|  |  |
| --- | --- |
| **Name** | Shubhan Singh |
| **UID no.** | 2022300118 |
| **Experiment No.** | 6-A |

|  |  |
| --- | --- |
| **PROBLEM STATEMENT :** | **Write a program which will read a string and rewrite it in the alphabetical order. For example, the word STRIN should be written as GINRST** |
| **THEORY:** | **Some key points about strings in Java:**  **1. String Creation:**  In Java, strings can be created using string literals or the `String` class constructor. String literals are created by enclosing characters within double quotes (" "):  ```  String str1 = "Hello"; // Using string literal  String str2 = new String("World"); // Using the String class constructor  ```  **2. String Immutability:**  In Java, strings are immutable, meaning their values cannot be changed once created. Any operation that appears to modify a string actually creates a new string. For example:  ```  String str = "Hello";  str = str + " World"; // Creates a new string, "Hello World"  ```  **3. Concatenating Strings:**  Strings can be concatenated using the `+` operator or the `concat()` method:  ```  String fullName = str1 + " " + str2; // Using the '+' operator  String concatenated = str1.concat(str2); // Using the 'concat()' method  ```  **4. String Length:**  The length of a string can be obtained using the `length()` method:  ```  int length = fullName.length(); // Returns the length of the string  ```  **5. Accessing Characters:**  Individual characters of a string can be accessed using the `charAt()` method:  ```  char firstChar = fullName.charAt(0); // Retrieves the character at index 0  ``` |
| **PROGRAM:** | import java.util.Arrays; import java.util.Scanner;  public class Stringhandling{  public static void main(String[] *args*) {  System.out.println("Enter a string:");  Scanner sc=new Scanner(System.in);  String s=sc.next();  int temp,j;  int[] strarr=new int[s.length()];  for(int i=0;i<s.length();i++){  strarr[i]=(int)s.charAt(i);  }  for(int i=1;i<s.length();i++){  temp=strarr[i];  j=i;  while(strarr[j-1]>temp){  strarr[j]=strarr[j-1];  j--;  if(j==0){  break;  }  }  strarr[j]=temp;  }  String snew=new String();  char string\_buffer;  for(int i=0;i<s.length();i++){  string\_buffer=(char)strarr[i];  snew=snew+string\_buffer;  }  System.out.println("The string sorted in alphabetical order is: "+snew);  sc.close();  } } |
| **RESULT:** | |