**LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA**

**JOBSHEET 6 ASD Sorting(bubble, selection, and insertion sort)**

Dosen Pengajar : Triana Fatmawati, S.T., M.T.



Muhammad Afiq Firdaus

2341760189 / 21

SIB-1E

**PROGRAM STUDI D-IV SISTEM INFORMASI BISNIS**

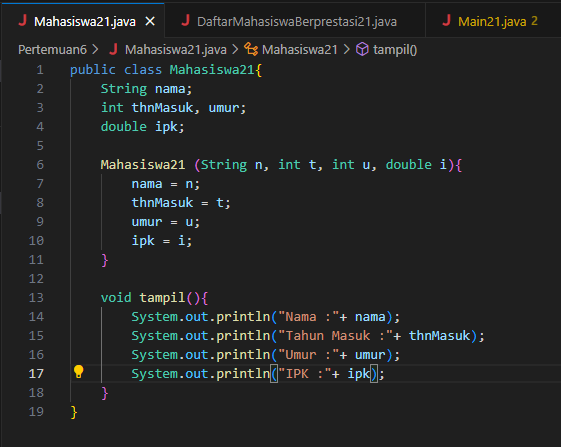
**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

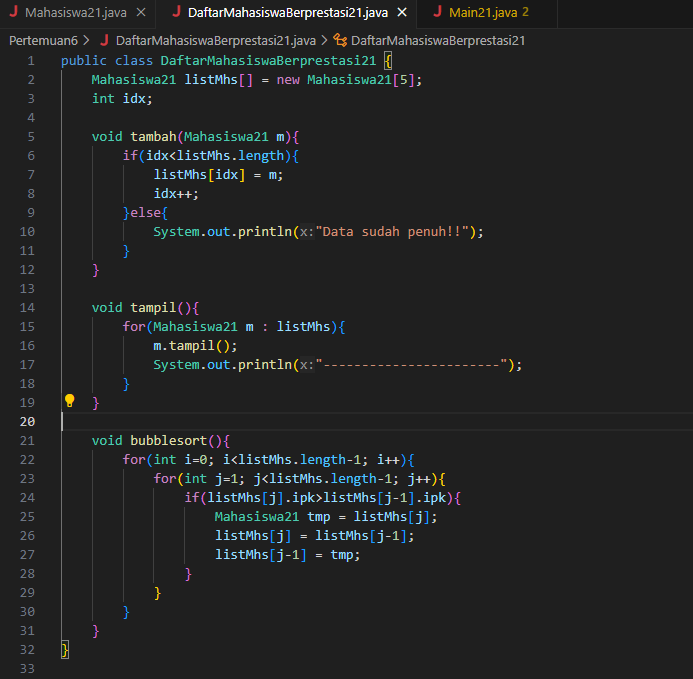
**2024**

**Percobaan 1 Bubble sort**

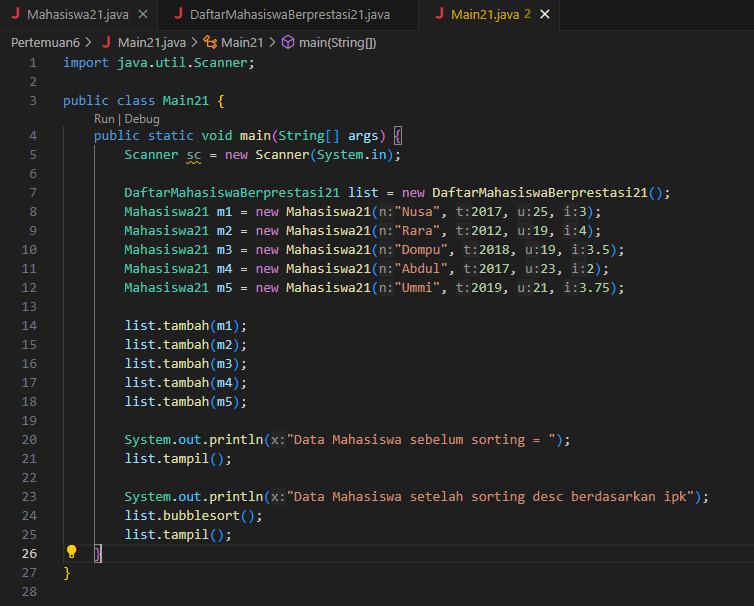
Berikut adalah code Mahasiswa berdasarkan jobsheet



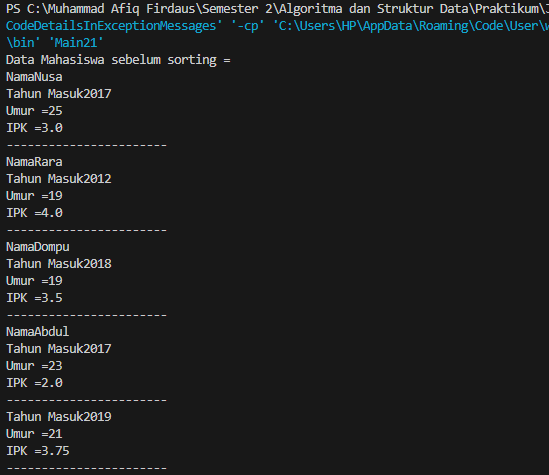
Berikut adalah code daftarMahasiswaBerprestasi berdasarkan jobsheet

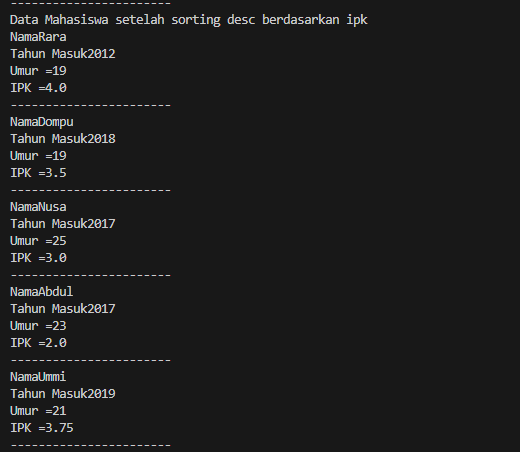


Berikut adalah code Main berdasarkan jobsheet



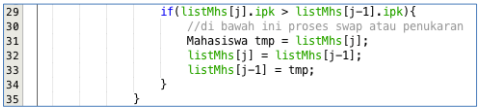
Berikut adalah hasil output dari code diatas





**Pertanyaan :**

1. Terdapat di method apakah proses bubble sort?
2. Di dalam method bubbleSort(), terdapat baris program seperti di bawah ini:



Untuk apakah proses tersebut?

1. Perhatikan perulangan di dalam bubbleSort() di bawah ini:



a. Apakah perbedaan antara kegunaan perulangan i dan perulangan j?

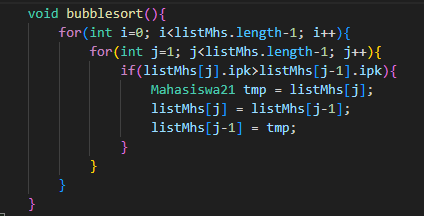
b. Mengapa syarat dari perulangan i adalah i<listMhs.length-1?

c. Mengapa syarat dari perulangan j adalah j

d. Jika banyak data di dalam listMhs adalah 50, maka berapakali perulangan i akan berlangsung? Dan ada berapa Tahap bubble sort yang ditempuh?

**Jawaban :**

1. Bubble sort berada di method bubble sort pada class DaftarMahasiswaBerprestasi



1. Proses tersebut untuk membandingkan dua elemen berturut-turut dalam array dan menukar posisinya jika elemen yang berada di indeks ke-j memiliki nilai IPK yang lebih tinggi daripada elemen yang berada di indeks ke-(j-1). Dengan cara ini, elemen dengan nilai IPK tertinggi akan "menggelembung" ke posisi yang benar secara berurutan, hingga seluruh array terurut.
2. a. Perbedaan antara variabel perulangan **i** dan **j** adalah bahwa **i** digunakan sebagai indeks untuk mengakses elemen-elemen array secara berurutan dari awal hingga akhir, sedangkan **j** digunakan sebagai indeks untuk membandingkan pasangan-pasangan elemen dalam algoritma bubble sort. Dalam implementasi ini, **i** digunakan untuk mengontrol perulangan luar yang mengatur jumlah iterasi keseluruhan, sementara **j** digunakan untuk mengontrol perulangan dalam yang digunakan untuk membandingkan elemen-elemen dan melakukan pertukaran.

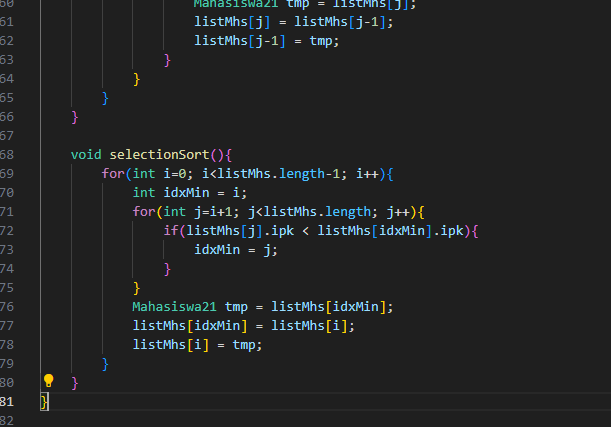
b. Syarat dari perulangan **i** adalah **i < listMhs.length-1** karena kita ingin memastikan bahwa perulangan tersebut berhenti sebelum mencapai elemen terakhir dalam array. Jika perulangan terus berlanjut hingga **i = listMhs.length**, itu berarti semua elemen sudah berada pada posisi yang benar dalam urutan terurut.

c. Syarat dari perulangan **j** adalah **j < listMhs.length-1** karena kita ingin memastikan bahwa kita hanya membandingkan elemen-elemen yang ada dalam array. Karena kita sudah membandingkan elemen-elemen pada posisi **j** dan **j-1**, maka kita tidak perlu membandingkan elemen pada posisi terakhir array (indeks **listMhs.length-1**), karena tidak ada elemen setelahnya.

d. Jika banyak data di dalam **listMhs** adalah 50, maka perulangan **i** akan berlangsung sebanyak 49 kali. Ini karena kita menggunakan kondisi **i < listMhs.length-1**, yang berarti perulangan akan terus berlanjut sampai **i** mencapai indeks terakhir sebelum akhir array. Tahap bubble sort yang ditempuh adalah sama dengan jumlah iterasi perulangan **i**, yaitu 49 tahap.

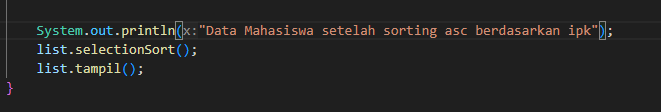
**Percobaan 2 Selection Sort**

Berikut adalah code yang ditambahkan pada percobaan 2



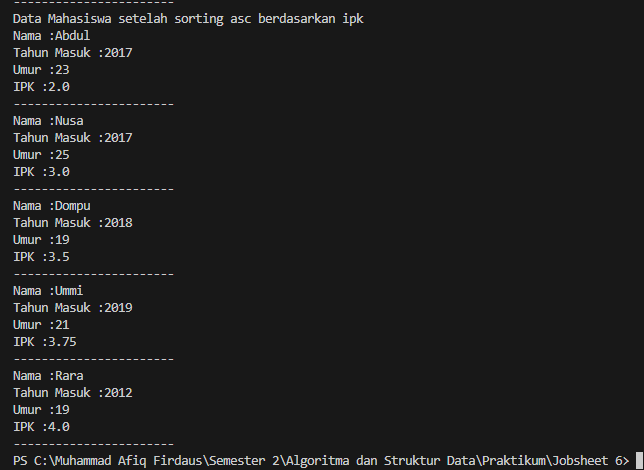
Yaitu ada penambahan selectionsort tetapi code sebelumnya masih menggunakan code yang sama.

Dan berikut adalah penambahan code pada percobaan 2



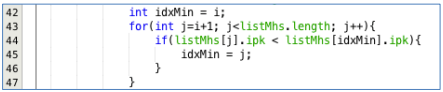
Yaitu pada class main tetapi code sebelumnya masih menggunakan code yang sama yang digunakan pada percobaan sebelumnya.

Dan berikut adalah hasil output dari code tersebut, output yang ditampilkan adalah hanya tambahan dari code percobaan 2



**Pertanyaan :**

Di dalam method selection sort, terdapat baris program seperti di bawah ini:



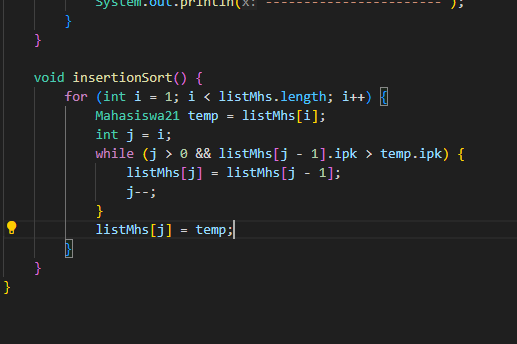
Untuk apakah proses tersebut, jelaskan!

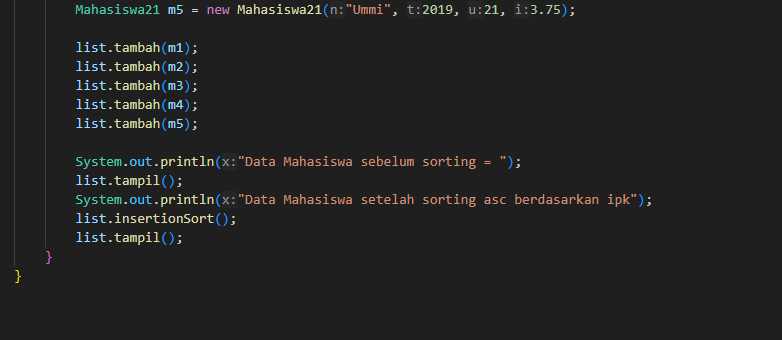
**Jawaban :**

Proses tersebut untuk mencari elemen terkecil dalam sublist yang belum diurutkan pada setiap iterasi dari algoritma selection sort. Setelah iterasi ke-i dari perulangan luar (yang mengontrol posisi awal dari sublist yang belum diurutkan), kita asumsikan bahwa elemen ke-i adalah elemen terkecil yang kita temui. Kemudian, kita melakukan perulangan dalam (yang dimulai dari i+1) untuk mencari elemen terkecil dalam sublist yang belum diurutkan. Setiap kali kita menemukan elemen yang lebih kecil dari elemen ke-i yang sedang kita pertimbangkan, kita memperbarui indeks elemen terkecil (idxMin) menjadi indeks elemen tersebut. Setelah perulangan dalam selesai, kita menukar elemen ke-i dengan elemen terkecil yang ditemukan (elemen di indeks idxMin). Hal ini memastikan bahwa setelah iterasi ke-i dari perulangan luar, elemen ke-i akan menjadi elemen terkecil dalam sublist yang belum diurutkan. Proses ini diulang hingga seluruh array terurut.

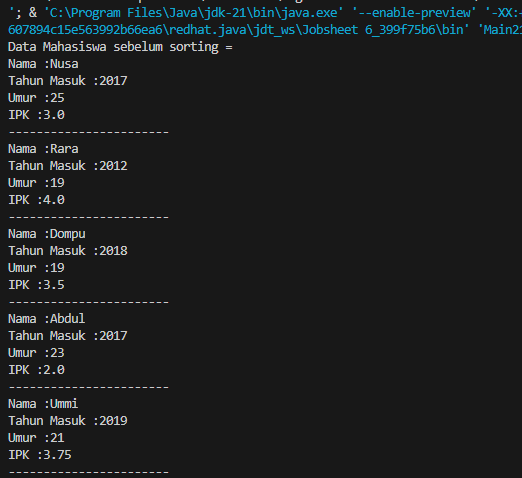
**Percobaan 3 Insertion Sort**

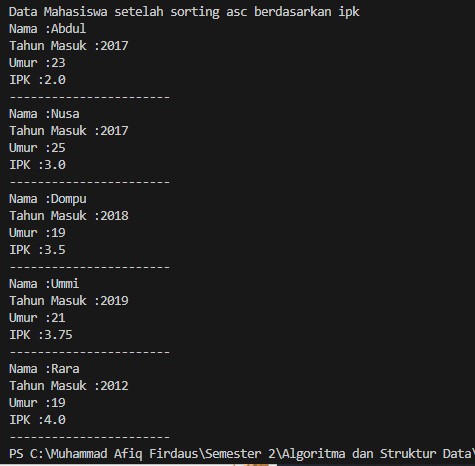
Berikut adalah hasil code percobaan 3 berdasarkan jobsheet, code yang ditampilkan adalah code tambahan saja pada percobaan 3





Berikut adalah hasil output dari code diatas

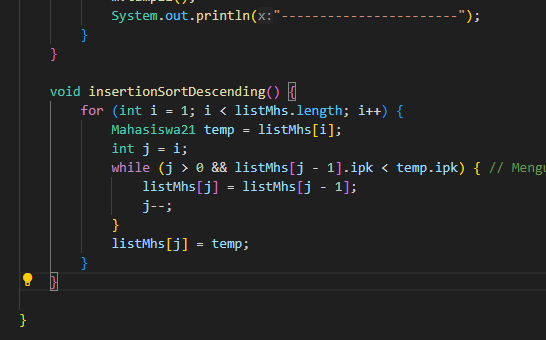




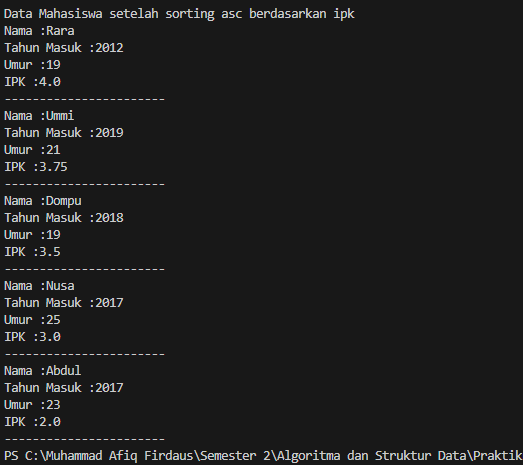
**Pertanyaan :**

Ubahlah fungsi pada InsertionSort sehingga fungsi ini dapat melaksanakan proses sorting dengan cara descending.

**Jawaban :**



Berikut adalah hasil output dari code diatas

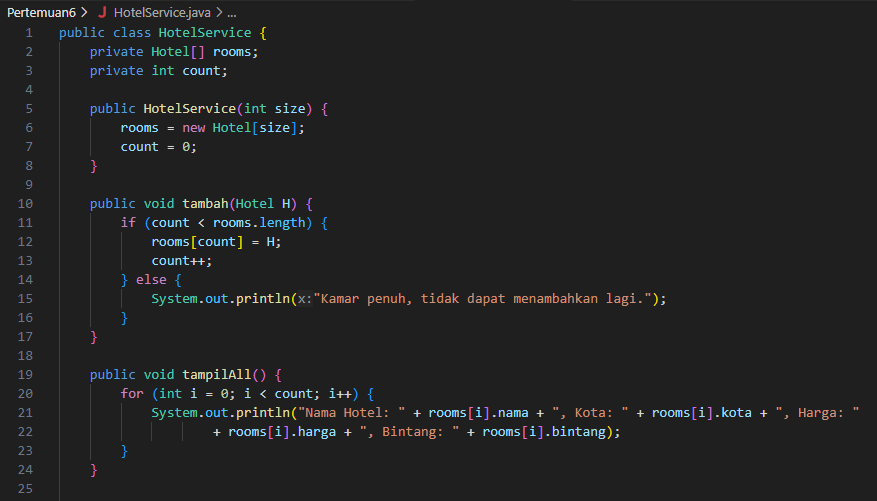


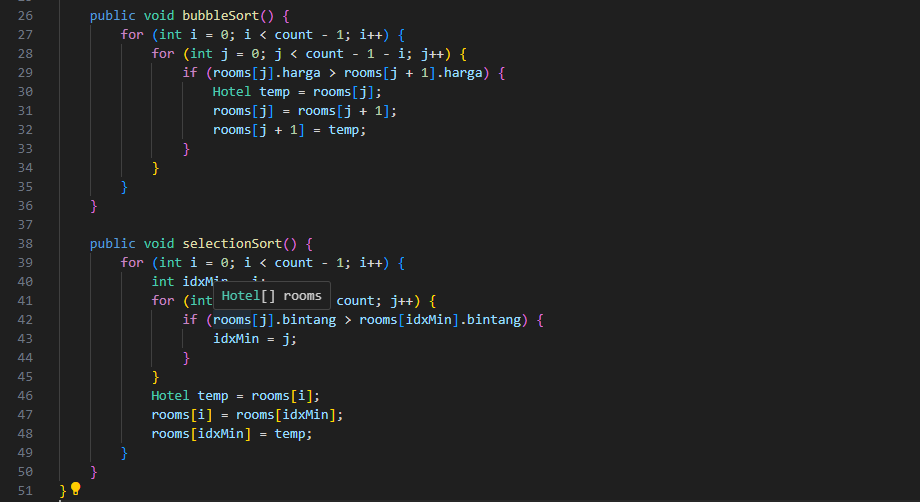
**5.5 Latihan Praktikum**

Sebuah platform travel yang menyediakan layanan pemesanan kebutuhan travelling sedang mengembangkan backend untuk sistem pemesanan/reservasi akomodasi (penginapan), salah satu fiturnya adalah menampilkan daftar penginapan yang tersedia berdasarkan pilihan filter yang diinginkan user. Daftar penginapan ini harus dapat disorting berdasarkan 1. Harga dimulai dari harga termurah ke harga tertinggi. 2. Rating bintang penginapan dari bintang tertinggi (5) ke terendah (1) Buatlah proses sorting data untuk kedua filter tersebut dengan menggunakan algoritma **bubble sort** dan **selection sort.**

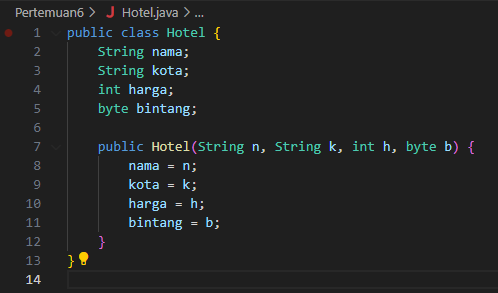
Berikut adalah code dari HotelService, Hotel, Main.

Code HotelService

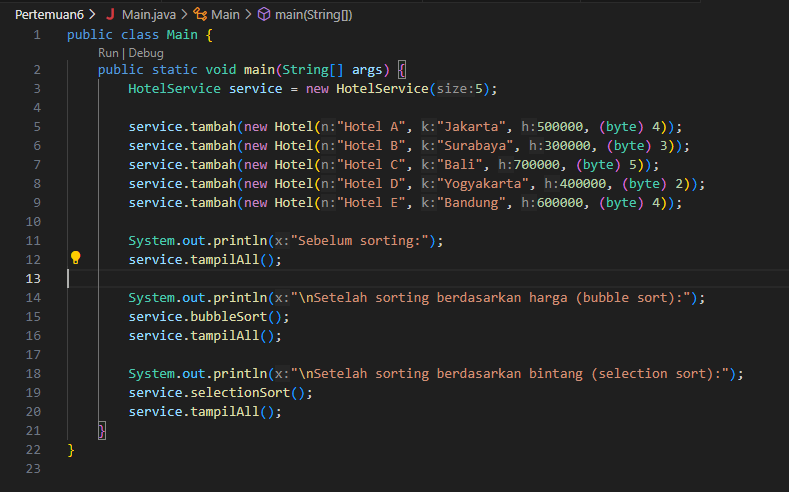




Code Hotel



Code Main



Berikut adalah output dari Code tersebut

