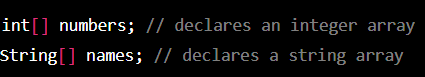
Arrays in Java

Assignment-

Question 1- What do you mean by an Array?

Ans-

An array is a data structure that allows you to store a fixed-size sequential collection of elements of the same data type. It is declared using square brackets after the data type, like this:



Single variable holding multiple values which improves readability of the program.

Question 2- How to create an Array?

Ans-

Array declarations:

1. 1-Dimensional Array
2. 2-Dimensional Array
3. Jagged Array

1-D Array: Declaration of array

. int[] a;//recommended to use as variable is separated from type.

. int a[];

. int []a;

. int[6] a; // compile time error. We cannot specify the size.

For example - to create an array of integer:



Question 3- can we change the size of an array at run time?

Ans-

NO, the size of an array in Java is fixed at the time of its creation and cannot be changed at runtime.

If we need to store a collection of data that can grow or shrink dynamically, you can use one of the collection classes provided by the Java API, such as ArrayList or LinkedList. These classes

Implement the list interface and provide methods to add, remove, and access elements in the list.

Question 4- can you declare an array without assigning the size of an array ?

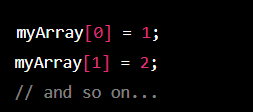
Ans-

No, in Java, we cannot declare an array without assigning the size of the array. When you declare an array in Java. you need to specify the size of the array explicitly.

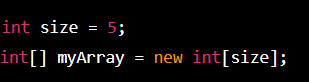
Let's take an example , to declare an array of integers with a size of 10, you would write:

int[] myArray = new int[10];

This creates an array of integers named ‘myArray’ with 10 elements. You can also assign values to the elements of the array:



If we want to create an array whose size is determined at runtime, you can use variable to specify the size. For example:



In this example given above the size of the array is determined by the value of the ‘size’ variable, which can be set at runtime.

Question 5- What is the default value of Array ?

Ans-

The default value of an array depends on the type of the array:

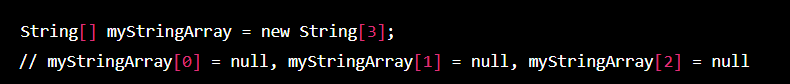
1. For integer arrays, the default value is 0 for each element.
2. For boolean arrays, the default value is false for each element .
3. For reference type arrays (e.g. String, Object), the default value is null for each element.

For example, if you declare an integer array without initializing its elements, all the elements of the array will be initialized to 0:

int[] myIntArray = new int[5];

// myIntArray[0] = 0, myIntArray[1] = 0, myIntArray[2] = 0, myIntArray[3] = 0, myIntArray[4] = 0

Similarly, if you declare a reference type array without initializing its elements, all the elements of the array will be initialized to null:



Question 6- What is a 1D array with an example ?

Ans-

A 1D array is a collection of values of the same data type that are stored in a contiguous block of memory and accessed using a single index. Here’s an example of how to create and initialize a 1D array of integers :



In this example, the array is named “numbers” and contains five integers: 1,2,3,4,and 5. The size of the array is implicitly determined by the number of elements in the initializer list enclosed in curly braces. The array elements are accessed using the index starting from 0, so ‘number[0]’ is 1, ‘numbers[1]’ is 2, and so on.

Question 7- Write a program on a 2D array ?

Ans-

ok , here’s an example of a program that creates and manipulates a 2D array in Java:

public class TwoDArrayExample {

public static void main(String[] args) {

int rows = 3;

int columns = 4;

int[][] myArray = new int[rows][columns];

// Fill the array with random values between 1 and 100

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

for (int j = 0; j < columns; j++) {

myArray[i][j] = (int)(Math.random() \* 100) + 1;

}

}

// Print the array

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

for (int j = 0; j < columns; j++) {

System.out.print(myArray[i][j] + " ");

}

System.out.println();

}

// Find the maximum value in the array

int max = myArray[0][0];

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

for (int j = 0; j < columns; j++) {

if (myArray[i][j] > max) {

max = myArray[i][j];

}

}

}

System.out.println("The maximum value in the array is " + max);

}

}

.