Kotlin Coroutines on Android

Software Development · Practice Test · 10 Questions

1. What is the primary purpose of using coroutines on Android?

A) To increase memory usage
B) To simplify asynchronous code
C) To replace the main thread
D) To disable network requests

2. Which Kotlin version introduced coroutines?

A) 1.0
B) 1.1
C) 1.3
D) 1.5

3. What is the main benefit of the 'lightweight' feature of coroutines?

A) They run faster than threads
B) They don't block the thread while suspended
C) They consume more memory
D) They require no CPU

4. What does 'structured concurrency' help prevent?

A) Memory leaks
B) UI freezes
C) Syntax errors
D) Network latency

5. Which component provides the 'viewModelScope' extension?

A) Activity
B) Fragment
C) ViewModel
D) Repository

6. What happens to coroutines launched in 'viewModelScope' when a ViewModel is destroyed?

A) They continue running
B) They are automatically canceled
C) They crash the app
D) They move to the background

7. What is the function of the 'suspend' keyword in Kotlin?

A) To run code on the main thread
B) To enforce that a function is called from a coroutine
C) To block the current thread
D) To delete a coroutine

8. Which dispatcher is recommended for I/O operations like network requests?

A) Dispatchers.Main
B) Dispatchers.Default
C) Dispatchers.IO
D) Dispatchers.Unconfined

9. What does the 'withContext' function achieve?

A) It changes the thread of a coroutine
B) It creates a new UI component
C) It stops the app from responding
D) It initializes the ViewModel

10. Why should you avoid network requests on the main thread?

A) It makes the code shorter
B) It leads to ANR (Application Not Responding) dialogs
C) It saves battery life
D) It is required by the OS