

Nama: Wahda Adella Putri Febriana

Kelas: 1B / 24

NIM: 2440700156

SORTING – BUBBLE SORT

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

```
public class Sorting24 {
    int []data;
    int jumData;

    Sorting24(int Data[], int jmlData) {
        jumData = jmlData;
        data=new int[jmlData];
        for(int i = 0; i < jumData; i++) {
            data[i] = Data[i];
        }
    }

    void bubbleSort() {
        int temp = 0;
        for(int i = 0; i < jumData-1; i++) {
            for(int j = 0; j < jumData; j++) {
                if(data[j-1] > data[j]) {
                    temp = data[j];
                    data[j] = data[j-1];
                    data[j-1] = temp;
                }
            }
        }
    }

    void tampil() {
        for (int i = 0; i < jumData; i++) {
            System.out.print(data[i] + " ");
        }
        System.out.println();
    }
}
```

2. Buat SortingMain24.java dan panggil class sorting

```
SortingMain24.java > SortingMain24
public class SortingMain24 {
    public static void main(String[] args) {
        int a[] = {20, 10, 2, 7, 12};
        Sorting24 dataUrut1 = new Sorting24(a, a.length);

        System.out.println(x: "Data awal 1");
        dataUrut1.tampil();
        dataUrut1.bubbleSort();
        System.out.println(x: "Setelah Data Diurutkan menggunakan bubble sort");
        dataUrut1.tampil();
    }
}
```

3. Hasil run

```
Data awal 1
20 10 2 7 12
Setelah Data Diurutkan menggunakan bubble sort
2 7 10 12 20
```

SORTING – BUBBLE SORT

1. Tambahkan method baru Bernama selectionSort

```
void selectionSort() {  
    for(int i = 0; i < jumData - 1; i++) {  
        int min = i;  
        for(int j = i + 1; j < jumData; j++) {  
            if(data[j] < data[min]) {  
                min = j;  
            }  
        }  
        int temp = data[i];  
        data[i] = data[min];  
        data[min] = temp;  
    }  
}
```

2. Tambahkan kodingan ini di sortingmain24.java

```
int b[] = {30, 20, 2, 8, 14};  
Sorting24 dataUrut2 = new Sorting24(b, b.length);  
  
System.out.println(x:"Data awal 2");  
dataUrut2.tampil();  
dataUrut2.selectionSort();  
System.out.println(x:"Setelah Data Diurutkan menggunakan selection sort");  
dataUrut2.tampil();
```

3. Hasil setelah di run

```
Data awal 2  
30 20 2 8 14  
Setelah Data Diurutkan menggunakan selection sort  
2 8 14 20 30
```

SORTING – INSERTION SORT

1. Tambahkan function insertionSort di sorting24.java

```
void insertionSort() {  
    for(int i = 1; i <= data.length-1; i++) {  
        int temp = data[i];  
        int j = i - 1;  
        while (j >= 0 && data[j] > temp) {  
            data[j+1] = data[j];  
            j--;  
        }  
        data[j+1] = temp;  
    }  
}
```

2. Panggil functionnya di sortingmain24.java

```
int c[] = {40, 10, 4, 9, 3};
Sorting24 dataUrut3 = new Sorting24(c, c.length);

System.out.println(x:"Data awal 3");
dataUrut3.tampil();
dataUrut3.insertionSort();
System.out.println(x:"Setelah Data Diurutkan menggunakan insertion sort");
dataUrut3.tampil();
```

3. Hasil Run

```
Data awal 3
40 10 4 9 3
Setelah Data Diurutkan menggunakan insertion sort
3 4 9 10 40
```

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

PERCOBAAN 2

1. Buat file Mahasiswa24.java dan isi code nya

```
public class Mahasiswa24 {
    String nim, nama, kelas;
    double ipk;

    Mahasiswa24(String nim, String nama, String kelas, double ipk) {
        this.nim = nim;
        this.nama = nama;
        this.kelas = kelas;
        this.ipk = ipk;
    }

    void tampilInformasi() {
        System.out.println("Nama\t: " + nama);
        System.out.println("NIM\t: " + nim);
        System.out.println("kelas\t: " + kelas);
        System.out.println("IPK\t: " + ipk);
    }
}
```

2. Buat file MahasiswaBerprestasi24.java dan isi filenya

```
public class MahasiswaBerprestasi {
    Mahasiswa24[] listMhs = new Mahasiswa24[5];
    int idx;

    void tambah(Mahasiswa24 m) {
        if (idx < listMhs.length) {
            listMhs[idx] = m;
            idx++;
        } else {
            System.out.println("Data sudah penuh");
        }
    }

    void tampil() {
        for (Mahasiswa24 m : listMhs) {
            m.tampilInformasi();
            System.out.println("-----");
        }
    }

    void bubbleSort() {
        for (int i = 0; i < listMhs.length - 1; i++) {
            for (int j = 1; j < listMhs.length - i; j++) {
                if (listMhs[j].ipk > listMhs[j - 1].ipk) {
                    Mahasiswa24 tmp = listMhs[j];
                    listMhs[j] = listMhs[j - 1];
                    listMhs[j - 1] = tmp;
                }
            }
        }
    }
}
```

3. Buat file MahasiswaDemo24.java dan code nya

```
public class MahasiswaDemo24 {
    Run | Debug
    public static void main(String[] args) {
        MahasiswaBerprestasi list = new MahasiswaBerprestasi();
        Mahasiswa24 m1 = new Mahasiswa24(nim:"123", nama:"Zidan", kelas:"2A", ipk:3.2);
        Mahasiswa24 m2 = new Mahasiswa24(nim:"124", nama:"Ayu", kelas:"2A", ipk:3.5);
        Mahasiswa24 m3 = new Mahasiswa24(nim:"125", nama:"Sofi", kelas:"2A", ipk:3.1);
        Mahasiswa24 m4 = new Mahasiswa24(nim:"126", nama:"Sita", kelas:"2A", ipk:3.9);
        Mahasiswa24 m5 = new Mahasiswa24(nim:"127", nama:"Miki", kelas:"2A", ipk:3.7);

        list.tambah(m1);
        list.tambah(m2);
        list.tambah(m3);
        list.tambah(m4);
        list.tambah(m5);

        System.out.println("Data Mahasiswa Sebelum Sorting: ");
        list.tampil();

        System.out.println("Data Mahasiswa Setelah di Sorting");
        list.bubbleSort();
        list.tampil();
    }
}
```

4. Hasil run

<pre>sa785ace0@reinal: java (JDK-11.0.2) Data Mahasiswa Sebelum Sorting: Nama : Zidan NIM : 123 kelas : 2A IPK : 3.2 ----- Nama : Ayu NIM : 124 kelas : 2A IPK : 3.5 ----- Nama : Sofi NIM : 125 kelas : 2A IPK : 3.1 ----- Nama : Sita NIM : 126 kelas : 2A IPK : 3.9 ----- Nama : Miki NIM : 127 kelas : 2A IPK : 3.7 -----</pre>	<pre>Data Mahasiswa Setelah di Sorting Nama : Sita NIM : 126 kelas : 2A IPK : 3.9 ----- Nama : Miki NIM : 127 kelas : 2A IPK : 3.7 ----- Nama : Ayu NIM : 124 kelas : 2A IPK : 3.5 ----- Nama : Zidan NIM : 123 kelas : 2A IPK : 3.2 ----- Nama : Sofi NIM : 125 kelas : 2A IPK : 3.1 -----</pre>
---	---

PERTANYAAN

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

```
public class MahasiswaDemo24 {
    Run | Debug
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        MhasiswBerprestasi list = new MhasiswBerprestasi();

        for(int i = 0; i < 5; i++) {
            System.out.println("Masukkan Data Mahasiswa Ke-" + (i+1));
            System.out.print(s:"Masukkan NIM\t: ");
            String nim = input.next();
            System.out.print(s:"Masukkan Nama\t: ");
            String nama = input.next();
            System.out.print(s:"Masukkan Kelas\t: ");
            String kelas = input.next();
            System.out.print(s:"Masukkan IPK\t: ");
            double ipk = input.nextDouble();
            Mahasiswa24 m = new Mahasiswa24(nim, nama, kelas, ipk);
            list.tambah(m);
            System.out.println();
        }

        System.out.println(x:"Data Mahasiswa Sebelum Sorting: ");
        list.tampil();

        System.out.println(x:"Data Mahasiswa Setelah di Sorting");
        list.bubbleSort();
        list.tampil();

        input.close();
    }
}
```

Hasil Run

Masukkan Data Mahasiswa Ke-1 Masukkan NIM : 123 Masukkan Nama : Wahda Masukkan Kelas : 1B Masukkan IPK : 4,0	Data Mahasiswa Sebelum Sorting: Nama : Wahda NIM : 123 kelas : 1B IPK : 4.0	-----
Masukkan Data Mahasiswa Ke-2 Masukkan NIM : 124 Masukkan Nama : Galih Masukkan Kelas : 1C Masukkan IPK : 2,1	----- Nama : Galih NIM : 124 kelas : 1C IPK : 2.1	-----
Masukkan Data Mahasiswa Ke-3 Masukkan NIM : 125 Masukkan Nama : Sandy Masukkan Kelas : 1H Masukkan IPK : 1,1	----- Nama : Sandy NIM : 125 kelas : 1H IPK : 1.1	-----
Masukkan Data Mahasiswa Ke-4 Masukkan NIM : Aldy Masukkan Nama : Aldy Masukkan Kelas : 1G Masukkan IPK : 3,2	----- Nama : Aldy NIM : Aldy kelas : 1G IPK : 3.2	-----
Masukkan Data Mahasiswa Ke-5 Masukkan NIM : 127 Masukkan Nama : Wahyu Masukkan Kelas : 1I Masukkan IPK : 2,8	----- Nama : Wahyu NIM : 127 kelas : 1I IPK : 2.8	-----
		Data Mahasiswa Setelah di Sorting 1225 hasilnya Nama : Wahda NIM : 123 kelas : 1B IPK : 4.0 ----- Nama : Aldy NIM : Aldy kelas : 1G IPK : 3.2 ----- Nama : Wahyu NIM : 127 kelas : 1I IPK : 2.8 ----- Nama : Galih NIM : 124 kelas : 1C IPK : 2.1 ----- Nama : Sandy NIM : 125 kelas : 1H IPK : 1.1 -----

PERCOBAAN 3

1. Tambahkan method selectionsort di mahasiswaberprestasi24.java

```
void selectionSort() {  
    for(int i = 0; i < listMhs.length - 1; i++) {  
        int idxMin = i;  
        for(int j = i + 1; j < listMhs.length; j++) {  
            if(listMhs[j].ipk > listMhs[idxMin].ipk) {  
                idxMin = j;  
            }  
        }  
        Mahasiswa24 temp = listMhs[idxMin];  
        listMhs[idxMin] = listMhs[i];  
        listMhs[i] = temp;  
    }  
}
```

2. File mainSum

```
System.out.println(x:"Data Mahasiswa Setelah di Sorting menggunakan Selection sort");  
list.selectionSort();  
list.tampil();
```

3. Hasil Run

```
Data Mahasiswa Setelah di Sorting menggunakan Selection sort
Nama : Sita
NIM : 126
kelas : 2A
IPK : 3.9
-----
Nama : Miki
NIM : 127
kelas : 2A
IPK : 3.7
-----
Nama : Ayu
NIM : 124
kelas : 2A
IPK : 3.5
-----
Nama : Zidan
NIM : 123
kelas : 2A
IPK : 3.2
-----
Nama : Sofi
NIM : 125
kelas : 2A
IPK : 3.1
```

PERTANYAAN

1. idxmin di gunakan untuk menyimprn 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

```
void insertionSort() {
    for(int i = 1; i < listMhs.length; i++) {
        Mahasiswa24 temp = listMhs[i];
        int j = i;
        while (j > 0 && listMhs[j-1].ipk > temp.ipk) {
            listMhs[j] = listMhs[j-1];
            j--;
        }
        listMhs[j] = temp;
    }
}
```

2. Tambahkan pemanggilan method di mahasiswademo24.java

```
System.out.println(x:"Data Mahasiswa Setelah di Sorting menggunakan Insertion sort");
list.insertionSort();
list.tampil();
```

3. Hasil Run

```
-----
Data Mahasiswa Setelah di Sorting menggunakan Insertion sort
Nama   : Sofi
NIM    : 125
kelas  : 2A
IPK    : 3.1
-----
Nama   : Zidan
NIM    : 123
kelas  : 2A
IPK    : 3.2
-----
Nama   : Ayu
NIM    : 124
kelas  : 2A
IPK    : 3.5
-----
Nama   : Miki
NIM    : 127
kelas  : 2A
IPK    : 3.7
-----
Nama   : Sita
NIM    : 126
kelas  : 2A
IPK    : 3.9
-----
```

PERTANYAAN

1. Modifikasi kode agar bisa descending

```
void insertionSort() {
    for(int i = 1; i < listMhs.length; i++) {
        Mahasiswa24 temp = listMhs[i];
        int j = i;
        while (j > 0 && listMhs[j-1].ipk < temp.ipk) {
            listMhs[j] = listMhs[j-1];
            j--;
        }
        listMhs[j] = temp;
    }
}
```

Hasil run

```
-----
Data Mahasiswa Setelah di Sorting menggunakan Insertion sort
Nama   : Sita
NIM    : 126
kelas  : 2A
IPK    : 3.9
-----
Nama   : Miki
NIM    : 127
kelas  : 2A
IPK    : 3.7
-----
Nama   : Ayu
NIM    : 124
kelas  : 2A
IPK    : 3.5
-----
Nama   : Zidan
NIM    : 123
kelas  : 2A
IPK    : 3.2
-----
Nama   : Sofi
NIM    : 125
kelas  : 2A
IPK    : 3.1
-----
```


TUGAS

Kode Dosen24.java

```
jen24.java > Dosen24 > tampil()
public class Dosen24 {
    String kode, nama;
    boolean jenisKelamin;
    int usia;

    Dosen24(String kode, String nama, boolean jenisKelamin, int usia) {
        this.kode = kode;
        this.nama = nama;
        this.jenisKelamin = jenisKelamin;
        this.usia = usia;
    }

    void tampil() {
        System.out.println("Kode\t\t: " + kode);
        System.out.println("Nama\t\t: " + nama);
        System.out.println("Jenis Kelamin\t: " + (jenisKelamin ? "Perempuan" : "Laki Laki"));
        System.out.println("Usia\t\t: " + usia);
    }
}
```

Kode DataDosen24.java

```
public class DataDosen24 {
    Dosen24[] dataDosen = new Dosen24[10];
    int idx;

    void tambah(Dosen24 dosen) {
        if(dataDosen.length >= idx) {
            dataDosen[idx] = dosen;
            idx++;
        } else {
            System.out.println(x:"Data Sudah Penuh");
        }
    }

    void tampil() {
        for(Dosen24 dosen : dataDosen) {
            dosen.tampil();
            System.out.println(x:"-----");
        }
    }

    void sortingASCBubbleSort() {
        for(int i = 0; i < dataDosen.length - 1; i++) {
            for(int j = 1; j < dataDosen.length - i; j++) {
                if(dataDosen[j].usia < dataDosen[j-1].usia) {
                    Dosen24 tmp = dataDosen[j];
                    dataDosen[j] = dataDosen[j-1];
                    dataDosen[j-1] = tmp;
                }
            }
        }
    }

    void sortingDSCSelectionSort() {
        for(int i = 0; i < dataDosen.length - 1; i++) {
            int index = i;
            for(int j = i + 1; j < dataDosen.length; j++) {
                if(dataDosen[j].usia > dataDosen[index].usia) {
                    index = j;
                }
            }
            Dosen24 tmp = dataDosen[index];
            dataDosen[index] = dataDosen[i];
            dataDosen[i] = tmp;
        }
    }

    void sortingDSCInsertionSort() {
        for(int i = 1; i < dataDosen.length; i++) {
            Dosen24 tmp = dataDosen[i];
            int j = i;
            while(j > 0 && dataDosen[j-1].usia < tmp.usia) {
                dataDosen[j] = dataDosen[j-1];
                j--;
            }
            dataDosen[j] = tmp;
        }
    }
}
```

Kode DosenDemo24.java

```
public class DosenDemo24 {
    Run | Debug
    public static void main(String[] args) {

        DataDosen24 dataDosen = new DataDosen24();

        String data[][] = {
            {"d1", "dosen1", "true", "19"},
            {"d2", "dosen2", "false", "20"},
            {"d3", "dosen3", "true", "30"},
            {"d4", "dosen4", "false", "25"},
            {"d5", "dosen5", "false", "20"},
            {"d6", "dosen6", "true", "29"},
            {"d7", "dosen7", "true", "33"},
            {"d8", "dosen8", "true", "40"},
            {"d9", "dosen9", "false", "33"},
            {"d10", "dosen10", "true", "25"},
        };

        for (int i = 0; i < data.length; i++) {
            Dosen24 dosen = new Dosen24(
                data[i][0], data[i][1],
                Boolean.parseBoolean(data[i][2]),
                Integer.parseInt(data[i][3]));
            dataDosen.tambah(dosen);
        }

        dataDosen.tampil();

        System.out.println();

        System.out.println(x:"Data setelah di sorting asc dengan bubble sort");
        dataDosen.sortingASCBubbleSort();
        dataDosen.tampil();

        System.out.println();

        System.out.println(x:"Data setelah di sorting dsc dengan selection sort");
        dataDosen.sortingDSCSelectionSort();
        dataDosen.tampil();

        System.out.println();

        System.out.println(x:"Data setelah di sorting dsc dengan insertion sort");
        dataDosen.sortingDSCInsertionSort();
        dataDosen.tampil();
    }
}
```

Hasil Run

Sebelum Di sorting

```
Kode      : d1
Nama      : dosen1
Jenis Kelamin : Perempuan
Usia      : 19
-----
Kode      : d2
Nama      : dosen2
Jenis Kelamin : Laki Laki
Usia      : 20
-----
Kode      : d3
Nama      : dosen3
Jenis Kelamin : Perempuan
Usia      : 30
-----
Kode      : d4
Nama      : dosen4
Jenis Kelamin : Laki Laki
Usia      : 25
-----
Kode      : d5
Nama      : dosen5
Jenis Kelamin : Laki Laki
Usia      : 20
-----
Kode      : d6
Nama      : dosen6
Jenis Kelamin : Perempuan
Usia      : 29
-----
Kode      : d7
Nama      : dosen7
Jenis Kelamin : Perempuan
Usia      : 33
-----
Kode      : d8
Nama      : dosen8
Jenis Kelamin : Perempuan
Usia      : 40
-----
Kode      : d9
Nama      : dosen9
Jenis Kelamin : Laki Laki
Usia      : 33
-----
Kode      : d10
Nama      : dosen10
Jenis Kelamin : Perempuan
Usia      : 25
-----
```

Sorting ASC dengan bubble sort

```
Data setelah di sorting asc dengan bubble sort
Kode      : d1
Nama      : dosen1
Jenis Kelamin : Perempuan
Usia      : 19
-----
Kode      : d2
Nama      : dosen2
Jenis Kelamin : Laki Laki
Usia      : 20
-----
Kode      : d5
Nama      : dosen5
Jenis Kelamin : Laki Laki
Usia      : 20
-----
Kode      : d4
Nama      : dosen4
Jenis Kelamin : Laki Laki
Usia      : 25
-----
Kode      : d10
Nama      : dosen10
Jenis Kelamin : Perempuan
Usia      : 25
-----
Kode      : d6
Nama      : dosen6
Jenis Kelamin : Perempuan
Usia      : 29
-----
Kode      : d3
Nama      : dosen3
Jenis Kelamin : Perempuan
Usia      : 30
-----
Kode      : d7
Nama      : dosen7
Jenis Kelamin : Perempuan
Usia      : 33
-----
Kode      : d9
Nama      : dosen9
Jenis Kelamin : Laki Laki
Usia      : 33
-----
Kode      : d8
Nama      : dosen8
Jenis Kelamin : Perempuan
Usia      : 40
-----
```

Sorting DSC dengan Selection sort

Data setelah di sorting dsc dengan selection sort	
Kode	: d8
Nama	: dosen8
Jenis Kelamin	: Perempuan
Usia	: 40

Kode	: d7
Nama	: dosen7
Jenis Kelamin	: Perempuan
Usia	: 33

Kode	: d9
Nama	: dosen9
Jenis Kelamin	: Laki Laki
Usia	: 33

Kode	: d3
Nama	: dosen3
Jenis Kelamin	: Perempuan
Usia	: 30

Kode	: d6
Nama	: dosen6
Jenis Kelamin	: Perempuan
Usia	: 29

Kode	: d10
Nama	: dosen10
Jenis Kelamin	: Perempuan
Usia	: 25

Kode	: d4
Nama	: dosen4
Jenis Kelamin	: Laki Laki
Usia	: 25

Kode	: d2
Nama	: dosen2
Jenis Kelamin	: Laki Laki
Usia	: 20

Kode	: d5
Nama	: dosen5
Jenis Kelamin	: Laki Laki
Usia	: 20

Kode	: d1
Nama	: dosen1
Jenis Kelamin	: Perempuan
Usia	: 19

Sorting DSC dengan Insertion sort

Data setelah di sorting dsc dengan insertion sort	
Kode	: d8
Nama	: dosen8
Jenis Kelamin	: Perempuan
Usia	: 40

Kode	: d7
Nama	: dosen7
Jenis Kelamin	: Perempuan
Usia	: 33

Kode	: d9
Nama	: dosen9
Jenis Kelamin	: Laki Laki
Usia	: 33

Kode	: d3
Nama	: dosen3
Jenis Kelamin	: Perempuan
Usia	: 30

Kode	: d6
Nama	: dosen6
Jenis Kelamin	: Perempuan
Usia	: 29

Kode	: d10
Nama	: dosen10
Jenis Kelamin	: Perempuan
Usia	: 25

Kode	: d4
Nama	: dosen4
Jenis Kelamin	: Laki Laki
Usia	: 25

Kode	: d2
Nama	: dosen2
Jenis Kelamin	: Laki Laki
Usia	: 20

Kode	: d5
Nama	: dosen5
Jenis Kelamin	: Laki Laki
Usia	: 20

Kode	: d1
Nama	: dosen1
Jenis Kelamin	: Perempuan
Usia	: 19
