**1. Основы языка программирования (Java/Kotlin)**

* базовые конструкции языка: переменные, операторы, циклы, условия;
* коллекции (List, Set, Map);
* ООП: инкапсуляция, наследование, полиморфизм, абстракция;
* исключения и их обработка (try-catch, finally);
* работа с потоками (Threads, Coroutines);
* работа с лямбда-выражениями и функциональными возможностями (Kotlin).

**2. Android SDK и архитектура Android**

* жизненный цикл Activity и Fragment;
* Intent, BroadcastReceiver, Service, ContentProvider;
* View, ViewGroup, RecyclerView, ConstraintLayout;
* работа с ресурсами (res, strings.xml, styles.xml);
* хранение данных: SharedPreferences, Room, DataStore;
* взаимодействие с API: Retrofit, OkHttp;
* работа с асинхронными задачами: AsyncTask, Handler, LiveData, Coroutines.

**3. Основы многопоточности**

* Thread, Runnable, ExecutorService;
* Handler, Looper, MessageQueue;
* RxJava или Kotlin Coroutines.

**4. Архитектура Android-приложений**

* MVVM, MVP, Clean Architecture;
* ViewModel, LiveData;
* Dependency Injection (Dagger, Hilt, Koin).

**5. Работа с API и базами данных**

* REST API (Retrofit, OkHttp);
* JSON и Gson / Moshi;
* локальные базы данных: SQLite, Room.

**6. Jetpack и современные инструменты разработки**

* ViewModel, LiveData;
* Navigation Component;
* WorkManager;
* DataBinding / ViewBinding;
* Paging, RecyclerView.

**7. Git и системы контроля версий**

* основные команды Git (clone, pull, push, commit, merge, rebase);
* работа с ветками (branch, checkout, merge).

**8. CI/CD и DevOps-базовые знания**

* работа с Gradle;
* настройка ProGuard;
* автоматическая сборка (GitHub Actions, Jenkins).

**9. Тестирование (по желанию, но плюс)**

* юнит-тесты (JUnit, Mockito);
* UI-тестирование (Espresso);
* интеграционное тестирование.

**10. Публикация приложений**

* подготовка приложения к релизу;
* загрузка в Google Play (подготовка .apk или .aab);
* работа с Firebase Analytics и Crashlytics.

**Что важно для Junior?**

* знание базовых концепций Android SDK;
* уверенное владение Kotlin или Java;
* понимание основных паттернов разработки;
* работа с REST API и базами данных.

**Что важно для Middle?**

* уверенное использование MVVM и Clean Architecture;
* знание многопоточности (Coroutines, RxJava);
* опыт работы с Jetpack библиотеками;
* написание тестов (юнит, интеграционных, UI);
* работа с CI/CD и Gradle.