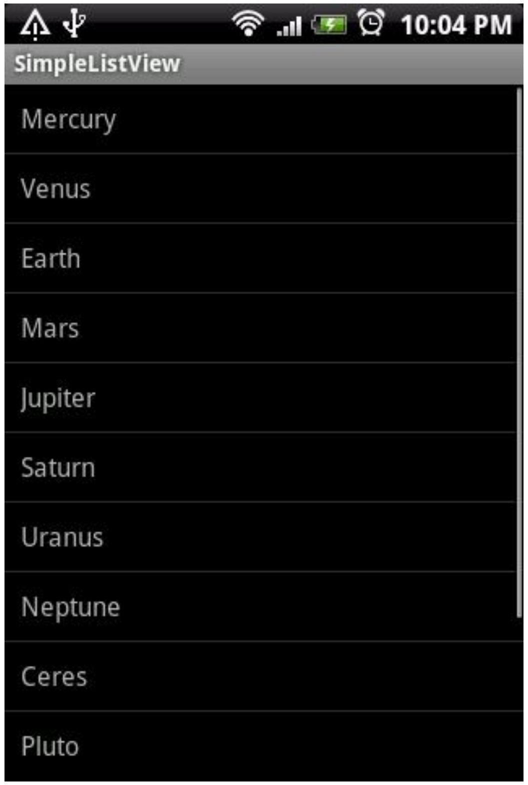
Android Application Development

Session: 19

Recycleview

“Recyclerview is a view group that displays a list of scrollable items”

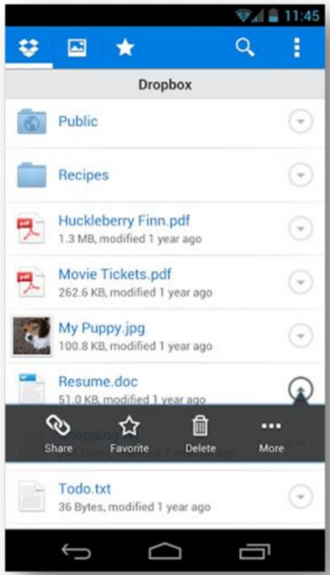
Listview



10:04 PM

SimpleListView

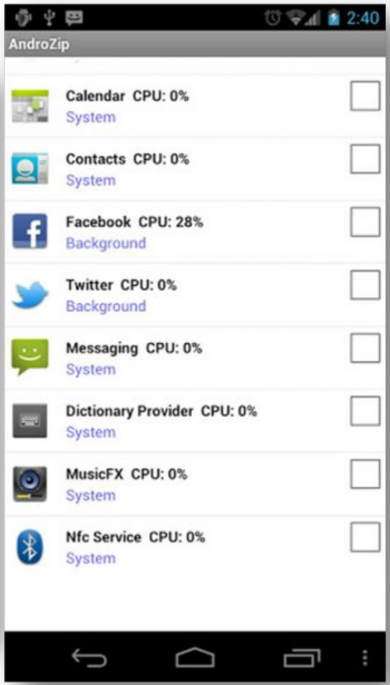
- Mercury
- Venus
- Earth
- Mars
- Jupiter
- Saturn
- Uranus
- Neptune
- Ceres
- Pluto



11:45

Dropbox

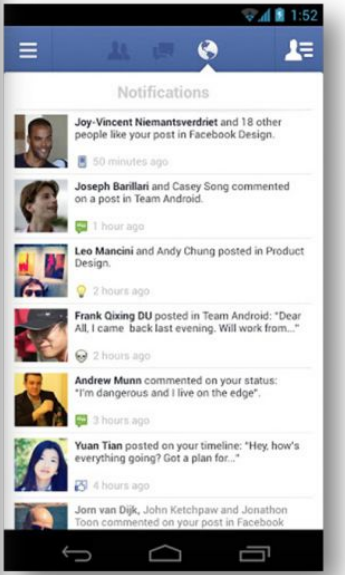
- Public
- Recipes
- Huckleberry Finn.pdf
  1.3 MB, modified 1 year ago
- Movie Tickets.pdf
  262.6 KB, modified 1 year ago
- My Puppy.jpg
  100.8 KB, modified 1 year ago
- Resume.doc
  51.0 KB, modified 1 year ago
- Todo.txt
  36 Bytes, modified 1 year ago



2:40

AndroZip

- Calendar CPU: 0%
  System
- Contacts CPU: 0%
  System
- Facebook CPU: 28%
  Background
- Twitter CPU: 0%
  Background
- Messaging CPU: 0%
  System
- Dictionary Provider CPU: 0%
  System
- MusicFX CPU: 0%
  System
- Nfc Service CPU: 0%
  System



1:52

Notifications

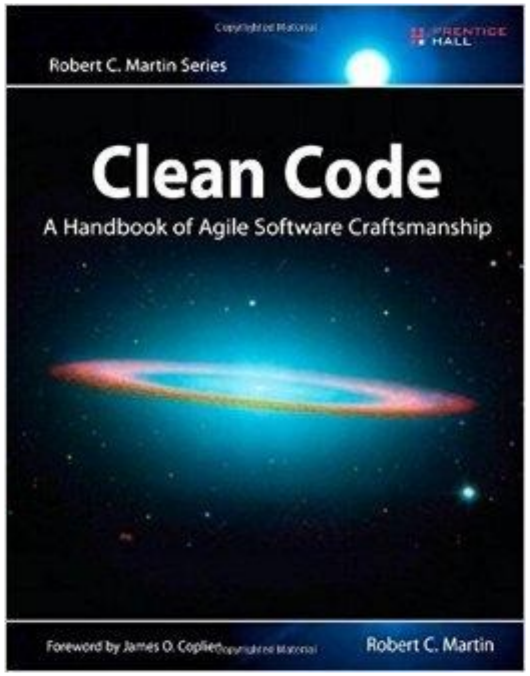
- **Joy-Vincent Niemantsverdriet** and 18 other people like your post in Facebook Design.
  50 minutes ago
- **Joseph Barillari** and Casey Song commented on a post in Team Android.
  1 hour ago
- **Leo Mancini** and Andy Chung posted in Product Design.
  2 hours ago
- **Frank Qixing DU** posted in Team Android: "Dear All, I came back last evening. Will work from..."
  2 hours ago
- **Andrew Munn** commented on your status: "I'm dangerous and I live on the edge".
  3 hours ago
- **Yuan Tian** posted on your timeline: "Hey, how's everything going? Got a plan for..."
  4 hours ago
- **Jorn van Dijk,** John Ketchpaw and Jonathon Toon commented on your post in Facebook

Why Google ?

Why Google ?

- Listview codebase so complex
- Duplicate functionality
    - Itemclicklistener vs onClickListener
- Hard to create animation
- And others (Google IO 2016)

  https://www.youtube.com/watch?v=LqBlYJTfLP4


Robert C. Martin Series
PRENTICE HALL
Clean Code
A Handbook of Agile Software Craftsmanship
Foreword by James O. Coplien
Robert C. Martin

Setup Recyclerview

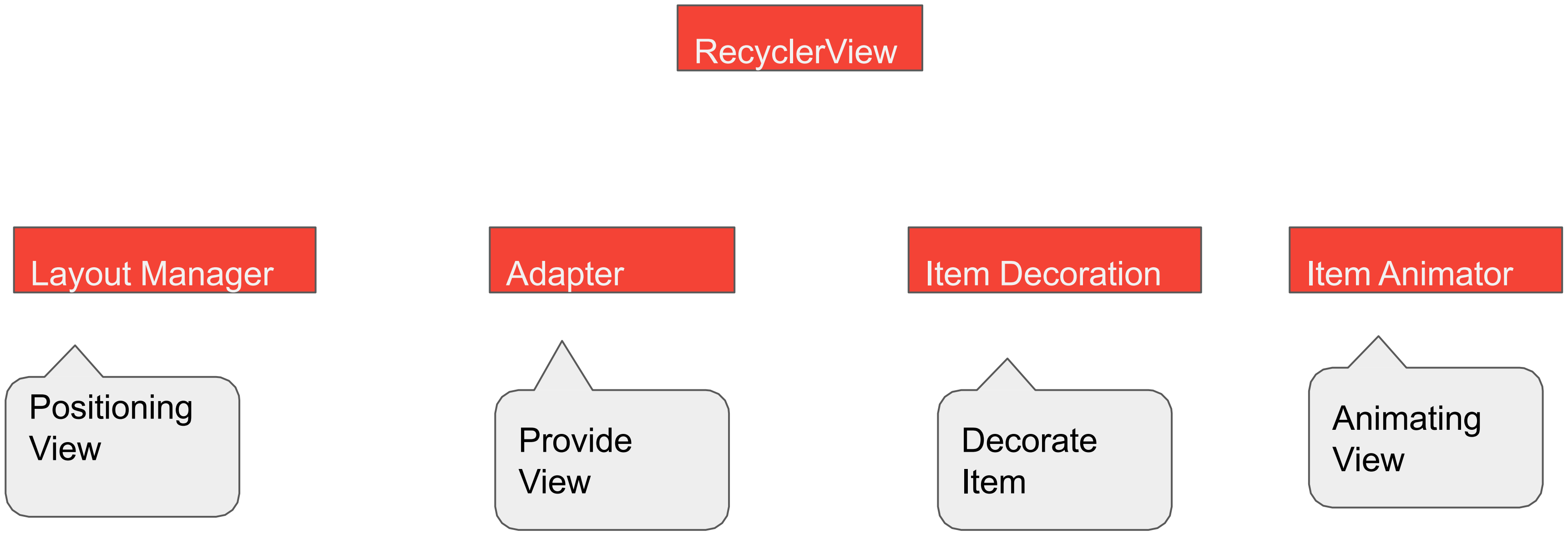
1. app/build.gradle

```
dependencies {
 compile 'com.android.support:appcompat-v7:23.4.0'
 compile 'com.android.support:design:23.4.0' // recyclerview-v7
}
```

2. Layout

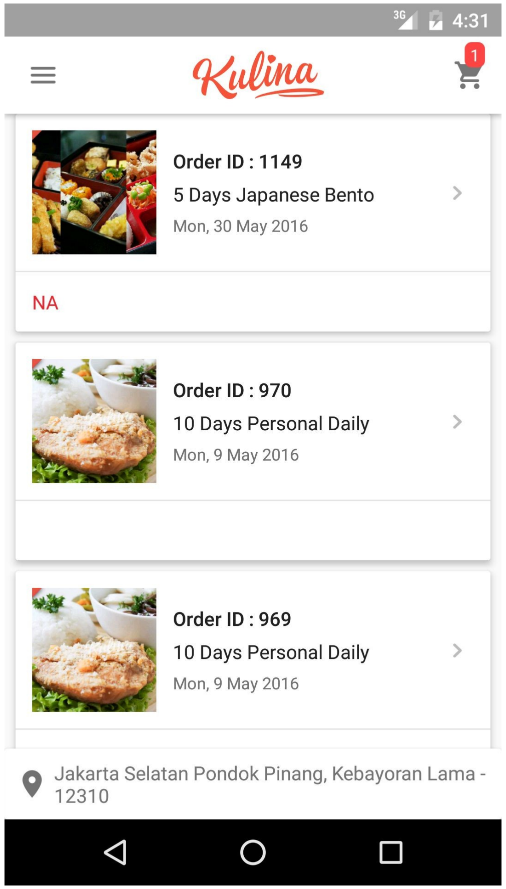
```

<android.support.v7.widget.RecyclerView
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   />
```

RecyclerView Component



Layout Manager

LinearLayoutManager



3G 4:31

Kulina

Order ID : 1149
5 Days Japanese Bento
Mon, 30 May 2016

NA

Order ID : 970
10 Days Personal Daily
Mon, 9 May 2016

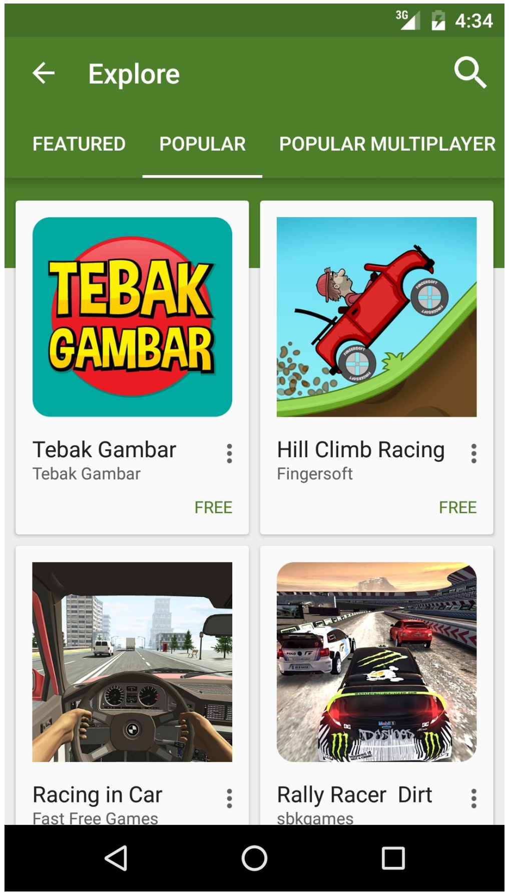
Order ID : 969
10 Days Personal Daily
Mon, 9 May 2016

Jakarta Selatan Pondok Pinang, Kebayoran Lama - 12310

```
layoutManagerVertical = new LinearLayoutManager(context);

recyclerView.setLayoutManager(layoutManagerVertical);
```

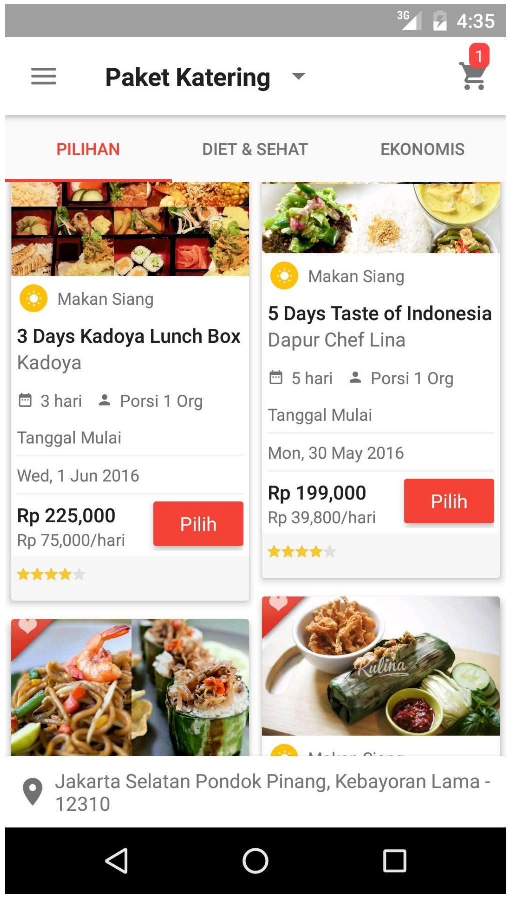
```
layoutManagerHorizontal =
   new LinearLayoutManager(context, LinearLayoutManager.
HORIZONTAL, false);

recyclerView.setLayoutManager(layoutManagerHorizontal);
```

GridlayoutManager



```
gridLayoutManager = new GridLayoutManager(context, SPAN_COUNT);
recyclerView.setLayoutManager(gridLayoutManager);
```

StaggeredLayoutManager


4:35

Paket Katering

PILIHAN
DIET & SEHAT
EKONOMIS

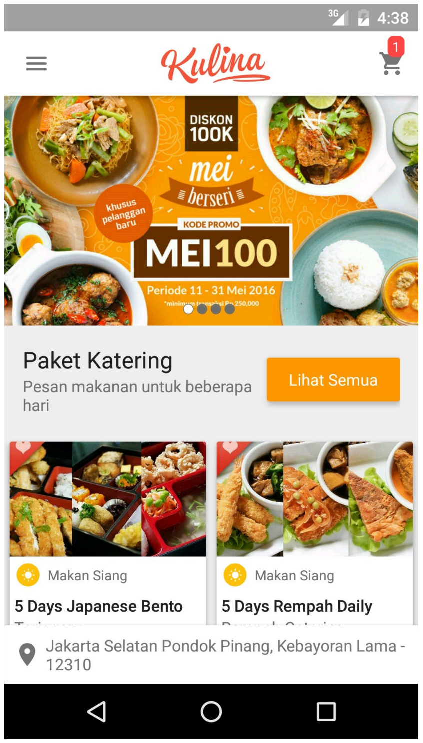
Makan Siang
3 Days Kadoya Lunch Box
Kadoya
3 hari Porsi 1 Org
Tanggal Mulai
Wed, 1 Jun 2016
Rp 225,000
Rp 75,000/hari
Pilih

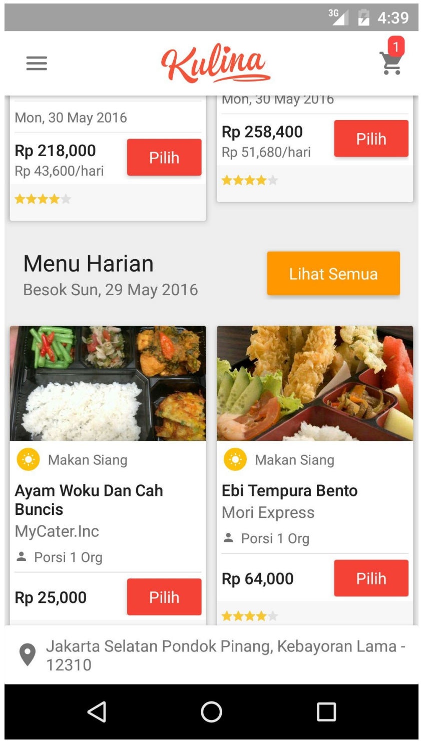
Makan Siang
5 Days Taste of Indonesia
Dapur Chef Lina
5 hari Porsi 1 Org
Tanggal Mulai
Mon, 30 May 2016
Rp 199,000
Rp 39,800/hari
Pilih

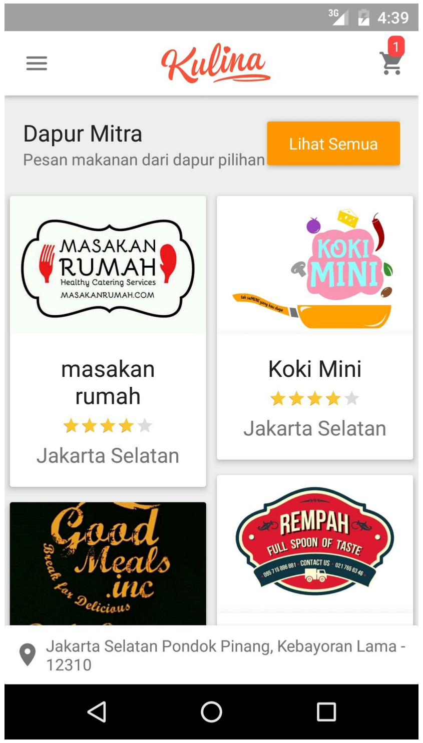
Jakarta Selatan Pondok Pinang, Kebayoran Lama - 12310

```
gridLayoutManager = new StaggeredGridLayoutManager(2,
StaggeredGridLayoutManager.VERTICAL);
recyclerView.setLayoutManager(gridLayoutManager);
```

Multitype Item Layout


4:38
Kulina
Paket Katering
Pesan makanan untuk beberapa hari
Lihat Semua
Makan Siang
5 Days Japanese Bento
Makan Siang
5 Days Rempah Daily
Jakarta Selatan Pondok Pinang, Kebayoran Lama - 12310


4:39
Kulina
Mon, 30 May 2016
Rp 218,000
Rp 43,600/hari
Pilih
Mon, 30 May 2016
Rp 258,400
Rp 51,680/hari
Pilih
Menu Harian
Besok Sun, 29 May 2016
Lihat Semua
Makan Siang
Ayam Woku Dan Cah Buncis
MyCater.Inc
Porsi 1 Org
Rp 25,000
Pilih
Makan Siang
Ebi Tempura Bento
Mori Express
Porsi 1 Org
Rp 64,000
Pilih
Jakarta Selatan Pondok Pinang, Kebayoran Lama - 12310


4:39
Kulina
Dapur Mitra
Pesan makanan dari dapur pilihan
Lihat Semua
masakan rumah
Jakarta Selatan
Koki Mini
Jakarta Selatan
Jakarta Selatan Pondok Pinang, Kebayoran Lama - 12310

Adapter

Adapter Component

- Create View and Viewholder
- Bind item to ViewHolder
- Notify Recyclerview about changes
- Item Interaction handling (click, etc)
- Multiple view types

Sample Adapter Clas

```
public class LinearLayoutAdapter extends RecyclerView.Adapter<LinearLayoutAdapter.
LinearLayoutViewHolder>{

 @Override public LinearLayoutViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
   return null;
 }

 @Override public void onBindViewHolder(LinearLayoutViewHolder holder, int position) {}

 @Override public int getItemCount() {return 0;}


 public class LinearLayoutViewHolder extends RecyclerView.ViewHolder{
   public LinearLayoutViewHolder(View itemView) { super(itemView); }
 }
}
```

Viewholder Class

```java
public class LinearHolder extends RecyclerView.ViewHolder {

 @BindView(R.id.img) ImageView img;
 @BindView(R.id.name) TextView name;
 @BindView(R.id.location) TextView location;

 public LinearHolder(View itemView) {
   super(itemView);
   ButterKnife.bind(this, itemView);
 }

 public void bindItem(Mountain mountain) {
   name.setText(mountain.name);
   location.setText(mountain.location);
   ImageLoader.LoadImage(context, img, mountain.img);
 }
}
```

Create View and Bind View

```
@Override public LinearHolder onCreateViewHolder(ViewGroup parent, int viewType) {
 return new LinearHolder(
     LayoutInflater.from(context).inflate(R.layout.item_linear_vertical, parent, false));
}

@Override public void onBindViewHolder(LinearHolder holder, int position) {
 if (mountainList != null && mountainList.size() > 0) {
   holder.bindItem(mountainList.get(position));
 }
}
```

Handle Click Listener

Create interface

```java
public interface ClickListener {
 void onItemClick(int pos);
}
```

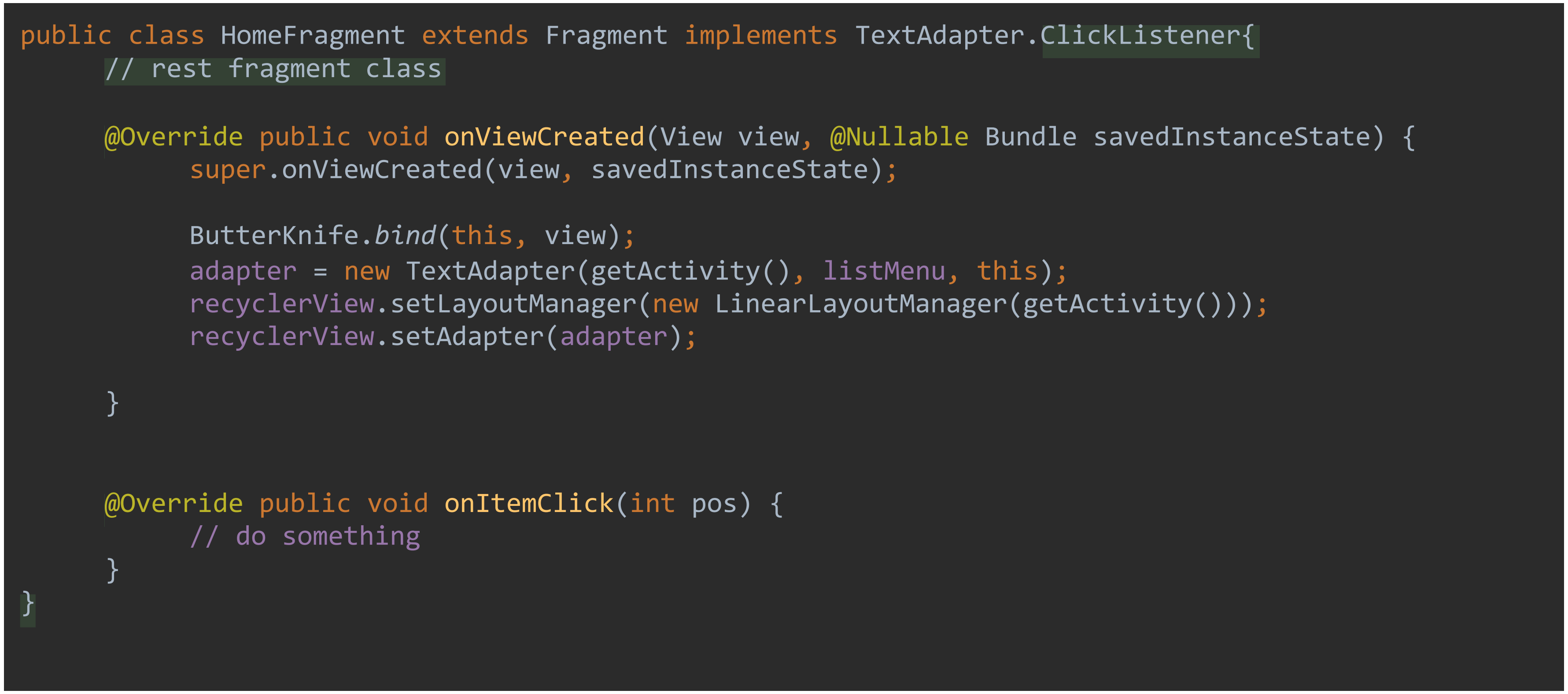
Pass Listener into Adapter

```
public interface ClickListener {
 void onItemClick(int pos);
}

private Context context;
private List<String> strings;
private ClickListener listener;

public TextAdapter(Context context, List<String> strings, ClickListener listener)
 {   this.context = context;
 this.strings = strings;
 this.listener = listener;
}
```

Set Clicklistener at ViewHolder

```
public class MainHolder extends RecyclerView.ViewHolder {

 @BindView(R.id.txt_title) TextView txtTitle;
 @BindView(R.id.container_text) LinearLayout container;

 public MainHolder(View itemView) {
   super(itemView);
   ButterKnife.bind(this, itemView);
 }

 public void bindData(String string, final int pos) {
   txtTitle.setText("" + string);
   container.setOnClickListener(new View.OnClickListener() {
     @Override public void onClick(View view) {
       listener.onItemClick(pos);
     }
   });
 }
}
```

Implement Listener into Activity/Fragment



```
public class HomeFragment extends Fragment implements TextAdapter.ClickListener{
    // rest fragment class

    @Override public void onViewCreated(View view, @Nullable Bundle savedInstanceState) {
        super.onViewCreated(view, savedInstanceState);

        ButterKnife.bind(this, view);
        adapter = new TextAdapter(getActivity(), listMenu, this);
        recyclerView.setLayoutManager(new LinearLayoutManager(getActivity()));
        recyclerView.setAdapter(adapter);

    }


    @Override public void onItemClick(int pos) {
        // do something
    }
}
```

Item Decoration

Simple Item Decoration

```java
public class SimpleDividerItemDecoration extends RecyclerView.ItemDecoration {
 private Drawable mDivider;

 public SimpleDividerItemDecoration(Context context) {
   mDivider = ContextCompat.getDrawable(context, R.drawable.line_divider);
 }

 @Override public void onDrawOver(Canvas c, RecyclerView parent, RecyclerView.State state) {
   int left = parent.getPaddingLeft();
   int right = parent.getWidth() - parent.getPaddingRight();

   int childCount = parent.getChildCount();
   for (int i = 0; i < childCount; i++) {
     View child = parent.getChildAt(i);

     RecyclerView.LayoutParams params = (RecyclerView.LayoutParams) child.getLayoutParams();

     int top = child.getBottom() + params.bottomMargin;
     int bottom = top + mDivider.getIntrinsicHeight();

     mDivider.setBounds(left, top, right, bottom);
     mDivider.draw(c);
   }}}
```

Simple Item Decoration

```
recyclerView.addItemDecoration(new SimpleDividerItemDecoration(getActivity()));
```

Item Animator

Simple Item Animator

```
recyclerView.setItemAnimator(new DefaultItemAnimator());
```

Other item animator ?

Take a look

https://github.com/wasabeef/recyclerview-animators ?;)

Multitype Viewholder

Multiviewtype Holder

```java
@Override public int getItemViewType(int position) {
 if (position == 0) {
   return VIEW_TYPE_HEADER;
 } else {
   return VIEW_TYPE_ITEM;
 }
}

@Override public void onBindViewHolder(RecyclerView.ViewHolder holder, int position) {
 if (holder.getItemViewType() == VIEW_TYPE_HEADER) {

   HeaderAdapterHolder headerHolder = (HeaderAdapterHolder) holder;

 }else if (holder.getItemViewType() == VIEW_TYPE_ITEM) {

   ItemAdapterHolder itemHolder = (ItemAdapterHolder) holder;


 }
}
```

Discuss and QA