## My NanoDegree – Android

## Intro to Java – Codecademy

## https://www.codecademy.com/en/courses/learn-java

## [Prerequisites and Requirements](https://www.udacity.com/course/android-developer-nanodegree--nd801)

This will be a challenging and rewarding journey that will take a novice programmer 9 months or longer to complete, spending an estimated 10 hours per week on the coursework. Do you have the dedication and mindset to sustain you through it? If you're not sure, please check out the courses and projects, listed under**Nanodegree Structure**, before making this commitment.

### Programming Experience

**This is not a "Zero to Hero" program.** Entering students are expected to have prior experience building applications (web or mobile) in Java or another object-oriented programming language.

**You should have at least 1-2 years of experience** in Java or another object-oriented programming language prior to enrolling.

**If you have no prior programming experience**, we recommend you take our [Android Development for Beginners](https://www.udacity.com/course/android-development-for-beginners--ud837" \t "_blank) course, also built with Google. You should also learn foundational Java programming concepts before enrolling in this Nanodegree. Our [Intro to Java](https://www.udacity.com/course/intro-to-java-programming--cs046" \t "_blank) course may be a good place to start.

### Technical Skills and Requirements

You will submit your projects on GitHub, and must have a GitHub handle and know how to share code prior to enrolling. Take our course on [How to Use Git and GitHub](https://www.udacity.com/course/how-to-use-git-and-github--ud775" \t "_blank) if you need to learn how to do this.

You will build your apps in Android Studio, and will need to install it on your machine to complete your Nanodegree projects. If you don't have Android Studio installed, see our mini-course on [How to Install Android Studio](https://www.udacity.com/course/how-to-install-android-studio--ud808" \t "_blank) for instructions.

### Dedication and Mindset

In addition to 1-2 years of prior programming experience and intermediate technical skills, students are expected to demonstrate the following characteristics:

 **Resourcefulness**: Ability to search for and find solutions in documentation, backed by the belief that all problems in code are discoverable;

 **Grit**: Ability to work through challenges and persevere when code breaks and tests fail.

 **Growth Mindset**: Belief that intelligence is NOT a fixed entity, and can be boosted by hard work in the process of learning and practice.

See the [Technology Requirements](https://www.udacity.com/tech-requirements" \t "_blank) for using Udacity.

## Nanodegree Program Structure

### Nanodegree Courses

* [Developing Android Apps: Android Fundamentals](https://www.udacity.com/course/developing-android-apps--ud853)
* [Advanced Android App Development: Productionize and Publish Your App](https://www.udacity.com/course/advanced-android-app-development--ud855" \t "_blank)
* Google Play Services
  + [Google Play Services: Location and Context](http://www.udacity.com/course/ud876-1" \t "_blank)
  + [Google Play Services: Analytics](http://www.udacity.com/course/ud876-2)
  + [Google Play Services: AdMob](http://www.udacity.com/course/ud876-3)
  + [Google Play Services: Maps](http://www.udacity.com/course/ud876-4)
  + [Google Play Services: Identity](http://www.udacity.com/course/ud876-5)
* [Gradle for Android and Java: Build Better Apps Through Automation](https://www.udacity.com/course/gradle-for-android-and-java--ud867" \t "_blank)
* [Android Design for Developers: Make Your Apps Material](https://www.udacity.com/course/android-design-for-developers--ud862" \t "_blank)
* [Android Ubiquitous Computing: Extend Apps to Wearables, TV and Auto](https://www.udacity.com/course/android-ubiquitous-computing--ud875" \t "_blank)

## Nanodegree Project Portfolio

### Personal App Portfolio

Design and build an app that shows off the apps that you'll create as part of this Nanodegree.

### [Popular Movies, Stage 1](https://www.udacity.com/course/viewer" \l "!/c-nd801/l-4256658707/m-4242848655)

Build an app to help users discover popular and recent movies. You will build a clean UI, sync to a server, and present information to the user.

### [Popular Movies, Stage 2](https://www.udacity.com/course/viewer" \l "!/c-nd801/l-4324689102/m-4304038833)

Add onto your project from Stage 1 with a detail view for each movie, allowing users to 'favorite' movies, and adding a tablet layout.

### [Super Duo: Productionize Two Apps](https://www.udacity.com/course/viewer" \l "!/c-ud855/l-3993648704/m-4291808735" \t "_blank)

Productionize two apps, taking them from a functional state to a production-ready state. To do this, you will find and handle error cases, add accessibility features, allow for localization, add widgets, and add a library.

### [Build It Bigger](https://www.udacity.com/course/viewer" \l "!/c-ud867/l-4328489198/m-4382778568" \t "_blank)

Use Gradle to build a joke-telling app, factoring functionality into libraries and flavors to keep the build simple. You'll also configure a Google Cloud Endpoints development server to supply the jokes.

### [Make Your App Material](https://www.udacity.com/course/viewer#!/c-ud855/l-4326960273/m-4329086234)

Implement Android design concepts and transform a functional newsreader app using material design techniques.

### [Go Ubiquitous](https://www.udacity.com/course/viewer#!/c-ud875/l-4643648560/m-4634049761)

Let users access the weather at a glance by building a Sunshine watch face for Android Wear.

### [Capstone, Stage 1: Design, Scope and Plan Your App](https://www.udacity.com/course/viewer#!/c-ud876-1/l-4330969190/m-4333036077)

Create a detailed plan, including UI mocks and technical specifications, for building your own app.

### [Capstone, Stage 2: Build and Polish Your App for Production](https://www.udacity.com/course/viewer#!/c-ud876-1/l-4333029280/m-4376351203)

This is your chance to take the skills that you've learned across your Nanodegree journey and apply it to an app idea of your own.