**6. Detailed Code With Relevant comments :->**

package switchsorting;

import java.util.Scanner;

public class switchsortingg {

// function for sorting by Bubble sort method

public void bubblesort(int A[], int n) {

int temp = 0;

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

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

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

temp = A[j];

A[j] = A[j + 1];

A[j + 1] = temp;

}

}

}

System.out.print("Sorted Array is =");

for (int i = 0; i < n; i++) {

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

}

}

// function for sorting by Insertion sort method

public void insertionsort(int A[], int n) {

int temp = 0;

for (int i = 0; i < n; i++) {

temp = A[i];

int j = i - 1;

while (j >= 0 && A[j] > temp) {

A[j + 1] = A[j];

j--;

}

A[j + 1] = temp;

}

System.out.print("Sorted Array is =");

for (int i = 0; i < n; i++) {

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

}

System.out.println(" ");

}

// function for sorting by Quick sort method

public void Quicksort(int A[], int lb, int ub) {

if (lb < ub) {

int loc = partition(A, lb, ub);

Quicksort(A, lb, loc - 1);

Quicksort(A, loc + 1, ub);

}

}

public int partition(int A[], int lb, int ub) {

int temp, temp1;

int p = A[lb];

int s = lb;

int e = ub;

while (s <= e) {

while (A[s] <= p) {

s++;

}

while (A[e] > p) {

e--;

}

if (s < e) {

temp = A[s];

A[s] = A[e];

A[e] = temp;

}

}

temp1 = A[lb];

A[lb] = A[e];

A[e] = temp1;

return e;

}

// main function takes size of the array and elements of array.

public static void main(String[] args) {

switchsortingg ss = new switchsortingg();

Scanner sc = new Scanner(System.in);

int ch;

System.out.print("Enter size of array = ");

int n = sc.nextInt();

int A[] = new int[n];

System.out.print("Enter array element = ");

for (int i = 0; i < n; i++) {

A[i] = sc.nextInt();

}

// for a while loop

while (true) {

// List of options for choice

System.out.println("Menu for searching");

System.out.println("1.bubble Sort ");

System.out.println("2.insertion Sort");

System.out.println("3.Quick Sort");

System.out.println("0.For Exit");

System.out.println("Enter youre choice = ");

ch = sc.nextInt();

switch (ch) {

case 1:

// function call for bubble sort

ss.bubblesort(A, n);

System.out.println();

break;

case 2:

// function call for insertion sort

ss.insertionsort(A, n);

System.out.println();

break;

case 3:

// function call for Quick sort

ss.Quicksort(A, 0, n - 1);

System.out.print("The sorted array is = ");

for (int i = 0; i < n; i++) {

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

}

System.out.println();

break;

case 0:

// To end while loop

System.exit(0);

break;

}

}

**7. Results:->**

Menu for searching

1.bubble Sort

2.insertion Sort

3.Quick Sort

0.For Exit

Enter youre choice =

0

Process finished with exit code 0

Enter size of array = 6

Enter array element = 5

4

3

2

1

8

Menu for searching

1.bubble Sort

2.insertion Sort

3.Quick Sort

0.For Exit

Enter youre choice =

3

The sorted array is = 1 2 3 4 5 8

Enter size of array = 5

Enter array element = 5

4

3

2

1

Menu for searching

1.bubble Sort

2.insertion Sort

3.Quick Sort

0.For Exit

Enter youre choice =

2

Sorted Array is =1 2 3 4 5

Enter size of array = 5

Enter array element = 5 4 3 2 1

Menu for searching

1. bubble Sort

2.insertion Sort

3.Quick Sort

0.For Exit

Enter youre choice =

1

Sorted Array is =1 2 3 4 5