**Sorting in JAVA**

1. **Bubble Sort**

Idea: if arr[i] > arr[i+1] swap them. To place the element in their respective position, we have to do the following operation N-1 times.

Time Complexity: O(N2)

**Code**

import java.util.\*;

class Sorting {

public static void printArray(int arr[]) {

for(int i=0; i<arr.length; i++) {

System.out.print(arr[i]+" ");

}

System.out.println();

}

public static void main(String args[]) {

int arr[] = {7, 8, 1, 3, 2};

//bubble sort

for(int i=0; i<arr.length-1; i++) {

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

if(arr[j] > arr[j+1]) {

//swap

int temp = arr[j];

arr[j] = arr[j+1];

arr[j+1] = temp;

}

}

}

printArray(arr);

}

}

1. **Selection Sort**

Idea: The inner loop selects the minimum element in the unsorted array and places the elements in increasing order.

Time complexity: O(N2)

**Code**

**import java.util.\*;**

**class Sorting {**

**public static void printArray(int arr[]) {**

**for(int i=0; i<arr.length; i++) {**

**System.out.print(arr[i]+" ");**

**}**

**System.out.println();**

**}**

**public static void main(String args[]) {**

**int arr[] = {7, 8, 1, 3, 2};**

**//selection sort**

**for(int i=0; i<arr.length-1; i++) {**

**int smallest = i;**

**for(int j=i+1; j<arr.length; j++) {**

**if(arr[j] < arr[smallest]) {**

**smallest = j;**

**}**

**}**

**//swap**

**int temp = arr[smallest];**

**arr[smallest] = arr[i];**

**arr[i] = temp;**

**}**

**printArray(arr);**

**}**

**}**

1. **Insertion Sort**

Idea: Take an element from the unsorted array, place it in its corresponding position in the sorted part, and shift the elements accordingly.

Time Complexity: O(N2)

**Code**

**import java.util.\*;**

**class Sorting {**

**public static void printArray(int arr[]) {**

**for(int i=0; i<arr.length; i++) {**

**System.out.print(arr[i]+" ");**

**}**

**System.out.println();**

**}**

**public static void main(String args[]) {**

**int arr[] = {7, 8, 1, 3, 2};**

**//insertion sort**

**for(int i=1; i<arr.length; i++) {**

**int current = arr[i];**

**int j = i - 1;**

**while(j >= 0 && arr[j] > current) {**

**//Keep swapping**

**arr[j+1] = arr[j];**

**j--;**

**}**

**arr[j+1] = current;**

**}**

**printArray(arr);**

**}**

**}**

**`**