Assignment-1

1.Sort string using bubble sort and insertion sort

```
import java.util.*;
public class StringComparsionBubbleSort {

	public static void main(String[] args) {
		System.out.println("Enter how many string u need");
		Scanner scan=new Scanner(System.in);
		int n=scan.nextInt();
		scan.nextLine();
		System.out.println("Enter array of strings");
		String arr[]=new String[n];
		for(int i=0;i<n;i++) {
			arr[i]=scan.nextLine();
		}
		bubbleSort(arr);
		insertionSort(arr);

	}
	public static void bubbleSort(String[] s) {
		System.out.println("String sort using bubble
sort");
		for(int i=0;i<s.length-1;i++) {
			for(int j=0;j<s.length-1-i;j++) {
				if(s[j].compareTo(s[j+1])>0) {
					String temp=s[j];
					s[j]=s[j+1];
					s[j+1]=temp;
				}
			}
		}
		for(int i=0;i<s.length;i++) {
			System.out.print(s[i]+" ");
		}
		System.out.println();
	}
```

```
	public static void insertionSort(String[] s) {
		System.out.println("String sort using insertion
sort");
		for(int i=1;i<s.length;i++) {
			String key=s[i];
			int j=i-1;
			while(j>=0 && s[j].compareTo(key)>0) {
				s[j+1]=s[j];
				j--;
			}
			s[j+1]=key;

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

}
```

2.Displaying initial of a name

```
import java.util.*;
public class DisplayInitialOfName {

	public static void main(String[] args) {
		System.out.println("Enter name");
		Scanner scan=new Scanner(System.in);
		String name=scan.nextLine();
		displayInitial(name);
	}
	public static void displayInitial(String name) {
		String initial="";
		for(int i =0;i<name.length();i++) {
```

```
                if(name.charAt(i)>64 && name.charAt(i)<97)
                    initial+=name.charAt(i);
            }
            System.out.println(initial);
        }

}
```

3.computing password using initials and age

```
import java.util.*;
public class GeneratePasswordInitial {

    public static void main(String[] args) {
        Scanner scan=new Scanner(System.in);
        System.out.println("Enter first name");
        String fname=scan.nextLine();
        System.out.println("Enter middle name");
        String mname=scan.nextLine();
        System.out.println("Enter last name");
        String lname=scan.nextLine();
        System.out.println("Enter age");
        int age=scan.nextInt();
        String fullname=fname+mname+lname;
        String initial="";
        for(int i=0;i<fullname.length();i++) {
```

```
                if(fullname.charAt(i)>64 &&
fullname.charAt(i)<97)
                  initial+=fullname.charAt(i);
            }
            generatePassword(initial,age);

      }
      public static void generatePassword(String initial,int
age) {
            String comb=initial+age;
            char pass[]=new char[8];
            for(int i=0;i<8;i++) {
                  char
c=comb.charAt((int)(Math.random()*comb.length()));
                  pass[i]=c;
            }
            System.out.println(pass);
      }

}
```

4.interchanging last names of two name

```
import java.util.*;
public class InterchangingLastName {

	public static void main(String[] args) {
		System.out.println("Enter two names");
		Scanner scan=new Scanner(System.in);
		String firstName=scan.nextLine();
		String secondName=scan.nextLine();
		interchangeLastName(firstName,secondName);


	}
	public static void interchangeLastName(String
fname,String sname) {
		String firstName[]=findLastName(fname);
		String secondName[]=findLastName(sname);
		String firstFullName="";
		String secondFullName="";
		for(int i=0;i<Integer.valueOf(firstName[1]);i++)
			firstFullName+=fname.charAt(i);
		for(int i=0;i<Integer.valueOf(secondName[1]);i++)
			secondFullName+=sname.charAt(i);
		System.out.println(firstFullName+secondName[0]);
		System.out.println(secondFullName+firstName[0]);


	}
```

```
    public static String[] findLastName(String s) {
        int index=0;
        String lname="";
        String re[]=new String[2];
        for(int i=0;i<s.length();i++) {
            if(s.charAt(i)>64 && s.charAt(i)<97)
                index=i;
        }
        for(int i=index;i<s.length();i++) {
            lname+=s.charAt(i);
        }
        re[0]=lname;
        re[1]=String.valueOf(index);
        return re;
    }


}
```

5.Analysing compareTo

```
public class CompareTwoMethod {
    static String s1="Hello World";
     public static void main(String[] args) {
          String s2="Hello";
          System.out.println(s1.compareTo(s2));

     }
  public int compareTo(String anotherString) {
       int len1=s1.length();
       int len2=anotherString.length();
       int lim=Math.min(len1, len2);
       char v1[]=s1.toCharArray();
       char v2[]=anotherString.toCharArray();
       int k=0;
       while(k<lim) {
            char c1=v1[k];
            char c2=v2[k];
            if(c1!=c2)
                 return c1-c2;
            k++;
       }
       return len1-len2;

  }

}
```

6.validate usn

```
import java.util.*;
import java.util.regex.Matcher;
import java.util.regex.Pattern;
public class ValidateUsn {

	public static void main(String[] args) {
		Scanner scan=new Scanner(System.in);
		System.out.println("Enter usn");
		String usn=scan.nextLine();
		validateUsn(usn);

	}
	public static void validateUsn(String usn) {
		String regx="[1-2][A-Z]{2}[0-
9]{2}[CS||IS||EC||ME]{2}[0-9]{3}";
		boolean f=usn.matches(regx);
		if(f)
			System.out.println("Success");
		else
			System.out.println("Failure");
	}

}
```

7.Reverse a words in sentence without reversing number

```
import java.util.*;
import java.util.regex.Pattern;
public class ReverseSentence {
	public static void main(String[] args) {
		Scanner scan=new Scanner(System.in);
```

```
        System.out.println("Enter sentence");
        String s=scan.nextLine();
       String split[]=s.split(" ");
       for(int i=0;i<split.length;i++) {
        displayReversed(split[i]);
       }
    }
    public static void displayReversed(String s) {
        String regex="[A-Za-z]{1,}";
        if(s.matches(regex)) {
            System.out.print(reverseString(s));
        }
        else {
            System.out.print(s);
        }
        System.out.print(" ");
    }
    public static String reverseString(String rev) {
        String reverse="";
        for(int i=rev.length()-1;i>=0;i--) {
            reverse+=rev.charAt(i);
        }
        return reverse;
    }


}
```

8.computing consecutive characters

```
import java.util.*;
public class ConsecutiveSequence {

    public static void main(String[] args) {
        Scanner scan=new Scanner(System.in);
        System.out.println("Enter a Sentence");
        String s=scan.nextLine();
       printConsecutiveCharacters(s);
    }
```

```
     public static void printConsecutiveCharacters(String ss)
{
          HashMap<String,Integer> map=new HashMap<>();
          String s=ss.replaceAll("-","");
          for(int i=0;i<s.length()-1;i++) {
               if(s.charAt(i)+1==s.charAt(i+1)) {
                    String
temp=String.valueOf(s.charAt(i))+String.valueOf(s.charAt(i+1
));
                    if(map.containsKey(temp)) {
                         int ans=map.get(temp);
                         map.put(temp, ans+1);
                    }
                    else {
                         map.put(temp, 1);
                    }
               }
          }
          System.out.println(map);
     }

}
```

9.String compression

```
import java.util.*;
public class StringCompression {

     public static void main(String[] args) {
          Scanner scan=new Scanner(System.in);
          System.out.println("Enter a string");
          String s=scan.nextLine();
         stringComp(s);
     }
     public static void stringComp(String ss) {
          String s=ss.toLowerCase();
          String comp="";
          int count=1;
```

```
            for(int i=0;i<s.length()-1;i++) {
                 if(s.charAt(i)==s.charAt(i+1)) {
                      count++;
                 }else {
                      comp+=String.valueOf(s.charAt(i))+count;
                      count=1;
                 }
            }
            if(s.charAt(s.length()-1)!=s.charAt(s.length()-2))
{
                 comp+=String.valueOf(s.charAt(s.length()-
1))+"1";
            }
            else {
                 int c=1;
                 for(int i=s.length()-1;i>=0;i--) {
                      if(s.charAt(i)!=s.charAt(i-1))
                           break;
                      c++;
                 }
                 comp+=String.valueOf(s.charAt(s.length()-
1))+c;
            }
            System.out.println(comp);
      }

}
```

10.Triplets

```
import java.util.*;
public class Triplets {

	public static void main(String[] args) {
		Scanner scan=new Scanner(System.in);
		System.out.println("enter size n");
		int n=scan.nextInt();
		int a[]=new int[n];
		for(int i=0;i<n;i++) {
			a[i]=scan.nextInt();
		}
		for(int i=0;i<n;i++) {
			for(int j=i+1;j<n;j++) {
				for(int k=j+1;k<n;k++) {
					if(a[i]+a[j]==a[k])
```

```
        System.out.println("<"+a[i]+","+a[j]+","+a[k]+" >");
                        }
                    }
                }

        }

}
```

11.Retail Store

```
import java.util.*;
public class RetailStore {

        public static void main(String[] args) {
            int itemId[]= {5001,5002,5003,5004,5005};
            int quantity[]= {1,2,3,4,5};
            double price[]= {1000,2000,3000,4000,5000};
            double discount[]= {4,10,20,15,5};
           display(itemId,quantity,price,discount,5002);
        }
        public static void display(int[] itemId,int[]
quantity,double[] price,double[] discount,int purchaseId)
        {
        boolean flag=false;
        int temp=0,customerId=0,billId=0;
        for(int i=0;i<itemId.length;i++)
        {
        if(purchaseId==itemId[i])
        {
        flag=true;
        temp=i;
        customerId++;
        billId++;
        break;
        }
        }
        if(flag)
        {
```

```
        double billAmount=price[temp]*quantity[temp];
        double discountAmount=billAmount*(discount[temp]/100);
        System.out.println("bill id : "+billId);
        System.out.println("Customer id : "+customerId);
        System.out.println("purchase id : "+purchaseId);
        System.out.println("quantity purchased :
"+quantity[temp]);
        System.out.println("discount amount : "+discountAmount);
        System.out.println("bill amount : "+(billAmount-
discountAmount));
        }
        else
        System.out.println("item not found");
        }


}
```