**Answers for interview 1**

**Android platform architecture? Android Compiler?**

Android is an open source, Linux-based software stack created for a wide array of devices and form factors.

Architecture is below:

-System Apps

-Java API Framework

-Native C/C++ libraries

-Hardware abstraction layer (HAL)

-Linux Kernel

Compiler converts the source code in to DEX files, which includes the byte code that runs on Android devices.

**Supporting multiple languages**

We can create many strings.xml files, each stored in a specific directory as:

res/values/strings.xml 🡺 contains English text

res/values-fr/string.xml 🡺 contains French text.

res/values-ja/string.xml 🡺 contains Japanese text.

If the device is set to French, Android loads title from the res/values-fr/strings.xml file.

**View/View group? When to use which ViewGroup?**

Views are usually called widgets. It draws something that user can see and interact with it as Button, or TextView, EditText, CheckBox etc. ViewGroup is invisible to the user. It is called *Layouts*. it defines the layout structure for View and other ViewGroup objects. When we use ViewGroup when we want to specify the layout we want to use such as Linear Layout, Relative Layout, Web View etc.

**What is context? What are the types?**

Context is an Interface to global information about an application environment.

Types: Application context, Activity context.

**What are the components in android? Elaborate each?**

There are four different components: activity, services, broadcast receivers and content providers.

***Activity***is a single screen that the user is interacting with. It is said to be an entry point for the interacting user.

***Service*** is a general-purpose entry point that keeps an app running in the background. For example playing music in the background while doing something else in the foreground.

***Broadcast receiver***enables the system to deliver events to the app. For example battery low notification, picture was captured, or screen turned off. The app gets notified by broadcast receiver for these events.

***Content provider*** is a component that supplies data from one app to others on request. For example managing user’s contact information.

**How to handle configuration changes?**

When configuration change happens(changing screen orientation) the activity can be destroyed and removed from the memory and re-created again. For this reason we need to properly save and restore the state. In another words Android provides callbacks to save the application state before destroying it and restoring the application state when creating it. We need to activity callback methods called onSavedInstanceState() and onRestoreInstanceState().

**How to avoid activity from getting destroyed on orientation change?**

The answer is the same as the previous question’s answer.

**What is intent? What the types on intents?**

It is an abstract description of an operation to be performed.

**Serializable vs parcelable?**

Serializable is part of standard Java Interface. It is not part of the android SDK. But parcelable is part of Android SDK. Parcelable is more faster then serializable because it is optimized. Because serializable interface create a lot of temporary objects and cause quite a bit of garbage collection.

**How does the build system work?**

-Converts the source file into DEX files.

-The APK Packager combines the DEX files and compiled resources into a single APK.

-The APK Packager signs the APK using either the debug or release keystore

**What are the different ways to persist data in Android?**

[***Internal file storage***](https://developer.android.com/guide/topics/data/data-storage.html#filesInternal)**:** Store app-private files on the device file system.

[External file storage](https://developer.android.com/guide/topics/data/data-storage.html#filesExternal): Store files on the shared external file system. This is usually for shared user files, such as photos.

[***Shared preferences***](https://developer.android.com/guide/topics/data/data-storage.html#pref): Store private primitive data in key-value pairs.

[***Databases***](https://developer.android.com/guide/topics/data/data-storage.html#db)**:** Store structured data in a private database.

**How is data stored in shared pref?**

The key-value pairs are written to XML files that persist across user sessions. We can manually specify a name for the file or use per-activity files to save the data.

**Can we have multiple files for shared prep? Why? How to write /read data on files?**

**Elaborate the impl for SQL database in Android?**

Android contains SQLite database to manage their own private databases. We need to create a class that extends SQLiteOpenHandler, then execute the SQLite command by override onCreate() method, then call create the database by calling getWritableDatabase() method and add object to it. Finally we can use curser to to get all the elements in a list.

**How do achieve multithreading in Android?**

We can do it by implementing runnable interface of we can create a class that extends Thread class and this subclass should override run method of class Thread the we can create an instance of thread and start is by .start() method so runnable is runnable.

What are different ways to pass data to the main thread?

Threads running in pool don’t have access to the UI objects. To move the data from background thread to the UI thread (main thread) we need to use Handler that runs on the UI thread. Handler object receives messages runs code to handle messages.

**How to use and Handler with Looper class?**

-Instantiate the Handler object in the constructor for the class that creates pool threads. This constructor uses looper class and the handler runs on the same thread as the Looper. Inside the handler handle Message() must be overridden.

**What is a Thread pool executor? Blocking queue?**

Thread pool executor is a class that implements ExecutorService which gives fine control on the thread pool. Blocking queue

Blocking queue is an interface in java. There are threads, which insert objects to the queue, and other threads taking the objects out. When the queue is full the threads that are trying to insert new objects to the queue is automatically blocked.

**Difference between Final, Finally and Finalize?**

*Final* is a key word on a method, objects (as String) and primitive types that they cannot be overridden. The *finally* block always executes when the try block exits.

*Finalize* is Called by the garbage collector on an object when garbage collection determines that there are no more references to the object.

**How achieve thread safety?**

It’s a concept that the program behaves correctly when multiple threads use the same source. The simplest means of making something thread safe is to only access the state from a single thread. Java has a synchronized key word to do this.

Mutex lock is designed for this. It means only one thread can have the lock and access the source.

**Answers for Interview 2**

**What is a ListView? How do we implement adapter class?**

ListView is a view that displays vertically scrollable of the views. It is modern and flexible.

To implement adapter class??

**How does a View gets inflated in a ListView?**

Let say we have list about any anything.

Get the ListView by calling findViewById method.

Create an ArrayAdapter and associate it with the list.

Then call myListView.setAdapter(arrayAdapter);

**What is the difference between ListView and RecyclerView?**

ListView shows all the elements of the list but RecyclerView provides limited window on a large dataset.

**What are the some layout managers that you can use in RV?**

LinearLayoutManager, StagedGridLayoutManager, GridLayoutManager.

**How to define a ViewType in a RcyclerView?**

We need to override getItemViewType() this returns an integer. Override onCreateViewHolder() then we can use the integer that is returned by getItemViewType() method and do switch cases to use a layout that is specified by the integer

**Elaborate the implementation of RecyclerView.Adapter class?**

The view holder objects are managed by an adapter that extends Recycler.Adapter class. Recycler.Adapter class creates view holders and bind view to data by calling onBindViewHolder() method. This method uses the view holders position to determine what the contents should be.

**How to load images/items in RV with smooth scrolling?**

**How to achieve the swipe/drag on RV items?**

This can be done by ItemTouchHelper class we can add swipe to dismiss, drag&drop support to RecyclerView. ItemTouchHelper.SipmleCallback provides certain callback method like onMove(), onChildDraw(), onSwiped() when the row is swiped.

**What are Fragments? What are they used for?**

Fragments are the portions of the UI in a FragmentActivity. They are used to contribute its own layout to the activity.

**What is a fragment lifecycle?**

onAttach 🡺 onCreate🡺 onCreateVew 🡺 onActivityCreated 🡺 onStart 🡺 onResume 🡺 onPause 🡺 onStop 🡺 onActivityDestroyed 🡺 onDestroy 🡺 onDetach

**When is the onActivityCreated called in a fragment?**

It gets called right after onCreateView.

**How to add a fragment on an Activity?**

By creating a new fragment xml file.

**Can a fragment exists own their own?**

A fragment cannot exist its own. It has to be in activity.

**Does the fragment need a UI?**

Not necessarily

**How to communicate among fragments?**

Communications go through activity

**How to remove all fragments on an activity?**

By making Fragment transaction.

**What are services? Types?**

Services are application components that they can perform long-running operations in the background.

Types are: Foreground, background and Bound services.

**What is the difference between IntentService and Service?**

Service is a base class for all services but IntentService is a subclass of Service that uses worker thread instead of main thread to handle multiple requests simultaneously.

**How do we communicate with a bound service?**

Through service connection

**How do we communicate with an intent service?**

Through ResultReceiver

**Can we run an intent service in parallel?**

No, all the consecutive Intents will go into the message queue for the worker thread and will execute sequentially.