

## Study Guide

You will be required to complete programming tasks (6 short tasks) in addition to answering 25 multiple choice questions.

Read everything from <https://developer.android.com/guide/index.html>

(Or at least the main bullet point concepts listed below)

- Git
  - Basic git commands
  - git commit
  - git add
  - git checkout
  - git stash
  - git branch
  - git push/pull
- Android Components
  - Activity
  - Content Provider
  - Broadcast Receiver
  - Service
- AndroidManifest
  - Permissions
  - What gets declared?
- Activity Lifecycle
  - onCreate, onStart, onResume, onPause, onStop, onDestroy
  - When do these get called?
  - If the home button is pressed, which methods get called?
  - App can be killed at any moment by the system (due to low memory)
- Intents
  - Implicit vs Explicit
  - Used to start an Activity, start a Service, broadcast an event
  - Actions (ACTION\_PICK, ACTION\_VIEW, ...)
  - IntentFilters—Used to specify the types of Intents to respond to
  - Extras - Key/Value pairs sent along with the intent.
- Jetpack Compose
  - Text, TextField, Button, Column, Row
  - Events (Button click, long-press, Text input...)
  - MutableState, remember
  - ViewModels
  - LaunchedEffect
  - Modifiers, font size, font color, padding, width, height, alignment ...
- Database (Room)
  - SQLite - How is it different than MySQL or PostgreSQL?

- Where is your database file located?
- How is your database created?
- Room - How do you write queries?
- Services - Background (No UI) work
  - Service - By default, runs on the UI thread. Can be bound or started
  - Worker - Managed by the WorkManager.
  - JobScheduler - A system service that schedules services from *all* apps on the device. Tries to optimize for efficient use of battery.
- Content Providers
  - Purpose? - Share data with other components/apps
  - Provides a secure, structured way to share data
  - Requires permissions
  - Query using a ContentResolver using the ContentProvider's URI.
- Coroutines
  - All networking must be done on a background thread
  - Must not block the UI thread for more than 5 secs.
  - Coroutines
    - How do you launch a new coroutine?
    - How do you execute a coroutine on the main thread? IO thread?
    - How do you declare functions that are callable within a coroutine?
- Navigation Framework
  - How do you navigate from one Composable to another?
  - How do you declare your navigation routes?
- Notifications
  - How do you create one? (Use the Notification.Builder object. Call the "notify" method on the NotificationManager with the object and an ID (integer)).
  - How do you update or remove a notification? (Update by creating a *new* Notification object and calling "notify" on the NotificationManager using the *same* id that was used to originally create the first notification.)
- Code:
  - Write code to display a Column of Text
  - Write code to display a LazyColumn of items
  - Write code to handle a button click
  - Write code that uses a ViewModel
  - Write code that uses Navigation
  - Write code that uses a Room database
  - Write code that uses coroutines
- Kotlin
  - Nullable types. String? Vs String, Int vs Int?
  - Lambda. myList.forEach { item → doSomething(item) }
  - Functions. Top level (functions that exist outside a class)
  - Immutable vs mutable. var is mutable. val is immutable.
  - Type inference. val x: String = "Hello" vs val x = "World"