Kotlin MCQ Questions and Answers

1. What is Kotlin?
    a. A programming language for web development
    b. A programming language for mobile app development
    c. A programming language for desktop application development
    d. A markup language for web development

    Answer: B

2. What is the extension of a Kotlin source file?
    a. .java
    b. .kt
    c. .kotlin
    d. .jar

    Answer: B

3. Which of the following is not a basic data type in Kotlin?
    a. Int
    b. String
    c. Double
    d. Decimal

    Answer: D

4. In Kotlin, what is the difference between a var and a val keyword?
    a. var is used for mutable variables while val is used for immutable variables
    b. val is used for mutable variables while var is used for immutable variables
    c. var and val are interchangeable keywords
    d. None of the above

    Answer: A

5. What is the entry point of a Kotlin program?
    a. fun main()
    b. fun start()
    c. fun init()
    d. fun run()

    Answer: A

6. Which of the following is not a control structure in Kotlin?
    a. if expression
    b. for loop
    c. while loop
    d. goto statement

    Answer: D

7. Which of the following is a correct way to declare a nullable variable in Kotlin?
    a. var x: Int
    b. var x: Int?
    c. var x: Nullable<Int>
    d. var x: Optional<Int>

   Answer: B

8. Which of the following is a correct way to declare a function in Kotlin?
    a. fun myFunction(): String
    b. function myFunction(): String
    c. fun myFunction() String
    d. fun myFunction() -> String

   Answer: A

9. What is the range operator in Kotlin?
    a. ..
    b. :
    c. ,
    d. ;

   Answer: A

10. What is the purpose of the elvis operator in Kotlin?
    a. To check for nullability of a variable
    b. To assign a default value if a variable is null
    c. To perform string interpolation
    d. To perform bitwise operations

   Answer: B

11. Which of the following is a correct way to declare a class in Kotlin?
    a. class MyClass {
   }
   B. class MyClass()
   C. class MyClass {
   fun myFunction(): String
   }
   D. All of the above
   Answer: A

12. What is the default visibility modifier for a class in Kotlin?
    a. public
    b. private
    c. protected
    d. internal

   Answer: D

13. Which of the following is a correct way to create an instance of a class in Kotlin?
    a. val obj = MyClass
    b. val obj = MyClass()
    c. val obj = new MyClass()
    d. val obj: MyClass = null

    Answer: B

14. Which of the following is a correct way to create a secondary constructor in Kotlin?
    a. constructor()
    b. secondary constructor()
    c. constructor(args: List<String>)
    d. init(args: List<String>)

    Answer: C

15. What is the purpose of a companion object in Kotlin?
    a. To define static methods and properties
    b. To define constants
    c. To define abstract classes
    d. To define private methods

    Answer: A

16. Which of the following is a correct way to declare an interface in Kotlin?
    a. interface MyInterface {
    }
    B. class MyInterface {
    }
    C. abstract class MyInterface {
    }
    D. None of the above

    Answer: A

17. Which of the following is a correct way to implement an interface in Kotlin?
    a. class MyClass implements MyInterface {
    }
    B. class MyClass : MyInterface {
    }
    C. abstract class MyClass implements MyInterface {
    }
    D. None of the above

    Answer: B

18. What is the purpose of a data class in Kotlin?
    a. To define abstract data types
    b. To define classes for working with data
    c. To define interfaces for data storage
    d. To define classes for data encryption

    Answer: B

19. Which of the following is a correct way to define a data class in Kotlin?
    a. data MyClass {
    }
    B. data class MyClass {
    }
    C. class MyClass {
    data
    }
    D. None of the above
    Answer: B

20. What is the purpose of the init block in Kotlin?
    a. To initialize the properties of a class
    b. To define static methods and properties
    c. To define constants
    d. To define private methods

    Answer: A

21. Which of the following is a correct way to declare a lambda expression in Kotlin?
    a. { x -> x + 1 }
    b. (x) => x + 1
    c. fun(x) { x + 1 }
    d. lambda(x) -> x + 1

    Answer: A

22. What is the purpose of higher-order functions in Kotlin?
    a. To pass functions as arguments to other functions
    b. To define functions within other functions
    c. To define functions with a variable number of arguments
    d. To define functions with default arguments

    Answer: A

23. Which of the following is a correct way to declare a higher-order function in Kotlin?
    a. fun myFunction(): (Int) -> Int
    b. fun myFunction(x: Int, f: (Int) -> Int): Int
    c. fun myFunction(x: Int, f: Int): (Int) -> Int
    d. fun myFunction(x: Int, f: Int): Int -> Int

    Answer: B

24. What is the purpose of an extension function in Kotlin?
    a. To add new functionality to an existing class
    b. To define a function that can be called without an instance of a class
    c. To define a function that can only be called from within a class
    d. To define a function that can only be called by specific classes

    Answer: A

25. Which of the following is a correct way to define an extension function in Kotlin?
    a. fun MyClass.myFunction(): String {
    }
    B. fun MyClass(): String.myFunction {
    }
    C. fun myFunction(): String.MyClass {
    }
    D. fun MyClass(): String.myFunction() {
    }

    Answer: A

26. What is the purpose of a sealed class in Kotlin?
    a. To define a class that cannot be extended
    b. To define a class that can only be instantiated once
    c. To define a class hierarchy with a limited number of subclasses
    d. To define a class that can only be used within a specific package

    Answer: C

27. Which of the following is a correct way to declare a sealed class in Kotlin?
    a. sealed MyClass {
    }
    B. sealed class MyClass {
    }
    C. abstract sealed class MyClass {
    }
    D. None of the above

    Answer: B

28. What is the purpose of a typealias in Kotlin?
    a. To define an alias for a type
    b. To define an alias for a function
    c. To define an alias for a class
    d. To define an alias for a variable

    Answer: A

29. Which of the following is a correct way to declare a nullable variable in Kotlin?
    a. var myVariable: Int
    b. var myVariable: Int?
    c. val myVariable: Int?
    d. None of the above

    Answer: C

30. What is the purpose of the Elvis operator (?:) in Kotlin?
    a. To assign a default value to a nullable variable
    b. To concatenate strings
    c. To define a lambda expression
    d. To define a higher-order function

    Answer: A

    Which of the following is a correct way to use the Elvis operator in Kotlin?
    e. var myVariable = nullableVariable ?: "default value"
    f. var myVariable = nullableVariable ? "default value" :
    g. var myVariable = nullableVariable : "default value" ?
    h. None of the above

    Answer: A

31. What is the purpose of the safe call operator (?.) in Kotlin?
    a. To call a method on a nullable object if it is not null
    b. To assign a value to a variable
    c. To define a lambda expression
    d. To define a higher-order function

    Answer: A

32. Which of the following is a correct way to use the safe call operator in Kotlin?
    a. nullableObject?.myMethod()
    b. nullableObject:myMethod()?
    c. nullableObject.myMethod()?
    d. None of the above

    Answer: A

33. What is the purpose of the non-null assertion operator (!!) in Kotlin?
    a. To cast a nullable variable to a non-null type
    b. To assign a value to a variable
    c. To define a lambda expression
    d. To define a higher-order function

    Answer: A

34. Which of the following is a correct way to use the non-null assertion operator in Kotlin?
    a. var myVariable = nullableVariable!!
    b. var myVariable = !!nullableVariable
    c. var myVariable = nullableVariable!:
    d. None of the above

    Answer: A

35. What is the purpose of the when expression in Kotlin?
    a. To replace the switch statement in Java
    b. To define a loop
    c. To define a lambda expression
    d. To define a higher-order function

    Answer: A

36. Which of the following is a correct way to use the when expression in Kotlin?
    a. when (myVariable) {
    0 -> "zero"
    1 -> "one"
    else -> "unknown"
    }
    B. myVariable.when {
    0 -> "zero"
    1 -> "one"
    else -> "unknown"
    }
    C. when myVariable {
    0 -> "zero"
    1 -> "one"
    else -> "unknown"
    }
    D. None of the above
    Answer: A

37. What is the purpose of the range operator (..) in Kotlin?
    a. To create a range of numbers
    b. To define a loop
    c. To define a lambda expression
    d. To define a higher-order function

    Answer: A

38. Which of the following is a correct way to use the range operator in Kotlin?
    a. for (i in 1..10) { }
    b. for (i in 1 to 10) { }
    c. for (i in range(1, 10)) { }
    d. None of the above

    Answer: A

39. What is the output of the following Kotlin code?

```
fun main() {
val myList = listOf(1, 2, 3)
myList.forEach { print("$it ") }
}
```

    a. 1 2 3
    b. [1, 2, 3]
    c. 1,2,3
    d. None of the above

    Answer: A

40. What is the output of the following Kotlin code?

```
fun main() {
val myList = mutableListOf(1, 2, 3)
myList.add(4)
myList.forEach { print("$it ") }
}
```

    a. 1 2 3
    b. [1, 2, 3, 4]
    c. 1,2,3,4
    d. None of the above

    Answer: B

41. What is the output of the following Kotlin code?

```
fun main() {
val myMap = mapOf("a" to 1, "b" to 2, "c" to 3)
println(myMap["a"])
}
```

a. 1
b. "a"
c. "b"
d. None of the above

Answer: A

42. What is the output of the following Kotlin code?

```
fun main() {
val myMap = mutableMapOf("a" to 1, "b" to 2, "c" to 3)
myMap["d"] = 4
println(myMap)
}
```

a. {a=1, b=2, c=3}
b. {a=1, b=2, c=3, d=4}
c. {d=4}
d. None of the above

43. Answer: B

What is the output of the following Kotlin code?

```
fun main() {
val mySet = setOf(1, 2, 3, 3)
println(mySet)
}
```

a. {1, 2, 3}
b. {1, 2, 3, 3}
c. [1, 2, 3]
d. None of the above

Answer: A

44. What is the output of the following Kotlin code?

```
fun main() {
val mySet = mutableSetOf(1, 2, 3, 3)
mySet.add(4)
println(mySet)
}
```

a. {1, 2, 3}
b. {1, 2, 3, 4}
c. [1, 2, 3, 4]
d. None of the above

Answer: B

45. Which of the following is a correct way to declare a function with parameters in Kotlin?
a. fun myFunction() { }
b. fun myFunction(param1: Int, param2: String) { }
c. fun myFunction(param1, param2) { }
d. None of the above

Answer: B

46. What is the output of the following Kotlin code?

```
fun main() {
val myFunction = { param: Int -> param * 2 }
println(myFunction(3))
}
```

a. 3
b. 6
c. 9
d. None of the above

Answer: B

47. What is the output of the following Kotlin code?

```
fun main() {
val myArray = arrayOf(1, 2, 3)
println(myArray[1])
}
```

a. 1
b. 2
c. 3
d. None of the above

Answer: B

48. What is the output of the following Kotlin code?

```
fun main() {
val myString = "Hello, world!"
println(myString.substring(7))
}
```

a. "Hello,"
b. "world!"
c. "Hello, world!"
d. "world"

Answer: "world!"

49. What is the output of the following Kotlin code?

```
fun main() {
val myList = mutableListOf("apple", "banana", "orange")
myList.remove("banana")
myList.forEach { print("$it ") }
}
```

A. apple banana orange
B. apple orange
C. [apple, orange]
D. None of the above

Answer: B

50. What is the output of the following Kotlin code?

```
fun main() {
val myMap = mapOf("a" to 1, "b" to 2, "c" to 3)
val myMap2 = myMap.filter { it.value % 2 == 0 }
println(myMap2)
}
```

A. {b=2}
B. {a=1, b=2, c=3}
C. {a=1, c=3}
D. None of the above

Answer: A