Experiment No. 6
Implement a program on 2D array & strings functions.
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:

Code for 2D array:

```
import java.util.Scanner;
public class ArrayEx
{
 public static void main(String[] args)
 {
   Scanner sc = new Scanner(System.in);
   System.out.println("Enter the Number of Elements:");
   int size = sc.nextInt();
   int arr[] = new int[size];
   System.out.println("enter the Element of the array:");
   for(int i =0;i<size;i++)
   {
     arr[i] = sc.nextInt();
   }
   System.out.println("The Element Entered Are:");
   for(int i =0;i<size;i++)
```

```
    {
       System.out.println(arr[i]+" ");
    }
  }
}
```

Output:

```
C:\Users\Mitanksh\Documents\java practical[1]\AIDS S1 14 Mitanksh>java 2DArray.java
Enter Number of rows :
3
Enter number of columns:
3
Enter the elements  :
1 2 3

4 5 6

7 8 9

elements in the array are:
1 2 3
4 5 6
7 8 9

C:\Users\Mitanksh\Documents\java practical[1]\AIDS S1 14 Mitanksh>
```

Code for String Function:

1.CompareTo:

```
import java.util.Scanner;


public class Main
{
   public static void main(String[] args)
  {
      Scanner scanner = new Scanner(System.in);
      System.out.println("Enter the first string:");
      String string1 = scanner.nextLine();
      System.out.println("Enter the second string:");
      String string2 = scanner.nextLine();
      int comparisonResult = string1.compareTo(string2);
```

```
        if(comparisonResult < 0)
        {
            System.out.println("The first string is less than the second string.");
        }
        else if(comparisonResult > 0)
        {
            System.out.println("The first string is greater than the second string.");
        }
        else
        {
            System.out.println("The strings are equal.");
        }
        scanner.close();
    }
}
```

Output:

```
C:\Users\Mitanksh\Documents\java practical[1]\AIDS S1 14 Mitanksh\String Function>java CompareTo.java
Enter the first string:
10 20 30 40 50
Enter the second string:
10 30 50
The first string is less than the second string.
```

2.EqualString

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter the first string:");
        String string1 = scanner.nextLine();

        System.out.println("Enter the second string:");
        String string2 = scanner.nextLine();
```

```
        if(string1.equals(string2)) {
            System.out.println("The strings are equal.");
        } else {
            System.out.println("The strings are not equal.");
        }

        scanner.close();
    }
}
```

Output:

```
C:\Users\Mitanksh\Documents\java practical[1]\AIDS S1 14 Mitanksh\String Function>java EqualString.java
Enter the first string:
10 20 30 40
Enter the second string:
10 30 50 60
The strings are not equal.
```

3.StringLength:

```
import java.util.Scanner;

public class Main
{
    public static void main(String[] args)
    {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter a string:");
        String userInput = scanner.nextLine();
        System.out.println("The length of the string is: " + userInput.length());
        for(int i = 0; i < userInput.length(); i++)
        {
            System.out.println("Character at position " + i + " is: " + userInput.charAt(i));
        }

        scanner.close();
    }
```

}

Output:

```
C:\Users\Mitanksh\Documents\java practical[1]\AIDS S1 14 Mitanksh\String Function>java StringLength.java
Enter a string:
10 20 30 40 50
The length of the string is: 14
Character at position 0 is: 1
Character at position 1 is: 0
Character at position 2 is:
Character at position 3 is: 2
Character at position 4 is: 0
Character at position 5 is:
Character at position 6 is: 3
Character at position 7 is: 0
Character at position 8 is:
Character at position 9 is: 4
Character at position 10 is: 0
Character at position 11 is:
Character at position 12 is: 5
Character at position 13 is: 0
```

4.IsEmpty:

```
import java.util.Scanner;


public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter a string:");
        String userInput = scanner.nextLine();

        if(userInput.isEmpty()) {
            System.out.println("The string is empty.");
        } else {
            System.out.println("The string is not empty.");
        }

        scanner.close();
    }
}
```

Output:

```
C:\Users\Mitanksh\Documents\java practical[1]\AIDS S1 14 Mitanksh\String Function>java IsEmpty.java
Enter a string:
10 20 30 40
The string is not empty.
```

Conclusion:

Comment on how you have used the concept of string and 2D array.

I have used the concept of string and 2D array in various programming tasks and projects. Some examples are:

- I have used a string array to store and manipulate a list of words or sentences. For example, I have used a string array to sort, search, reverse, concatenate, or compare strings. I have also used a string array to store the names of students, countries, colors, etc.
- I have used a 2D array to store and process data in a tabular or matrix form. For example, I have used a 2D array to perform matrix operations, such as addition, multiplication, transpose, inverse, etc. I have also used a 2D array to store the values of a game board, an image, a spreadsheet, etc.
- I have used a 2D array of strings to store and display data that has both rows and columns. For example, I have used a 2D array of strings to store the information of students, such as their name, roll number, marks, etc. I have also used a 2D array of strings to store the history of web pages visited by a user.