Algorithms and Data Structures
ALGO 08006

LIMERICK INSTITUTE OF TECHNOLOGY

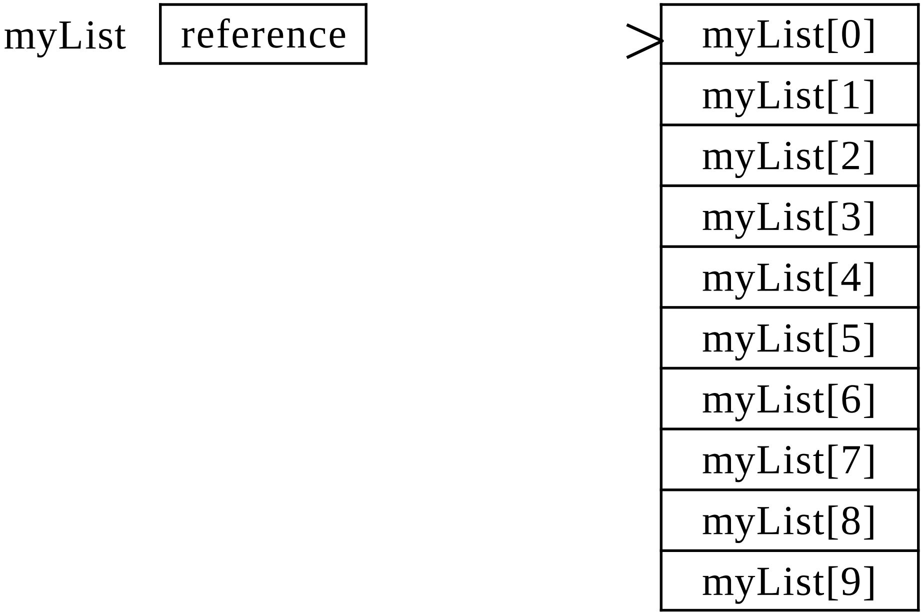
SCHOOL OF SCIENCE, ENGINEERING & I.T.

Department of Information Technology

Introducing Arrays

Array is a data structure that represents a collection of the same types of data. Java treats these arrays as objects.

```
double[] myList = new double[10];
```



myList reference

myList[0]
myList[1]
myList[2]
myList[3]
myList[4]
myList[5]
myList[6]
myList[7]
myList[8]
myList[9]

An Array of 10 Elements
of type `double`

Declaring Array Variables

- `datatype[] arrayname;`

  Example:

  ```
  int[] myList;
  ```

- `datatype arrayname[];`

  Example:

  ```
  double myList[];
  ```

Creating Arrays

```
arrayName = new datatype[arraySize];
```

Example:

```
myList = new double[10];
```

Declaring and Creating in One Step

```
datatype[] arrayname =
                 new datatype[arraySize];
```

Example:

```
double[] myList = new double[10];
```

```
datatype arrayname[] = new
     datatype[arraySize];
```

Example:

```
double myList[] = new double[10];
```

Initialising Arrays

- Using a loop:

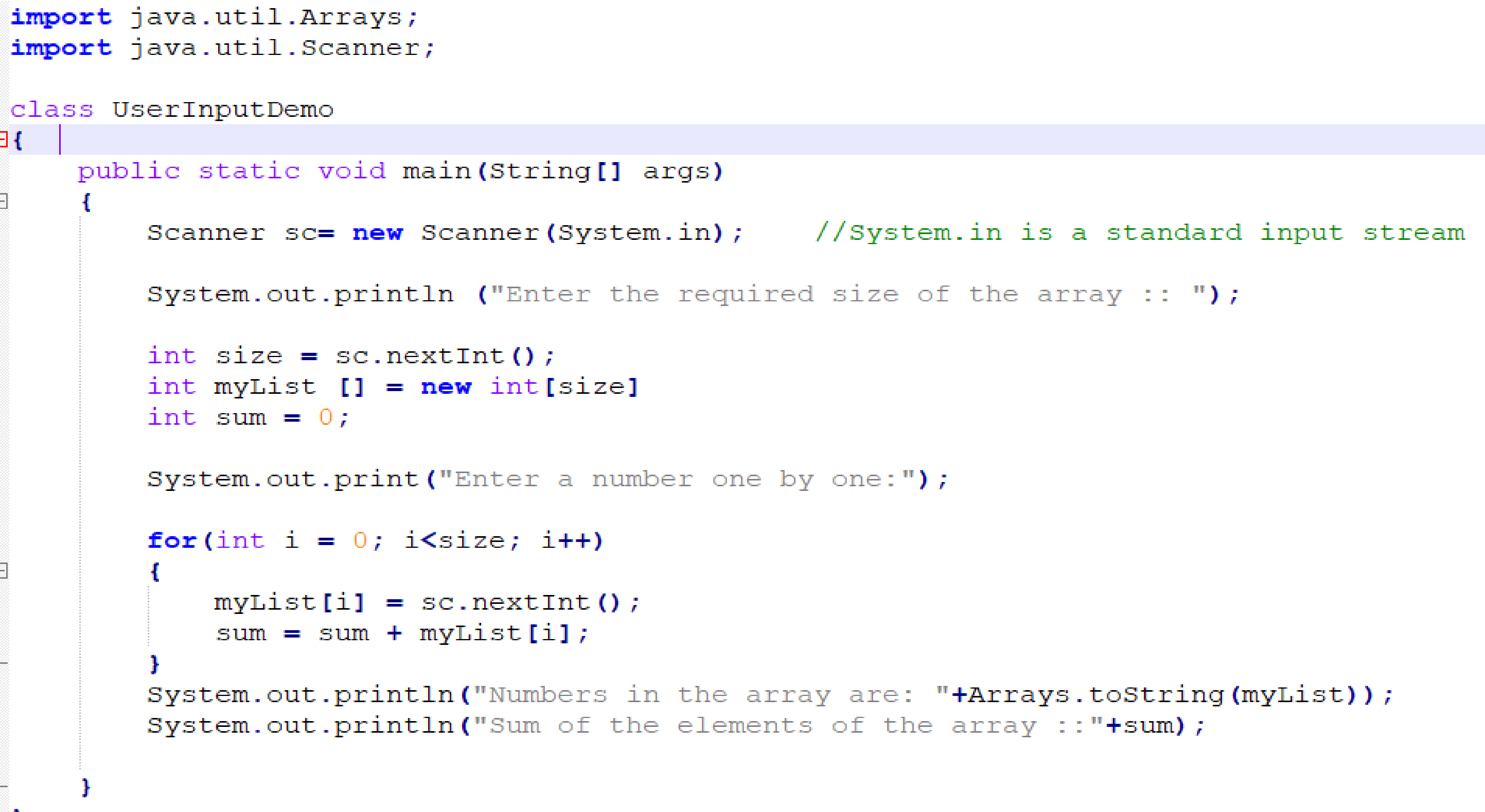
```
for (int i = 0; i < myList.length; i++)
   myList[i] = (double)i;
```

- Declaring, creating, initializing in one step:

```
double[] myList = {1.9, 2.9, 3.4, 3.5};
```

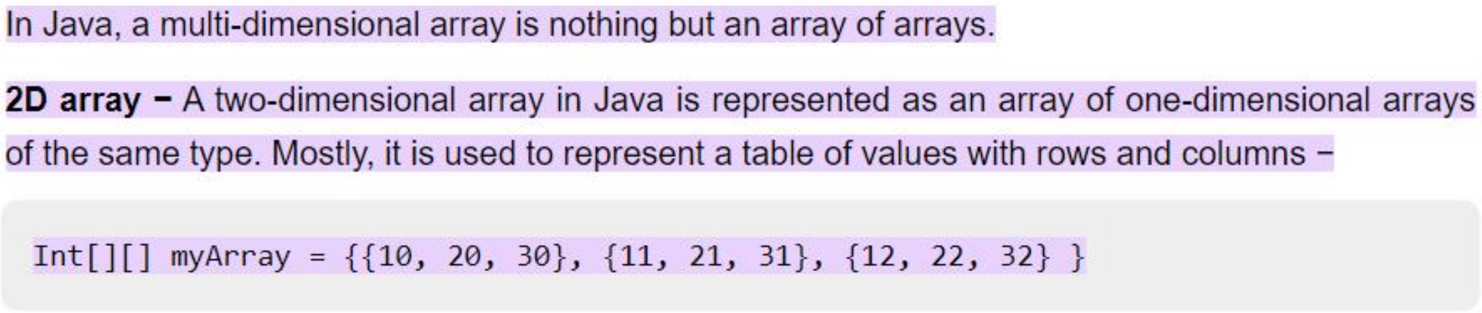
Example

Objective: The program receives numbers as input and sums the number in the array. The size of the array is specified by creating a variable called “size”.



```
import java.util.Arrays;
import java.util.Scanner;

class UserInputDemo
{
    public static void main(String[] args)
    {
        Scanner sc= new Scanner(System.in);    //System.in is a standard input stream

        System.out.println ("Enter the required size of the array :: ");

        int size = sc.nextInt();
        int myList [] = new int[size]
        int sum = 0;

        System.out.print("Enter a number one by one:");

        for(int i = 0; i<size; i++)
        {
            myList[i] = sc.nextInt();
            sum = sum + myList[i];
        }
        System.out.println("Numbers in the array are: "+Arrays.toString(myList));
        System.out.println("Sum of the elements of the array ::"+sum);

    }
```

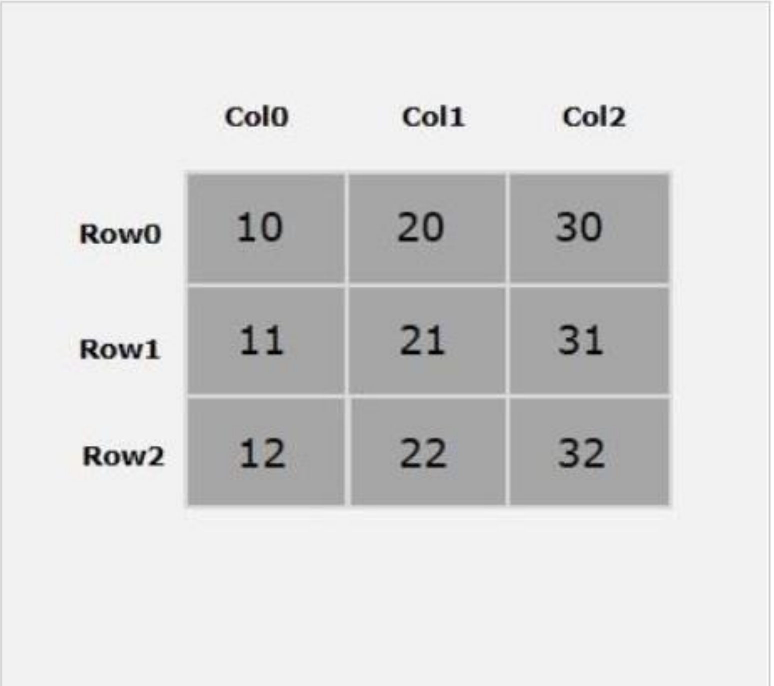
Multidimensional Arrays



In Java, a multi-dimensional array is nothing but an array of arrays.

2D array – A two-dimensional array in Java is represented as an array of one-dimensional arrays of the same type. Mostly, it is used to represent a table of values with rows and columns −

```
Int[][] myArray = {{10, 20, 30}, {11, 21, 31}, {12, 22, 32} }
```



	Col0	Col1	Col2
Row0	10	20	30
Row1	11	21	31
Row2	12	22	32

Initialising Multidimensional Arrays

```
int[][] matrix = new int[10][10];
```

or

```
int matrix[][] = new int[10][10];

for (int i=0; i<matrix.length; i++)
  for (int j=0; j<matrix[i].length; j++)
  {
    matrix[i][j] = (int)(Math.random()*1000);
  }
```

Ragged Arrays

Each row in a two-dimensional array is itself an array. So, the rows can have different lengths. Such an array is known as *a ragged array*. For example,

```
int[][] matrix =
{
  {1, 2, 3, 4, 5},
  {2, 3, 4, 5},
  {3, 4, 5},
  {4, 5},
  {5}
};
```

Exercise 1

Write a program that creates an array with 100 elements.

You must initialise each element in the array to a random value in the range 1 - 1000;

Your program must then calculate and display (using a method to achieve each of the following tasks)

1. The largest number in the array.
2. The smallest number in the array.
3. The sum of all the numbers.
4. The average of all the numbers.
5. The frequency of the number 7 in the array.

Exercise 2

Write a program that creates an array with 5000 elements.

Each element in this array must be initialised with a random value in the range 0 – 29.

Your program must then output the frequency of each number in the range 0 - 29 which is stored in the array.

You must also output the modal value (the modal value is the value which occurs most frequently in the array.

Exercise 3

Write a program that creates an array with 5000 elements.

Each element in this array must be initialised with a random value in the range 5 – 50.

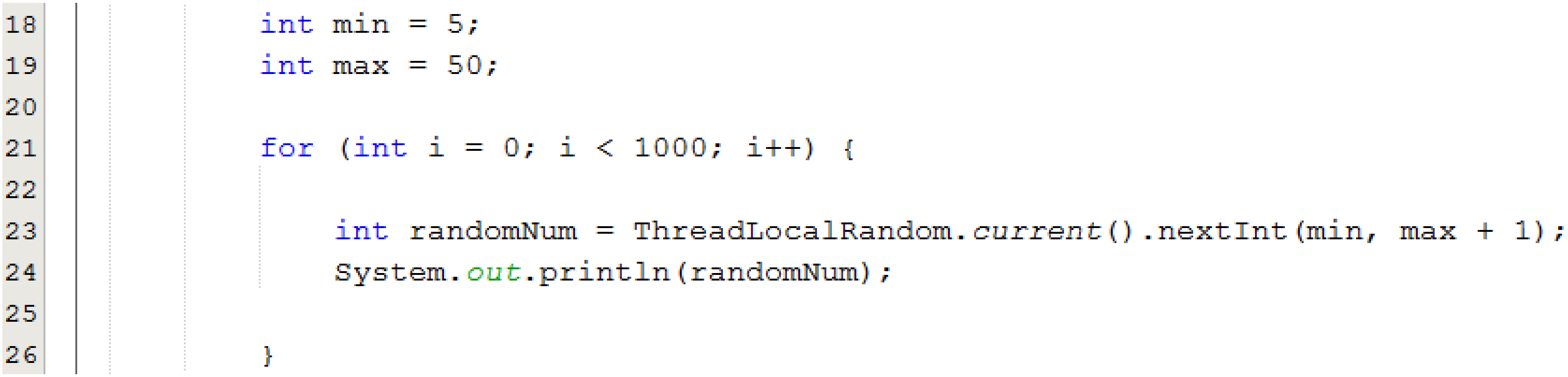
Your program must firstly output the arrays contents in columns of 20 values.

You must then output the numbers in the array which are multiples of 6, 7 and 8 (if any) along with the index of the array in which they appear.

If there are no multiples of 6, 7 and 8 in the array, your program must print a suitable message.

Generating Random Numbers

There are a number of ways in which you can generate random numbers. Here’s one.



```java
int min = 5;
int max = 50;

for (int i = 0; i < 1000; i++) {

    int randomNum = ThreadLocalRandom.current().nextInt(min, max + 1);
    System.out.println(randomNum);

}
```