1. Tools

1.1. Buttons and Drawables

http://angrytools.com/android/button/

2. Name convention

https://source.android.com/source/code-style

Suggested By Udacity Forum Mentor

https://github.com/TreyCai/AndroidNamingConvention
https://github.com/NexMM/android-naming-conventions

Other references:

https://github.com/futurice/android-best-practices
https://github.com/ribot/android-guidelines/blob/master/project_and_code_guidelines.md
https://github.com/ribot/android-boilerplate

https://github.com/futurice/android-best-practices

3. RecyclerView

3.1. Reusable Template: TakeOffRecyclerView

Tags: Reusable RecyclerView template

1. Download, extract, place.
2. Go to new -> other -> recyclerview -> done.
RecyclerView with adapter and dummy data.
https://github.com/TakeoffAndroid/RecyclerViewTemplate

3.2. LoadMore EndLess Infinite RecyclerView

Tags: infinite, endless, loadmore, load, more, recyclerview, recycler view

3.2.1. Android Pagin Library

https://medium.com/@sharmadhiraj.np/android-paging-library-step-by-step-implementation-guide-75417753d9b9

https://proandroiddev.com/8-steps-to-implement-paging-library-in-android-d02500f7fffe

3.2.2. Old way:

The old idea is:

1. We add null as the last element of the list if the size of the data to add in a list is equal to pre-defined pagination limit.
2. We use different viewHolder for null item in RecyclerView.Adapter
3. We use scroll listener, layout manager, threshold, visible items etc... to detect if the user is reached at the bottom of the recyclerView.
4. Once the user is reached to the bottom of the recyclerView, we add more data (next page data) to the list and before we insert new data, we simply remove that null element.
5. Repeat 1.
6. Note: Currently we are using setLoaded boolean in scroll listener.

We are using a combination of:
https://guides.codepath.com/android/endless-scrolling-with-adapterviews-and-recyclerview
https://gist.github.com/nesquena/8a976dd3d6f866518db2cfe7f9cb0db7
https://baraabytes.com/android-endless-scrolling-with-recyclerview/

Other useful resources to get the idea and broad insight:
http://android-pratap.blogspot.in/2015/01/endless-recyclerview-onscrolllistener.html

https://guides.codepath.com/android/using-the-recyclerview#scrolling-to-new-items

https://guides.codepath.com/android/Endless-Scrolling-with-AdapterViews-and-RecyclerView#implementing-with-recyclerview

Which method for what

http://msobhy.me/2015/09/05/infinite_scrolling_recyclerview/

https://codentrick.com/load-more-recyclerview-bottom-progressbar/

3.3. setHasFixedSize

Understood setHasFixedSize(boolean true | false) method of recycler view. Below is the link:

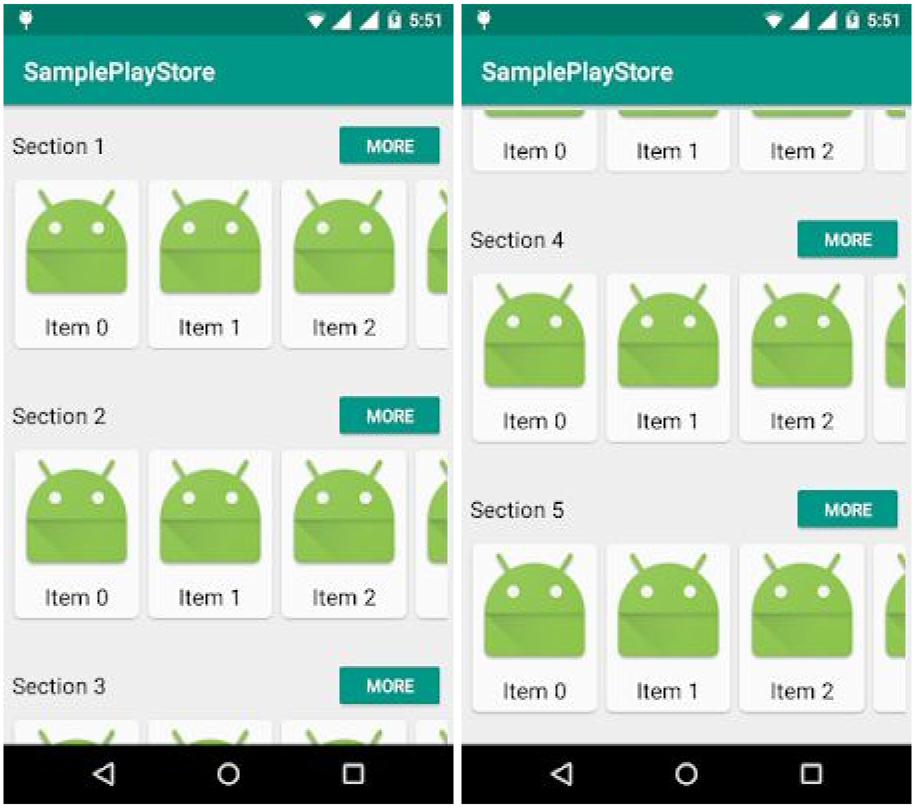
http://stackoverflow.com/questions/28709220/understanding-recyclerview-sethasfixedsize

https://developer.android.com/reference/android/support/v7/widget/RecyclerView.html#setHasFixedSize(boolean)

Both recommends to keep it true when CRUD kind of operations being performed on items of recycler view.

3.4. Horizontal recyclerView inside Vertical recyclerView like Google play store

Tags: Nested recyclerView recyclerView inside recyclerView



http://android-pratap.blogspot.in/2015/12/horizontal-recyclerview-in-vertical.html

https://irpdevelop.wordpress.com/2016/02/10/horizontal-recyclerview-inside-a-vertical-recyclerview/

https://medium.com/@mgn524/optimizing-nested-recyclerview-a9b7830a4ba7

https://github.com/msahakyan/nested-recycler-view

3.5. RecyclerView performance tips

https://medium.com/master-of-code-global/recyclerview-tips-and-recipes-476410fa12fd

https://stackoverflow.com/questions/33845846/why-is-adding-an-onclicklistener-inside-onbindviewholder-of-a-recyclerview-adapt

3.6. DiffUtil for RecyclerView

https://android.jlelse.eu/smart-way-to-update-recyclerview-using-diffutil-345941a160e0

https://medium.com/@iammert/using-diffutil-in-android-recyclerview-bdca8e4fbb00

3.7. SmoothScrollToPosition

Tags: scroll to position

https://stackoverflow.com/questions/31235183/recyclerview-how-to-smooth-scroll-to-top-of-item-on-a-certain-position/43505830#43505830

3.8. RecyclerView above soft keyboard show last item

addOnLayoutChangeListener (View.OnLayoutChangeListener)

https://stackoverflow.com/questions/34102741/recyclerview-not-scrolling-to-end-when-keyboard-opens/34103263

--1-- scroll to position of an element of child recyclerView…

recyclerView.addOnLayoutChangeListener (View.OnLayoutChangeListener)

recyclerView.smoothScrollToPosition,

((LinearLayoutManager)recyclerView.getLayoutManager).scrollToPositionWithOffset (pos, offset)

recyclerView.scrollToPosition, recyclerView.scrollTo (explicitly specified, fixed distance from the screen or top or distance from parent, from 0 to?), recyclerView.scrollBy (explicitly specified, fixed distance to travel from the current state, current position)

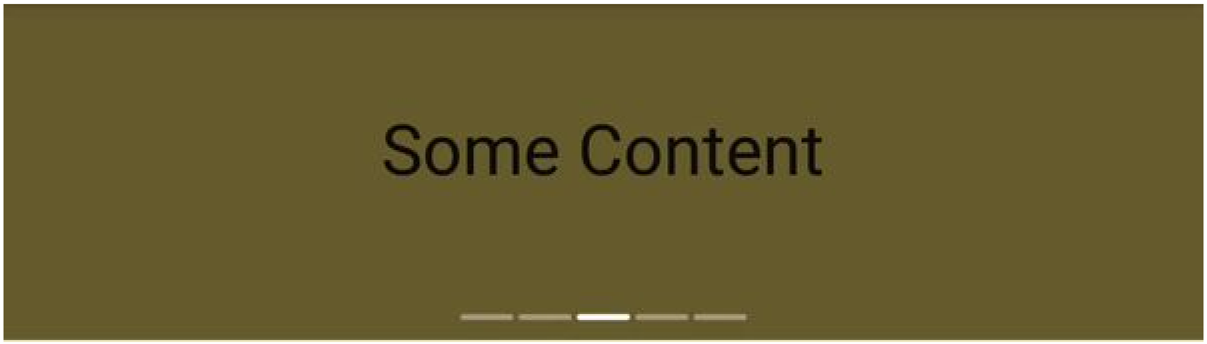
3.9. Attach to root : ViewHolder

Can give remove view error if true for nested?

Read the answer given by Rohit Singh

https://stackoverflow.com/questions/12567578/what-does-the-layoutinflater-attachtoroot-parameter-mean

3.10. RecyclerView like ViewPager with bottom indicator like tabs



```
// add pager behavior
recyclerView.setOnFlingListener(null);
PagerSnapHelper snapHelper = new PagerSnapHelper();
snapHelper.attachToRecyclerView(recyclerView);
```

https://blog.davidmedenjak.com/android/2017/06/24/viewpager-recyclerview.html

https://stackoverflow.com/questions/44043501/an-instance-of-onflinglistener-already-set-in-recyclerview

3.11. RecyclerView to ViewPager

Shared element transition for recyclerview to viewpager, sprinig rebound effect for recyclerview to viewpager

https://github.com/google/android-transition-examples

3.12. Animation: RecyclerView to ViewPager

Shared element transition for recyclerview to viewpager, sprinig rebound effect for recyclerview to viewpager.

A google repository:

https://github.com/google/android-transition-examples

https://github.com/alexvasilkov/GestureViews

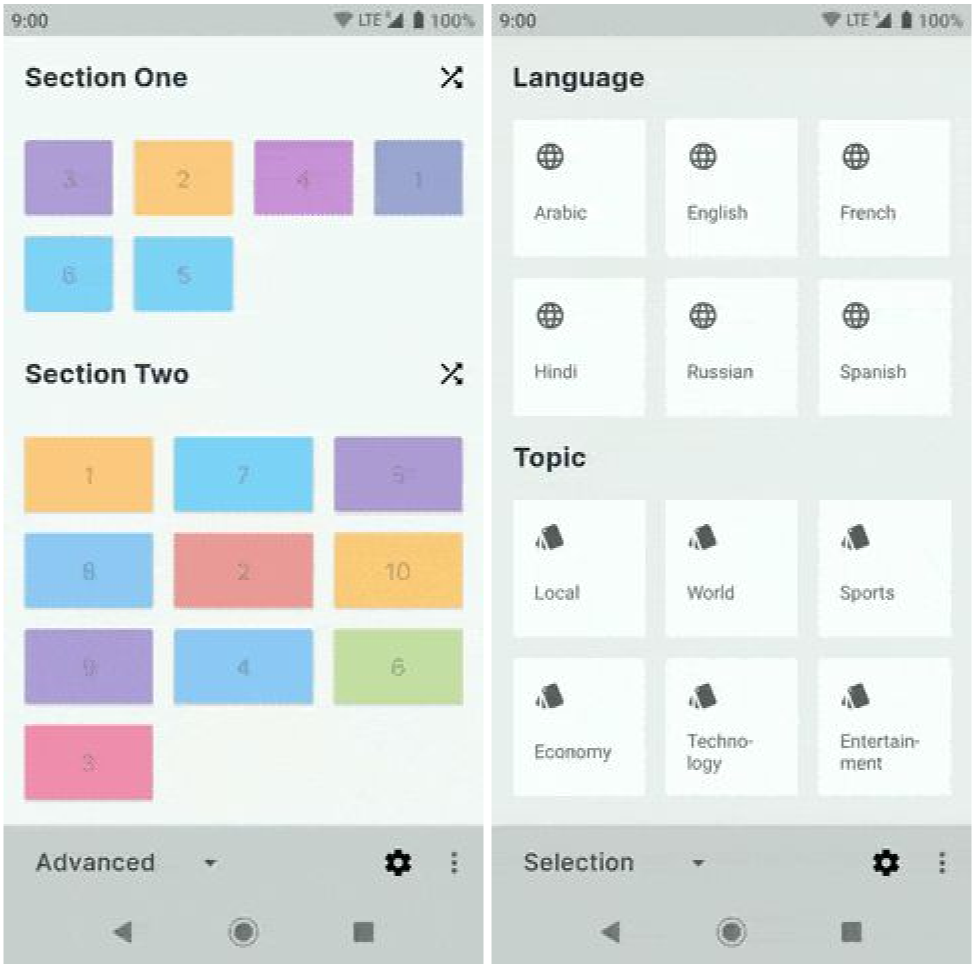
3.13. Stop auto scrolling of RecyclerView as soon as screen gets launched

Tags: auto scrolling of recyclerView, recyclerView reaches top instead of nestedScrollView, nestedScrollView goes automatically up followed by recyclerView, coordinator

Simply put the xml attribute, `focusableInTouchMode = "true"` in recyclerView or it's parent

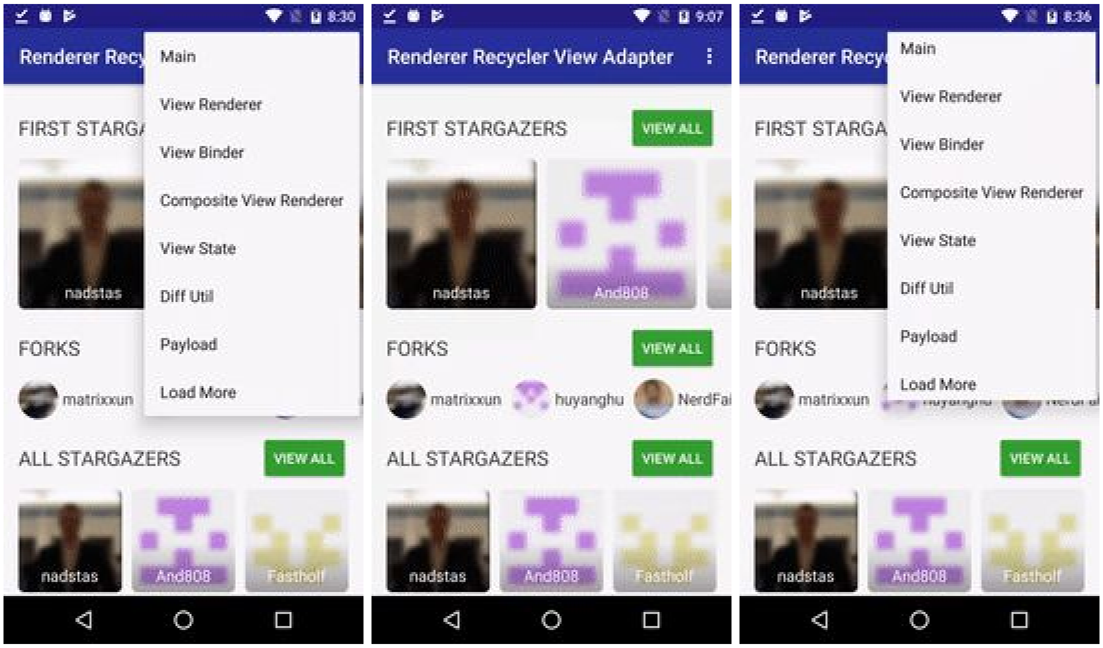
3.14. Custom RecyclerView

https://github.com/DevAhamed/MultiViewAdapter



3.15. Custom RecyclerView

https://github.com/vivchar/RendererRecyclerViewAdapter



3.16. Nested RecyclerView: Horizontal inside Vertical Rv

https://github.com/msahakyan/nested-recycler-view

3.17. Troubleshooting

3.17.1. Unable to see recyclerView list item

1. Make sure that your recyclerView is visible.
2. Make sure that you have set layout manager and adapter to your recyclerView.
3. Make sure that you have some items in your recyclerView adapter.
4. Make sure that you have notified the adapter after any changes in list item with the help of one of the proper notify methods of recyclerView adapter.
5. Make sure that your getItemCount method in recyclerView adapter is returning proper value (list.size()).
6. Make sure that your getItemViewType method is returning proper value.

4. Ubuntu Linux

4.1. Update Android Studio

Tags: android studio, ubuntu, linux, update

Tried to update android studio. Found problem. Followed below link/solution and it worked:

https://stackoverflow.com/documentation/android-studio/1998/android-studio-updates/6539/updating-android-studio-in-ubuntu

If you are able to download an update of Android Studio, but after it restarts nothing happens, check out the following example:

After the patch was downloaded and Android Studio closed, open the terminal
Go to your android studio folder, e.g. cd ~/android-studio
Go to bin subfolder: cd bin
Make sure your studio.sh file has run permissions: chmod +x studio.sh
Run Android Studio from here: ./studio.sh
After that Android Studio will find the patch and install it. Then you may close Android Studio and run it the regular way (whichever way you prefer, I run it from Launcher pad).

5. Fragments

For viewpager, tablayout and communication onResume-onPause, check 14. ViewPager TabLayout

5.1. Beautiful Theory and Explanation

How backstack, onCreate, onCreateView, onResume, Recreation, Relaunch etc… works in fragment:

https://bricolsoftconsulting.com/state-preservation-in-backstack-fragments/

What each method is supposed to do and how each method is supposed to behave and in which order:

http://www.silverbaytech.com/2014/03/24/android-fragment-lifecycle-in-detail/

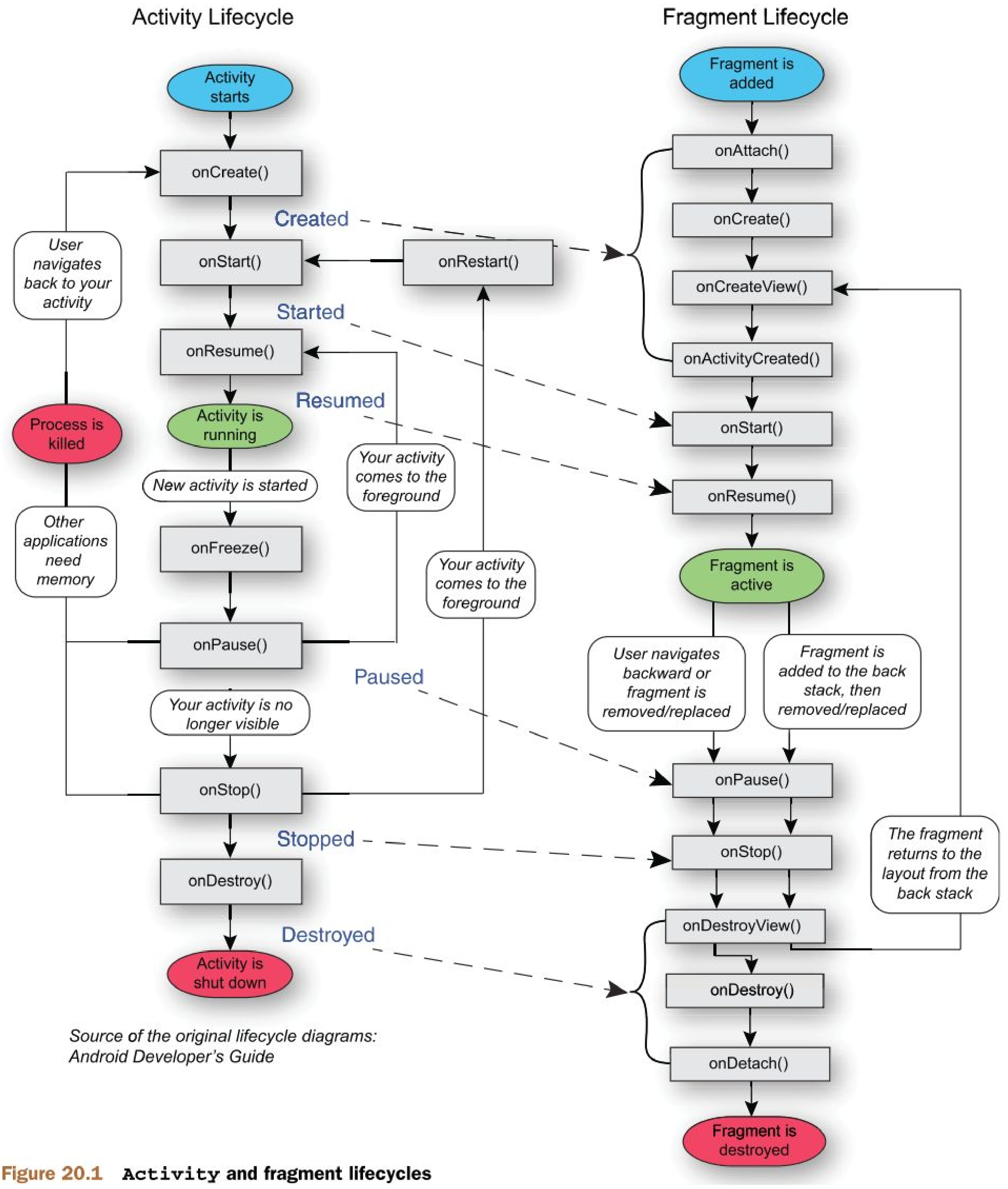
Check an answer given by Marcin to understand fragment life cycle regarding to back press

http://stackoverflow.com/questions/13543835/android-fragments-lifecycle-method-always-called-when-fragment-is-displayed

5.2. life Cycle

Tags: screen orientation change, fragment state, fragment, state, screen rotation

https://github.com/xxv/android-lifecycle



Source of the original lifecycle diagrams: Android Developer's Guide

Figure 20.1 `Activity` **and fragment lifecycles**

5.3. Fragment: How to use properly?

http://stackoverflow.com/questions/7951730/viewpager-and-fragments-whats-the-right-way-to-store-fragments-state

http://stackoverflow.com/questions/13305861/fool-proof-way-to-handle-fragment-on-orientation-change

https://techbandhu.wordpress.com/2013/07/02/android-headless-fragment/
http://stackoverflow.com/questions/19219458/fragment-on-screen-rotation

Beautiful explanation of the DO's and DON'Ts:_

http://code.hootsuite.com/orientation-changes-on-android/

5.4. Back stack - Push-Pop Transaction

http://stackoverflow.com/questions/34276627/android-return-to-specific-fragment-and-pop-off-everything-after-it

http://stackoverflow.com/questions/38852930/pop-specific-fragment-from-stack-and-remove-others

http://stackoverflow.com/questions/32603990/how-to-navigate-to-a-specific-fragment-from-all-other-fragment

http://stackoverflow.com/questions/34803872/android-back-button-navigate-to-specific-fragment

http://stackoverflow.com/questions/6374170/how-to-set-a-fragment-tag-by-code

http://vinsol.com/blog/2014/09/19/transaction-backstack-and-its-management/

https://www.codota.com/android/methods/android.app.FragmentManager/popBackStack

6. Animation

6.1. StateListAnimation

https://blog.stylingandroid.com/statelistanimator/

6.2. Animation Scope and Terms

Animation : Translation Animation etc...
Value Animator - > Object Animator. Property animation / ViewPropertyAnimator / PropertyValueHolder Animator
LayoutAnimation Controller
PhysicsAnimation : Dynamic, Spring, Fling etc... (smooth transition with change in target)

Scene transition
sharedElement transition
Motion layout
constraintLayout and constraintSet animations

Animator set to play more than one animations together or in sequence...

6.3. Object Animator, PropertyValueAnimation and AnimatorSet

```
private void animateObject(int position, int itemCount, View... views) {

        boolean even = timesClicked % 2 == 0;
        if (android.os.Build.VERSION.SDK_INT >= android.os.Build.VERSION_CODES.LOLLIPOP) {
            for (final View view : views) {
                view.setEnabled(false);
                final AnimatorSet animSetXY = new AnimatorSet();

            /*ObjectAnimator y = ObjectAnimator.ofFloat(view,
                    "translationY", view.getY(), getCenter().get(1) - (view.getY() +
(view.getHeight() / 2)));*/

            /*if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.JELLY_BEAN_MR2) {
                y.setAutoCancel(true);
            }*/

            /*ObjectAnimator x = ObjectAnimator.ofFloat(view,
                    "translationX", view.getX(), getCenter().get(0) - (view.getX() +
(view.getWidth() / 2)));*/

            /*if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.JELLY_BEAN_MR2) {
                x.setAutoCancel(true);
            }*/

                PropertyValuesHolder pvhX = PropertyValuesHolder.ofFloat(View.TRANSLATION_X,
/*view.getX(),*/ getCenter().get(0) - (view.getX() + (view.getWidth() / 2)));
                PropertyValuesHolder pvhY = PropertyValuesHolder.ofFloat(View.TRANSLATION_Y,
/*view.getY(),*/ getCenter().get(1) - (view.getY() + (view.getHeight() / 2)));
                PropertyValuesHolder pvhZ = PropertyValuesHolder.ofFloat(View.TRANSLATION_Z,
/*view.getY(),*/ position == 0 && !even ? Animation.ZORDER_TOP : !even ? Animation.ZORDER_BOTTOM
: Animation.ZORDER_NORMAL);

//              PropertyValuesHolder pvhR = PropertyValuesHolder.ofFloat(View.ROTATION, 0, -720);
//              PropertyValuesHolder pvhsX = PropertyValuesHolder.ofFloat(View.SCALE_X, 1, 0, 1);
//              PropertyValuesHolder pvhsY = PropertyValuesHolder.ofFloat(View.SCALE_Y, 1, 0, 1);
//              PropertyValuesHolder pvhA = PropertyValuesHolder.ofFloat(View.ALPHA, 1, 0, 1);


                final ObjectAnimator propertyAnimation =
ObjectAnimator.ofPropertyValuesHolder(view, pvhX, pvhY, pvhZ/*, pvhR, pvhsX, pvhsY, pvhA*/);
//              animation.setDuration(500);

            /*if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.JELLY_BEAN_MR2) {
                animation.setAutoCancel(true);
            }*/
//              animation.setInterpolator(new AccelerateDecelerateInterpolator());
//              animation.setInterpolator(new AnticipateInterpolator());
```

```
                Rect rectf = new Rect();

                //For coordinates location relative to the parent
                view.getLocalVisibleRect(rectf);

                //For coordinates location relative to the screen/display
                view.getGlobalVisibleRect(rectf);

                Log.d("WIDTH        :", String.valueOf(rectf.width()));
                Log.d("HEIGHT       :", String.valueOf(rectf.height()));
                Log.d("left         :", String.valueOf(rectf.left));
                Log.d("right        :", String.valueOf(rectf.right));
                Log.d("top          :", String.valueOf(rectf.top));
                Log.d("bottom       :", String.valueOf(rectf.bottom));

                int[] location = new int[2];
                view.getLocationOnScreen(location);

//              ObjectAnimator centerChangeAnim = ObjectAnimator.ofObject(view, "centerpoint", new PointEvaluator(), view.getX(), btnCenter.getX());

                animSetXY.playTogether(propertyAnimation);
                animSetXY.setInterpolator(new OvershootInterpolator());
                animSetXY.setDuration(5000);
                animSetXY.setStartDelay(position * 100);
                animSetXY.start();

           /* new Handler().postDelayed(new Runnable() {
                @Override
                public void run() {
                    view.clearAnimation();
                    animSetXY.cancel();
                }
            }, 6000);*/

                new Handler().postDelayed(new Runnable() {
                    @Override
                    public void run() {
                        if (timesClicked % 2 == 0) {
                            customGridLayoutManager.setScrollEnabled(true);
                        }
                        enableAllViews();
                    }
                }, 5000 + (itemCount * 100));
            }
        }
    }
```

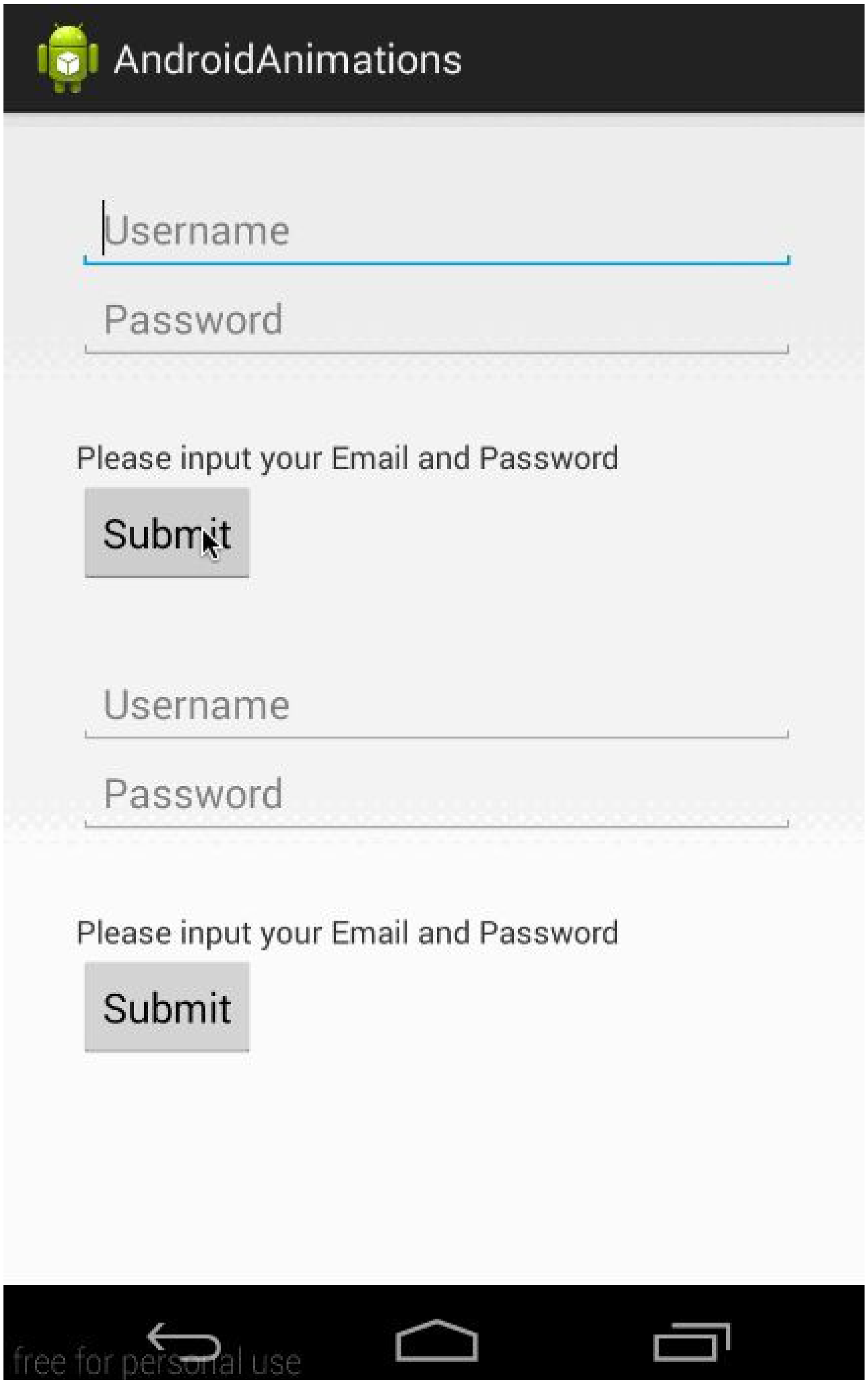
So, using the link below, we can simply make an animation on an item that has just been selected to delete. RollOutAnimation, RotateOutAnimation, ZoomOutAnimation etc... are kind of we were looking for.

Understand each term of Animation in simple possible way:

http://blogs.innovationm.com/scale-animation-in-android-how-it-works/

This library makes it easy to apply any available animation to any view:

https://github.com/daimajia/AndroidViewAnimations



Please input your Email and Password

Please input your Email and Password

https://www.numetriclabz.com/animation-on-custom-view-using-androidviewanimations-library-in-android-tutorial/

From the above info, we can apply ZoomOutAnimation on an item that has just been selected to delete and at the same time, we can apply Shake/swing/wobble Animation to dustbin icon. So, it can give a visual effect like item is being moved to trash bin and trash bin has just been filled up with it!

Now at the same time, or before animation on delete icon, we can simply show a gif over there that will show the dustbin opened and then closed. After which, that trash icon will have shake animation.

So it is like animation series:_ item ZoomOutUpAnimation -> dustbin gif -> dustbin shake, swing or wobble

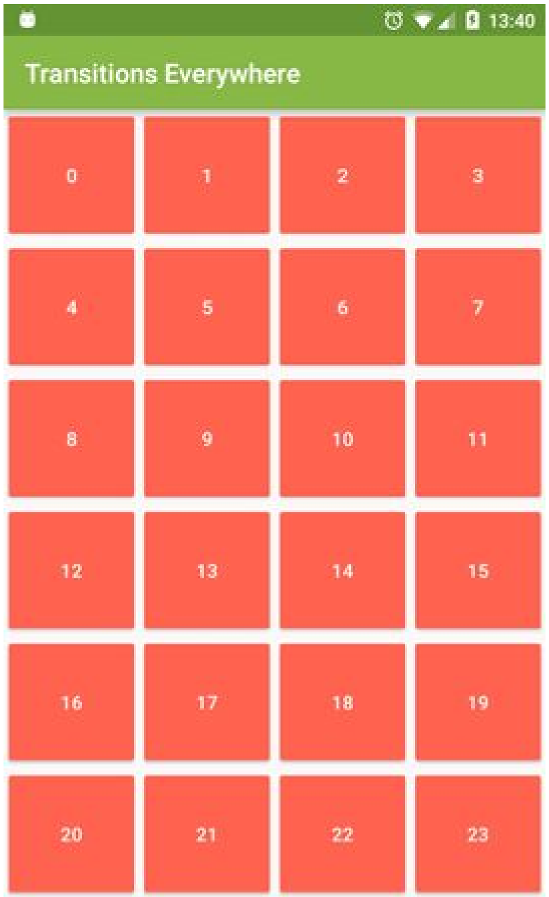
6.4. Names (Terms) of animation in Android:

Tags: animation properties field property names.
These names are useful when we define the animation on our own:
https://developer.android.com/guide/topics/graphics/prop-animation.html#views

6.5. Animations: Transition animation

This is another library which not only on single particular view can operate on but also in group of views or for recyclerView also:

https://github.com/andkulikov/Transitions-Everywhere



https://medium.com/@andkulikov/animate-all-the-things-transitions-in-android-914af5477d50

6.6. Animations:

https://github.com/daimajia/AndroidViewAnimations

6.7. Animations: Spring and fling physics based animations

https://proandroiddev.com/introduction-to-physics-based-animations-in-android-1be27e468835

6.8. Animations: Animate all recyclerView items at once

Tags: animate recyclerview item, animation for recyclerView animation?

https://stackoverflow.com/questions/43791565/animate-all-recycler-view-items-at-once

6.9. Shared Element Transition Animation

http://mikescamell.com/shared-element-transitions-part-1/

https://github.com/lgvalle/Material-Animations#c-start-an-activity-with-a-shared-element

6.10. Lottie from AirBnb

https://github.com/airbnb/lottie-android

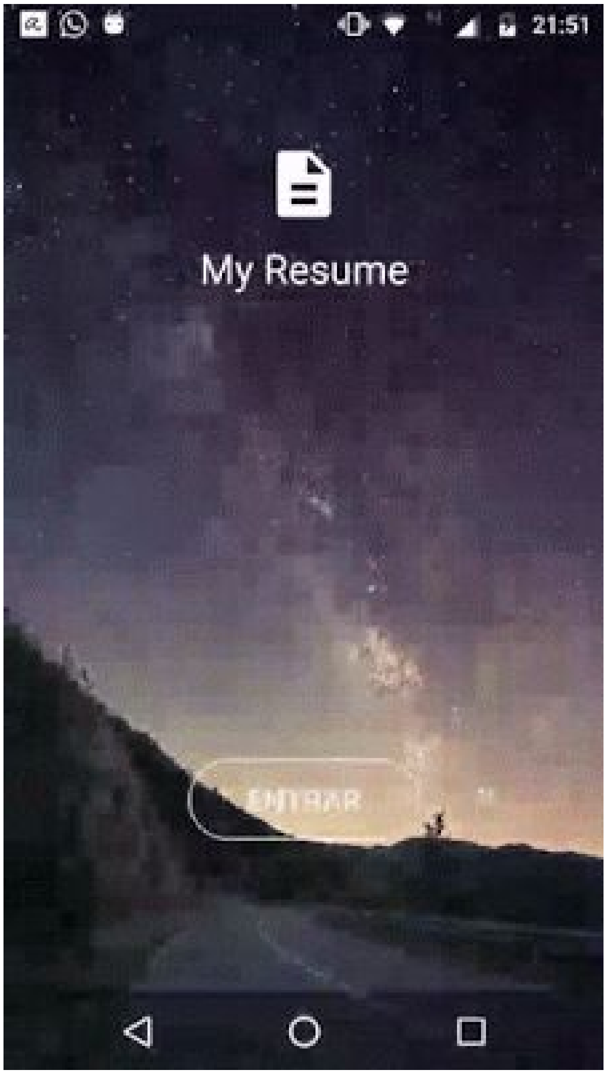
http://airbnb.io/lottie/android/android.html

https://www.lottiefiles.com/

6.11. Circular Reveal Animation

Tags: Reveal activity reveal fragment reveal screen let me android sheggy

https://android.jlelse.eu/a-little-thing-that-matter-how-to-reveal-an-activity-with-circular-revelation-d94f9bfcae28



https://stackoverflow.com/questions/30958834/circular-reveal-transition-for-new-activity

Find useful other animations too for view:

https://stackoverflow.com/questions/36089963/android-reveal-animation-to-start-an-activity

https://medium.com/snapp-mobile/meaningful-motion-circular-reveal-shared-elements-ea495b99adf4

https://developers.pagesjaunes.fr/2017/12/22/reveal-your-app-part2/

This is what vishal might followed: (onTouch and exitReveal)

https://guides.codepath.com/android/circular-reveal-animation

6.12. Create svg vector drawable or vector icon animation : Switch shifter

Tags: icon drawable animation animator online tool

https://romannurik.github.io/AndroidAssetStudio/

7. Motion Event

7.1. Long Press

How to detect long press?
Make new GestureDetector, override onLongPress.
Set onTouch listener on the view, pass the event to gestureDetector:

https://stackoverflow.com/questions/7919865/detecting-a-long-press-with-android

```
final GestureDetector gestureDetector = new GestureDetector(context, new
GestureDetector.SimpleOnGestureListener() {
     public void onLongPress(MotionEvent e) {
       Log.e("", "Longpress detected");
     }
});

public boolean onTouchEvent(MotionEvent event) {
   return gestureDetector.onTouchEvent(event);
};
```

Note: If constructor used in above method is deprecated, you can use any other constructor from here:
https://developer.android.com/reference/android/view/GestureDetector.html#public-constructors

8. Paging Library

Usually we don't fetch all the data at once from server. We get data for each page one by one as the user scrolls down to the bottom of the list. It is also known as pagination, endless scrolling, loadmore, infinite scrolling etc…
We used to implement it as described here:
LoadMore, Endless, Infinite Scrolling

However, Android framework now provides this functionality natively. You can read more on this from here:

https://proandroiddev.com/8-steps-to-implement-paging-library-in-android-d02500f7fffe

9. Networking and data parsing

9.1. Retrofit

https://www.androidhive.info/2016/05/android-working-with-retrofit-http-library/

10. Instagram

https://github.com/TorzsaCristian/Instagram-API-Android-Tutorial
http://siteoftutorials.com/instagram-api-android-tutorial/

11. Google OpenSource Apps

Tags: google open source apps github google github
https://android.googlesource.com/platform/packages/apps/

12. Gradle

Learn how we can configure and optimize our gradle files:
https://developer.android.com/studio/build

Learn how to debug gradle
https://android.jlelse.eu/how-to-fix-android-studio-3-1-error-debugging-issue-8d06ad9f49a9

13. Material Style and Design

Here you will find all components:

https://github.com/material-components/material-components-android

14. ViewPager TabLayout

Tags: fragment onResume visibilityHint onPause viewpager tabLayout visible call method from to

Make interface. Implement in fragment. In tablayout or viewpager listener, cast listener from current fragment and call interface method.

https://stackoverflow.com/questions/17845641/alternative-for-the-onresume-during-fragment-switching
https://stackoverflow.com/questions/10853611/viewpager-with-fragments-onpause-onresume/10860670

15. List Collections Guava gson and Advanced Java

15.1. Filter the list

Tag: get the list of object from another list by conditions on field/s of object

Suppose you have a list of user class and you want to get only those elements from that list where user has either null or empty id. Here, id is the field of user object.

So, we can do something like below:

```
Collection<User> result
                    = Collections2.filter(userList, input -> input != null &&
StringUtils.isNullOrEmpty(user.getId()));
```

Note that we have used custom Predicate to define our custom rule for the filter purpose. For more examples, head over to:

https://www.baeldung.com/guava-filter-and-transform-a-collection

15.2. Transform a list

Tags: Get list of particular field from the list of object
Extract all fields for each object of list
E.g. List<String> userIds = userList.transform(User, userId);
https://www.baeldung.com/guava-filter-and-transform-a-collection

```
Public static <S, T> List<T> transformList(List<S> originalList, Function<S, T>
function) {
      Collection<T> result = Collections2.transform(originalList, function);
      return new ArrayList<>(result);
}
```

The function would look something like below:

```
 Function<User, String> func = new Function<User, String>(){
        @Override
        public String apply(User input) {
            return input != null && StringUtils.isNotNullNotEmpty(input.getUserId) ?
input.getUserId() : “ ”
        }
    };
```

15.3. FluentIterable: Conditional predicate and transform

E.g. List<User> -> Extract only those users who has userId and then make a list of userName, i.e. List<String> userNames from filtered users.

https://www.baeldung.com/guava-filter-and-transform-a-collection

15.4. Multiple Generic Type parameter in class

Tags: generics type parameters

https://stackoverflow.com/questions/1759549/java-generics-multiple-generic-parameters

15.5. Generic Type parameter as field for parcelable class

Tags: generics, type parameters, parcelable, creator, write, in, read, class loader

https://stackoverflow.com/questions/18797399/is-it-possible-to-parcel-a-generic-class

15.6. Custom Serialization and Deserialization

Serialization is used during post data and deserialization is used during parsing.

Sometimes we want to exclude several fields of an object in post_data (request) and sometimes we want to make a custom rule for a field/s with null, empty or 0 value. This can be done too in gson. Google custom serialization and deserialization for gson, custom serializer for gson, register custom adapter for gson serialization and deserialization etc… You can either annotate particular field with annotation @jsonAdapter or you can set your custom adapter or register custom adapter while building gson from GsonBuilder.

You can also rename certain properties (fields, keys) too using naming strategy!

https://futurestud.io/tutorials/gson-advanced-custom-serialization-part-1

https://stackoverflow.com/questions/43047823/gson-parameter-get-serialised-even-though-it-has-exposeserialize-false

Gson normally does not include null values but if you want to include those fields also then:

https://stackoverflow.com/questions/3923759/gson-ignoring-map-entries-with-value-null

This is how we register custom adapter @JsonAdapter to define rule for particular value such as what to do for null value that we expect as default double value in request:

https://stackoverflow.com/questions/42278197/assigning-nulls-in-the-gson-custom-serialization-for-double-fields

https://stackoverflow.com/questions/6856937/gson-custom-serializer-in-specific-case

15.7. Tips

1. Consider return type as an object sometimes so that we can use the same method in different possible ways to support as many cases as we can for a single business logic
   https://stackoverflow.com/questions/8557716/how-to-return-multiple-values

2.
3. S
4. Z
5. z

https://medium.com/@ankit.sinhal/core-and-advance-features-of-gson-library-5cc79625ac37

https://futurestud.io/tutorials/gson-advanced-custom-serialization-part-1

15.8. About Google Gson

https://github.com/google/gson

- Provides simple `toJson()` and `fromJson()` methods to convert Java objects to JSON and vice-versa
- Allow pre-existing unmodifiable objects to be converted to and from JSON
- Extensive support of Java Generics

- Allow custom representations for objects
- Support arbitrarily complex objects (with deep inheritance hierarchies and extensive use of generic types)
- Easy post data for nested object that contains nested structure of json object and json arrays as a result of nested children

https://github.com/google/gson/blob/master/UserGuide.md
https://futurestud.io/tutorials/gson-getting-started-with-java-json-serialization-deserialization

15.9. Response to object

Tags: Convert response into object, myclass, data type, custom object, list of my object, list of particular data type, data type, list of custom data type

1. Use appropriate @SerializedName and @Expose annotation in your custom data type (object) class to map the fields from response
2. Note that you can define multiple @SerializedName for a particular field of an object in case there is a possibility where you get the different keys for the same field. For more info:
3. Make your object with default constructor (empty constructor, object with no state, no field)
4. Convert linkedTreeMap to JsonObject, Convert that JsonObject to a String.
5. Use parameterized type token and pass the name of the class of your custom data type (object)
6. Re-assign the empty object to gson.fromJson output - Done.

```
Public static Object getType (LinkedTreeMap linkedTreeMap, Object newEmptyObject, Class
classNameOfObject) {
      Gson gson = new Gson();
      String dataMap = Utils.getStringFromLinkedTreeMap(linkedTreeMap);
      Type type = TypeToken.getParameterized(classNameOfObject,
classNameOfObject).getType();
      gson.toJson(newEmptyObject, type);
      newEmptyObject = gson.fromJson(dataMap, type);
      return newEmptyObject;
}
```

Points to take care of:

- Do not cast gson or type into generic Object.

15.10. Gson converter

Tags: This is how gson converter works, google gson conversion

https://stackoverflow.com/questions/5128442/how-to-convert-a-string-to-jsonobject-using-gson-library/39636232

https://sites.google.com/site/gson/gson-user-guide#TOC-Primitives-Examples

```
Gson gson = new Gson(); YourClass yourClassObject = new YourClass(); String jsonString =
gson.toJson(yourClassObject);
```

15.11. LinkedTreeMap to String

Tags: Convert linkedTreeMap to string using google gson

https://stackoverflow.com/questions/38057172/how-do-i-convert-a-linkedtreemap-to-gson-jsonobject

https://stackoverflow.com/questions/17651395/convert-jsonobject-to-string

```
Gson gson = new Gson();
YourClass yourClassObject = new YourClass();
String jsonString = gson.toJson(yourClassObject);
```

OR

```
val jsonString:String = jsonObject.toString()
val gson:Gson = GsonBuilder().setPrettyPrinting().create()
val json:JsonElement = gson.fromJson(jsonString,JsonElement.class)
val jsonInString:String= gson.toJson(json) println(jsonInString)
```

15.12. LinkedTreeMap to JsonObject

https://stackoverflow.com/questions/38057172/how-do-i-convert-a-linkedtreemap-to-gson-jsonobject

```
LinkedTreeMap<?,?> yourMap = ...; JsonObject jsonObject =
gson.toJsonTree(yourMap).getAsJsonObject();
```

Note that, if you can, just deserialize the JSON directly to a `JsonObject` with

```
gson.fromJson(theJson, JsonObject.class);
```

15.13. gson Cast exception

https://stackoverflow.com/questions/32444863/google-gson-linkedtreemap-class-cast-to-myclass/32494079

```
Gson gson = new Gson();
JSONObject jsonObj = new JSONObject(object.get(DO_SERVICES).toString());
Type type = new TypeToken<MyDto>() {}.getType();
servDto = gson.fromJson(jsonObj.toString(),type);
```

In the above method, for the type token you can use below approach for common function (method):

15.14. Generic google gson Type token

Tags: get generic google gson type token from method function parameter argument

https://stackoverflow.com/questions/14139437/java-type-generic-as-argument-for-gson

```
TypeToken.getParameterized(List.class, myType).getType();
```

15.15. Convert child list to parent list:

```
List<Parent> parentList = new ArrayList<>(childList);
```

15.16. Convert list for similar fields (parent to child is possible!)

If two classes share common fields, you can use gson and type parameter to seamless transformation from one type of list to other type of list as below:

```
List<Car> carList = Utils.getTypeList(List<Vehicle> vehicleList, new Car(), Car.class)
```

Convert vehicleList into string, use Car.class for parameterized type and assign the return value as your expected variable.

This can also be applied as:

```
Car car = Utils.getTypeList(Vehicle vehicle, new Car(), Car.class)
```

You can convert any type of object into any other type of object as long as they share common fields.

15.17. Gson Expose Vs SearializedName

Tags: Gson Expose and SerializedName

@SerializedName to map the key from response
@Expose to explicitly include the field for serialization and deserialization

https://stackoverflow.com/questions/34752200/gson-expose-vs-serializedname

15.18. Multiple @SerializedName

Tags: Multiple Alternate Alternative @SerializedName
https://stackoverflow.com/questions/25707728/multiple-gson-serializedname-per-field

```
@SerializedName(value="name", alternate={"person", "user"})
Private String title;
```

As you can see, above gson will look for key *name, person or user* and once it finds any of them in response, it will map it to the field *title.*

15.19. Clone list

Java (at least up to 1.6) does not support deep copy of a list. There are basically two types of copy: One is known as shallow copy and other is deep copy. All the java methods for copy like copy constructor, collection, clone etc… do shallow copy. Shallow copy means though reference of both original and copied lists are different, underlying elements remain referenced. Hence, when we change any element in either original or copied list, the element of other list is also changed. However, by doing a deep copy, we can get two completely different lists where modifying any element in one list does not affect other list.

We can do deep copy in android java using Google gson.

https://stackoverflow.com/questions/715650/how-to-clone-arraylist-and-also-clone-its-contents

Read the answer given by sagits there:

```
List<CategoryModel> originalList = new ArrayList<>(); // add some items later
String listAsJson = gson.toJson(originalList);
List<CategoryModel> newList = new Gson().fromJson(listAsJson, new
TypeToken<List<CategoryModel>>() {}.getType());
```

We can use above concept and make general method as below:

```
Public static List cloneList(List list, Class classNameOfObjectType) {
      String listAsJson = new Gson().toJson(list);
      Return new Gson.fromJson(listAsJson, TypeToken.getParameterized(list,
classNameOfObjectType)).getType();
}
```

Use above method as below:

```
List<User> clonedList = new ArrayList<>(Utils.cloneList(List<User>
existingReferenceUserList, User.class))
```

Read more about copy clone issue in Java:

https://stackoverflow.com/questions/14742437/copy-of-arraylist-keeps-getting-modified-to-the-values-of-the-original

https://stackoverflow.com/questions/19572953/why-does-one-arraylist-change-when-a-copy-of-it-is-modified

https://stackoverflow.com/questions/15798268/how-to-copy-an-arraylist-to-not-reflecting-changes-in-the-original-list https://stackoverflow.com/questions/15778645/passing-arraylist-as-value-only-and-not-reference

Different ways of copying or cloning a list in android java:

https://stackoverflow.com/questions/27464781/creating-distinct-list-from-existing-list-in-java-7-and-8

https://www.journaldev.com/17129/java-deep-copy-object

https://stackoverflow.com/questions/7042182/how-to-make-a-deep-copy-of-java-arraylist

15.20. List to JsonArray

https://stackoverflow.com/questions/18857884/how-to-convert-arraylist-of-custom-class-to-jsonarray-in-java

```
List<Customer> customerList = CustomerDB.selectAll(); Gson gson = new Gson(); JsonElement
element = gson.toJsonTree(customerList, new TypeToken<List<Customer>>() {}.getType());
if (! element.isJsonArray()) {
// fail appropriately throw new SomeException();
}

JsonArray jsonArray = element.getAsJsonArray();
```

15.21. JsonArray to List

You can consider parameterized type below (Refer 15.14. Generic google gson Type token)

```
List<MyModel> myModelList = gson.fromJson(jsonArray.toString(), listType);
```

15.22. Remove key-value pair from JsonObject, JsonElement, JsonArray

https://stackoverflow.com/questions/23920740/remove-empty-collections-from-a-json-with-gson

https://stackoverflow.com/questions/47193364/using-gson-convert-java-object-into-jsonobject

16. Microsoft Outlook API Integration

Sample Projects:
https://github.com/Azure-Samples/ms-identity-android-native
https://github.com/Azure-Samples/active-directory-android
https://github.com/AzureAD/azure-activedirectory-library-for-android/wiki
https://developer.microsoft.com/en-us/graph
https://docs.microsoft.com/en-us/graph/outlook-mail-concept-overview

...but then we are redirected to previous-versions...
https://docs.microsoft.com/en-us/previous-versions/office/office-365-api/api/version-2.0/mail-rest-operations#GetMessages

....However, the explore link uses graph api instead of office for mail
https://developer.microsoft.com/en-us/graph/graph-explorer

Getting started:
After registration, you will get link of demo project:
https://docs.microsoft.com/en-us/outlook/rest/

Call outlook mail api:
https://docs.microsoft.com/en-us/outlook/rest/get-started#calling-the-mail-api

Outlook mail permissions:
https://docs.microsoft.com/en-us/graph/permissions-reference#mail-permissions

Manage permissions for your app:
https://docs.microsoft.com/en-us/graph/auth-v2-service#2-configure-permissions-for-microsoft-graph

List messages:
https://docs.microsoft.com/en-us/graph/api/user-list-messages?view=graph-rest-1.0&tabs=http

Various ops on mail:
https://docs.microsoft.com/en-us/graph/api/resources/message?view=graph-rest-1.0

Delete mail:
https://docs.microsoft.com/en-us/graph/api/message-delete?view=graph-rest-1.0&tabs=http

Send mail:
https://docs.microsoft.com/en-us/graph/api/user-sendmail?view=graph-rest-1.0&tabs=httphttps://docs.microsoft.com/en-us/graph/api/user-sendmail?view=graph-rest-1.0&tabs=http

Get a message:
https://docs.microsoft.com/en-us/previous-versions/office/office-365-api/api/version-2.0/mail-rest-operations

Send mail will require an object of type **message** to be sent as below:
https://docs.microsoft.com/en-us/graph/api/resources/message?view=graph-rest-1.0

Document for various api calls and their response:
https://docs.microsoft.com/en-us/graph/api/user-list-messages?view=graph-rest-1.0&tabs=http

Explore api calls through this graph:
https://developer.microsoft.com/en-us/graph/graph-explorer

16.1. Outlook

https://github.com/OfficeDev/Outlook-SDK-Android
https://developer.microsoft.com/en-us/graph
https://developer.microsoft.com/en-us/graph/gallery/?filterBy=Samples,SDKs,Android

16.2. Microsoft OpenSource Projects

Tags: sharePoint
https://github.com/SharePoint

16.3. Manage Permissions

401 Unauthorized
https://stackoverflow.com/questions/47870502/microsoft-graph-401-unauthorized-with-access-token
https://docs.microsoft.com/en-us/graph/auth-v2-service

17. Instagram

https://web.archive.org/web/20181106200753/https://github.com/TorzsaCristian/Instagram-API-Android-Tutorial

http://siteoftutorials.com/instagram-api-android-tutorial/

18. Video compressor

https://github.com/fishwjy/VideoCompressorhttps://github.com/fishwjy/VideoCompressor

19. Useful libraries

Useful libraries in android:_

https://github.com/codepath/android_guides/wiki/Must-Have-Libraries

https://github.com/wasabeef/awesome-android-ui

20. Custom Camera

https://github.com/CameraKit/camerakit-android

21. ExoPlayer

Simplest and easy to understand explanation by Ali
https://android.jlelse.eu/how-to-deal-with-exoplayer-67e528e2cbcc