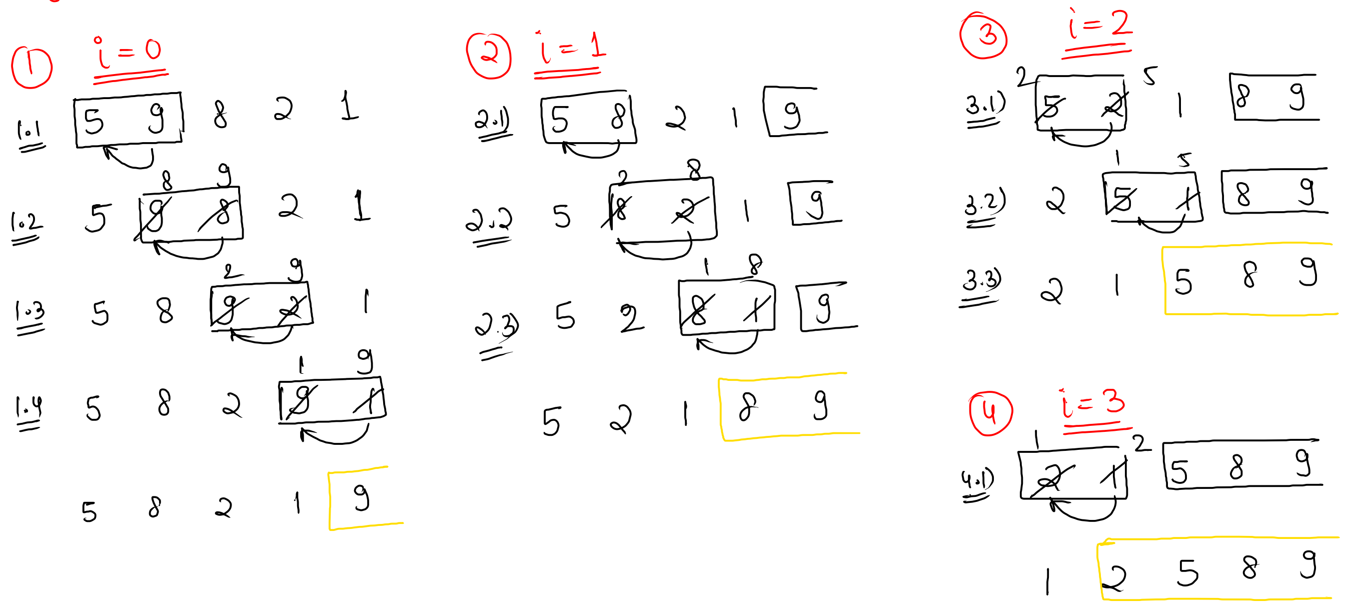
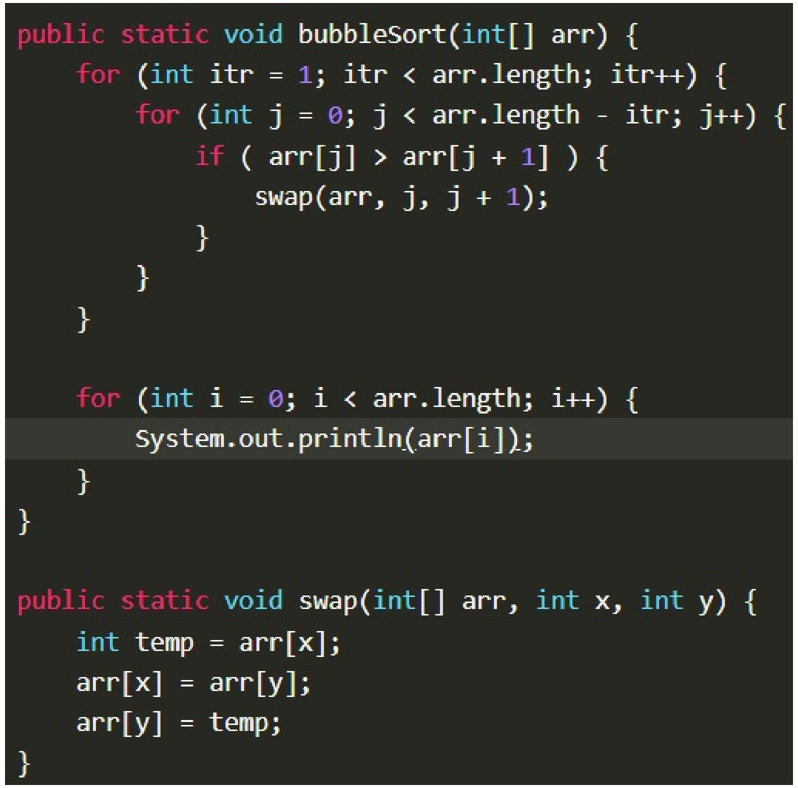
⇒ Sorting (asc/des)
- ↳ Bubble sort
- ↳ Insertion sort
- ↳ Selection sort

⇒ Bubble sort (trying to pick largest element and take it rightmost part)

$O(N^2)$

e.g., 5 9 8 2 1





```
public static void bubbleSort(int[] arr) {
    for (int itr = 1; itr < arr.length; itr++) {
        for (int j = 0; j < arr.length - itr; j++) {
            if ( arr[j] > arr[j + 1] ) {
                swap(arr, j, j + 1);
            }
        }
    }

    for (int i = 0; i < arr.length; i++) {
        System.out.println(arr[i]);
    }
}

public static void swap(int[] arr, int x, int y) {
    int temp = arr[x];
    arr[x] = arr[y];
    arr[y] = temp;
}
```

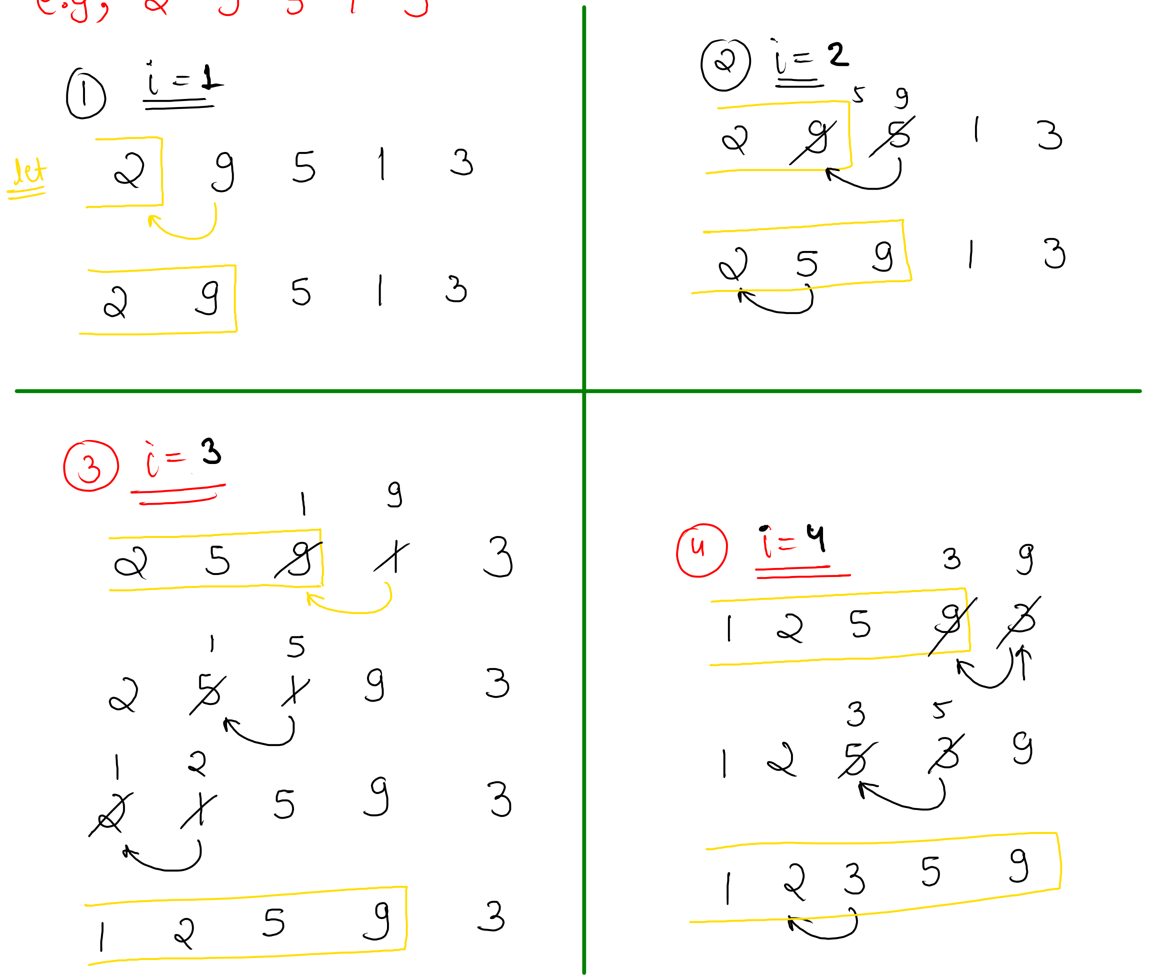
compare consiqutive elements and checks if right one is smaller the swap

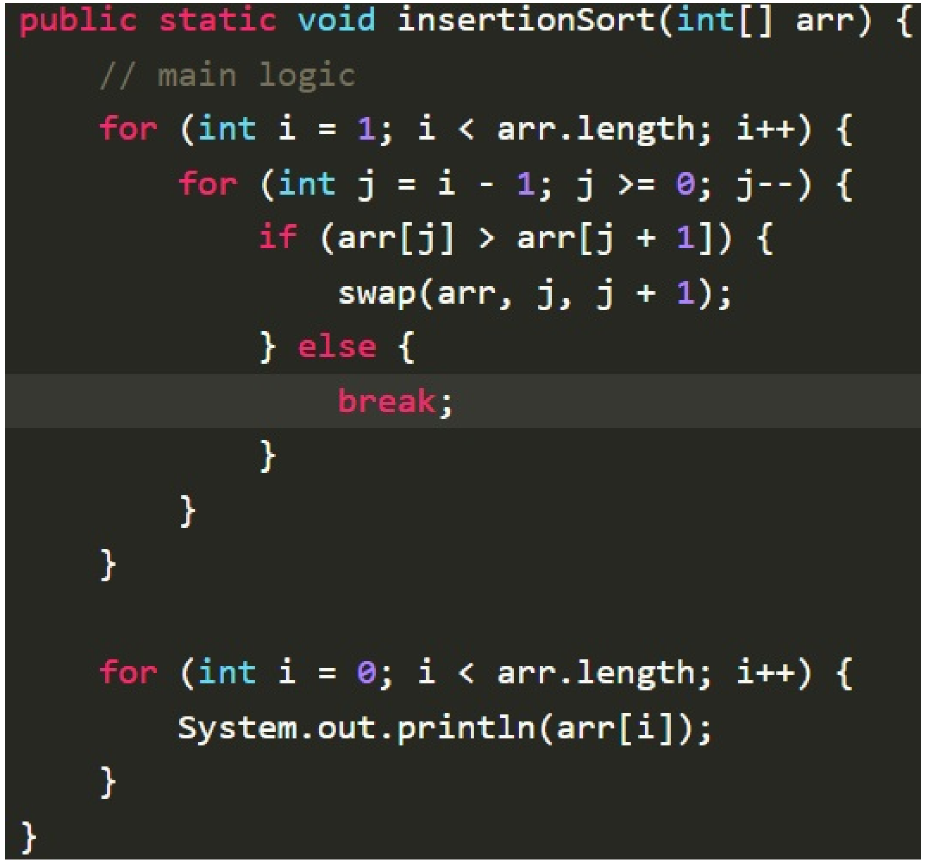
⇒ Insertion Sort $O(N^2)$

↳ pick the first element from unsorted part of array and place it at the correct position in sorted part.

(Keep on swapping the element until it reached it's correct position)

e.g, 2 9 5 1 3



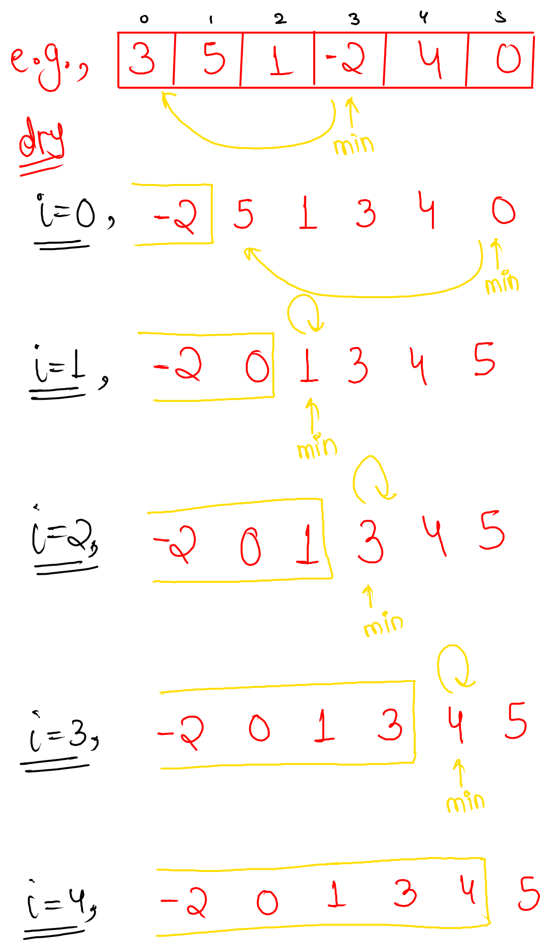


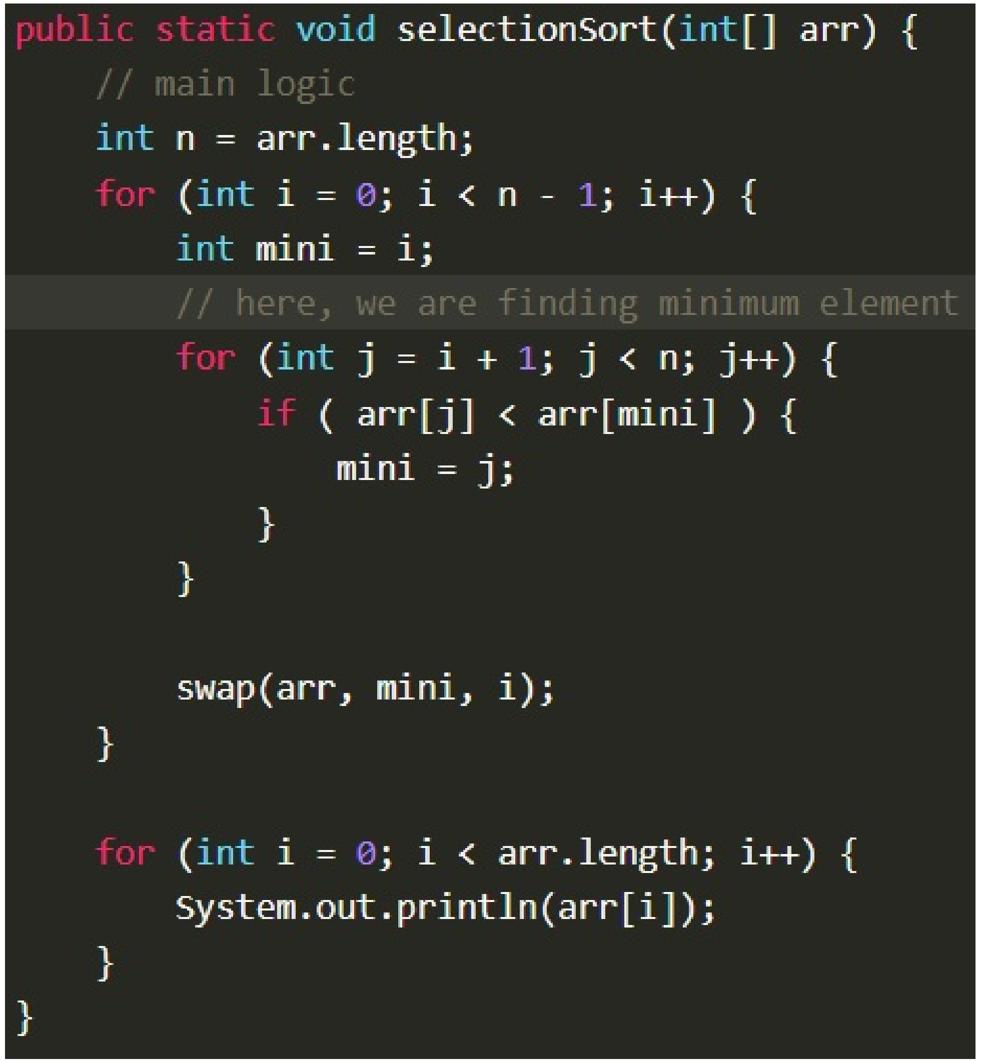
```java
public static void insertionSort(int[] arr) {
    // main logic
    for (int i = 1; i < arr.length; i++) {
        for (int j = i - 1; j >= 0; j--) {
            if (arr[j] > arr[j + 1]) {
                swap(arr, j, j + 1);
            } else {
                break;
            }
        }
    }

    for (int i = 0; i < arr.length; i++) {
        System.out.println(arr[i]);
    }
}
```

⇒ Selection Sort

↳ each time find the smallest element from unsorted part of array and swap it with first unsorted value.





```java
public static void selectionSort(int[] arr) {
    // main logic
    int n = arr.length;
    for (int i = 0; i < n - 1; i++) {
        int mini = i;
        // here, we are finding minimum element
        for (int j = i + 1; j < n; j++) {
            if ( arr[j] < arr[mini] ) {
                mini = j;
            }
        }

        swap(arr, mini, i);
    }

    for (int i = 0; i < arr.length; i++) {
        System.out.println(arr[i]);
    }
}
```

Que K th largest element

```java
public static void kthLargest(int[] arr, int k, int n) {
    bubbleSort(arr);
    int ans = arr[n - k];
    System.out.println(ans);
}

public static int[] bubbleSort(int[] arr) {
    for (int itr = 1; itr < arr.length; itr++) {
        for (int j = 0; j < arr.length - itr; j++) {
            if ( arr[j] > arr[j + 1] ) {
                swap(arr, j, j + 1);
            }
        }
    }

    return arr;
}

public static void swap(int[] arr, int x, int y) {
    int temp = arr[x];
    arr[x] = arr[y];
    arr[y] = temp;
}
```

also

```java
public static void kthLargest(int[] arr, int k, int n) {
    bubbleSort(arr);
    int ans = arr[k - 1];
    System.out.println(ans);
}
```