**Array Definition :**

Array is a fixed size index based data structure containing similar type of objects.

For example :

int[] a = new int[10] -> It is an array of 10 integers.

char[] c = new char[15] -> It is an array of 15 characters.

String[] s = new String[20] -> It is an array of 20 strings.

**Array Structure :**

Arrays in Java use zero-based indexing to store the elements where first element is stored at 0th index, second element at 1st index, third element at 2nd index and so on.

**Array Declaration :**

There are two ways to declare arrays in Java.

int[] intArray;

int intArray[];

**Array Initialization :**There are three ways to initialize array elements.

**1) Initializing individual elemets:-**

int[] intArray = new int[5];

intArray[0] = 21; intArray[1] = 15; intArray[2] = 37; intArray[3] = 53;

**2) Passing all elements at a time with new operator:-**

int[] intArray = new int[] {21, 15, 37, 53, 17};

**3) Passing all elements at a time without new operator:-**

int[] intArray = {21, 15, 37, 53, 17};

**Array Traversal :** There are two ways to traverse an array.

**1) Using for loop :**

int[] intArray = {21, 15, 37, 53, 17};

for (int i = 0; i < intArray.length; i++)

{ System.out.println(i); }

**2) Using advanced for loop / for each loop:-**

int[] intArray = {21, 15, 37, 53, 17};

for (int i : intArray)

{ System.out.println(i); }

**Multidimensional Arrays :**

Multidimensional arrays can be defined as arrays of arrays. Arrays can be two dimensional, three dimensional, four dimensional and so on.

**Two Dimensional Array :** Two – dimensional array is the simplest form of a multidimensional array.

Elements in two-dimensional arrays are commonly referred by x[i][j] where ‘i’ is the row number and ‘j’ is the column number.

int[][] arr = { { 1, 2 }, { 3, 4 } };

**Three dimentionarray:-** Three – dimensional array is a complex form of a multidimensional array.

Elements in three-dimensional arrays are commonly referred by x[i][j][k] where ‘i’ is the array number, ‘j’ is the row number and ‘k’ is the column number.

int[][][] arr = { { { 1, 2 }, { 3, 4 } }, { { 5, 6 }, { 7, 8 } }};

->000=1|001=2|010=3|011=4|100=5|101=6|110=7|111=8

**Jagged Arrays :-**Jagged arrays are also multidimensional arrays containing arrays of different length.

A jagged array in Java is a collection of arrays where each array may contain a varied number of elements. A two-dimensional array, in contrast, requires all rows and columns to have the same length. They can be created by specifying the size of each array in the declaration.

int[][] jaggedArray = {

{1, 2, 3}, // first row has three columns

{4, 5}, // second row has two columns

{6, 7, 8, 9} // third row has four columns

};

**Anonymous Array :**Anonymous array is an array without reference or name.

 System.out.println(Arrays.toString(**new** **int**[] {1, 2, 3, 4, 5}));

**java.util.Arrays Class :**

java.util.Arrays class is an utility class which contains many static methods to perform basic operations on an array.

|  |  |
| --- | --- |
| **Methods** | **Description** |
| sort() | Used to sort an array. |
| stream() | Returns a stream containing all elements of an array. |
| spliterator() | Returns spliterator of an array. |
| setAll() | Initializes all elements of an array. |
| fill() | Fills the given array with the given value. |
| copyOf() | Creates copy of an array. |
| asList() | Returns a list containing all elements of an array. |
| binarySearch() | It is used to search an array for the given value. |

**Array To List :**

List<String> list = Arrays.asList("One", "Two", "Three", "Four", "Five");

**Array To Stream :**

IntStream stream = Arrays.stream(new int[] {1, 2, 3, 4, 5});

**Array Length :**

System.out.println(intArray.length); //Output : 5

**Array Pros And Cons :**

**Pros :**

Easy to use and implement

Can hold both primitive types as well as objects

Faster data retrieval

**Cons :**

Fixed size

Not type safe

No in-built methods