**LAPORAN PRAKTIKUM JOBSHEET 5**

**SORTING (BUBBLE, SELECTION, DAN INSERTION SORT)**

**MATA KULIAH ALGORITMA DAN STRUKTUR DATA**



**Disusun Oleh :**

**Jami’atul Afifah (2341760102)**

**SIB-1F**

**PROGRAM STUDI D4 SISTEM INFOEMASI BISNIS**

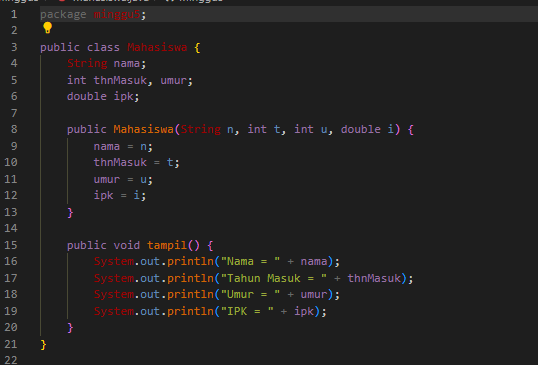
**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

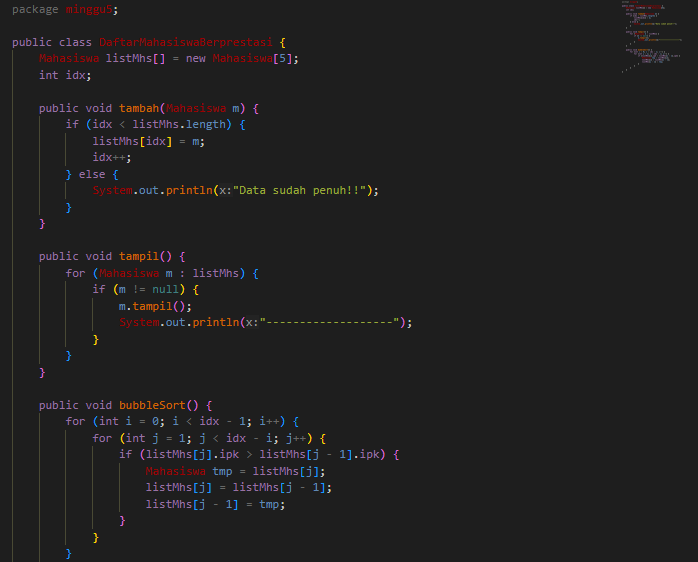
**2024**

**Mengurutkan Data Mahasiswa Berdasarkan IPK Menggunakan Bubble Sort**

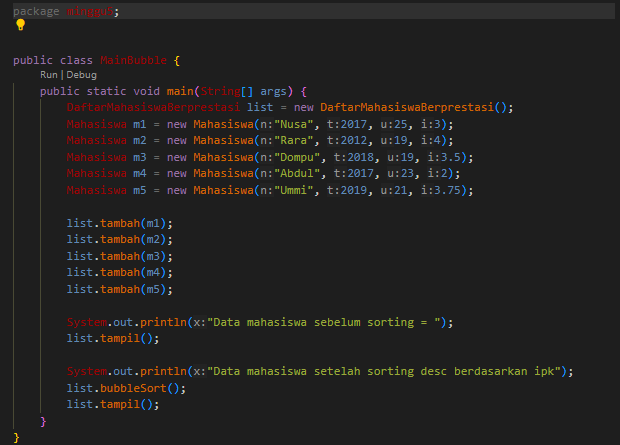
1. Buat project baru dengan nama “bubble-selection-insertion”, kemudian buat package dengan nama “jobsheet6”.
2. Buatlah sebuah class dengan nama Mahasiswa
3. Sesuaikan class Mahasiswa dengan melihat class diagram di atas dengan menambahkan attribute, konstruktor, dan fungsi atau method. Untuk lebih jelasnya class tersebut dapat dilihat pada potongan kode di bawah ini



1. Buat class DaftarMahasiswaBerprestasi seperti di bawah ini!
2. Tambahkan method tambah() di dalam class tersebut! Method tambah() digunakan untuk menambahkan objek dari class Mahasiswa ke dalam atribut listMhs.
3. Tambahkan method tampil() di dalam class tersebut! Method tampil() digunakan untuk menampilkan semua data mahasiswa-mahasiswa yang ada di dalam class tersebut! Perhatikan penggunaan sintaks for yang agak berbeda dengan for yang telah dipelajari sebelumnya, meskipun secara konsep sebenarnya mirip.
4. Tambahkan method bubbleSort() di dalam class tersebut!

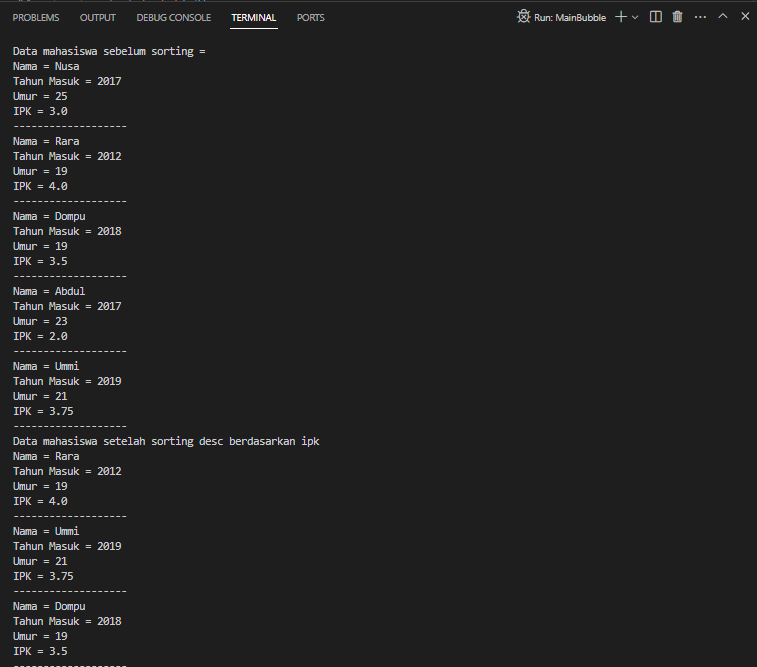


1. Buat class Main dan didalamnya buat method main() seperti di bawah ini!
2. Di dalam method main(), buatlah sebuah objek DaftarMahasiswaBerprestasi dan buatlah 5 objek mahasiswa kemudian tambahkan semua objek mahasiswa tersebut dengan memanggil fungsi tambah pada objek DaftarMahasiswaBerprestasi. Silakan dipanggil fungsi tampil() untuk melihat semua data yang telah dimasukan, urutkan data tersebut dengan memanggil fungsi bubbleSort() dan yang terakhir panggil fungsi tampil kembali.



**Verifikasi Hasil Percobaan**

Cocokan hasilnya dengan yang terdapat pada tampilan di bawah ini

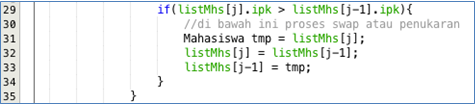


**Pertanyaan**

1. Terdapat di method apakah proses bubble sort?

Proses bubble sort terdapat di dalam method bubbleSort() di file DaftarMahasiswaBerprestasi.java.

1. Di dalam method bubbleSort(), terdapat baris program seperti di bawah ini:



Untuk apakah proses tersebut?

Proses tersebut adalah bagian dari algoritma bubble sort yang bertanggung jawab untuk menukar posisi dua elemen dalam array jika kondisi tertentu terpenuhi, yaitu jika elemen ke-j memiliki IPK yang lebih besar dari elemen ke-(j-1). Ini dilakukan untuk mengurutkan elemen dari yang terkecil ke yang terbesar.

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



* 1. Apakah perbedaan antara kegunaan perulangan i dan perulangan j?

Perbedaan antara penggunaan perulangan i dan j adalah bahwa perulangan i digunakan untuk mengontrol iterasi melalui seluruh array, sementara perulangan j digunakan untuk membandingkan dan menukar elemen-elemen terkait dalam setiap iterasi i.

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

Syarat dari perulangan i adalah i < listMhs.length - 1 karena setelah setiap iterasi i, elemen terakhir sudah pasti berada di posisi yang benar setelah berjalannya algoritma bubble sort. Oleh karena itu, tidak perlu membandingkan elemen terakhir di setiap iterasi

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

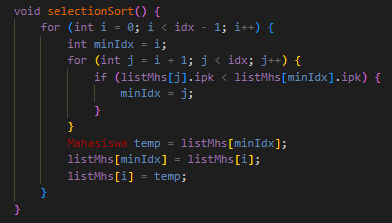
Syarat dari perulangan j adalah j < listMhs.length - i karena setiap iterasi i akan mengurangi jumlah elemen yang perlu dibandingkan di setiap iterasi j. Pada awalnya, perulangan j akan membandingkan semua elemen di array, tetapi setiap kali iterasi i berjalan, elemen terakhir dalam array sudah pasti berada di posisi yang benar, sehingga tidak perlu dibandingkan lagi.

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

Jika banyak data di dalam listMhs adalah 50, maka perulangan i akan berlangsung sebanyak 49 kali. Ini karena setiap iterasi i mengurangi jumlah elemen yang harus dibandingkan. Tahap bubble sort yang ditempuh adalah sebanyak 49 tahap, karena setiap iterasi i memindahkan setidaknya satu elemen ke posisi yang benar.

**Mengurutkan Data Mahasiswa Berdasarkan IPK Menggunakan Selection Sort**

1. Lihat kembali class DaftarMahasiswaBerprestasi, dan tambahkan method selectionSort() di dalamnya! Method ini juga akan melakukan proses sorting secara ascending, tetapi menggunakan pendekatan selection sort



1. Setelah itu, buka kembali class Main, dan di dalam method main() tambahkan baris program untuk memanggil method selectionSort() tersebut!



1. Coba jalankan kembali class Main, dan amati hasilnya! Apakah kini data mahasiswa telah tampil urut menaik berdasar ipk?

Ya hasilnya urut sesuai ipk dari terbesar ke terkecil

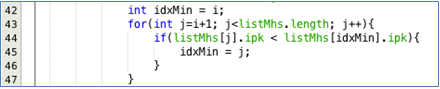
**Verifikasi Hasil Percobaan**

Pastikan output yang ditampilkan sudah benar seperti di bawah ini



**Pertanyaan**

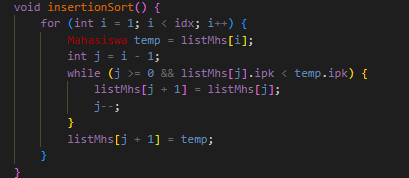
1. Di dalam method selection sort, terdapat baris program seperti di bawah ini!

  
Untuk apakah proses tersebut, jelaskan!

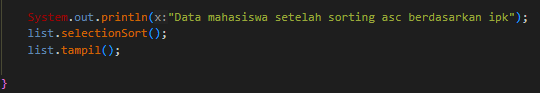
* 1. Inisialisasi Variabel idxMin: Pada awal setiap iterasi, variabel idxMin diinisialisasi dengan nilai i. Ini dilakukan untuk menandai indeks dari elemen yang memiliki nilai minimum yang terpilih saat ini dalam iterasi tersebut.
  2. Perbandingan dengan Elemen Berikutnya: Selanjutnya, dalam loop for kedua, kita membandingkan elemen pada indeks j dengan elemen pada indeks idxMin. Jika nilai ipk dari elemen pada indeks j lebih kecil dari nilai ipk dari elemen pada indeks idxMin, maka kita perbarui nilai idxMin dengan nilai j.
  3. Pembaruan idxMin: Jika ditemukan elemen dengan nilai ipk yang lebih kecil dari elemen yang sebelumnya dianggap sebagai nilai minimum (idxMin), maka idxMin akan diperbarui dengan indeks yang menunjuk ke elemen tersebut. Ini dilakukan untuk memastikan bahwa pada akhir iterasi, idxMin akan menunjuk ke indeks yang berisi nilai minimum dari seluruh elemen yang belum diurutkan.
  4. Dengan cara ini, langkah-langkah ini terus diulangi untuk setiap elemen dalam array sampai seluruh array terurut, yaitu dengan menempatkan elemen terkecil pada posisi yang benar secara berurutan. Setelah selesai iterasi, elemen dengan nilai minimum akan dipindahkan ke posisi yang tepat pada iterasi tersebut.

**Mengurutkan Data Mahasiswa Berdasarkan IPK Menggunakan Insertion Sort**

1. Lihat kembali class DaftarMahasiswaBerprestasi, dan tambahkan method insertionSort() di dalamnya. Method ini juga akan melakukan proses sorting secara ***ascending***, tetapi menggunakan pendekatan Insertion Sort



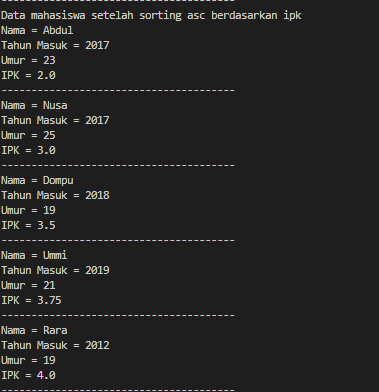
1. Setelah itu, buka kembali class Main, dan di dalam method main() tambahkan baris program untuk memanggil method insertionSort() tersebut!



1. Coba jalankan kembali class Main, dan amati hasilnya! Apakah kini data mahasiswa telah tampil urut menaik berdasar ipk?

**Verifikasi Hasil Percobaan**

Pastikan output yang ditampilkan sudah benar seperti di bawah ini

****

**Pertanyaan**

Ubahlah fungsi pada InsertionSort sehingga fungsi ini dapat melaksanakan proses sorting

dengan cara descending

tambahkan kode :  
// Di dalam kelas DaftarMahasiswaBerprestasi

void insertionSort() {

for (int i = 1; i < idx; i++) {

Mahasiswa key = listMhs[i];

int j = i - 1;

// Geser elemen yang lebih besar dari key ke kanan

while (j >= 0 && listMhs[j].ipk < key.ipk) {

listMhs[j + 1] = listMhs[j];

j--;

}

listMhs[j + 1] = key;

}

}

**Latihan Praktikum**

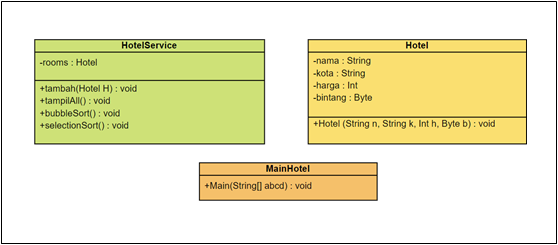
Sebuah platform travel yang menyediakan layanan pemesanan kebutuhan travelling sedang mengembangkan backend untuk sistem pemesanan/reservasi akomodasi (penginapan), salah satu fiturnya adalah menampilkan daftar penginapan yang tersedia berdasarkan pilihan filter yang diinginkan user. Daftar penginapan ini harus dapat disorting berdasarkan

1. Harga dimulai dari harga termurah ke harga tertinggi.

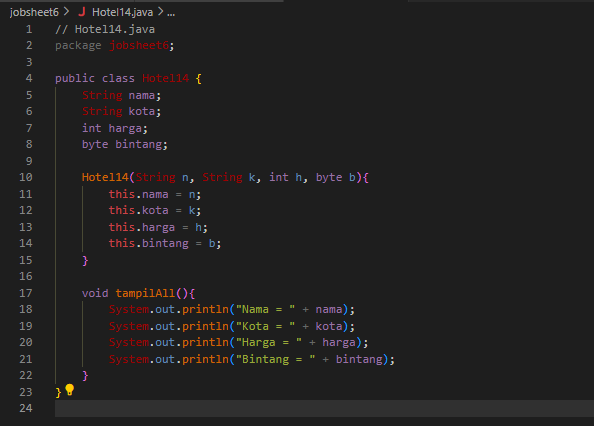
2. Rating bintang penginapan dari bintang tertinggi (5) ke terendah (1)

Buatlah proses sorting data untuk kedua filter tersebut dengan menggunakan algoritma

bubble sort dan selection sort.



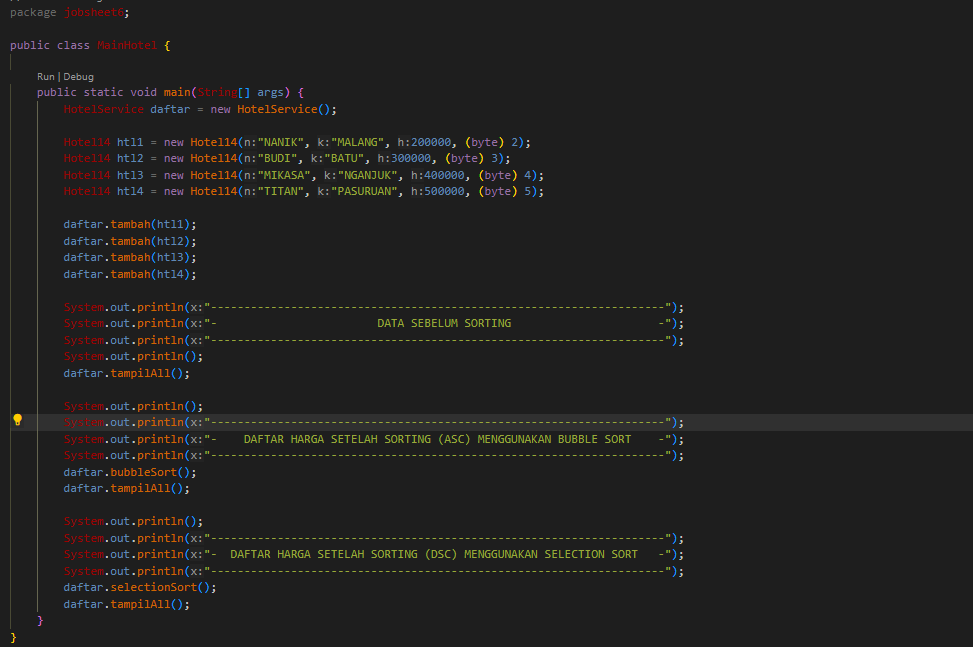
Hotel



HotelService



MainHotel



Hasil:

