**Author :**

Akshay Chavan

akshay.chavan@finiq.com

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Comma Separated Numbers:**

String.format("%,d", NumberFormat.getNumberInstance(Locale.US).parse(yourNumberString.trim()))

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Gradient Line:**

Drawable-

<?xml version="1.0" encoding="utf-8"?>

<shape xmlns:android="http://schemas.android.com/apk/res/android"

android:shape="rectangle"

>

<gradient

android:endColor="@android:color/transparent"

android:centerColor="@color/text\_color"

android:startColor="@android:color/transparent"

android:angle="0"

android:type="linear"/>

</shape>

XML-

<ImageView

android:id="@+id/hr"

android:layout\_width="match\_parent"

android:layout\_height="1dp"

android:background="@drawable/gradient\_line"

android:gravity="center\_horizontal"

/>

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Custom Arraylist Sorting:**

Collections.sort(sharesList, new Comparator<yourArrayListType>() {

@Override

public int compare(yourArrayListType item1, yourArrayListType item2) {

String s1 = item1.getfieldName();

String s2 = item2.getfieldName();

return s1.compareToIgnoreCase(s2);

}

});

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Textview with Marquee effect:**

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="0dp"

android:fadingEdge="horizontal"

android:singleLine="true"

android:ellipsize="marquee"

android:marqueeRepeatLimit="marquee\_forever"

android:padding="@dimen/text\_inner\_padding"

android:scrollHorizontally="true"

android:textSize="13dp" />

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Horizontal Line:**

<View

android:layout\_width="match\_parent"

android:layout\_height="1dp"

android:layout\_marginTop="10dp"

android:background="#494949" />

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Vertical Line:**

<View

android:layout\_width="0dp"

android:layout\_height="100dp"

android:layout\_gravity="center"

android:layout\_weight="0.01"

android:background="#494949" />

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Decimal Points Formatting for Number:**

String.format("%.4f", val);

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Decimal Points Formatting for an Input Field:**

@Override

public void afterTextChanged(Editable s) {

if (s.toString().length() != 0) {

et\_ko\_percentage.removeTextChangedListener(this);

int integerPlaces = s.toString().indexOf('.'); // 122.58 => 3

int decimalPlaces = s.toString().length() - integerPlaces - 1; // 6 - 3 - 1

// Log.e(TAG,"Integer Places: "+integerPlaces+" Decimal Places: "+decimalPlaces);

float koPerc = Float.parseFloat(s.toString());

//if (Float.parseFloat(String.format("%.2f", koPerc)) != koPerc)

if (s.toString().contains(".") && decimalPlaces > 2) {

et\_ko\_percentage.setText(String.format("%.2f", koPerc));

}

et\_ko\_percentage.setSelection(et\_ko\_percentage.getText().length());

et\_ko\_percentage.addTextChangedListener(this);

}

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Input field value formatting:**

et\_popup\_notional.addTextChangedListener(new TextWatcher() {

@Override

public void beforeTextChanged(CharSequence s, int start, int count, int after) {

}

@Override

public void onTextChanged(CharSequence s, int start, int before, int count) {

}

@Override

public void afterTextChanged(Editable s) {

// number formatting for nominal amount

Number nominal = null;

try {

// The comma in the format specifier does the trick

nominal = NumberFormat.getNumberInstance(java.util.Locale.US).parse(s.toString().trim());

Log.e(TAG, "Nominal->" + nominal);

} catch (NumberFormatException | ParseException e) {

}

// Set s back to the view after temporarily removing the text change listener

et\_popup\_notional.removeTextChangedListener(this);

et\_popup\_notional.setText(String.format("%,d", nominal));

et\_popup\_notional.setSelection(et\_popup\_notional.getText().length());

et\_popup\_notional.addTextChangedListener(this);

}

});

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**API Calls Optimization:**

public void saveArrayList(ArrayList<ShareListPojo> sharesList) {

SharedPreferences prefs = PreferenceManager.getDefaultSharedPreferences(AveragingAutocall.this);

SharedPreferences.Editor editor = prefs.edit();

Gson gson = new Gson();

String json = gson.toJson(sharesList);

editor.putString("SHARES\_LIST\_LOADED", json);

editor.apply();

MySingletonClass.getInstance().setSharesLoadedFlag(true);

} // end writeJsonToLocal

public ArrayList<ShareListPojo> getArrayList() {

Toast.makeText(this, "Loading previous shares list", Toast.LENGTH\_SHORT).show();

SharedPreferences prefs = PreferenceManager.getDefaultSharedPreferences(AveragingAutocall.this);

Gson gson = new Gson();

String json = prefs.getString("SHARES\_LIST\_LOADED", null);

Type listType = new TypeToken<ArrayList<ShareListPojo>>() {

}.getType();

// ArrayList<ShareListPojo> mSomeArraylist = gson.fromJson(json, listType);

return gson.fromJson(json, listType);

//Toast.makeText(this, "Test->"+mSomeArraylist.get(0).getLongName(),Toast.LENGTH\_SHORT).show();

} // end getArrayList()

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Synchronous Retrofit Call:**

try {

// ------------------- CurrentVal\_PortfolioRequest -------------------------

EQGetWatchListDataAPIRequest.put("WatchListName", this.watchlistName); // Default -> "Watchlist1"

EQGetWatchListDataAPIRequest.put("CustomerID", "32768");

EQGetWatchListDataAPIRequest.put("EntityID", "4");

EQGetWatchListDataAPIRequest.put("LoginID", "Dealer1");

// ------------------- mainObj -------------------------

mainObj.put("FinIQRequestHeader", FinIQRequestHeader);

mainObj.put("EQGetWatchListDataAPIRequest", EQGetWatchListDataAPIRequest);

Log.e(TAG, "Get Watchlist Product Data Shares API Input parameters -> " + mainObj.toString());

ApiInterface apiInterface = ApiClient.getClient().create(ApiInterface.class);

Call<GetWatchlistDataSharesPojo> call = apiInterface.getWatchlistDataShares(mainObj.toString());

// ----------------------------------- Synchronous Call ---------------------------------------------

StrictMode.ThreadPolicy policy = new StrictMode.ThreadPolicy.Builder().permitAll().build();

StrictMode.setThreadPolicy(policy);

Response<GetWatchlistDataSharesPojo> response = call.execute();

GetWatchlistDataSharesPojo getWatchlistDataSharesPojo = response.body();

for (GetWatchlistDataResponseItem item : getWatchlistDataSharesPojo.getEQGetWatchListDataAPIResponse().getItems()) {

callGetProductInfoSharesAPI(item.getNoteMasterId());

}

} catch (IOException | JSONException e) {

Toast.makeText(getContext(), "TRY CATCH ERROR!\n>>" + e.getMessage(), Toast.LENGTH\_SHORT).show();

}