Arrays in Java

Assignment

1. What do you mean by an Array?

Ans.1 An Array is a data structure that stores a collection of elements, such as numbers or strings, in a specific order. Each element in an array is identified by an index, which represents its position in the array. Arrays are commonly used in programming to organize and manipulate data efficiently, as elements can be accessed and modified based on their index. The size of an array is usually fixed at the time of creation, meaning you need to specify the number of elements it can hold.

2. How to create an Array?

Ans.2 Steps to create an array:-

1. Declare the array: Declare on the data type and declare the array variable using square brackets “[ ]” after the data type.

2. Initialize the array: Allocate memory for the array and set its initial values. You can either specify the values when creating the array or later on.

Example:

```
// Declaring and initializing an integer array with values
int nums [ ] = {1,2,3,4,5};
```

In Java, Arrays have a fixed size, and you need to specify the size when you create them.

3. Can we change the size of an array at run time?

Ans3. In languages like Python, we can change the size of a list (similar to an array) at runtime by using built-in function like ‘append( ) ‘ or ‘resize( ) ‘. However, in languages like C or Java,arrays have a fixed size and cannot be resized directly. To simulate dynamic sizing, you’d need to create a new array with the desired size and copy the elements from the old array to the new one.

4. Can you declare an array without assigning the size of an array?

Ans4. Yes, you can declare an array without assigning the size of an array without assigning the size directly.

Syntax:

```
DataType[ ] arrayName;
```

For example, to declare an array of integers without specifying the size, you can use:

```
Int nums[ ];
```

However, you will need to assign the size of the array before using it, otherwise, you’ll encounter a ‘NullPointerException’ when trying to access elements. You can initialize the array with a specific size later in your code using ‘new’ keyword, like this:

```
nums = new int[10] ; // Here, the array size is set to 10
```

Alternatively,you can also declare and initialize an array in a single line like this:

```
int nums[ ] = new int [10]; // Array with size 10
```

5. What is the default value of Array?

Ans5. In many languages like C,C++, Java and JavaScript, the default value for an array of numeric types is usually 0. For arrays of other types e.g., strings or objects , the default value is often “null” or an equivalent value that represents the absence of a value. However, it’s important to note that not all programming languages have a concept of default values for arrays, and some may leave the elements uninitialized, containing random or garbage values.

6. What is a 1D array with an example?

Ans6. A 1D array is a linear data structure that stores elements of the same data type in a single row. It can be visualized as a list of items arranged in a single line. Here’s an example of a 1D array in Python:

```
numbers = [1, 2, 3, 4, 5]
```

In this example, ‘numbers’ is a 1D array that holds five elements: 1,2,3,4 and 5. Each elements can be accessed using its index in the array (e.g., ‘numbers[0]’ gives 1, ‘numbers[1]’ gives 2, and so on).

7. Write a program on a 2D array?

Ans7.

```
import java.util.Scanner;

public class TwoDArrayExample
        {
            public static void main(String[ ] args)
                {
                   // Define the dimensions of the 2D array
                  int rows = 3;
```

```
                int columns = 4;
                // Create a 2D array with the specified dimensions
                int [ ][ ] myArray = new int[rows] [columns];
                 // Initialize the 2D array with some values
                int value =1;
            for (int i=0;i<rows;i++)
            {
                for (int j = 0; j < columns; j++)
                {
                   myArray[i][j] = value;
                value++;
                 }
               }
          // Display the elements of the 2D array
    System.out.println("Elements of the 2D array : ");
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < columns; j++)
        {
           System.out.print(myArray[i][j] + " ");
         }
         System.out.println( );
     }
  }
}
```