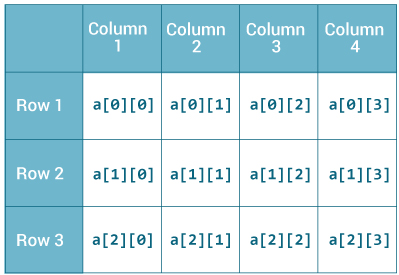
2D Arrays

int[][] a = new int[3][4];

Here, a is a two-dimensional (2d) array. The array can hold maximum of 12 elements of type int.



Remember, Java uses zero-based indexing, that is, indexing of arrays in Java starts with 0 and not 1.

Similarly, you can declare a three-dimensional (3d) array. For example,

String[][][] personalInfo = new String[3][4][2];

Here, personalInfo is a 3d array that can hold maximum of 24 (3\*4\*2) elements of type String.

**How to initialize a 2d array in Java?**

Here's an example to initialize a 2d array in Java.

int[][] a = {

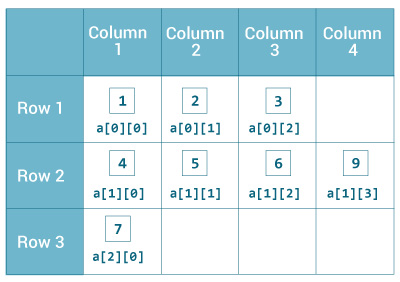
{1, 2, 3},

{4, 5, 6, 9},

{7},

};

As mentioned, each component of array a is an array in itself, and length of each rows is also different.



### Example: Print all elements of 2d array Using Loop

class MultidimensionalArray {

public static void main(String[] args) {

int[][] a = {

{1, -2, 3},

{-4, -5, 6, 9},

{7},

};

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

for(int j = 0; j < a[i].length; ++j) {

System.out.println(a[i][j]);

}

}

}

}

It's better to use [for..each loop](https://www.programiz.com/java-programming/enhanced-for-loop) to iterate through arrays whenever possible. You can perform the same task using for..each loop as:

class MultidimensionalArray {

public static void main(String[] args) {

int[][] a = {

{1, -2, 3},

{-4, -5, 6, 9},

{7},

};

for (int[] innerArray: a) {

for(int data: innerArray) {

System.out.println(data);

}

}

}

}