Android Development

Page No.
Date

ch-3 : Development Tools

Layout – An app is comprised of one or more screens we can define what each screen look like using a layout to define its appearance.
Layout can defined using XML, and can include GUI components such as buttons, text fields etc.
→ (activity_main.xml)

Activity – A special Java class called an activity decides which layout to use & tells the app what to do, by writing Java code. on how to respond to the user.
eg:- Layout includes a button, you need to write a java code in the activity to define what the button should do when user press it. → (MainActivity.java)

Q1. Why can't we use ordinary JDE in Android ?
Ans. Android devices don't run .class & .jar files. To improve speed & battery performance android devices use their own optimised formats for compiled code. So, we need a special tools to convert compiled code into an Android format.

- **Android SDK** – Android Software Development Kit contains the libraries & tools you need to develop android apps. It includes –

a) SDK Platform – present for each version of android
b) SDK Tools – Tools for debugging & testing.
c) Sample apps – It will help you on how to use some of the APIs.

JRE → Java Runtime Edition
API → Application Program Interface

d) Google Play Billing :- Allow you in integrate billing service in your app.
e) Android Support :- Provides extra APIs that aren't available in standard platforms.
f) Documentation :- Get to the latest API documentation offline.

- Android Studio uses the "Company Domain" & "Application Name" to form the name of the package. is important as it is used by android devices to "uniquely identify your app" & used to manage multiple versions of the same app.
- Minimum SDK → is the lowest version of android app will support.

- How Activities & Layout work together to create a user interface :-

1). Device lauch app → create activity object.
2). Activity object specifies a layout. & display the layout on the screen.
3). User interacts layout on device.
4) Activity responds to these interactions by running applicate code.
5). Activity update → user can see on device.

- Android is a open source platform based on Linux & championed by Google.

Android studio projects uses the gradle build system to compile & deploy the app.

Page No.
Date

(IN Android Studio)

- Types of folder structure includes different types of files-

1. Java & xml source files - Activity & Layout files created for the development of app.
2. Android generated Java files - There are some extra java files that Android studio generates automatically.
3. Resource files - Include default image files for icons, styles & string values.
4. Android Libraries - Minimum SDK version of your app, the studio make sure it includes the relevant android libraries for this version.
5. Configuration files - tell the Android what's actually in the app & how it should run.

- Imp files :-

1. root/folder → has the same name as your project.
2. Build/folder → contain files that and. studio creates for you. We don't usually edit anything in this.
3. src/folder → contains source code, we write & edit.
4. java/folder → contains any Java code. any activity you create it will live here.
5. res/folder → to find system resources in this.
   layout/folder → contain layout.
   values/folder → contain resource file for value such as strings.
6. AndroidManifest.xml → The manifest file contains essential info about the app such as components, libraries, etc. every android app must include a file called Android Manifest.xml. at its root. It lives in the app/src/main folder.

It is used to separate out text values from layouts & activities & supports localisation.

7. Strings.xml → contains string id / value pairs. such as application name & default text values.

8. activity_main.xml → defines a layout.

9. MainActivity.java → defines a activity.

10. R.java → Every android project need this file which is presend in the generated folder. Android uses it to keep track of resources in the app.

- Two ways to edit the files 1) Code editor 2) Design editor.
- AVD → Android Virtual Device behaves just like an android physical device. ~~It allows the emulator to run by providing the environment to check the built app.~~ AVD runs in the Android emulator.
- Emulator — Allow you to run your app on an Android VD Emulator built on existing emulator called QEMU similar to other virtual machine. like Virtual Box or VM Ware. It built into the Android SDK.
- AVD Manager — Allows you to setup new AVD & view & edit one's that you have already created.
- APK file — Android Packaging Kit is an android application package. It's basically a jar/zip file. It includes the compiled Java files, bytecode, libraries & resources needed by your app. You install an app on device by installing APK.

Page No.
Date

• What happens when you run the app :-

1). Java file compiled bytecode
2) APK will be created.
3) Emulator gets launched with AVD.
4) once emulator & AVD is active, the APK file is uploaded to AVD & installed.
5) AVD starts mainactivity, App gets displayed on AVD screen.

• Android Studio Console - Gives you a blow by blow account of what the gradle build system is doing & if it encounters any errors, get highlighted in text.

• ART - Android Runtime. Android app runs in separate processes using the Android RT. It comes with a set of core libraries that implement from JAVA.

• Android App is comprised of Activities, Layouts, & resource files. They are just a bunch of files in particular directories.

• Android Studio - special version of IntelliJ IDEA that interfaces with Android SDK & gradle build system It provides tools, editor, libraries, templates for development

• Version of API have a version no, API level, code name.

• How to create a AVD :-

1) open AVD Manager (Tools → Android → AVD Manager)
2). select the hardware
3) select a system image.
4) Verify AVD.

- string.xml → is a default resource file used to hold name/value pairs of strings so that they can be referenced throughout the app.

<string> - identifies the name/value pair's strings.
<resource> - identifies the content of the file as resources.

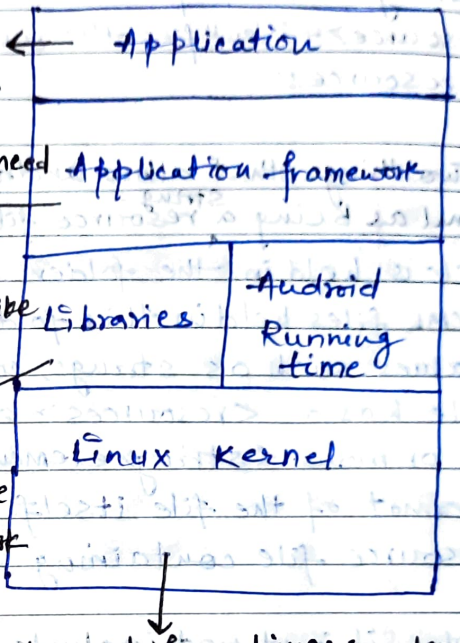
① Two things that allow android to recognize strings.xml as being a string resource file :-

1) file is held in the folder app/src/main/res/values. XML files held in this folder contains simple values such as strings, color.
2) file has a <resources> element, which contains one or more <string> elements.
format of the file itself indicates that it is a resource file containing strings.

① Why it just not include simple text?
Localisation.
Using string.xml as a central resource for text values also makes it easier to make global changes in your whole application.

eg:- You want to make your app available internationally & for different language. rather than to change harcoded text values, you can simply replace string.xml file with an internationalized version.

• Android Platform :- is made up of a no: of different components, includes core application, set of API to help you control what your app look & how it behaves & a whole load of supporting files.



Set of core application eg→ contacts, calendar, maps, etc.

When you build your app you need to access same APIs used by the core application to control what your apps look like & how it behaves.

Set of C & C++ libraries. These exposed to us through framework APIs.

Android relies on kernel for drivers, also core services such as security & memory management.

Emulator - Application that recreates the exact hardware env. of an android device from its CPU & memory through to the sound chips & video display.