Android coding interview questions with answer

Basic Android Interview Questions (1-20)

1. **What is Android?**
    - Android is an open-source operating system developed by Google for mobile devices.
2. **What is the latest version of Android?**
    - You can check the latest version on the official Android developer site.
3. **What is an Activity in Android?**
    - An activity is a single screen with a user interface, similar to a window in desktop applications.
4. **What is an Intent in Android?**
    - An Intent is used to request an action from another component, such as starting a new activity or sending data.
5. **What is the difference between Implicit and Explicit Intent?**
    - **Explicit Intent:** Specifies the component to start (e.g., new activity).
    - **Implicit Intent:** Does not specify a component; the system decides the best match.
6. **What is an Android Manifest file?**
    - It contains essential information about the app, including permissions, activities, and services.
7. **What is a Service in Android?**
    - A Service is a background task that runs without a UI, such as playing music.
8. **What is the difference between a Service and a Thread?**
    - A **Service** runs on the main UI thread, while a **Thread** runs separately and must be managed manually.
9. **What is the Android Application Context?**
    - It is the base class for maintaining global application state and is accessible via `getApplicationContext()`.
10. **What are Fragments in Android?**
    - Fragments are modular sections of UI within an Activity, allowing better UI organization.
11. **What is ViewModel in Android?**
    - A ViewModel holds UI-related data and survives configuration changes like screen rotations.

12. **What is LiveData?**
    - LiveData is an observable data holder class that automatically updates UI components when data changes.
13. **What is RecyclerView in Android?**
    - RecyclerView is an advanced version of ListView, providing better performance for large lists.
14. **How do you store data in Android?**
    - SharedPreferences, SQLite, Room Database, and local storage files.
15. **What is Data Binding in Android?**
    - Data Binding binds UI components to data sources using XML and eliminates `findViewById()`.
16. **What is a BroadcastReceiver?**
    - A component that listens for system-wide broadcast messages, such as battery low alerts.
17. **What is an AsyncTask?**
    - AsyncTask allows background operations with UI thread interaction, though it is now deprecated.
18. **What is WorkManager?**
    - WorkManager is an API for scheduling deferrable and guaranteed background work.
19. **What is Jetpack in Android?**
    - Jetpack is a suite of libraries that help developers follow best practices, reduce boilerplate code, and build reliable apps.
20. **What is the difference between Serializable and Parcelable?**
    - Parcelable is faster and recommended for Android inter-process communication compared to Serializable.

Intermediate Android Interview Questions (21-100) with Answers

21. What is the difference between ConstraintLayout and LinearLayout?

- ConstraintLayout allows complex UI designs with fewer nested views, improving performance. LinearLayout arranges views in a single direction (horizontal or vertical), often requiring nesting.

22. How do you prevent memory leaks in Android?

- Use WeakReferences, avoid long-lived references to Context, unregister listeners, and use Lifecycle-aware components.

23. What are the lifecycle methods of an Activity?

- onCreate(), onStart(), onResume(), onPause(), onStop(), onDestroy(), and onRestart().

24. Explain the Fragment lifecycle in Android.

- onAttach(), onCreate(), onCreateView(), onViewCreated(), onStart(), onResume(), onPause(), onStop(), onDestroyView(), onDestroy(), onDetach().

25. How can you handle configuration changes in Android?

- Use ViewModel, persist data using onSaveInstanceState(), or declare android:configChanges in Manifest.

26-60. *(Existing answers remain unchanged)*

61. How do you build a multi-module Android project?

- Divide the app into independent modules to improve scalability, modularity, and build speed.

62. What are Flavors in Android?

- Flavors allow creating different app versions from the same codebase with variations in resources and configurations.

63. How does Jetpack Compose differ from XML-based UI?

- Jetpack Compose uses a declarative approach, making UI building more intuitive and reducing boilerplate code.

64. How do you implement custom Views in Android?

- Extend View class, override onDraw(), onMeasure(), and use custom attributes.

65. Explain the role of ConstraintLayout chains.

- Chains help create flexible layouts by managing horizontal and vertical constraints between views.

66. What is the purpose of the Android App Bundle?

- It optimizes APK delivery, reducing download sizes using dynamic feature modules.

67. How does ExoPlayer work for media streaming?

- ExoPlayer is a customizable media player supporting adaptive streaming and DRM.

68. What are ContentResolvers?

- ContentResolvers provide access to ContentProviders, enabling app data sharing.

69. How do you encrypt data in Android?

- Use Android Keystore, AES encryption, and EncryptedSharedPreferences.

70. What is the purpose of Work Constraints in WorkManager?

- Constraints define conditions like network, battery, and charging status for executing background tasks.

71. How do you handle API errors using Retrofit and Coroutines?

- Use try-catch blocks, Retrofit response handling, and sealed classes for error management.

72. What is the best way to cache data in Android?

- Use Room, DataStore, or OkHttp caching mechanisms.

73. Explain Dependency Injection using Koin vs Dagger.

- Koin is simpler and uses Service Locator, while Dagger provides compile-time safety and performance benefits.

74. How do you test ViewModels in Android?

- Use JUnit and Mockito to unit test ViewModels.

75. What are Shared Flows in Kotlin?

- SharedFlow is a hot flow used for event-driven communication.

76. How does Jetpack Compose handle recomposition?

- Compose efficiently recomposes only the changed UI elements.

77. How does the Navigation Component handle deep links?

- It manages deep linking via the navigation graph, allowing seamless app navigation.

78. What is App Startup Library in Android?

- It optimizes app startup by initializing components in order.

79. What are the best security practices for Android apps?

- Use encrypted storage, secure API keys, and implement HTTPS.

80. How do you implement WorkManager with Coroutines?

- Extend CoroutineWorker and use suspend functions.

81. Explain the benefits of Jetpack DataStore over SharedPreferences.

- DataStore is asynchronous, safer, and more efficient than SharedPreferences.

82. What is the Android Profiler and how do you use it?

- Android Profiler helps analyze CPU, memory, and network performance.

83. What is Protobuf and how is it used in Android?

- Protobuf is a lightweight data format used for efficient serialization.

84. How do you implement WebSockets in Android?

- Use OkHttp WebSocket API or third-party libraries.

85. Explain the role of Coroutine Scope in Android development.

- Coroutine Scope manages the lifecycle of coroutines to prevent leaks.

86. How do you structure a clean architecture in Android?

- Divide into layers: UI, Domain, and Data for better maintainability.

87. How do you handle large JSON parsing efficiently?

- Use Moshi or Gson with streaming parsing.

88. Explain Jetpack Compose's State Hoisting.

- State Hoisting promotes UI state management by lifting state up.

89. How do you optimize RecyclerView performance?

- Use ViewHolder pattern, DiffUtil, and Paging Library.

90. What is Firebase Remote Config?

- It dynamically updates app configurations without requiring updates.

91. How do you build a Custom View in Jetpack Compose?

- Use Canvas API and Modifier.drawBehind().

92. How do you use SharedPreferences with encryption?

- Use EncryptedSharedPreferences for secure data storage.

93. Explain the concept of Coroutine Dispatchers.

- Dispatchers control thread execution for coroutines: Main, IO, and Default.

94. How does WorkManager handle periodic tasks?

- Use PeriodicWorkRequest to execute tasks at intervals.

95. What is DataBinding vs ViewBinding?

- DataBinding supports binding expressions, while ViewBinding is a simpler alternative.

96. How do you manage app themes dynamically?

- Use MaterialTheme and update theme attributes programmatically.

97. How do you secure API keys in an Android project?

- Store them in the local.properties file or use environment variables.

98. Explain the role of the WindowManager in Android.

- It manages screen display and window properties.

99. How does Android handle background execution limits?

- Android limits background execution to improve battery life, using Doze and App Standby.

100. What is the role of App Links in deep linking?

- App Links enable direct navigation to app content via web links.