**Minimal requirements for interns: Android**

1. **BASIC MOBILE PRINCIPLES:**
   1. Version control - [Git](https://git-scm.com/book/ru/v2)
   2. [UX/UI](https://habr.com/ru/post/321312/)
   3. [Difference between UX/UI](https://habr.com/ru/company/pixli/blog/324794/)
2. **JAVA CORE:**
   1. [Basics](https://www.journaldev.com/7153/core-java-tutorial#core-java-tutorial-8211-getting-started)
   2. [Object Oriented Concepts](https://www.journaldev.com/7153/core-java-tutorial#core-java-tutorial-8211-object-oriented-concepts)
   3. [Data Types and Operators](https://www.journaldev.com/7153/core-java-tutorial#core-java-tutorial-8211-data-types-and-operators)
   4. [Interface, Abstract Class](https://www.journaldev.com/7153/core-java-tutorial#core-java-tutorial-8211-interface-abstract-class)
   5. [String Manipulation](https://www.journaldev.com/7153/core-java-tutorial#core-java-tutorial-8211-string-manipulation)
   6. [Arrays](https://www.journaldev.com/7153/core-java-tutorial#core-java-tutorial-8211-arrays)
   7. [Annotation and Enum](https://www.journaldev.com/7153/core-java-tutorial#core-java-tutorial-8211-annotation-and-enum)
   8. [Collection & Generics](https://www.journaldev.com/7153/core-java-tutorial#core-java-tutorial-8211-collection-038-generics)
   9. [Exception Handling](https://www.journaldev.com/7153/core-java-tutorial#core-java-tutorial-8211-exception-handling)
3. **KOTLIN CORE:**
   1. [Basics](https://kotlinlang.org/docs/reference/basic-types.html)
   2. [Classes and Object](https://kotlinlang.org/docs/reference/classes.html)
   3. [Functions and Lambdas](https://kotlinlang.org/docs/reference/functions.html)
   4. [Collections](https://kotlinlang.org/docs/reference/collections-overview.html)
   5. [Equality](https://kotlinlang.org/docs/reference/equality.html#equality)
   6. [Null Safety](https://kotlinlang.org/docs/reference/null-safety.html)
   7. [Exceptions](https://kotlinlang.org/docs/reference/exceptions.html)
4. **ANDROID**:
   1. [Application Fundamentals](https://developer.android.com/guide/components/fundamentals)
   2. [Activity](https://developer.android.com/guide/components/activities/intro-activities)
      1. [Lifecycle](https://developer.android.com/guide/components/activities/activity-lifecycle)
      2. [Parcelables and Bundles](https://developer.android.com/guide/components/activities/parcelables-and-bundles)
   3. [Fragment](https://developer.android.com/guide/components/fragments)
   4. Navigation
      1. [Tasks and back stack](https://developer.android.com/guide/components/activities/tasks-and-back-stack)
      2. [Fragment](https://developer.android.com/training/basics/fragments/fragment-ui)
   5. User Interface
      1. [Layouts](https://developer.android.com/guide/topics/ui/declaring-layout)
      2. [ConstraintLayout](https://developer.android.com/training/constraint-layout) + [ConstraintLayout Params](https://developer.android.com/reference/android/support/constraint/ConstraintLayout)
      3. [RecyclerView](https://developer.android.com/guide/topics/ui/layout/recyclerview)
   6. [App permissions](https://developer.android.com/guide/topics/permissions/overview)
      1. [Request app permissions](https://developer.android.com/training/permissions/requesting)
   7. [Retrieve user location](https://developer.android.com/training/location/retrieve-current)
   8. [MVP](https://medium.com/cr8resume/make-you-hand-dirty-with-mvp-model-view-presenter-eab5b5c16e42)
   9. [Material design & components](https://developer.android.com/guide/topics/ui/look-and-feel)
      1. [Icons](https://materialdesignicons.com/)
      2. [Bottom navigation](https://material.io/develop/android/components/bottom-navigation-view/)

**LIBRARIES**

1. [RxJava - reactive programming & async operations](https://proandroiddev.com/exploring-rxjava-in-android-e52ed7ef32e2)
   1. [Different types of Observables](https://proandroiddev.com/exploring-rxjava-in-android-different-types-of-observables-f23b3c78aeb6)
   2. [Schedulers](https://proandroiddev.com/understanding-rxjava-subscribeon-and-observeon-744b0c6a41ea)
2. [Retrofit](https://square.github.io/retrofit/) - [HTTP client](https://habr.com/ru/post/429058/)
   1. [Add Custom Request Header](https://futurestud.io/tutorials/retrofit-add-custom-request-header)
3. [Glide - management and image loading](https://github.com/bumptech/glide)