Scala Array

Array is a collection of mutable values. It is an index based data structure which starts from 0 index to n-1 where n is length of array.

Scala arrays can be generic. It means, you can have an Array[T], where T is a type parameter or abstract type. Scala arrays are compatible with Scala sequences - you can pass an Array[T] where a Seq[T] is required. It also supports all the sequence operations.

Following image represents the structure of array where first index is 0, last index is 9 and array length is 10.

Scala Array 1

Scala Types of array

Single dimensional array

Multidimensional array

Scala Single Dimensional Array

Single dimensional array is used to store elements in linear order. Array elements are stored in contiguous memory space. So, if you have any index of an array, you can easily traverse all the elements of the array.

Syntax for Single Dimensional Array

var arrayName : Array[arrayType] = new Array[arrayType](arraySize); or

var arrayName = new Array[arrayType](arraySize) or

var arrayName : Array[arrayType] = new Array(arraySize); or

var arrayName = Array(element1, element2 ... elementN)

Scala Array Example: Single Dimensional

class ArrayExample{

```
  var arr = Array(1,2,3,4,5)      // Creating single dimensional array
  def show(){
    for(a<-arr)                   // Traversing array elements
      println(a)
    println("Third Element  = "+ arr(2))       // Accessing elements by using index
  }
}

object MainObject{
  def main(args:Array[String]){
    var a = new ArrayExample()
    a.show()
  }
}
```

Scala Example 2: Single Dimensional

In this example, we have created an array by using new keyword which is used to initialize memory for array. The entire array elements are set to default value, you can assign that later in your code.

```
class ArrayExample{
  var arr = new Array[Int](5)        // Creating single dimensional array
  def show(){
    for(a<-arr){                    // Traversing array elements
      println(a)
    }
    println("Third Element before assignment = "+ arr(2))       // Accessing elements by
using index
```

```
    arr(2) = 10                                    // Assigning new element at 2 index
    println("Third Element after assignment = "+ arr(2))
  }
}

object MainObject{
  def main(args:Array[String]){
    var a = new ArrayExample()
    a.show()
  }
}
```

Scala Passing Array into Function

You can pass array as an argument to function during function call. Following example illustrate the process how we can pass an array to the function.

```
class ArrayExample{
  def show(arr:Array[Int]){
    for(a<-arr)            // Traversing array elements
      println(a)
    println("Third Element = "+ arr(2))       // Accessing elements by using index
  }
}

object MainObject{
  def main(args:Array[String]){
```

```
    var arr = Array(1,2,3,4,5,6)    // creating single dimensional array
    var a = new ArrayExample()
    a.show(arr)                 // passing array as an argument in the function
  }
}
```

Scala Array Example: Iterating By using Foreach Loop

You can also iterate array elements by using foreach loop. Let's see an example.

```
class ArrayExample{
  var arr = Array(1,2,3,4,5)     // Creating single dimensional array
  arr.foreach((element:Int)=>println(element))      // Iterating by using foreach loop
}

object MainObject{
  def main(args:Array[String]){
    new ArrayExample()
  }
}
```