

# Two Dimensional Array

When we declare multi-dimensional arrays, the declared type can itself be an array, and this is how Java supports two-dimensional arrays:

```
int[][] myArray = new int[3][];    // Declares and instantiates an array of 3 integer arrays,  
//                               whose sizes are not specified
```

```
Dog[][] myDogs = new Dog[3][];    // Declares and instantiates an array of 3 arrays,  
//                               which will have Dog elements, again, the sizes of the inner arrays aren't specified
```

Type and length of array	Possible Element Values (each element is an array and can be any length)
int[3][]	[5, 7, 9, 10] [3, 6] [11, 21, 31]
Dog[3][]	[pug, rottweiler] [germanShephard, poodle, cavapoo] [beagle, boxer, bulldog, yorkie]

# Multi Dimensional Array

---

We can take that even further, the outer array can have references to any kind of array itself.

In this example, we have an outer array with three elements.

```
Object[] multiArray = new Object[3];  
multiArray[0] = new Dog[3];  
multiArray[1] = new Dog[3][];  
multiArray[2] = new Dog[3][][];
```

The first element is itself a single-dimension array.

The second element is a two-dimensional array.

And lastly, the third element is a three-dimensional array.