

# What is Kotlin

Kotlin was introduced in 2011 by JetBrains, who were using Java earlier. There were two reasons for creating Kotlin, first and foremost is productivity. While the second one is the technology that demands evolution each day, which results in coding for the developers also needing some abstraction. The [Kotlin Programming Language](#) is an abstraction of Java with the core principles like Readability, conciseness, safety and ability to tool the language. The goal behind the development of Kotlin was to create an industry language that is used to develop a desktop, web server or mobile application so that Kotlin can be used in any application. Kotlin focuses on efficiency and dealing with the clutter of software development; adding up on that is now it is officially adopted as a supported language by Android. This has left the companies to look for a candidate with the knowledge of Kotlin. So, below are a few **Kotlin Interview questions** that will provide you with the overall structure of Kotlin, its components, and other basics. These questions might help you crack the interview with ease and get your hands on the job.

## Q1. [What is Kotlin?](#)

**Kotlin** is an open-source programming language that combines object-oriented programming features. The features like Range Expression, Extension Function, Companion Object, Smart casts, Data classes are considered to be a surplus of the Kotlin Language.

## Q2. [Which type of Programming does Kotlin support?](#)

Kotlin supports only two types of programming, and they are:

- Procedural programming
- Object-oriented programming

## Q3. [Why is Kotlin preferred over Java?](#)

Kotlin eases the coding process as it is simpler than [Java](#) and has many features required, that is not provided by Java yet like Extension functions, Null Safety, range expressions etc.

In Kotlin, we code approximately 40% less number of code lines as compared with Java.

## Q4. [Where does the Kotlin run and what is the entry point of Kotlin?](#)

The Kotlin program once compiled, can run on standard JVM like other programming codes. And, like many other programming languages `main()` function is the entry point of the Kotlin.

## Q5. [What are the different types of constructors in Kotlin?](#)

There are two types of constructors in Kotlin:

- **Primary constructor:** It is a section of the Class header and is declared after the class name.
- **Secondary constructor:** This constructor is declared inside the body.

Note: There can be more secondary constructors for a class.

## Q6. Can you execute Kotlin code without JVM?

JVM, which stands for Java Virtual Machine is a feature of Kotlin. This feature compiles a Kotlin code into a native code, which can be done without JVM too.

## Q7. Mention the structural expressions in Kotlin?

There are three Structural expressions in Kotlin.They are:

- **Return:** It returns from the nearest enclosing function or anonymous function by default.
- **Break:** This expression terminates the closest enclosing loop.
- **Continue:** This expression proceeds you to the next closest enclosing loop.

## Q8. Explain the data classes in Kotlin?

In programming, we use classes to hold data and these classes are called as data classes.

An object can be initialized in the data class and to access the individual parameters of these data classes, we use component functions.

## Q9. What are the modifiers that are available in Kotlin?

**Access modifier in Kotlin** provides the developer to customize the declarations as per the requirements. Kotlin provides four modifiers. They are:

**Private:** This makes the declaration visible only inside the file containing a declaration.

**Public:** It is by default, which means that the declarations will be visible everywhere.

**Internal:** This makes the declaration visible everywhere in the same modules.

**Protected:** This keeps the declaration protected and is not available for top-level declarations.

## Q10. Can you migrate the code from Java to Kotlin? If yes how do you do it?

Yes, we can migrate the code from Java to Kotlin.This can be done using JetBrains IDEA, which facilitates the conversion of Java code to Kotlin code.

## Q11. State the differences between Val and Var?

**Val:** Val, which is the short form of value, is a constant and it cannot be changed once assigned.

**Var:** Var, which is the short form of variable, is a storage location that accepts the reassignment of values that have the same data types.

## **Q12. Explain Kotlin's Null safety?**

In Kotlin, the Null safety is used to eliminate the risk of countering the NullPointerException in real time.

## **Q13. List the Basic data types of Kotlin?**

Data types of a constant or variable decide what type of variable it is and how much space is required to store it. The basic data types in Kotlin are:

- Numbers
- Characters
- Strings
- Arrays
- Booleans

## **Q14. What are the types of strings available in Kotlin? And, what do you mean by Kotlin String Interpolation?**

Strings are a collection of characters together. Kotlin features two types of strings, and they are:

- Raw string
- Escaped string

In Kotlin String, templates can be evaluated. This evaluation of string templates is called as the string template interpolation.

## **Q15. State the advantages and disadvantages of Kotlin?**

### **Advantages:**

Kotlin is simple and easy to learn as its syntax is similar to that of Java.

It is the functional language that is based on **JVM (Java Virtual Machine)**, which removes the boilerplate codes. Upon all this, Kotlin is considered as an expressive language that is easily readable and understandable and the performance is substantially good.

It can be used by any desktop, web server or mobile based applications.

### **Disadvantages:**

Kotlin does not provide the static modifier, which causes problems for conventional java developer.

In Kotlin, the function declaration can be done in many places in the application, which creates the trouble for the developer to understand which function is being called.

### **Q16. How to Declare a Variable in Kotlin?**

In Kotlin, you can declare a variable using var or val which followed by an optional datatype.

Variable declaration in Kotlin looks like:

```
val s: String = "Hi"  
var x = 5
```

### **Q17. What is the difference between declaration variable using val or var in Kotlin?**

In Kotlin a variable declared using val keyword is cannot be changed. It is similar to the final modifiers in Java whereas the variables declared using var keywords can be reassigned.

### **Q18. How to convert a String to an Int in Kotlin?**

toInt() method is used to convert a string value to integer or INT in Kotlin. Below is example uses

```
fun main(args: Array) {  
    val s: String = "Kotlin"  
    var x = 10  
    x = "8".toInt()  
}
```

### **Q19. Explain Functions In Kotlin?**

Kotlin functions are first-class functions that are easily stored in variables and data structures and can be pass as arguments and returned from other higher-order functions.

Sample function declaration and usage in Kotlin

```
fun double(x: Int): Int {  
    return 2 * x  
}  
val result = double(2)
```

### **Q20. Explain Higher-Order Functions in Kotlin?**

Higher-Order Functions: A higher-order function is a function that takes functions as parameters, or returns a function.

**Hot Tip:**

Kotlin is a flavor of [Java](#) that goes alongside Java. Many of the Kotlin applications can be built with the integrated parts of Java code.

Thus, most of the **Kotlin questions** can be answered if you have already acquired the knowledge of Java.

Tags: [Advanced kotlin interview questions](#) , [kotlin interview questions medium](#), [kotlin programming questions](#), [kotlin coroutines interview questions](#)