

1. Write a program to take an integer array from the user and give the user a choice to sort using bubble sort (or) selection sort.
//Sort the array elements according to the selected algorithm of the user and display the sorted array.

```
package sba4;
import java.util.Scanner;

public class Ques1 {
    void bubbleSort(int arr[])
    {
        System.out.println("bubble sorting started");
        int n = arr.length;
        for (int i = 0; i < n-1; i++)
            for (int j = 0; j < n-i-1; j++)
            {
                if (arr[j] > arr[j+1])
                {
                    int temp = arr[j];
                    arr[j] = arr[j+1];
                    arr[j+1] = temp;
                }
            }
    }

    void bubbleprint(int arr[]) {
        System.out.println("bubble Sorted array :");

        int n = arr.length;
        for (int i=0; i<n; ++i)
            System.out.print(arr[i] + " ");
    }

    void selectionsort(int arr[])
    {
        System.out.println("selection sorting started");
        int n =arr.length;
        for (int i =0; i<n-1; i++)
        {
            int min_idx=i;
            for (int j=i+1; j<n;j++)
            {
                if (arr[min_idx]>arr[j])
                    min_idx=j;
            }
            int temp=arr[min_idx];
            arr[min_idx]=arr[i];
            arr[i]=temp;
        }
    }

    void selectionprint(int arr[]) {
        System.out.println("selection Sorted array");
        int n = arr.length;
```

```

        for (int i=0; i<n; ++i)
            System.out.print(arr[i]+" ");
    }

    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        System.out.println("enter the number of elements");
        int n=sc.nextInt();
        int [] arr=new int[n];
        System.out.println("enter the numbers");
        for( int i=0;i<n;i++) {
            arr[i]=sc.nextInt();
        }

        System.out.println("enter the mode of sorting :either press '1' for bubble
sort or '2' for selection sort");
        int a=sc.nextInt();
        Ques1 ob=new Ques1();
        if(a==1)
        {
            ob.bubbleSort(arr);
            ob.bubbleprint(arr);
        }
        else{
            ob.selectionsort(arr);
            ob.selectionprint(arr);
        }
    }
}

```

//Output

```

<terminated> Ques1 (2) [Java Application] C:\Users\bizz-IT\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.
enter the number of elements
3
enter the numbers
36
25
49
enter the mode of sorting :either press '1' for bubble sort or '2' for selection sort
1
bubble sorting started
bubble Sorted array :
25 36 49

```

2. Write a program to implement insertion sort

```
package sba4;
public class Ques2 {

    public static void insertionSort(int array[]) {
        int n=array.length;
        for (int j =1; j<n;j++) {
            int key= array[j];
            int i=j-1;
            while ( ( i > -1) && ( array [i] > key ) ) {
                array [i+1] = array [i];
                i--;
            }
            array[i+1] = key;
        }
    }

    public static void main(String a[]){
        int[] arr1 = {2,5,0,12,29,13,62};
        System.out.println("Before Insertion Sort");
        for(int i:arr1){
            System.out.print(i+" ");
        }
        System.out.println();

        insertionSort(arr1);//sorting array using insertion sort

        System.out.println("After Insertion Sort");
        for(int i:arr1){
            System.out.print(i+" ");
        }
    }
}
```

//Output

```
<terminated> Ques2 (2) [Java Application] C:\Users\bizz-II\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86
Before Insertion Sort
2 5 0 12 29 13 62
After Insertion Sort
0 2 5 12 13 29 62
```

3. Write a program to implement Hashtable and add atleast 4 values into it, implement the putIfAbsent() method

```
package sba4;
import java.util.Hashtable;

public class Ques3 {

    public static void main(String[] args) {
        Hashtable<Integer,String> map=new Hashtable<Integer,String>();
        map.put(100,"Amit");
        map.put(102,"Ravi");
        map.put(101,"Vijay");
        map.put(103,"Rahul");
        System.out.println("Before remove: "+ map);
        // Remove value for key 102
        map.remove(102);
        System.out.println("After remove: "+ map);
        //checking empty or not
        System.out.println("map is empty? "+map.isEmpty());
        //Here, we specify the if and else statement as arguments of the method
        System.out.println(map.getDefault(101, "Not Found"));
        System.out.println(map.getDefault(105, "Not Found"));
        //Inserts, as the specified pair is unique
        map.putIfAbsent(102,"Gaurav");
        System.out.println("Updated Map: "+map);
        //Returns the current value, as the specified pair already exist
        map.putIfAbsent(101,"Dhamu");
        System.out.println("Updated Map: "+map);
        //Replace the value at key 100
        map.replace(100,"Kelu");
        System.out.println("Updated Map: "+map);
        //Checking values in map
        System.out.println("Dhamu in map? "+map.contains("Dhamu"));
        System.out.println("Kelu in map? "+map.contains("Kelu"));
        //Checking key in map and getting the value
        if(map.containsKey(101)==true) {
            System.out.println("Vlaue of key 101 is "+map.get(101));
        }
        //printing all values in map
        System.out.println(map.values());
        if(map.replace(103,"rahul","Raju")==true) {
            System.out.println("Replaced Rahul...");
            System.out.println("Updated Map: "+map);
        }
    }
}

//Output
```

```

Before remove: {103=Rahul, 102=Ravi, 101=Vijay, 100=Amit}
After remove: {103=Rahul, 101=Vijay, 100=Amit}
map is empty? false
Vijay
Not Found
Updated Map: {103=Rahul, 102=Gaurav, 101=Vijay, 100=Amit}
Updated Map: {103=Rahul, 102=Gaurav, 101=Vijay, 100=Amit}
Updated Map: {103=Rahul, 102=Gaurav, 101=Vijay, 100=Kelu}
Dhamu in map? false
Kelu in map? true
Vlaue of key 101 is Vijay
[Rahul, Gaurav, Vijay, Kelu]

```

4.Create a class of Books with attributes:

- a)id
- b)name
- c)author
- d)publisher
- e)quantity sold.

Implement a Hashtable to implement the objects of Books type. Print all the details of books by traversing through the Hashtable

```

package sba4;
import java.util.Hashtable;
import java.util.Map;

class Book {
int id;
String name,author,publisher;
int quantity;
public Book(int id, String name, String author, String publisher, int quantity) {
    this.id = id;
    this.name = name;
    this.author = author;
    this.publisher = publisher;
    this.quantity = quantity;
}
}

public class Ques4 {
    public static void main(String[] args) {
        //Creating map of Books
        Hashtable<Integer,Book> map=new Hashtable<Integer,Book>();
        //Creating Books
        Book b1=new Book(101,"Let us C","Yashwant Kanetkar","BPB",8);
    }
}

```

```

        Book b2=new Book(102,"Data Communications & Networking","Forouzan","Mc
Graw Hill",4);
        Book b3=new Book(103,"Operating System","Galvin","Wiley",6);
        //Adding Books to map
        map.put(1,b1);
        map.put(2,b2);
        map.put(3,b3);
        //Traversing map
        for(Map.Entry<Integer, Book> z:map.entrySet()){
            int key=z.getKey(); //key=3
            Book b=z.getValue(); //b=b3
            System.out.println(key+" Details:");
            System.out.println(b.id+" "+b.name+" "+b.author+" "+b.publisher+"
"+b.quantity);
        }
    }
}

```

//Output

```

<terminated> Ques4 (2) [Java Application] C:\Users\bizz-II\p2\pool\plugins\org.eclipse.justj.op
3 Details:
103 Operating System Galvin Wiley 6
2 Details:
102 Data Communications & Networking Forouzan Mc Graw Hill 4
1 Details:
101 Let us C Yashwant Kanetkar BPB 8

```