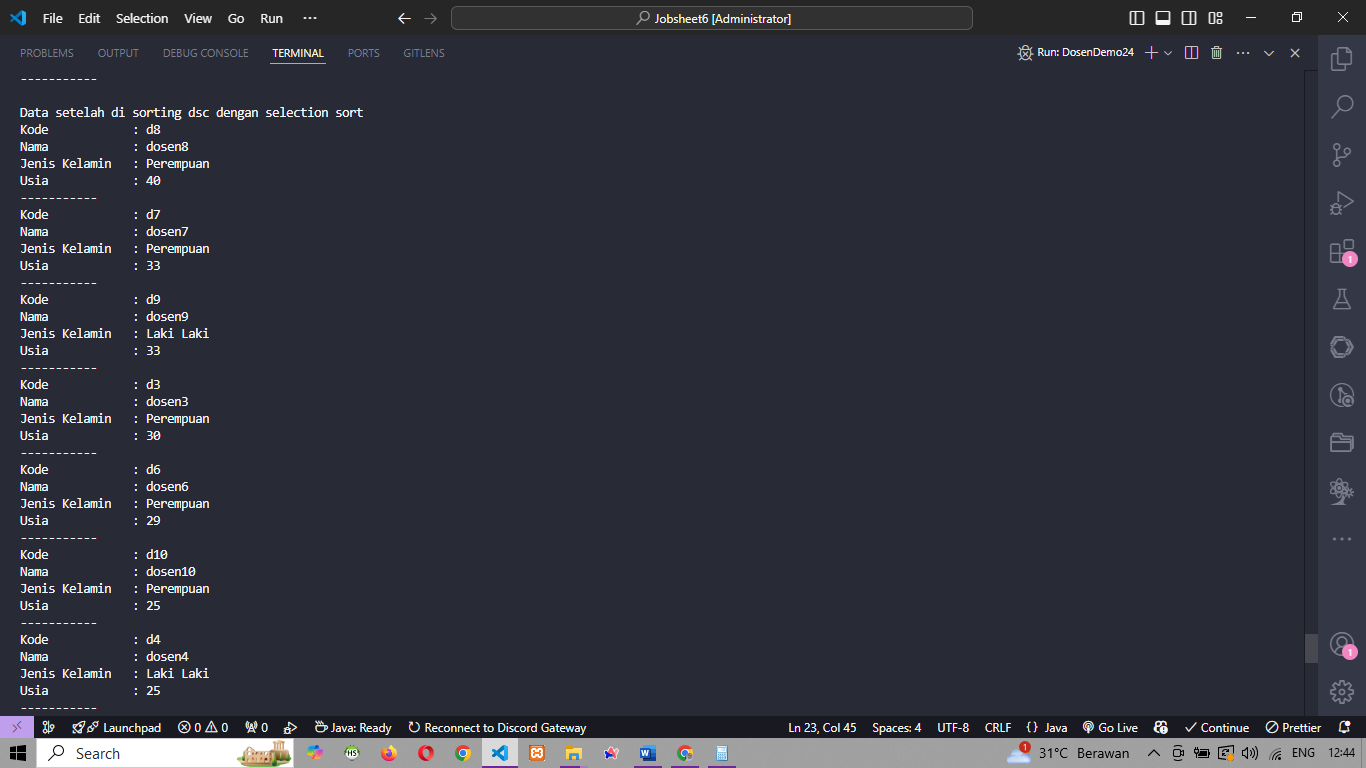
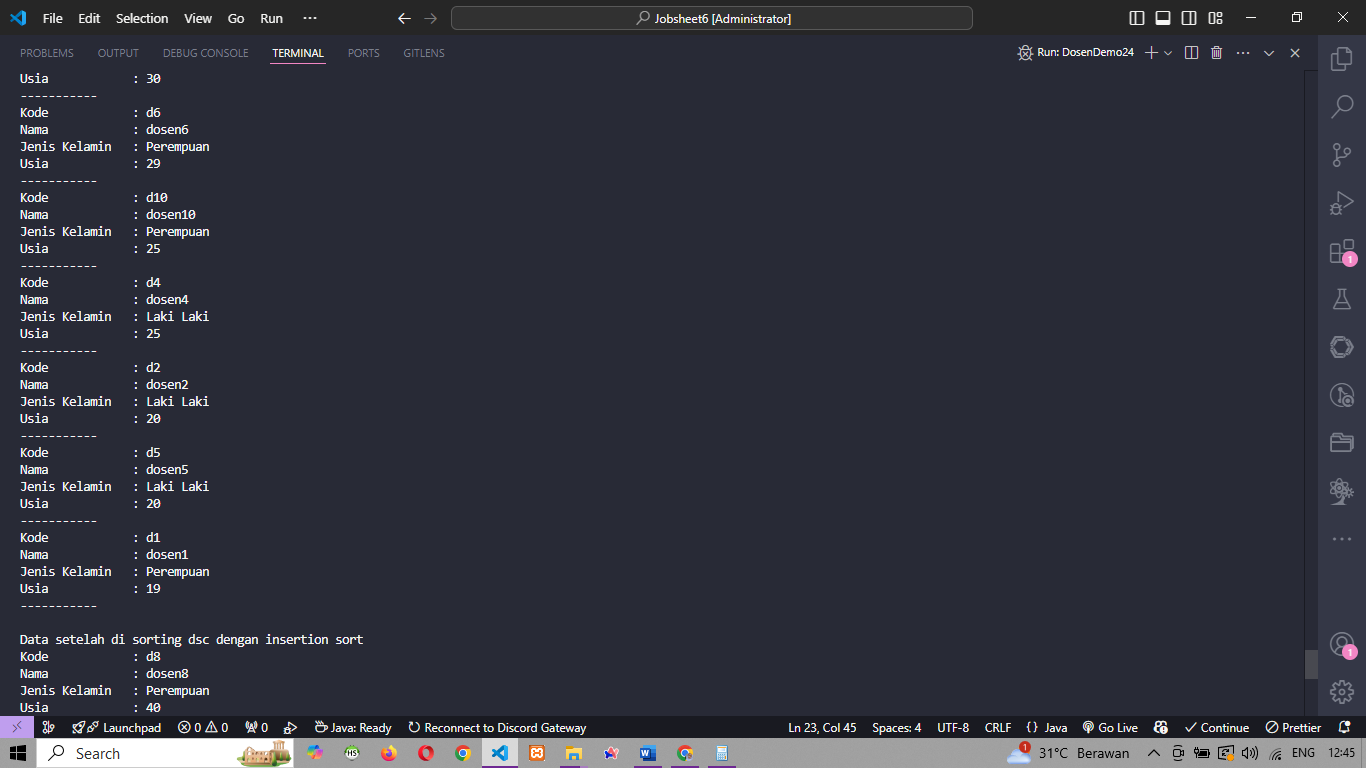
Sebelum Di sorting



Sorting ASC dengan bubble sort

Sorting DSC dengan Selection sort

Sorting DSC dengan Insertion sort

