Google Developers
Training

Google Android Development Faculty Development Program Schedule

Contents

Unit 1 – Get started

- Lesson 1. Build your first app
  - 1.0: Introduction to Android
  - 1.1: Your first Android app
  - 1.2: Layouts and resources for the UI
  - 1.3: Text and scrolling views
  - 1.4: Resources to help you learn
- Lesson 2. Activities and intents
  - 2.1: Activities and intents
  - 2.2: Activity lifecycle and state
  - 2.3: Implicit intents
- Lesson 3. Testing, debugging, and using support libraries
  - 3.1: The Android Studio debugger
  - 3.2: C App testing
  - 3.3: The Android Support Library

Unit 2 – User experience

- Lesson 4. User interaction
  - 4.1: Buttons and clickable images
  - 4.2: Input controls
  - 4.3: Menus and pickers
  - 4.4: User navigation
  - 4.5: RecyclerView
- Lesson 5. Delightful user experience
  - 5.1: Drawables, styles, and themes
  - 5.2: Material Design
  - 5.3: Resources for adaptive layouts
- Lesson 6.Testing your UI

6.1: UI testing

Unit 3 – Working in the background

Lesson 7. Background tasks
- 7.1: AsyncTask and AsyncTaskLoader
- 7.2: Internet connection
- 7.3: Broadcast receivers
- 7.4.0: Services

Lesson 8. Alarms and schedulers
- 8.1: Notifications
- 8.2: Alarms
- 8.3: Efficient data transfer

Unit 4 – Saving user data

Lesson 9. Preferences and settings
- 9.0: Data storage
- 9.1: Shared preferences
- 9.2: App settings

Lesson 10. Storing data with Room
- 10.0: SQLite primer
- 10.1: Room, LiveData, and ViewModel

Unit 1 – Get started

This unit covers installing Android Studio, understanding project structure, building your first app, creating activities, testing your apps, and using the Android Support Library.

First, you deploy a simple Hello World app. You go on to create an app with a single activity, and then you create a multi-screen app that passes data between activities. You also learn how to use the Android Support Library to provide backward-compatibility with earlier versions of the Android system for your app.

Lesson 1. Build your first app **This lesson covers:** - Installing Android Studio. - Creating an Android app project. - Deploying the app to an emulator and a device. - Building a layout with UI elements including a scrolling list. - Learning where and how to get help with building applications.	Day - 1
1.0: Introduction to Android *Understand the motivation for developing Android apps*	Day – 1
1.1: Your first Android app *Understand the development process for building Android apps.*	Day – 1
1.2: Layouts and resources for the UI *Learn how to add interactive UI elements to your app and understand the range of UI elements available.*	Day – 1
1.3: Text and scrolling views *Understand the performance implications of different ways to make content in an app scrollable.*	Day - 1

1.4: Resources to help you learn *Understand the ways that developers can learn for themselves.*	Day - 1

Lesson 2. Activities and intents **This lesson covers:** - Activities, which are the major building blocks of your app's user interface. - Intents: learn about both implicit and explicit intents which are used to communicate between activities. - Callback events that perform tasks in each stage of the activity lifecycle.	Day – 1
2.1: Activities and intents *Understand how to send an explicit intent to start a specific activity.*	Day – 1
2.2: Activity lifecycle and state *Learn about activity lifecycles.*	Day – 2
2.3: Implicit intents *Understand how to send an implicit intent to sends a general request for an activity that can handle the request.*	Day – 2

Lesson 3. Testing, debugging, and using support libraries	Day - 2
3.1: The Android Studio debugger *Learn about the Android Studio debugger.*	Day - 2
3.2: C App testing *Overview of Android testing, and you learn about creating and running local unit tests in Android Studio with JUnit.*	Day - 2

3.3: The Android Support Library *Learn how to use the Android Support Library to get backward-compatible versions of new Android features*	Day - 2

Unit 2 – User experience

Create adaptive, responsive user interfaces that work across a wide range of devices. Create engaging, responsive interfaces that use material design principles. Test your app's user interface.

Lesson 4. User interaction **This lesson covers:** - Understanding and implementing different navigation paths through your application. - Customizing user input methods and controls. - Building responsive navigation. - Using buttons for navigation.	Day - 2
4.1: Buttons and clickable images *Learn how to create buttons and clickable images for triggering actions initiated by the user.*	Day – 2
4.2: Input controls *Learn about input controls such as switches, spinners and more..*	Day – 2
4.3: Menus and pickers *Learn about the types of menus, dialogs and pickers.*	Day - 2
4.4: User navigation *Learn about the different ways to enable users to navigate through your app.*	Day – 2
4.5: RecyclerView *Learn about RecyclerView, which displays items in a list in a way that uses memory efficiently.*	Day – 2

Lesson 5. Delightful user experience **This lesson covers:** - Using themes and styles. - Creating responsive user interfaces that use material design principles. - Creating layouts that work on different screen sizes and orientations. - Creating accessible and easily localizable apps.	Day - 3
5.1: Drawables, styles, and themes *Learn to use drawables, which are compiled images. Learn how styles and themes can give your app a consistent look with less XML code.*	Day - 3
5.2: Material Design *Learn about material design, a visual design philosophy that allows apps to include material design attributes, such as depth and elevation.*	Day - 3
5.3: Resources for adaptive layouts *Learn how to create layouts that work well for different screen sizes and orientations, different devices, different locales and languages, and different versions of Android.*	Day - 3

Lesson 6.Testing your UI **This lesson covers:** - An overview to UI Testing. - An introduction to the Espresso Framework. - Manual Testing. - Automated Testing. - Using Espresso and UI Automator. - Recording Tests.	Day - 3
6.1: UI testing *Learn about testing the user interface of your app.*	Day - 3

Unit 3 – Working in the background

This unit covers how to do background work, how to schedule tasks, and how to trigger events. It covers the performance implications of executing work in the background, as well as best practices for reducing battery drain. You learn how Android determines which apps to keep running and which to stop when resources run low.

You write an app that connects to the Internet in a background thread to find the author of any book. You also build apps that send notifications and schedule tasks, and you learn how to implement scheduling functionality for apps that run on earlier versions of Android.

Lesson 7. Background tasks **This lesson covers:** - Establishing an internet connection. - Sending an HTTP request. - Parsing a JSON response. - Running work asynchronously in the background. - Using AsyncTask and AsyncTaskLoader. - Working with Broadcast Receivers. - Understanding Services.	Day - 4
7.1: AsyncTask and AsyncTaskLoader *Learn about using AsyncTask to run work in the background so that the user does not have to wait for the task to complete.*	Day - 4
7.2: Internet connection *Learn how to enable your app to connect to the internet.*	Day – 4
7.3: Broadcast receivers *Learn about how to send and process broadcasts.*	Day - 4
7.4.0: Services *Learn about the different types of services, how to use them, and how to*	Day - 4

manage their lifecycles within your app.	

Lesson 8. Alarms and schedulers **This lesson covers:** - Scheduling and triggering background tasks. - Using alarms. - Understanding the impact of data transfer on battery power and performance. - Working with Job Scheduler.	Day - 4
8.1: Notifications *Learn how to create, deliver, and reuse notifications.*	Day - 4
8.2: Alarms *Learn how to schedule alarms.*	Day - 4
8.3: Efficient data transfer *Learn how data transfer can drain your users' batteries, and learn strategies for minimizing battery drain.* .	Day - 5

Unit 4 – Saving user data

This unit discusses how to store user data. You learn how to use shared preferences to save simple key value pairs, then you learn how to use the Room database to save, retrieve, and update user data. This unit also introduces you to the Android Architecture Components, which represent best practices for structuring your app.

Lesson 9. Preferences and settings **This lesson covers:** - ways to save data - using Shared Preferences as a way to save data - using the Settings activity to provide an interface that allowins users to set and save app settings	Day - 5
9.0: Data storage *Learn the different ways to store data in your app.*	Day - 5
9.1: Shared preferences *Learn when and how to use SharedPreferences to save data as key-value pairs.*	Day - 5
9.2: App settings *Learn how to allow users to update and save app settings.*	Day - 5

Lesson 10. Storing data with Room **This lesson covers:** - intro to SQLite - Architecture Components - using Room Database, ViewModel and Repository to save and manage data in your app	Day - 5

10.0: SQLite primer *Refresher and quick reference to interacting with an SQLlite relational database management system. Android provides a builtin SQLite database.*	Day - 5
10.1: Room, LiveData, and ViewModel *Introduction to Architecture Components, which provides guidelines and libraries to help you implement best practices in app architecture.*	Day - 5