

How to check if a String contains Specified String in Kotlin?

Kotlin – Check if String contains Specified String

To check if a string contains specified string in Kotlin, use `String.contains()` method.

Given a string `str1`, and if we would like to check if the string `str2` is present in the string `str1`, call `contains()` method on string `str1` and pass the the string `str2` as argument to the method as shown below.

```
str1.contains(str2)
```

`contains()` method returns true if the string `str2` is present in the string `str1`, else it returns false.

Example 1 – String contains Specified String

In this example, we will take two strings in `str1` and `str2`, and check if the string `str1` contains the string `str2` using `String.contains()` method.

Kotlin Program

```
fun main(args: Array<String>) {  
    val str1 = "abcdef"  
    val str2 = "cd"  
    val result = str1.contains(str2)  
    println("The result of str1.contains(str2) is: " + result)  
    if (result) {  
        println("String str1 contains the string str2.")  
    } else {  
        println("String str1 does not contain the string str2.")  
    }  
}
```

Output

```
The result of str1.contains(str2) is: true  
String str1 contains the string str2.
```

Example 2 – Specified String is not in Given String

In this example, we will take two strings in `str1` and `str2` such that `str2` is not present in `str1`. `str1.contains(str2)` should return false.

Kotlin Program

```
fun main(args: Array<String>) {  
    val str1 = "abcdef"  
    val str2 = "pq"  
    val result = str1.contains(str2)  
    println("The result of str1.contains(str2) is: " + result)  
    if (result) {  
        println("String str1 contains the string str2.")  
    } else {  
        println("String str1 does not contain the string str2.")  
    }  
}
```

Output

```
The result of str1.contains(str2) is: false  
String str1 does not contain the string str2.
```

Conclusion

In this [Kotlin Tutorial](#), we learned how to check if given string contains a specified string value in it, using `String.contains()` method, with the help of Kotlin example programs.

Kotlin Java

◆ [Kotlin Tutorial](#)

Getting Started

◆ [Setup Kotlin\(Java\) Project](#)

◆ [Kotlin Example Program](#)

◆ [Convert Java to Kotlin](#)

◆ [Kotlin Main Function](#)

◆ [Kotlin Loops](#)

◆ [Kotlin For Loop](#)

◆ [Kotlin While, Do While Loops](#)

◆ Kotlin Repeat

◆ Kotlin Ranges

◆ Kotlin When

Object Oriented Concepts

Classes

◆ Kotlin - Class, Primary and Secondary Constructors

◆ Kotlin Sealed Class

◆ Kotlin Data Class

◆ Kotlin Enum

◆ Kotlin - Extension Functions

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

