

ANDROID NOTES

classmate

Date _____

Page _____

* Android :

Android is a mobile operating system based on a modified version of the Linux kernel and other open source software, designed primarily for touchscreen mobile devices such as smartphones and tablets.

* Versions of Android :

- | | |
|---------------------|-----------------|
| 1. donut | 8. kitkat |
| 2. eclair | 9. lolipop |
| 3. froyo | 10. marshmallow |
| 4. gingerbread | 11. nought |
| 5. honeycomb | 12. Oreo |
| 6. icecreamsandwich | 13. pie |
| 7. jelly beans | 14. OS 10 |

* Android Architecture :

- Android architecture contains different number of components to support any android device needs.
- Android software contains an open-source Linux kernel having collection of number of c/c++ libraries which are exposed through an application, framework Services.

@ Curious _ . Programmer

* The main components of android architecture are following:

1. Applications :

top-layer - containing all applications like home, contact, gallery, camera.

2. Application Framework :

Content several important classes to develop applications.

3. Android Runtime :

Android Runtime environment is one of the most important part of Android. It contains components like core libraries and the Dalvik Virtual Machine (DVM)

4. Platform Libraries :

The Platform libraries includes various c/c++ core libraries and Java based libraries such as Media, Graphics, Surface Manager, OpenGL etc. to provide a support for android developement.

5. Linux kernel :

This is like a heart of android architecture. It contains security, memory management, process management, network stack, driver model.

* Software Required to develop Android Application:

1. java jdk 5 or jdk 6
2. android sdk
3. eclips or android studio
4. Android Development tools.
5. avd or android phone.

* Android Developer Tools:

Android developer tools create interactive and powerful application.

1. SDK Tools:

These are independant and no matter which android platform you working with.

2. Platform Tools:

Platform tools are customize to support the feature of the latest android platform.

* List of Platform Tools:

1. android debug bridge (ADB)

Android Debug Bridge is a versatile command line tool that lets you communicate with a device.

2. Android Interface Defination Language (AIDL)

3. adpt, dexdump and dex etc.

* Life cycle of Android Application:

• java file - java compiler (javac) - • class file - dex compiler (dx) - • dex file - packaging adpt - • apk file.

* Dalvik Virtual Machine (DVM):

- > The Dalvik Virtual Machine is a register based virtual machine optimised for the mobile devices.
- > It optimize the virtual machine for memory, battery life and performance.

* Android Virtual Device:

• An android virtual device is a configuration that defines the characteristics of an android phone tables, were os, android tv, and automative os.

• The AVD manager is an interface you can launch from Android Studio that helps you to create and manage AVDs.

* Emulator:

In computing, an emulator is hardware or software that enables one computer system (called the host) to behave like another computer system.

1) Android Emulator is a application that represents a Virtual Device.

2) This provides all the android applications in one device.

* Android Manifest:

The AndroidManifest.xml file contains information of your package, including components of the applications such as activities, services, broadcast receivers, content providers etc.

* Activity:

- An activity represents a single screen with a user interface.

- For example, An Email application might have one activity that shows a list of new emails, another activity to compose an email and another activity for reading emails.

- If an application has more than one activity, then one of them should be marked as the activity that is presented when the application is launched.

* Activity Life Cycle:

1. ACTIVITY LAUNCHED

2. onCreate(): the activity enters in a create state.

3. onStart(): this makes the activity visible to user.

4. onResume(): called when activity will start interacting with the user.

5. onPause(): called when activity is not visible to the user.

6. onStop(): called when activity is no longer visible to the user.

7. onDestroy(): called before the activity is destroyed.

* Services:

- A service is an application component that can perform long running operations in the background and it does not provide a user interface.

- Another application component can start a service, and it continues to run in the background even if the user switches to another applications.

* Types of Services:

1. Foreground:

- A foreground service performs some operation that is noticeable to the user.
- Ex: playing audio track in foreground.

2. Background:

- A background service performs operations that isn't directly noticed by the user.
- Ex: app used a service to compact its storage.

3. Bound:

- A service is bound when an application component binds to it by calling `bindService()`.

* Services States:

1. Started:

A service is started when an application component, such as an activity, starts it by calling `startService()`.

2. Bound:

A service is bound when an application component binds to it by calling `bindService()`.

* Life Cycle of Services:

UNBOUND SERVICE || bound

1] `startService()`:
This starts the service execution.

2. `bindService()`:
A service is bound when another component.

2] `onCreate()`:
The system invokes this method to perform one-time setup procedures when the service is initially created.

`onBind()`:
The system invokes this method by calling `bindService()` when another component wants to bind with the service.

3] `onStartCommand()`:
The system invokes this method by calling `startService()` when another component (such as an activity) requests that the service be started. If we implement this we must call `stopSelf()` or `stopService()` to stop the service.

onUnbind():

The system calls this method when all clients have disconnected from a particular interface published by the service.

OnRebind():

The system calls this method when new clients have connected to the service after it had previously been notified that all had disconnected in its on Unbind (Intent)

4] OnDestroy():

The system invokes this method when the service is no longer used and is being destroyed.

* CREATE A SERVICE:

1) startService (new Intent (getBaseContext(), MyService.class));

2) stopService (new Intent (getBaseContext(), MyService.class));

Android Manifest:

```
<service android:name=".MyService" />
```


* **TOAST:**

- Toast is used to display information for a period of time. It contains a message to be displayed quickly and disappears after specified period of time.
- Toast is a subclass of Object class.
- Toast notification in android always appears near the bottom of the screen.

`Toast.makeText (Context context, CharSequence text, int duration) -`

> This method is used to initiate the Toast.

1. **LENGTH - LONG :**

It is used to display the Toast for a long period of long time.

2. **LENGTH - SHORT :**

It is used to display the Toast for short period of short time.

* **Broadcast Receiver:**

• Android apps can send or receive broadcast messages from the android system and other Android apps.

• These broadcasts are sent when an event of interest occurs.