

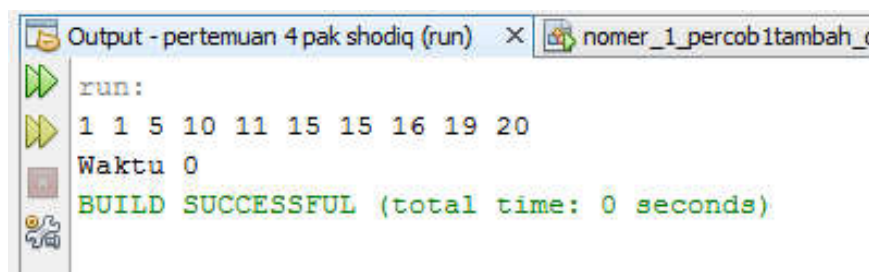
# PERCOBAAN

Source code

Nomer 1 : penggunaan insertion sort secara ascending

```
4  L  */
5      package PERCOBAAN;
6
7  L  /**
8      *
9      * @author Tomy
10     */
11     public class nomer_1_insertsecara_ascending {
12
13
14         public static <T extends Comparable<? super T>> void
15     insertionSort(T[] arr){
16         for (int i=arr.length-1 ; i>=0 ; i--) {
17             for (int j = i+1, k = i; j <arr.length ; j++){
18                 if (arr[j].compareTo(arr[k]) > 0) {
19                     break;
20                 } else {
21                     T temp = arr[k];
22                     arr[k] = arr[j];
23                     arr[j] = temp;
24                     k = j;
25                 }
26             }
27         }
28     }
29
30     public static <T> void tampil(T data[]) {
31         for (T objek : data) {
32             System.out.print(objek + " ");
33         }
34         System.out.println("");
35     }
36
37     public static void main(String[] args) {
38         Integer data[] = new Integer[10];
39         for(int a=0 ; a<data.length ; a++)
40         {
41             data[a]= (int) (Math.random ()*20+1);
42         }
43         long awal = System.currentTimeMillis();
44         insertionSort(data);
45         long sisaWaktu = System.currentTimeMillis() - awal;
46         tampil(data);
47         System.out.println("Waktu " + sisaWaktu);
48     }
49 }
```

Hasil run ( output ) :

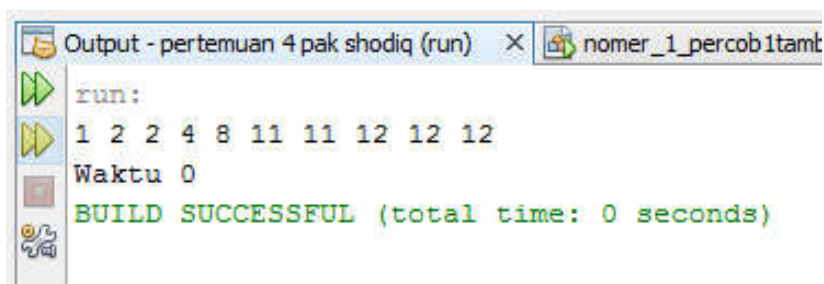


```
Output - pertemuan 4 pak shodiq (run) X  nomer_1_percoba1tambah_c
run:
1 1 5 10 11 15 15 16 19 20
Waktu 0
BUILD SUCCESSFUL (total time: 0 seconds)
```

## Nomer 2 : penggunaan selection sort secara ascending

```
4  L  */
5      package PERCOBAAN;
6
7  L  /**
8      *
9      * @author Tomy
10     */
11     public class nomer_2_selectsecara_ascending {
12
13
14         public static <T extends Comparable<? super T>> void
15     selectionSort(T[] arr) {
16         T temp;
17         for(int i=arr.length-1 ; i>=0 ; i--){
18             int max = i;
19             for(int j=i-1 ; j>=0 ; j--){
20                 if(arr[j].compareTo(arr[max]) > 0)
21                     max = j;
22             }
23             temp = arr[i];
24             arr[i] = arr[max];
25             arr[max] = temp;
26         }
27     }
28
29     public static <T> void tampil(T data[]) {
30         for (T objek : data) {
31             System.out.print(objek + " ");
32         }
33         System.out.println("");
34     }
35
36     /**
37     * @param args the command line arguments
38     */
39     public static void main(String[] args) {
40
41         Integer data[] = new Integer[10];
42         for(int a=0 ; a<data.length ; a++){
43             data[a]= (int) (Math.random()*13+1);
44         }
45         long awal = System.currentTimeMillis();
46         selectionSort(data);
47         long sisaWaktu = System.currentTimeMillis() - awal;
48         tampil(data);
49         System.out.println("Waktu " + sisaWaktu);// TODO code applica
50     }
```

Hasil run ( output ) :



```
Output - pertemuan 4 pak shodiq (run) X nomer_1_percoba1tamt
run:
1 2 2 4 8 11 11 12 12 12
Waktu 0
BUILD SUCCESSFUL (total time: 0 seconds)
```

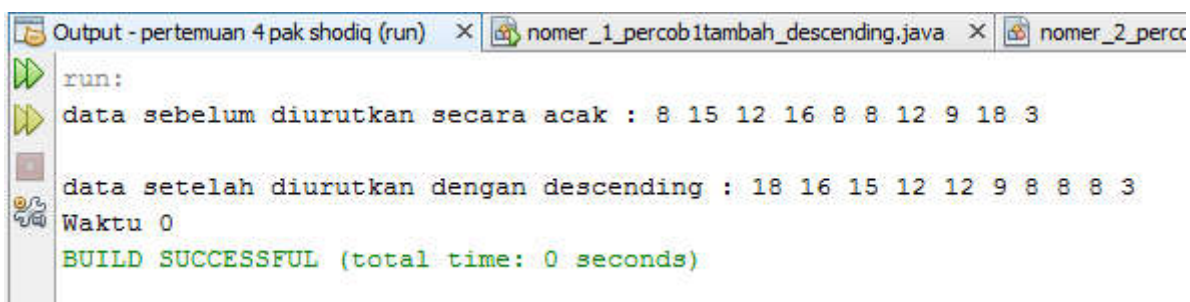
## LATIHAN

Source code

Nomer 1 : tambahkan descending pada percobaan nomer 1 ( insertion sort )

```
4  */
5  package LATIHAN;
6
7  /**
8   *
9   * @author Tomy
10  */
11  public class nomer_1_percobitambah_descending {
12
13      public static <T extends Comparable<? super T>> void
14      insertionSort(T[] arr){
15          for (int i=arr.length-1 ; i>=0 ; i--) {
16              for (int j = i+1, k = i; j <arr.length ; j++){
17                  if (arr[j].compareTo(arr[k]) < 0) {
18                      break;
19                  } else {
20                      T temp = arr[k];
21                      arr[k] = arr[j];
22                      arr[j] = temp;
23                      k = j;
24                  }
25              }
26          }
27      }
28
29      public static <T> void tampil(T data[]) {
30          for (T objek : data) {
31              System.out.print(objek + " ");
32          }
33          System.out.println("");
34      }
35
36      public static void main(String[] args) {
37          Integer data[] = new Integer[10];
38          for(int a=0 ; a<data.length ; a++)
39          {
40              data[a]= (int) (Math.random()*20+1);
41          }
42          System.out.print("data sebelum diurutkan secara acak : ");
43          tampil(data);
44          System.out.println(" ");
45          System.out.print("data setelah diurutkan dengan descending : ");
46
47          long awal = System.currentTimeMillis();
48          insertionSort(data);
49
50          long sisaWaktu = System.currentTimeMillis() - awal;
51          tampil(data);
52          System.out.println("Waktu " + sisaWaktu);
53      }
54  }
```

Hasil run ( output ) :

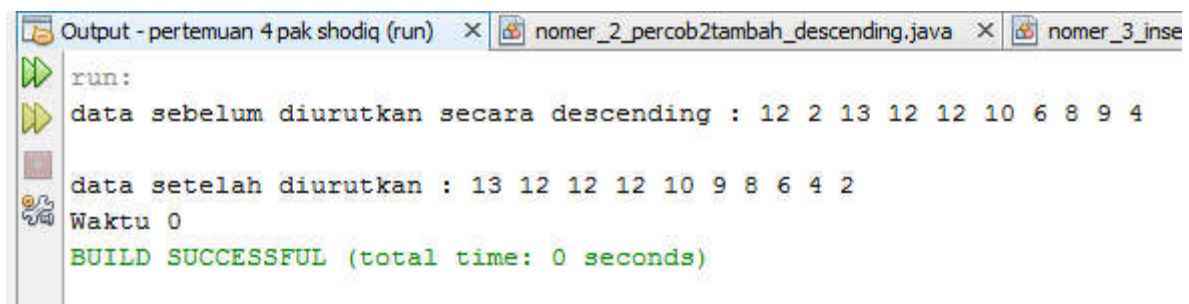


```
run:
data sebelum diurutkan secara acak : 8 15 12 16 8 8 12 9 18 3
data setelah diurutkan dengan descending : 18 16 15 12 12 9 8 8 8 3
Waktu 0
BUILD SUCCESSFUL (total time: 0 seconds)
```

Nomer 2 : tambahkan descending pada percobaan nomer 2 ( selection sort )

```
4  L  */
5  package LATIHAN;
6
7  /**
8   *
9   * @author Tomy
10  */
11  public class nomer_2_percob2tambah_descending {
12  public static <T extends Comparable<? super T>> void
13  selectionSort(T[] arr) {
14      T temp;
15      for(int i=arr.length-1 ; i>=0 ; i--){
16          int max = i;
17          for(int j=i-1 ; j>=0 ; j--){
18              if(arr[j].compareTo(arr[max]) < 0)
19                  max = j;
20          }
21          temp = arr[i];
22          arr[i] = arr[max];
23          arr[max] = temp;
24      }
25  }
26
27  public static <T> void tampil(T data[]) {
28      for (T objek : data) {
29          System.out.print(objek + " ");
30      }
31      System.out.println("");
32  }
33
34  /**
35   * @param args the command line arguments
36   */
37
38  public static void main(String[] args) {
39
40      Integer data[] = new Integer[10];
41      for(int a=0 ; a<data.length ; a++){
42          data[a]= (int) (Math.random() *13+1);
43      }
44      System.out.print("data sebelum diurutkan secara descending : ");
45      tampil(data);
46      System.out.println(" ");
47      System.out.print("data setelah diurutkan : ");
48
49      long awal = System.currentTimeMillis();
50      selectionSort(data);
51      long sisaWaktu = System.currentTimeMillis() - awal;
52      tampil(data);
53      System.out.println("Waktu " + sisaWaktu);// TODO code application
54  }
```

Hasil run ( output ) :



```
Output - pertemuan 4 pak shodiq (run) X nomer_2_percob2tambah_descending.java X nomer_3_inse
run:
data sebelum diurutkan secara descending : 12 2 13 12 12 10 6 8 9 4
data setelah diurutkan : 13 12 12 12 10 9 8 6 4 2
Waktu 0
BUILD SUCCESSFUL (total time: 0 seconds)
```

Nomer 3 : program insertion sort dengan ascending dan descending

```
4  L  */
5      package LATIHAN;
6  [ ] import java.util.Scanner;
7  [ ] /**
8      *
9      * @author Tomy
10     */
11     public class nomer_3_insertion_ascendingdescending {
12         public static <T extends Comparable<? super T>> void
13     [ ] insertionSort(T[] arr, int pilih){
14         if (pilih==1){
15             System.out.println("---urutan secara ascending---");
16             for (int i=arr.length-1 ; i>=0 ; i--) {
17                 for (int j = i+1, k = i; j <arr.length ; j++){
18                     if (arr[j].compareTo(arr[k]) > 0) {
19                         break;
20                     } else {
21                         T temp = arr[k];
22                         arr[k] = arr[j];
23                         arr[j] = temp;
24                         k = j;
25                     }
26                 }
27             }
28         }
29
30         else{
31             System.out.println("---urutan secara descending---");
32             for (int i=arr.length-1 ; i>=0 ; i--) {
33                 for (int j = i+1, k = i; j <arr.length ; j++){
34                     if (arr[j].compareTo(arr[k]) < 0) {
35                         break;
36                     } else {
37                         T temp = arr[k];
38                         arr[k] = arr[j];
39                         arr[j] = temp;
40                         k = j;
41                     }
42                 }
43             }
44         }
45     }
46     [ ] /**
47         * @param args the command line arguments
48         */
49     [ ] public static <T> void tampil(T data[]) {
50         for (T objek : data) {
51             System.out.print(objek + " ");
52         }
53         System.out.println("");
54     }
55     [ ] public static void main(String[] args) {
56         System.out.println("menu pilihan insertion sort : "
57             + "\n 1. urutan secara ascending "
58             + "\n 2. urutan secara descending ");
59         Scanner pilih = new Scanner(System.in);
60         Integer data[] = new Integer[10];
61         for(int a=0 ; a<data.length ; a++)
62         {
63             data[a]= (int) (Math.random ()*20+1);
64         }
65         System.out.println(" ");
```

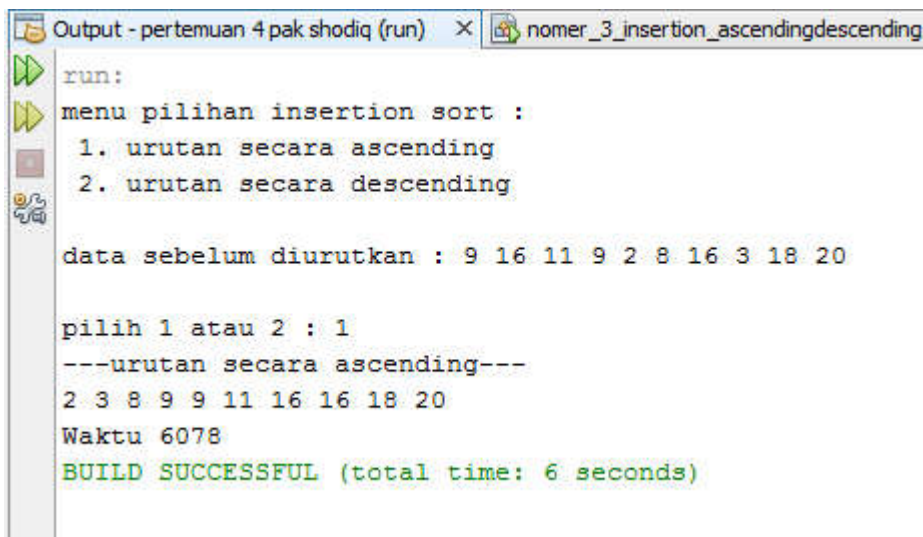
```

66     System.out.print("data sebelum diurutkan : ");
67     tampil(data);
68     System.out.println(" ");
69     System.out.print("pilih 1 atau 2 : ");
70     long awal = System.currentTimeMillis();
71     insertionSort(data,pilih.nextInt());
72
73     long sisaWaktu = System.currentTimeMillis() - awal;
74     tampil(data);
75     System.out.println("Waktu " + sisaWaktu);
76 }
77 }

```

Hasil run ( output ) :

Urutan secara ascending



```

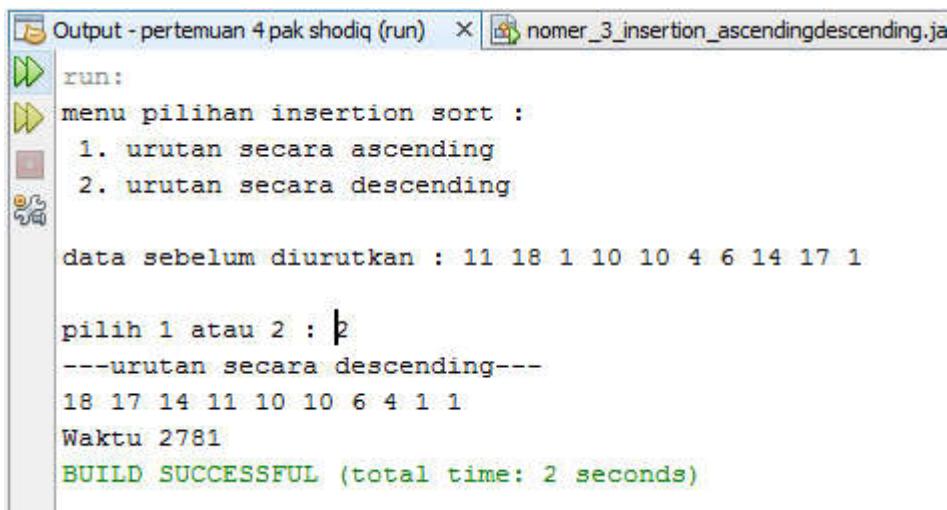
Output - pertemuan 4 pak shodiq (run)  X  nomer_3_insertion_ascendingdescending.ja
run:
menu pilihan insertion sort :
  1. urutan secara ascending
  2. urutan secara descending

data sebelum diurutkan : 9 16 11 9 2 8 16 3 18 20

pilih 1 atau 2 : 1
---urutan secara ascending---
2 3 8 9 9 11 16 16 18 20
Waktu 6078
BUILD SUCCESSFUL (total time: 6 seconds)

```

Urutan secara descending



```

Output - pertemuan 4 pak shodiq (run)  X  nomer_3_insertion_ascendingdescending.ja
run:
menu pilihan insertion sort :
  1. urutan secara ascending
  2. urutan secara descending

data sebelum diurutkan : 11 18 1 10 10 4 6 14 17 1

pilih 1 atau 2 : 2
---urutan secara descending---
18 17 14 11 10 10 6 4 1 1
Waktu 2781
BUILD SUCCESSFUL (total time: 2 seconds)

```

Analisa :

Untuk mengurutkan angka dari ascending dan descending dalam satu program maka harus menggunakan inputan dari keyboard, misalnya seperti yang saya gunakan yaitu scanner. Dengan begitu kita bisa memilih apakah ingin menggunakan urutan ascending ataupun descending. Dengan memasukkan fungsi if else. Yang pertama if, pada fungsi if saya memasukkan fungsi atau method untuk mengurutkan secara ascending dan pada fungsi else saya memasukkan fungsi method descending, maka dengan begitu jika kita pilih 1 maka yang muncul adalah urutan secara ascending dan sebaliknya.



Nomer 4 : program selection sort dengan ascending dan descending

```
4  L  */
5      package LATIHAN;
6
7  □ import java.util.Scanner;
8
9  □ /**
10     *
11     * @author Tomy
12     */
13     public class nomer_4_selection_ascendingdescending {
14
15
16         public static <T extends Comparable<? super T>> void
17     □ selectionSort(T[] arr,int pilih) {
18         T temp;
19         if(pilih==1){
20             System.out.println("----secara ascending----");
21             for(int i=arr.length-1 ; i>=0 ; i--){
22                 int max = i;
23                 for(int j=i-1 ; j>=0 ; j--){
24                     if(arr[j].compareTo(arr[max]) > 0)
25                         max = j;
26                 }
27                 temp = arr[i];
28                 arr[i] = arr[max];
29                 arr[max] = temp;
30             }
31         }
32         else{
33             System.out.println("----secara descending----");
34             for(int i=arr.length-1 ; i>=0 ; i--){
35                 int max = i;
36                 for(int j=i-1 ; j>=0 ; j--){
37                     if(arr[j].compareTo(arr[max]) < 0)
38                         max = j;
39                 }
40                 temp = arr[i];
41                 arr[i] = arr[max];
42                 arr[max] = temp;
43             }
44         }
45     }
46     □ public static <T> void tampil(T data[]) {
47         for (T objek : data) {
48             System.out.print(objek + " ");
49         }
50         System.out.println("");
51     }
52     □ /**
53         * @param args the command line arguments
54         */
55     □ public static void main(String[] args) {
56         System.out.println("menu pilihan selection sort : "
57             + "\n 1. urutan secara ascending| "
58             + "\n 2. urutan secara descending ");
59         Scanner pilih = new Scanner(System.in);
60         Integer data[] = new Integer[10];
61         for(int a=0 ; a<data.length ; a++){
62             data[a]= (int) (Math.random()*13+1);
63         }
64         System.out.println(" ");
65         System.out.print("data sebelum diurutkan : ");
66         tampil(data);
```

```

67     System.out.println(" ");
68     System.out.print("pilih 1 atau 2 : ");
69     long awal = System.currentTimeMillis();
70     selectionSort(data,pilih.nextInt());
71     long sisaWaktu = System.currentTimeMillis() - awal;
72     tampil(data);
73     System.out.println("Waktu " + sisaWaktu);// TODO code applica
74 }
75 }

```

Hasil run ( output ) :

Urutan secara ascending

```

Output - pertemuan 4 pak shodiq (run) X nomer_4_selection_ascendingdescending.java
run:
menu pilihan selection sort :
  1. urutan secara ascending
  2. urutan secara descending

data sebelum diurutkan : 5 2 7 9 1 11 2 11 8 6

pilih 1 atau 2 : 1
---secara ascending---
1 2 2 5 6 7 8 9 11 11
Waktu 205464
BUILD SUCCESSFUL (total time: 3 minutes 25 seconds)

```

Urutan secara descending

```

Output - pertemuan 4 pak shodiq (run) X nomer_4_selection_ascendingdescending.java
run:
menu pilihan selection sort :
  1. urutan secara ascending
  2. urutan secara descending

data sebelum diurutkan : 10 8 9 7 10 7 10 1 6 1

pilih 1 atau 2 : 2
---secara descending---
10 10 10 9 8 7 7 6 1 1
Waktu 2063
BUILD SUCCESSFUL (total time: 2 seconds)

```

Analisa :

Program insertion sort juga saya terapkan pada selection sort yaitu dengan cara masukkan dengan keyboard ( inputan ). Dan juga fungsi dari if else. Namun perbedaannya terdapat pada pendefinisian dari program selection, yaitu jika insertion dalam pengurutannya urut dari kiri kekanan dengan membandingkan satu angka per angka. Berbeda dengan selection sort. Dengan cara ini program langsung memilih dari bilangan terkecil dan membandingkan dengan angka yang paling kiri. Dan seterusnya. Maka oleh itu program insertion dan selection sort berbeda lantaran dalam proses pencariannya bilangannya ataupun pengurutannya berbeda satu sama lain.



### Kesimpulan :

Dari praktikum yang sudah saya lakukan saya dapat menganalisa bahwa program insertion sort dan selection sort sebenarnya fungsinya adalah sama yaitu untuk mengurutkan suatu data ataupun angka. Dari kiri kekanan. Untuk pengurutan ada dua jenis yaitu secara ascending yang berarti urutannya dari kecil ke besar ( ankanya ) dan juga urutan secara descending yaitu dengan urutan angka dari besar ke kecil. Untuk insertion dalam proses pengurutannya urut dari kiri kekanan dengan membandingkan satu angka per angka. Berbeda dengan selection sort. Dengan cara ini program langsung memilih dari bilangan terkecil dan membandingkan dengan angka yang paling kiri.