Java Array of Strings

Java Array of Strings

Java String Array is a Java Array that contains strings as its elements. Elements of no other datatype are allowed in this array.

In this tutorial, we will learn how to declare a Java String Array, how to initialize a Java String Array, how to access elements, etc.

How to declare a String Array?

Following is the syntax to declare an Array of Strings in Java.

```
string arrayName[];
```

or

```
string[] arrayName;
```

You can use any of these two notations.

How to initialize a String Array?

To initialize a string array, you can assign the array variable with new string array of specific size as shown below.

```
arrayName = new string[size];
```

You have to mention the size of array during initialization. This will create a string array in memory, with all elements initialized to their corresponding static default value.

The default value for a string is empty string “”.

Following is an example program to initialize a string array of size 10.

Java Program

```
public class ArrayExample {
    public static void main(String[] args) {
        String names[];
        names = new String[10];
    }
}
```

We have declared and initialized the string array in two different statements. But you can combine the declaration and initialization, to form the definition of string array, as shown below.

Java Program

```
public class ArrayExample {
    public static void main(String[] args) {
        String names[] = new String[10];
    }
}
```

In the above example, we have created a string array named **names**, and initialized it to a string array of size **10** with default values of empty strings.

You can also assign strings directly to the string array when declaring it.

In the following example, we have declared and initialized string array with elements.

Java Program

```
public class ArrayExample {
    public static void main(String[] args) {
        String names[] = {"apple", "banana", "cherry", "orange", "mango"};
    }
}
```

Now **names** is a String array with size of 4, because there are four elements in the array we assigned.

How to access String Array Elements?

Following is the syntax to access an element of an array using index.

```
arrayName[index]
```

The above statement can be used either to read an element at given index, or set an element at given index. The read or write operation depends on which side of assignment operator you are placing this.

For example, in the following program, we are reading the element of string array at index **2**.

Java Program

```
public class ArrayExample {
    public static void main(String[] args) {
        String names[] = {"apple", "banana", "cherry", "orange", "mango"};
        String name = names[2];
        System.out.println(name);
    }
}
```

Output

```
cherry
```

And in the following example, we are updating the element of string array at index **2** to “lychee”.

Java Program

```
public class ArrayExample {
    public static void main(String[] args) {
        String names[] = {"Akhil", "Honey", "Lucky", "Sushanth"};
        names[2] = "lychee";
        System.out.println(names[2]);
    }
}
```

Output

```
lychee
```

How to iterate over array elements?

We can use any of these looping statements like For Loop or While Loop to iterate over elements of a Java Array.

In the following example, iterate over elements of String Array using Java While Loop.

Java Program

```
public class ArrayExample {
    public static void main(String[] args) {
        String names[] = {"apple", "banana", "cherry", "orange", "mango"};
        int index = 0;
        while (index < names.length) {
            System.out.println(names[index]);
            index++;
        }
    }
}
```

Output

```
apple
banana
cherry
orange
mango
```

In the following example, we will iterate over elements of String Array using Java For Loop.

Java Program

```
public class ArrayExample {
    public static void main(String[] args) {
        String names[] = {"apple", "banana", "cherry", "orange", "mango"};
        for (int index = 0; index < names.length; index++) {
            System.out.println(names[index]);
        }
    }
}
```

Output

```
apple
banana
cherry
orange
mango
```

And in the following example, we will use Java for-each loop, to iterate over elements of string array. For each loop can be used to execute a set of statements for each string in string array.

Java Program

```
public class ArrayExample {
    public static void main(String[] args) {
        String names[] = {"apple", "banana", "cherry", "orange", "mango"};
        for (String name: names) {
            System.out.println(name);
        }
    }
}
```

Output

```
apple
banana
cherry
orange
mango
```

Conclusion

In this Java Tutorial, we learned about String Array in specific: how to declare a string array, how to initialize it, how to access elements, etc.

Related Tutorials

- How to Create String from Char Array in Java?
- How to Create String from Byte Array in Java?
- How to Join Elements of String Array with Delimiter in Java?