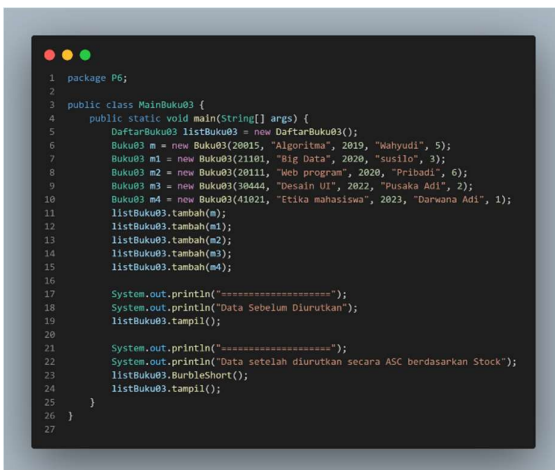
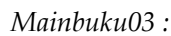
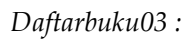


*FILE NAME =03_Alexander Agung Raya _1F_P6

SORTING (BUBBLE, SELECTION, DAN INSERTION SORT)

Buku03 :





NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SORTING (BUBBLE, SELECTION, DAN INSERTION SORT)

Hasil :

```
Judul Buku : Web program
Tahun Terbit : 2020
Pengarang : Prihadi joko
Stock : 6

=====
Kode Buku : 30444
Judul Buku : Desain UI
Tahun Terbit : 2022
Pengarang : Pusaka Adi
Stock : 2

=====
Kode Buku : 41021
Judul Buku : Etika mahasiswa
Tahun Terbit : 2023
Pengarang : Darwana Adi
Stock : 1

=====
Data setelah diurutkan secara ASC berdasarkan Stock
=====
Kode Buku : 41021
Judul Buku : Etika mahasiswa
Tahun Terbit : 2023
Pengarang : Darwana Adi
Stock : 1
```

Pertanyaan :

1. Terdapat di method apakah proses bubble sort?

Jawab :

Bubble Sort adalah algoritma pengurutan sederhana yang berulang kali melintasi daftar, membandingkan setiap elemen secara berpasangan, dan menukar elemen yang berdekatan jika mereka berada dalam urutan yang salah.

2. Di dalam method bubbleSort(), terdapat proses swapping, jelaskan proses tersebut!

Jawab :

Pengecekan kondisi: Pada setiap iterasi loop dalam, kondisi if memeriksa apakah elemen pada indeks j lebih besar dari elemen pada indeks $j+1$. Jika kondisi ini terpenuhi, maka artinya elemen pada indeks j harus ditukar dengan elemen pada indeks $j+1$ agar array berurutan dengan benar.

Penyimpanan sementara: Sebelum melakukan pertukaran, nilai elemen pada indeks j disimpan sementara dalam variabel bantu (temporary variable) untuk menghindari kehilangan data. Ini dilakukan dengan pernyataan `Buku03 tempBk = listBK[j];`.

Pertukaran: Setelah nilai elemen pada indeks j disimpan sementara, nilai elemen pada indeks $j+1$ ditransfer ke indeks j . Hal ini dilakukan dengan pernyataan `listBK[j] = listBK[j + 1];`.

Pemulihan nilai: Kemudian, nilai elemen yang disimpan sementara (`tempBk`) dipindahkan ke indeks $j+1$. Ini dilakukan dengan pernyataan `listBK[j + 1] = tempBk;`.

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

```
for (int i = 0; i < listBK.length - 1; i++) {
    for (int j = 0; j < listBK.length-i-1; j++) {
```

Jawab :

Perulangan i: Digunakan untuk mengulangi proses pengurutan untuk setiap elemen dalam daftar. Perulangan ini akan berhenti ketika nilai i mencapai `listBK.length-1`, yang berarti elemen terakhir dalam daftar telah diurutkan.

Perulangan j: Digunakan untuk membandingkan elemen dalam daftar dan menukar posisinya jika tidak urut. Perulangan ini akan berhenti ketika nilai j mencapai `listBK.length-i-1`, yang berarti elemen terakhir dalam daftar telah dibandingkan dengan semua elemen sebelumnya.



NAMA : Alexander Agung Raya

NIM : 2341720040

NO ABSEN : 03

KELAS : 1F

MATERI : SORTING (BUBBLE, SELECTION, DAN INSERTION SORT)

b. Syarat $i < \text{listBk.length} - 1$:

Syarat ini memastikan bahwa perulangan i tidak melebihi batas daftar. Jika perulangan i dijalankan sampai listBk.length , maka akan terjadi error karena tidak ada elemen pada index listBk.length .

c. Syarat $j < \text{listBk.length} - i - 1$:

Syarat ini memastikan bahwa perulangan j tidak membandingkan elemen yang sama dua kali. Contohnya, jika $i=0$, maka perulangan j akan membandingkan elemen ke-0 dengan elemen ke-1, ke-2, dan seterusnya. Jika syarat $j < \text{listBk.length} - i - 1$ tidak digunakan, maka perulangan j akan membandingkan elemen ke-0 dengan elemen ke-0 lagi, dan seterusnya, yang tidak diperlukan.

Penggunaan -i:

Penggunaan -i pada kode tersebut adalah untuk menyesuaikan batas perulangan j . Pada setiap iterasi perulangan i , elemen terakhir dalam daftar sudah diurutkan, sehingga tidak perlu dibandingkan lagi. Oleh karena itu, batas perulangan j dikurangi dengan i untuk menghindari perbandingan yang tidak perlu.

Jika -i dihilangkan:

Kode program tidak akan error, tetapi efisiensi program akan berkurang. Perulangan j akan melakukan perbandingan yang tidak perlu pada elemen terakhir dalam daftar pada setiap iterasi perulangan i .

d. Perulangan i dan tahap bubble sort:

Jika banyak data di dalam listBk adalah 50, maka perulangan i akan berlangsung sebanyak 49 kali.

Tahap bubble sort adalah jumlah iterasi perulangan i yang diperlukan untuk mengurutkan seluruh daftar. Dalam kasus ini, tahap bubble sort sama dengan jumlah perulangan i , yaitu 49.

Percobaan 2 :

DaftarBuku03 :

```
1 package PG;
2
3 public class DaftarBuku03 {
4     Buku03 listBK[] = new Buku03[5];
5     int idx;
6
7     void tambah(Buku03 m) {
8         if (idx < listBK.length) {
9             listBK[idx] = m;
10            idx++;
11        } else {
12            System.out.println("Data Sudah Penuh");
13        }
14    }
15
16    void tampil() {
17        for (Buku03 m : listBK) {
18            m.TampilDataBuku03();
19        }
20    }
21
22    void BubbleSort() {
23        for (int i = 0; i < listBK.length - 1; i++) {
24            for (int j = 0; j < listBK.length - i - 1; j++) {
25                if (listBK[j].stock > listBK[j + 1].stock) {
26                    Buku03 tempbk = listBK[j];
27                    listBK[j] = listBK[j + 1];
28                    listBK[j + 1] = tempbk;
29                }
30            }
31        }
32    }
33
34    void selectionSort() {
35        for (int i = 0; i < listBK.length - 1; i++) {
36            int idmx = i;
37            for (int j = i + 1; j < listBK.length; j++) {
38                if (listBK[j].stock > listBK[idmx].stock) {
39                    idmx = j;
40                }
41            }
42            Buku03 tempbkukm = listBK[idmx];
43            listBK[idmx] = listBK[i];
44            listBK[i] = tempbkukm;
45        }
46    }
47 }
```



NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SORTING (BUBBLE, SELECTION, DAN INSERTION SORT)
MainBuku03 :

```
1 // MainBuku03.java
2
3 import java.util.*;
4
5 public class MainBuku03 {
6     public static void main(String[] args) {
7         Scanner scanner = new Scanner(System.in);
8         List<Book> listBuku = new ArrayList<>();
9         Book b = new Book(1001, "Algoritma", 2019, "Mahyudi", 5);
10        listBuku.add(b);
11        b = new Book(1002, "Web Program", 2020, "Priadi", 6);
12        listBuku.add(b);
13        b = new Book(1003, "Big Data", 2020, "Susilo", 3);
14        listBuku.add(b);
15        b = new Book(1004, "Desain UI", 2022, "Adi", 2);
16        listBuku.add(b);
17
18        // Menampilkan daftar buku
19        System.out.println("Daftar Buku:");
20        for (Book b : listBuku) {
21            System.out.println(b);
22        }
23
24        // Melakukan sorting menggunakan Selection Sort
25        SelectionSort ss = new SelectionSort();
26        listBuku = ss.sort(listBuku);
27
28        // Menampilkan daftar buku setelah diurutkan
29        System.out.println("Daftar Buku setelah diurutkan:");
30        for (Book b : listBuku) {
31            System.out.println(b);
32        }
33    }
34}
```

Hasil :

```
TERMINAL
kode Buku : 30444
Judul Buku : Desain UI
Tahun Terbit : 2022
Pengarang : Pusaka Adi
Stock : 2
=====
kode Buku : 21101
Judul Buku : Big Data
Tahun Terbit : 2020
Pengarang : susilo
Stock : 3
=====
kode Buku : 20015
Judul Buku : Algoritma
Tahun Terbit : 2019
Pengarang : Mahyudi
Stock : 5
=====
kode Buku : 20111
Judul Buku : Web program
Tahun Terbit : 2020
Pengarang : Priadi joko
Stock : 6
```

Pertanyaan :

Di dalam method selection sort, jelaskan proses manakah yang melakukan pencarian stock terkecil dari listBuku yang ada! Jelaskan berdasarkan kode yang sudah Anda Buat!

Jawab :

Di looping menggunakan looping

Loop Luar (Pencarian untuk Setiap Elemen):

for (int i = 0; i < n-1; i++)

Loop ini melintasi semua elemen kecuali elemen terakhir karena elemen terakhir sudah pasti berada di posisi yang benar setelah operasi pengurutan.

Variabel n adalah jumlah total buku dalam listBuku03.

Penentuan Indeks Stok Terkecil:

int minIndex = i;

Pada setiap iterasi loop luar, kita asumsikan bahwa indeks saat ini (i) memiliki stok terkecil.

Loop Dalam (Pencarian Stok Terkecil):

for (int j = i+1; j < n; j++)

Loop ini dimulai dari indeks berikutnya (i+1) hingga elemen terakhir dari array.

Tujuannya adalah untuk mencari stok terkecil setelah indeks saat ini (i).

Pembandingan Stok:

if (listBK[j].stock < listBK[minIndex].stock)

Dalam loop dalam, dilakukan perbandingan antara stok buku pada indeks j dengan stok buku pada minIndex.



NAMA : Alexander Agung Raya

NIM : 2341720040

NO ABSEN : 03

KELAS : 1F

MATERI : SORTING (BUBBLE, SELECTION, DAN INSERTION SORT)

Jika stok buku pada indeks j lebih kecil dari stok buku pada minIndex, maka minIndex diperbarui menjadi j.

Pertukaran Stok Terkecil:

Buku03 temp = listBK[minIndex];

listBK[minIndex] = listBK[i];

listBK[i] = temp;

Setelah loop dalam selesai, jika minIndex telah berubah dari i, artinya kita menemukan stok terkecil setelah indeks i.

Langkah ini bertujuan untuk menukar elemen pada minIndex dengan elemen pada i, sehingga elemen dengan stok terkecil akan dipindahkan ke posisi yang benar.

Percobaan 3 :

Daftarbuku03 :

```
1 package P6;
2
3 public class DaftarBuku03 {
4     public static void main(String[] args) {
5         DaftarBuku03 listBK = new DaftarBuku03();
6         listBK.tambah(m1);
7         listBK.tambah(m2);
8         listBK.tambah(m3);
9         listBK.tambah(m4);
10        listBK.tampilkan();
11    }
12
13    void tambah(Buku03 m) {
14        listBK.add(m);
15    }
16
17    void tampilkan() {
18        System.out.println("Data Sebelum Diurutkan");
19        listBK.tampilkan();
20    }
21
22    void bubbleSort() {
23        for (int i = 0; i < listBK.size() - 1; i++) {
24            for (int j = i + 1; j < listBK.size(); j++) {
25                if (listBK.get(i).getStok() > listBK.get(j).getStok()) {
26                    Buku03 temp = listBK.get(i);
27                    listBK.set(i, listBK.get(j));
28                    listBK.set(j, temp);
29                }
30            }
31        }
32    }
33
34    void selectionSort() {
35        for (int i = 0; i < listBK.size() - 1; i++) {
36            int minIndex = i;
37            for (int j = i + 1; j < listBK.size(); j++) {
38                if (listBK.get(minIndex).getStok() > listBK.get(j).getStok()) {
39                    minIndex = j;
40                }
41            }
42            Buku03 temp = listBK.get(i);
43            listBK.set(i, listBK.get(minIndex));
44            listBK.set(minIndex, temp);
45        }
46    }
47
48    void insertionSort() {
49        for (int i = 1; i < listBK.size(); i++) {
50            Buku03 temp = listBK.get(i);
51            int j = i - 1;
52            while (j > 0 && listBK.get(j).getStok() > temp.getStok()) {
53                listBK.set(j + 1, listBK.get(j));
54            }
55            listBK.set(j + 1, temp);
56        }
57    }
58 }
```

Mainbuku03 :

```
1 package P6;
2
3 public class MainBuku03 {
4     public static void main(String[] args) {
5         DaftarBuku03 listBuku03 = new DaftarBuku03();
6         Buku03 m = new Buku03(20015, "Algoritma", 2019, "Mahyudi", 5);
7         Buku03 m1 = new Buku03(21101, "Big Data", 2020, "Huslilo", 3);
8         Buku03 m2 = new Buku03(20111, "Web program", 2020, "Prihadi Joko", 6);
9         Buku03 m3 = new Buku03(30444, "Desain UI", 2022, "Pustaka Adi", 2);
10        Buku03 m4 = new Buku03(41021, "Etika mahasiswa", 2023, "Darwana Adi", 1);
11        listBuku03.tambah(m);
12        listBuku03.tambah(m1);
13        listBuku03.tambah(m2);
14        listBuku03.tambah(m3);
15        listBuku03.tambah(m4);
16
17        System.out.println("=====");
18        System.out.println("Data Sebelum Diurutkan");
19        listBuku03.tampilkan();
20
21        System.out.println("=====");
22        System.out.println("Data setelah diurutkan secara ASC berdasarkan Stock");
23        listBuku03.bubbleSort();
24        listBuku03.tampilkan();
25
26        System.out.println("=====");
27        System.out.println("Data setelah diurutkan secara Desc berdasarkan Stock menggunakan selection sort");
28        listBuku03.selectionSort();
29        listBuku03.tampilkan();
30
31        System.out.println("=====");
32        System.out.println("Data setelah diurutkan secara ASC berdasarkan Stock menggunakan Insertion sort");
33        listBuku03.insertionSort();
34        listBuku03.tampilkan();
35    }
36 }
37 }
```



NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SORTING (BUBBLE, SELECTION, DAN INSERTION SORT)

Latihan :

Hotel03 :

```
1 package P6.Travel;
2
3 public class Hotel03 {
4     String nama;
5     int harga;
6     int kamar;
7     byte bintang;
8
9     public Hotel03(String n, int h, int k, byte b) {
10         this.nama = n;
11         this.harga = h;
12         this.kamar = k;
13         this.bintang = (byte) b;
14     }
15
16     public void tampilDetailHotel() {
17         System.out.println("-----");
18         System.out.println("Nama Hotel : " + nama);
19         System.out.println("Harga : " + harga);
20         System.out.println("Kamar : " + kamar);
21         System.out.println("Bintang : " + bintang);
22     }
23 }
```

Hotelservice03 :

```
1 package P6.Travel;
2
3 import java.util.*;
4
5 public class HotelService03 {
6     private List<Hotel03> daftarHotel;
7
8     public HotelService03() {
9         daftarHotel = new ArrayList<>();
10     }
11
12     public void tambah(Hotel03 h) {
13         daftarHotel.add(h);
14     }
15
16     public void tampilDetailHotel() {
17         System.out.println("-----");
18         for (Hotel03 h : daftarHotel) {
19             h.tampilDetailHotel();
20         }
21     }
22
23     public void bubbleSort() {
24         for (int i = daftarHotel.size() - 1; i > 0; i--) {
25             for (int j = 0; j < i; j++) {
26                 if (daftarHotel.get(j).harga > daftarHotel.get(j+1).harga) {
27                     Hotel03 temp = daftarHotel.get(j);
28                     daftarHotel.set(j, daftarHotel.get(j+1));
29                     daftarHotel.set(j+1, temp);
30                 }
31             }
32         }
33     }
34
35     public void selectionSort() {
36         for (int i = daftarHotel.size() - 1; i > 0; i--) {
37             int min = i;
38             for (int j = i + 1; j < daftarHotel.size(); j++) {
39                 if (daftarHotel.get(j).harga < daftarHotel.get(min).harga) {
40                     min = j;
41                 }
42             }
43             Hotel03 temp = daftarHotel.get(i);
44             daftarHotel.set(i, daftarHotel.get(min));
45             daftarHotel.set(min, temp);
46         }
47     }
48 }
```

Mainhotel03 :

```
1 package P6.Travel;
2 import P6.BUKU03.DaftarBuku03;
3
4 public class MainHotel03 {
5     public static void main(String[] args) {
6         HotelService03 kamar = new HotelService03();
7         Hotel03 k = new Hotel03("Hotel China", "Banyuwangi", 500000, 5);
8         Hotel03 k1 = new Hotel03("Hotel Kembang", "Jogja", 400000, 4);
9         Hotel03 k2 = new Hotel03("Hotel bibit", "Jakarta", 300000, 3);
10        Hotel03 k3 = new Hotel03("Hotel Crab", "Bikini Bottom", 200000, 2);
11        Hotel03 k4 = new Hotel03("Hotel hantu", "Surabaya", 100000, 1);
12        kamar.tambah(k);
13        kamar.tambah(k1);
14        kamar.tambah(k2);
15        kamar.tambah(k3);
16        kamar.tambah(k4);
17
18        System.out.println("=====");
19        System.out.println("Data Sebelum Diurutkan");
20        kamar.tampilAll();
21        System.out.println();
22
23        System.out.println("=====");
24        System.out.println("Data setelah diurutkan secara ASC berdasarkan Stock");
25        kamar.bubbleSortHarga();
26        kamar.tampilAll();
27        System.out.println();
28
29        System.out.println("=====");
30        System.out.println("Data setelah diurutkan secara Desc berdasarkan Stock menggunakan selection short");
31        kamar.selectionSortHarga();
32        kamar.tampilAll();
33
34    }
35 }
36 }
37 }
```



NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SORTING (BUBBLE, SELECTION, DAN INSERTION SORT)

Hasil :

```
Harga :400000
Rating :4
=====
Nama Hotel : Hotel bibit
Kota :jakarta
Harga :300000
Rating :3
=====
Nama Hotel : Hotel Crab
Kota :Bikini Bottom
Harga :200000
Rating :2
=====
Nama Hotel : Hotel hantu
Kota :Surabaya
Harga :100000
Rating :1
=====
Data setelah diurutkan secara ASC berdasarkan Stock
=====
Nama Hotel : Hotel hantu
Kota :Surabaya
Harga :100000
Rating :1
=====
Nama Hotel : Hotel Crab
Kota :Bikini Bottom
Harga :200000
```