OOPJ LAB ASSIGNMENT – 8

Name – Nishant Nahar

Roll No - 241551078

1. Write a Java program to create strings using literal and new keyword and compare them using == and .equals() methods.

```
class as_1_q_1 {
    public static void main(String[] a) {
        System.out.println("Nishant Nahar - 241551078");
        String s1 = "Hell";
        String s2 = "Hill";
        String s3 = new String("Hell");
        System.out.println(s1 == s2);
        System.out.println(s1 == s3);
        System.out.println(s1.equals(s3));
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java assign_1.java
Nishant Nahar - 241551078
false
false
true
```

2. Write a Java program to demonstrate immutability of the String class with suitable examples.

```
class as_1_q_2 {
    public static void main(String[] a) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "Hi";
        String s2 = s.concat(" Nishant");
        System.out.println(s);
        System.out.println(s2);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java assign_2.java
Nishant Nahar - 241551078
Hi
Hi Nishant
```

3. Write a Java program to count vowels, consonants, digits, and special characters in a given string.

```
class as_1_q_3 {
    public static void main(String[] a) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "Abc123@$$5rwQiuo#";
        int v = 0, c = 0, d = 0, sp = 0;
        for (int i = 0; i < s.length(); i++) {
            char ch = s.charAt(i);
            if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u' || ch
== 'A' || ch == 'E' || ch == 'I' || ch == 'O' || ch == 'U')
```

```
                v++;
            else if ((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z')) c++;
            else if (ch >= '0' && ch <= '9') d++;
            else sp++;
        }
        System.out.println(v + " " + c + " " + d + " " + sp);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java assign_3.java
Nishant Nahar - 241551078
4 5 4 4
```

4. Write a Java program to reverse a string using loop, StringBuilder, and recursion.

```
class as_1_q_4 {
    static String r(String s) {
        if (s.equals("")) return "";
        return r(s.substring(1)) + s.charAt(0);
    }
    public static void main(String[] a) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "Java";
        System.out.println(r(s));
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java assign_4.java
Nishant Nahar - 241551078
avaJ
```

5. Write a Java program to check whether a given string is a palindrome ignoring spaces and case sensitivity.

```
class as_1_q_5 {
    public static void main(String[] a) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "A man a plan a canal Panama".toLowerCase();
        int i = 0, j = s.length() - 1;
        boolean f = true;
        while (i < j) {
            char c1 = s.charAt(i);
            char c2 = s.charAt(j);
            if (c1 == ' ') {i++;continue;}
            if (c2 == ' ') {j--;continue;}
            if (c1 != c2) {f = false;break;}
            i++;j--;}
        System.out.println(f);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java assign_5.java
Nishant Nahar - 241551078
true
```

6. Write a Java program to count the number of words in a sentence without using the split() method.

```java
class as_1_q_6 {
    public static void main(String[] a) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "This is java code written by me ";
        s = s.trim();
        int c = 0;
        for (int i = 0; i < s.length(); i++) {
            if (s.charAt(i) == ' ') c++;}
        System.out.println(c + 1);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java assign_6.java
Nishant Nahar - 241551078
7
```

7. Write a Java program to remove duplicate characters from a string whilemaintaining insertion order.

```java
class as_1_q_7 {
    public static void main(String[] a) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "programming";
        String r = "";
        for (int i = 0; i < s.length(); i++) {
            char ch = s.charAt(i);
            if (r.contains(ch + "") == false) {
                r = r + ch;}}
        System.out.println(r);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java assign_7.java
Nishant Nahar - 241551078
progamin
```

8. Write a Java program to find the frequency of each character and display the character with maximum frequency.

```java
class as_1_q_8 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "nishant nahar";
        int max = 0;
        char cc = ' ';
        for (int i = 0; i < s.length(); i++) {
            int count = 0;
            for (int j = 0; j < s.length(); j++) {
                if (s.charAt(i) == s.charAt(j)) count++;}
            if (count > max) {
                max = count;
                cc = s.charAt(i);
            }}
```

```
        System.out.println(cc + " " + max);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_8.java
Nishant Nahar - 241551078
n 3
```

9. Write a Java program to check whether two strings are anagrams withoutusing sorting.

```
class as_1_q_9 {
    public static void main(String[] a) {
        System.out.println("Nishant Nahar - 241551078");
        String s1 = "listen";
        String s2 = "silent";
        if (s1.length() != s2.length()) {
            System.out.println("Not Anagram");
            return;}
        int count = 0;
        for (int i = 0; i < s1.length(); i++) {
            char ch = s1.charAt(i);
            for (int j = 0; j < s2.length(); j++) {
                if (ch == s2.charAt(j)) {
                    s2 = s2.substring(0, j) + s2.substring(j + 1);
                    count = count + 1;
                    break;}}}
        if (count == s1.length()) System.out.println("Anagram");
        else System.out.println("Not Anagram");
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_9.java
Nishant Nahar - 241551078
Anagram
```

10. Write a Java program to find the first non-repeating character in a string.

```
class as_1_q_10 {
    public static void main(String[] a) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "swiss";
        for (int i = 0; i < s.length(); i++) {
            int count = 0;
            for (int j = 0; j < s.length(); j++) {
                if (s.charAt(i) == s.charAt(j)) count = count + 1;
            }
            if (count == 1) {
                System.out.println(s.charAt(i));
                break;
            }}}}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_10.java
Nishant Nahar - 241551078
w
```

11. Write a Java program to demonstrate the difference between String, StringBuffer, and StringBuilder with performance comparison.

```java
class as_1_q_11 {
    public static void main(String[] a) {
        System.out.println("Nishant Nahar - 241551078");
        long t = System.currentTimeMillis();
        String s = "";
        for (int i = 0; i < 10000; i++) s = s + "a";
        System.out.println("String: " + (System.currentTimeMillis() - t));
        t = System.currentTimeMillis();
        StringBuffer sb = new StringBuffer();
        for (int i = 0; i < 10000; i++) sb.append("a");
        System.out.println("StringBuffer: " + (System.currentTimeMillis() - t));
        t = System.currentTimeMillis();
        StringBuilder sb2 = new StringBuilder();
        for (int i = 0; i < 10000; i++) sb2.append("a");
        System.out.println("StringBuilder: " + (System.currentTimeMillis() - t));
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_11.java
Nishant Nahar - 241551078
String: 33
StringBuffer: 2
StringBuilder: 1
```

12. Write a Java program to toggle the case of characters without using built- in case conversion methods.

```java
class as_1_q_12 {
    public static String toggleCase(String s) {
        String r = "";
        for (int i = 0; i < s.length(); i++) {
            char ch = s.charAt(i);
            if (ch >= 'a' && ch <= 'z') r = r + (char) (ch - 32);
            else if (ch >= 'A' && ch <= 'Z') r = r + (char) (ch + 32);
            else r = r + ch;}
        return r;}
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "Hola Amigos";
        String t = toggleCase(s);
        System.out.println(t);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_12.java
Nishant Nahar - 241551078
hOLA aMIGOS
```

13. Write a Java program to replace multiple spaces in a sentence with a single space.

```java
class as_1_q_13 {
    public static String rmv(String s) {
        boolean sp = false;
        String temp = "";
        for (int i = 0; i < s.length(); i++) {
            char c = s.charAt(i);
            if (c != ' ') {
                temp = temp + c;
```

```
                sp = false;
            } else {
                if (!sp) {
                    temp = temp + ' ';
                    sp = true;
                }}}
        s = temp;
        return s;}
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "How    are    you doing my    buddy   ?";
        s = rmv(s);
        System.out.println(s);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java assign_13.java
Nishant Nahar - 241551078
How are you doing my buddy ?
```

14. Write a Java program to find the longest and shortest word in a given sentence.

```
class as_1_q_14 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "Java programming is fun and powerful";
        String[] w = s.split(" ");
        String longWord = "", shortWord = w[0];
        for (int i = 0; i < w.length; i++) {
            if (w[i].length() > longWord.length()) longWord = w[i];
            if (w[i].length() < shortWord.length()) shortWord = w[i];
        }
        System.out.println("Longest: " + longWord);
        System.out.println("Shortest: " + shortWord);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_14.java
Nishant Nahar - 241551078
Longest: programming
Shortest: is
```

15. Write a Java program to implement basic string compression such as converting “aaabbcc” into “a3b2c2”

```
class as_1_q_15 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "aabbccddggffrryt";
        StringBuilder r = new StringBuilder();
        int c = 1;
        for (int i = 0; i < s.length(); i++) {
            if (i + 1 < s.length() && s.charAt(i) == s.charAt(i + 1)) c++;
            else {
                r.append(s.charAt(i)).append(c);
                c = 1;
            }}
```

```
        System.out.println(r);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_15.java
Nishant Nahar - 241551078
a2b2c2d2g2f2r2y1t1
```

16. Write a Java program to validate a password based on length, uppercase, lowercase, digit, and special character rules.

```
class as_1_q_16 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String p = "Abc123$";
        int count = 0;
        if (p.length() >= 6) {
            for (int i = 0; i < p.length(); i++) {
                char c = p.charAt(i);
                if (c >= 'A' && c <= 'Z') count++;
                else if (c >= 'a' && c <= 'z') count++;
                else if (c >= '0' && c <= '9') count++;
                else if ("!@#$%^&*()".indexOf(c) != -1) count++;
            }}
        if (count >= 5) System.out.println("Valid");
        else System.out.println("Invalid");
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_16.java
Nishant Nahar - 241551078
Valid
```

17. Write a Java program to check whether one string is rotation of another using only one string operation.

```
class as_1_q_17 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s1 = "nish";
        String s2 = "shni";
        int n = s1.length();
        int found = 0;
        if (s1.length() == s2.length()) {
            for (int i = 0; i < n; i++) {
                int j;
                for (j = 0; j < n; j++) {
                    if (s1.charAt((i + j) % n) != s2.charAt(j)) break;}
                if (j == n) {
                    found = 1;
                    break;
                }}}
        if (found == 1) System.out.println("Rotation");
        else System.out.println("Not Rotation");
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_17.java
Nishant Nahar - 241551078
Rotation
```

18. Write a Java program to print all possible substrings of a given string and count them.

```java
class as_1_q_18 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "abcfd";
        int count = 0;
        for (int i = 0; i < s.length(); i++) {
            for (int j = i + 1; j <= s.length(); j++) {
                System.out.println(s.substring(i, j));
                count++;
            }}
        System.out.println("Total substrings: " + count);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_18.java
Nishant Nahar - 241551078
a
ab
abc
abcf
abcfd
b
bc
bcf
bcfd
c
cf
cfd
f
fd
d
Total substrings: 15
```

19. Write a Java program to find the most repeated word in a paragraph ignoring case sensitivity.

```java
class as_1_q_19 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "Java is fun. Java is powerful. java was easy. Java is not
good.".toLowerCase();
        String[] w = s.split(" ");
        String max = w[0];
        int fmax = 0;
        for (int i = 0; i < w.length; i++) {
            int f = 0;
            for (int j = 0; j < w.length; j++) {
                if (w[i].equals(w[j])) f++;}
            if (f > fmax) { fmax = f; max = w[i];}}
        System.out.println(max);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_19.java
Nishant Nahar - 241551078
java
```

20. Write a Java program to check whether a string contains only unique characters.

```java
class as_1_q_20 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "abcdegwij";
        int n = s.length();
        int f = 0;
        for (int i = 0; i < n; i++) {
            for (int j = i + 1; j < n; j++) {
                if (s.charAt(i) == s.charAt(j)) {
                    f = 1;
                    break;
                }}
            if (f == 1) break; }
        if (f == 0) System.out.println("Unique");
        else System.out.println("Not Unique");
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_20.java
Nishant Nahar - 241551078
Unique
```

21. Write a Java program to determine whether a string is a valid IPv4 address.

```java
class as_1_q_21 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "192.168.1.1";
        String[] p = s.split("\\.");
        int f = 0;
        if (p.length == 4) {
            for (int i = 0; i < 4; i++) {
                int n = Integer.parseInt(p[i]);
                if (n < 0 || n > 255) {
                    f = 1;
                    break;}}
        } else f = 1;
        if (f == 0) System.out.println("Valid");
        else System.out.println("Invalid");
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_21.java
Nishant Nahar - 241551078
Valid
```

22. Write a Java program to implement your own version of index of() method without using built-in functions.

```java
class as_1_q_22 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "Nishant Pankaj Nahar";
        String t = "Pankaj Nahar";
        int n = s.length();
        int m = t.length();
        int index = -1;
```

```
        for (int i = 0; i <= n - m; i++) {
            int j;
            for (j = 0; j < m; j++) {
                if (s.charAt(i + j) != t.charAt(j)) break; }
            if (j == m) { index = i; break; }}
        System.out.println(index);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_22.java
Nishant Nahar - 241551078
8
```

23. Write a Java program to determine whether two strings differ by exactly one character.

```
class as_1_q_23 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s1 = "hello";
        String s2 = "hollo";
        int n1 = s1.length();
        int n2 = s2.length();
        int f = 0;
        if (n1 == n2) {
            for (int i = 0; i < n1; i++) {
                if (s1.charAt(i) != s2.charAt(i)) f++;}
        } else f = -1;
        if (f == 1) System.out.println("Yes, differ by 1");
        else System.out.println("No");
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_23.java
Nishant Nahar - 241551078
Yes, differ by 1
```

24. Write a Java program to generate all permutations of a given string.

```
class as_1_q_24 {
    static void perm(String p, String up) {
        if (up.length() == 0) {
            System.out.println(p);
            return;}
        for (int i = 0; i < up.length(); i++) {
            perm(p + up.charAt(i), up.substring(0, i) + up.substring(i + 1));}}
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "abc";
        perm("", s);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_24.java
Nishant Nahar - 241551078
abc
acb
bac
bca
cab
cba
```

25. Write a Java program to check whether a string follows a given pattern (e.g., “abba” matches “dog cat cat dog”).

```java
class as_1_q_25 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String p = "abba";
        String s = "dog cat cat dog";
        String[] w = s.split(" ");
        if (p.length() != w.length) {
            System.out.println("No");
            return;}
        for (int i = 0; i < p.length(); i++) {
            for (int j = i + 1; j < p.length(); j++) {
                if (p.charAt(i) == p.charAt(j) && !w[i].equals(w[j])) {
                    System.out.println("No");
                    return;}
                if (p.charAt(i) != p.charAt(j) && w[i].equals(w[j])) {
                    System.out.println("No");
                    return;
                }}}
        System.out.println("Yes");
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_25.java
Nishant Nahar - 241551078
Yes
```

26. Write a Java program to calculate the edit distance between two strings.

```java
class as_1_q_26 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s1 = "horse";
        String s2 = "ros";
        int n = s1.length();
        int m = s2.length();
        int[][] dp = new int[n + 1][m + 1];
        for (int i = 0; i <= n; i++) dp[i][0] = i;
        for (int j = 0; j <= m; j++) dp[0][j] = j;
        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= m; j++) {
                if (s1.charAt(i - 1) == s2.charAt(j - 1)) {
                    dp[i][j] = dp[i - 1][j - 1];
                } else {
                    int replace = dp[i - 1][j - 1];
                    int delete = dp[i - 1][j];
                    int insert = dp[i][j - 1];
                    int min = replace;
                    if (delete < min) min = delete;
                    if (insert < min) min = insert;
                    dp[i][j] = 1 + min;
                }}}
        System.out.println(dp[n][m]);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_26.java
Nishant Nahar - 241551078
3
```

27. Write a Java program to detect whether a string is a valid palindrome considering only alphanumeric characters.

```java
class as_1_q_27 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "A man, a plan, a canal: Panama";
        s = s.toLowerCase();
        String clean = "";
        for (int i = 0; i < s.length(); i++) {
            char c = s.charAt(i);
            if ((c >= 'a' && c <= 'z') || (c >= '0' && c <= '9')) clean += c;}
        int i = 0, j = clean.length() - 1;
        while (i < j) {
            if (clean.charAt(i) != clean.charAt(j)) {
                System.out.println("No");
                return;}
            i++;j--;}
        System.out.println("Yes");
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_27.java
Nishant Nahar - 241551078
Yes
```

28. Write a Java program to implement a simple Caesar Cipher encryption and decryption technique.

// Did not understand this

29. Write a Java program to find the longest palindromic substring in a given string.

```java
class as_1_q_29 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        String s = "bababd", r = "";
        for (int i = 0; i < s.length(); i++) {
            String o = e(s, i, i), v = e(s, i, i + 1);
            if (o.length() > r.length()) r = o;
            if (v.length() > r.length()) r = v;}
        System.out.println(r);}
    static String e(String s, int l, int r) {
        while (l >= 0 && r < s.length() && s.charAt(l) == s.charAt(r)) {
            l--;r++;}
        return s.substring(l + 1, r);
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java .\assign_29.java
Nishant Nahar - 241551078
babab
```

30. Write a Java program to design a menu-driven String Utility System that performs multiple string operations using modular methods.

```java
import java.util.Scanner;
class as_1_q_30 {
    public static void main(String[] args) {
        System.out.println("Nishant Nahar - 241551078");
        Scanner sc = new Scanner(System.in);
        String s = "Hello My Name is Nishant";
        while (true) {
            System.out.println("1.Length 2.Upper 3.Lower 4.Split 5.Reverse 6.Exit");
            int ch = sc.nextInt();
            sc.nextLine();
            if (ch == 1) System.out.println(s.length());
            else if (ch == 2) System.out.println(s.toUpperCase());
            else if (ch == 3) System.out.println(s.toLowerCase());
            else if (ch == 4) {
                String[] t = s.split("");
                for (String x : t) System.out.print(x + " ");
                System.out.println();
            } else if (ch == 5) System.out.println(r(s));
            else break;
        }
        sc.close();
    }
    static String r(String s) {
        String t = "";
        for (int i = s.length() - 1; i >= 0; i--) t += s.charAt(i);
        return t;
    }}
```

OUTPUT

```
PS B:\4th_sem\java_lab\assignment_8> java assign_30.java
Nishant Nahar - 241551078
1.Length 2.Upper 3.Lower 4.Split 5.Reverse 6.Exit
1
24
1.Length 2.Upper 3.Lower 4.Split 5.Reverse 6.Exit
2
HELLO MY NAME IS NISHANT
1.Length 2.Upper 3.Lower 4.Split 5.Reverse 6.Exit
3
hello my name is nishant
1.Length 2.Upper 3.Lower 4.Split 5.Reverse 6.Exit
4
H e l l o   M y   N a m e   i s   N i s h a n t
1.Length 2.Upper 3.Lower 4.Split 5.Reverse 6.Exit
5
tnahsiN si emaN yM olleH
1.Length 2.Upper 3.Lower 4.Split 5.Reverse 6.Exit
6
```