Experiment No. 6
Implement a program on 2D array and
Date of Performance:
Date of Submission:

Aim: To use 2D arrays and Strings for solving given problem.

Objective: To use 2D array concept and strings in java to solve real world problem

Theory:

- An array is used to store a fixed-size sequential collection of data of the same type.
- An array can be init in two ways:

1. Initializing at the time of declaration:

dataType[] myArray = {value0, value1, ..., valuek};

2. Dynamic declaration:

dataType[] myArray = new dataType[arraySize]; myArray[index]
= value;

- Two – dimensional array is the simplest form of a multidimensional array. Data of only same data type can be stored in a 2D array.Data in a 2D Array is stored in a tabular manner which can be represented as a matrix.
- A 2D Array can be declared in 2 ways:

1. Intializing at the time of declaration:

dataType[][] myArray = { {valueR1C1, valueR1C2...}, {valueR2C1, valueR2C2...},..}

2. Dynamic declaration:

dataType[][] myArray = new dataType[x][y];

myArray[row_index][column_index] = value;

In Java, string is basically an object that represents sequence of char values. An array of characters works same as Java string. Java String class provides a lot of methods to perform operations on strings such as compare(), concat(), equals(), split(), length(), replace(), compareTo(), intern(), substring() etc.

1.String literal

To make Java more memory efficient (because no new objects are created if it exists already in the string constant pool).

Example:

String demoString = "GeeksforGeeks";

2. Using new keyword

- String s = new String("Welcome");
- In such a case, JVM will create a new string object in normal (non-pool) heap memory and the literal "Welcome" will be placed in the string constant pool. The variable s will refer to the object in the heap (non-pool)

Example:

String demoString = new String ("GeeksforGeeks");

Code:

1)

MATR

IX :

```
 import java.util.*;   class Matrix {
public static void main(String[]
args) {   Scanner sc = new
Scanner(System.in);
 int a[][] = { { 3, 6 }, { 6,
2 } };   int b[][] = { { 5, 9
}, { 9, 3 } };   int c[][] =
new int[2][2];
 int i, j, k;
 System.out.println("\nGiven A Matrix
 is...");  for (i = 0; i < 2; i++) {  for (j = 0;
j < 2; j++) {   System.out.print(a[i][j] +
 "\t");

}
 System.out.println("\n");
 }
 System.out.println("\nGiven B Matrix is...");
 for (i = 0; i < 2;
i++) {   for (j = 0; j
< 2; j++) {
 System.out.print(b[i][j] + "\t");
 }
 System.out.println("\n");
 }   for (i = 0; i < 2; i++) {   for (j =
0; j < 2; j++) {c[i][j] = a[i][j] +
b[i][j];
 }
 }
 System.out.println("\nMatrix Addition is...");
 for (i = 0; i < 2;
i++) {   for (j = 0; j
< 2; j++) {
 System.out.print(c[i][j] + "\t");
 }
 System.out.println("\n");
 }   for (i = 0; i <
2; i++) {   for (j
= 0; j < 2; j++)
{   c[i][j] =
a[i][j] - b[i][j];
```

```
 }
 }
System.out.println("\nMatrix Subtraction is...");
for (i = 0; i < 2;
i++) {   for (j = 0; j
< 2; j++) {
System.out.print(c[
i][j] + "\t");
 }
 System.out.println("\n");
 }   for (i = 0; i < 2; i++) {   for (j = 0; j < 2; j++) {   for (k
= 0; k < 2; k++) {c[i][j] = c[i][j] + a[i][k] * b[k][j];
 }
 }
 }
 System.out.println("\nMatrix Multiplication is...");
 for (i = 0; i < 2;
i++) {   for (j = 0; j
< 2; j++) {
 System.out.print(c[i][j] + "\t");
 }
 System.out.println("\n");
 }
 }
}
2) STRING COMPARE :  public class Stringcomp {   public static void
   main(String[] args) {
 String a = "Apple";
 String b = "Apple";
 String c = "Strawberry";
 String d = new String("Apple");
```

```
System.out.println(a.equals(b));
System.out.println(a.equals(c)); System.out.println(a.equals(d));
}
}
```

3) STRING CONCATENATION:

```
 public class Stringconc {
public static void main(String[]
args) {
 String a1 = "Ice";
 String a2 = "Cream";
 String a3 = a1.concat(a2);
 System.out.println(a3);
 }
}
```

Conclusion:

The concept of strings and 2D arrays is widely used for various purposes. Strings are used to represent and manipulate textual data. They provide a range of methods for operations such as concatenation, substring extraction, and searching. 2D arrays, on the other hand, allow us to store and manipulate data in a tabular format with rows and columns. They are useful for tasks such as representing grids, matrices, or tables of data. By leveraging the power of strings and 2D arrays, we can efficiently handle and process textual and tabular data in our Java programs.