Kotlin List – Examples

Kotlin List

Kotlin List is one of the three Collections (List, Set, Map). List is a collection of elements whose position in the collection is preserved, whereas in Set or Map, there is no fixed position/index for an element. Hence, elements in a list can be accessed using index.

List is immutable, so, you cannot update individual elements of list or add items to the original list.

But if you want to create a list that could be mutable, Kotlin provides MutableList<T> class to define a mutable list.

In this tutorial, we shall learn with examples on how to access and modify items in a list, and how to convert it to other Collection types.

Kotlin List – Initialize

To initialize Kotlin List, use mutableListOf(vararg items : T) method. The method returns a MutableList<T>.

In the following example,

```
var listA = listOf<String>("Example", "Program", "Tutorial")
var listB = mutableListOf("Example", "Program", "Tutorial")
```

listA is an immutable list of type List<String>, since each of the element in the list is of type String.

listB is a mutable list of type MutableList<String>.

Kotlin List – Get Element

List.get() is used to read elements of a list using index.

You can also use array style accessing of elements using square brackets `[]`.

Kotlin Program – example.kt

```
fun main(args: Array<String>) {
    //initialize list
    var listA = listOf<String>("Example", "Program", "Tutorial")

    //accessing elements using square brackets
    println(listA[0])

    //accessing elements using get()
    println(listA.get(2))
}
```

Output

```
Example
Tutorial
```

In the above example, we have used square brackets to read element present at index `0` and used get() method to read element at index `2`.

Kotlin – Add Element to List

Since we cannot modify a List<T>, we shall learn how to add an element to a mutable list, MutableList<T>.

To append item to the list, use MutableList.add(item : T) method.

Kotlin Program – example.kt

```
fun main(args: Array<String>) {
    //initialize a mutable list
    var listA = mutableListOf("Example", "Program", "Tutorial")

    //add item to the list
    listA.add("Sample")

    //print the list
    println(listA)
}
```

Output

```
[Example, Program, Tutorial, Sample]
```

The item is added to the end of list.

Kotlin – Check if List Contains an Element

To check if an element is present in the list, use List.contains() method on the list. contains() return `true` if the element is present, else `false` is returned.

Kotlin Program – example.kt

```
fun main(args: Array<String>) {
    //create a list
    var listA = listOf<String>("Example", "Program", "Tutorial")

    //check if the list contains element "Program"
    val b = listA.contains("Program")

    //based on the result of contains(), print a message
    if(b){
        print("Program is present in the list.")
    } else{
        print("Program is not present in the list.")
    }
}
```

Output

```
Program is present in the list.
```

As the element is present in the list, b is set to true and hence the print statement with the string `"Program is present in the list."` is executed.

Kotlin – Find Element in List

You can use List.find() function to find the first element that satisfies a predicate provided to the find() method. Element of the list can be accessed using `it`. Observe the following example program and output, and you shall follow what we are talking about this find() function.

In the following example, we find the first element that contains string “am”. find() method returns the first element that satisfies the condition posed by predicate.

Kotlin Program – example.kt

```
fun main(args: Array<String>) {
    //initialize the list
    var listA = listOf<String>("Example", "Program", "Tutorial")

    //find the element
    val b = listA.find { it.contains("am") }

    //print the element
    print(b)
}
```

Output

```
Example
```

Kotlin – Filter Elements of List

List.filter() function can be used to filter elements of list based on a predicate. **it** in the predicate represents each element.

In the following example, we filter a list for the elements containing “am”. filter method returns new list with items filtered based on the predicate.

Kotlin Program – example.kt

```
fun main(args: Array<String>) {
    //initialize a list
    var listA = listOf<String>("Example", "Program", "Tutorial")

    //filter the list
    val b = listA.filter { it.contains("am") }

    //print the filtered list
    print(b)
}
```

Output

```
[Example, Program]
```

The filtered output contains elements that satisfy the predicate we passed to filter() function.

Kotlin List – For Loop

To iterate over all the elements of a list, for loop can be used. Following is an example to print all elements of a list using for loop.

Kotlin Program – example.kt

```
fun main(args: Array<String>) {
    //initialize a list
    var listB = listOf<String>("Example", "Program", "Tutorial")

    //for loop on list
    for(item in listB){
        println(item)
    }
}
```

Output

```
Example
Program
Tutorial
```

Conclusion

In this Kotlin Tutorial – **Kotlin List**, we have learnt to initialize a List with items, access them, filter them based on a predicate, find an element based on a predicate, etc.

Inheritance

- Kotlin Inheritance
- Kotlin Override Method of Super Class

Abstraction

- Kotlin Abstraction
- Kotlin Abstract Class
- Kotlin - Interfaces
- Kotlin Null Safety

Exception Handling

- Kotlin Try Catch
- Kotlin Throw Exception
- Kotlin Custom Exception

Fix Compilation Errors

- Kotlin - Variable must be initialized
- Kotlin - Primary Constructor call expected
- Kotlin - Null can not be a value of a non-null type String
- Kotlin - Cannot create an instance of an abstract class

Kotlin - String Operations

- Kotlin - Compare Strings
- Kotlin - Replace String
- Kotlin - Split String
- Kotlin - Split String to Lines
- Kotlin - String Capitalize

Kotlin - Functions

- Kotlin Function - Default Arguments
- Kotlin - Use Function

Kotlin Collections

Kotlin List

➪ Kotlin List

- ✦ Kotlin List forEach

Kotlin File Operations

- ✦ Kotlin - Create File
- ✦ Kotlin - Read File
- ✦ Kotlin - Read File as List of Lines
- ✦ Kotlin - Write to File
- ✦ Kotlin - Append Text to File
- ✦ Kotlin - Check if File Exists
- ✦ Kotlin - Copy a File to Other
- ✦ Kotlin - Iterate through all files in a directory
- ✦ Kotlin - Delete Recursively
- ✦ Kotlin - Get File Extension

Kotlin Interview Q/A

- ✦ Kotlin Interview Questions

Kotlin Android

- ✦ Kotlin Android Tutorial

Useful Resources

- ✦ How to Learn Programming