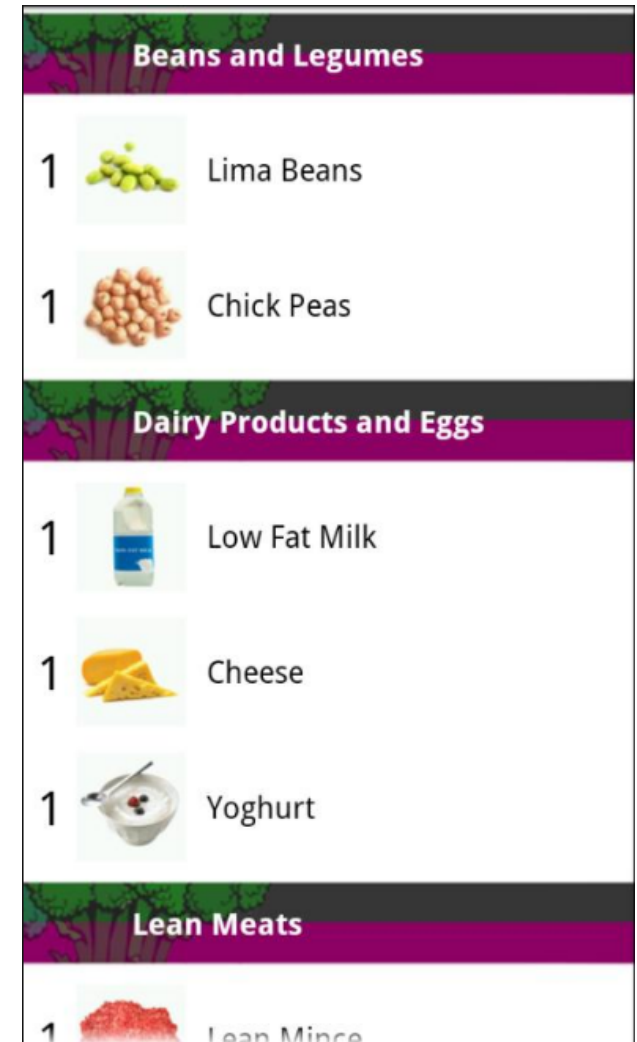


Разработка мобильных приложений

Android

Проблемы при создании больших списков

- Элементов в списке может быть больше 100, а иногда и больше 1 000 000
 - Каждый элемент может содержать объемную информацию (например, картинки)
 - Каждый элемент должен обрабатывать нажатия
- Наивные способы реализации:
 - Создавать элементы списка по мере необходимости (когда появляются на экране)
 - Создать все элементы списка сразу

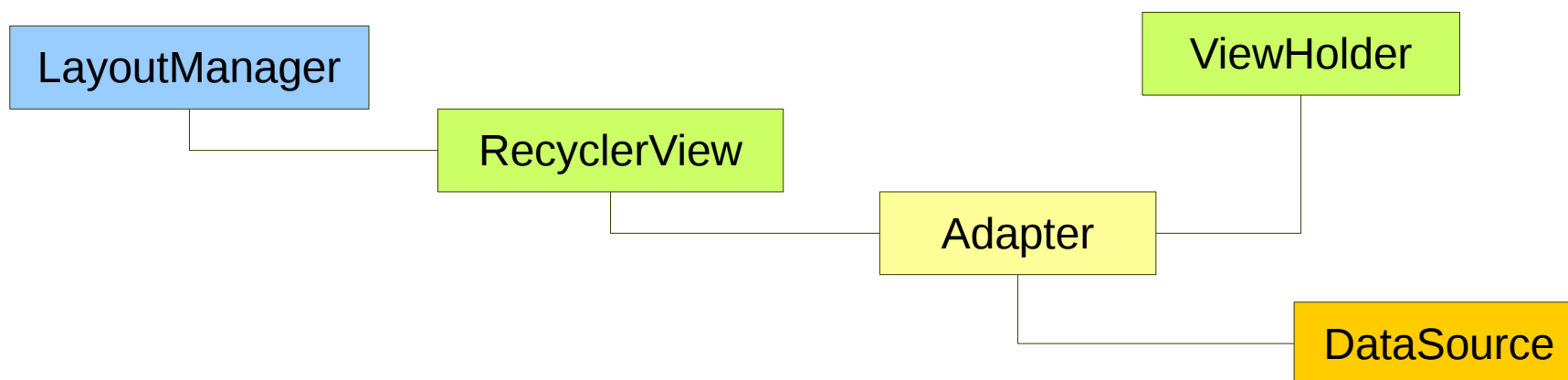


RecyclerView

- Позволяет эффективно представлять наборы из большого количество элементов

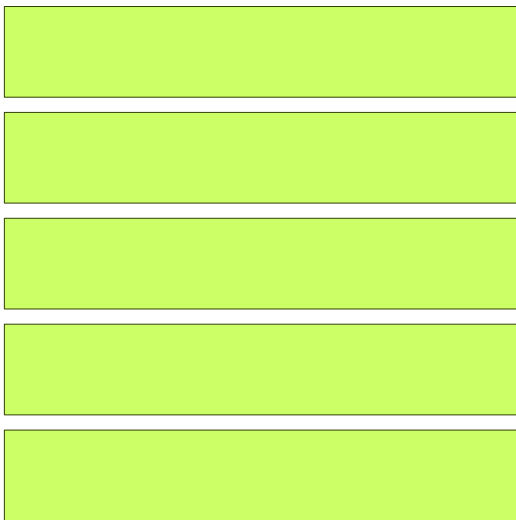
Например: списки сообщений в социальных сетях, списки писем в почтовых ящиках, ленты новостей, списки товаров в интернет-магазинах

- Общая схема работы

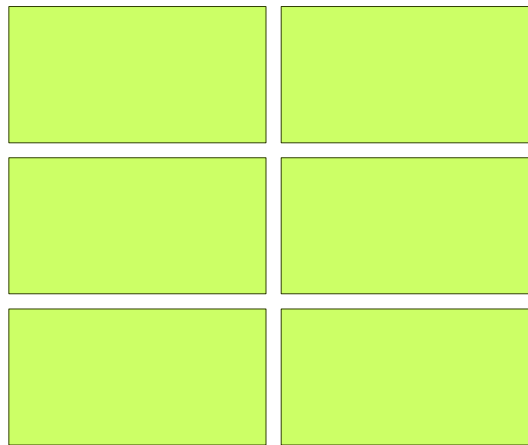


Варианты размещения элементов RecyclerView

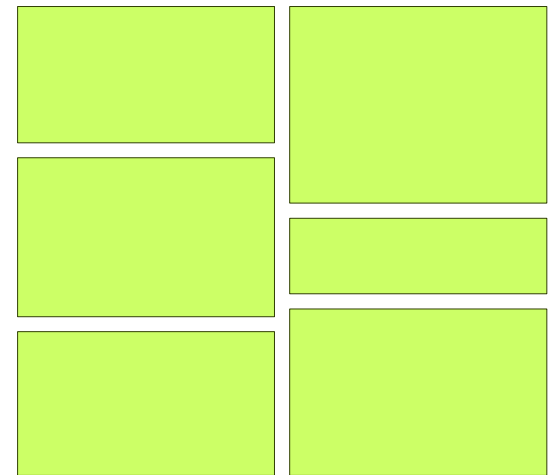
LinearLayoutManger



GridLayoutManger



StaggeredGridLayoutManger



- Либо реализовать свой LayoutManger

Применение RecyclerView

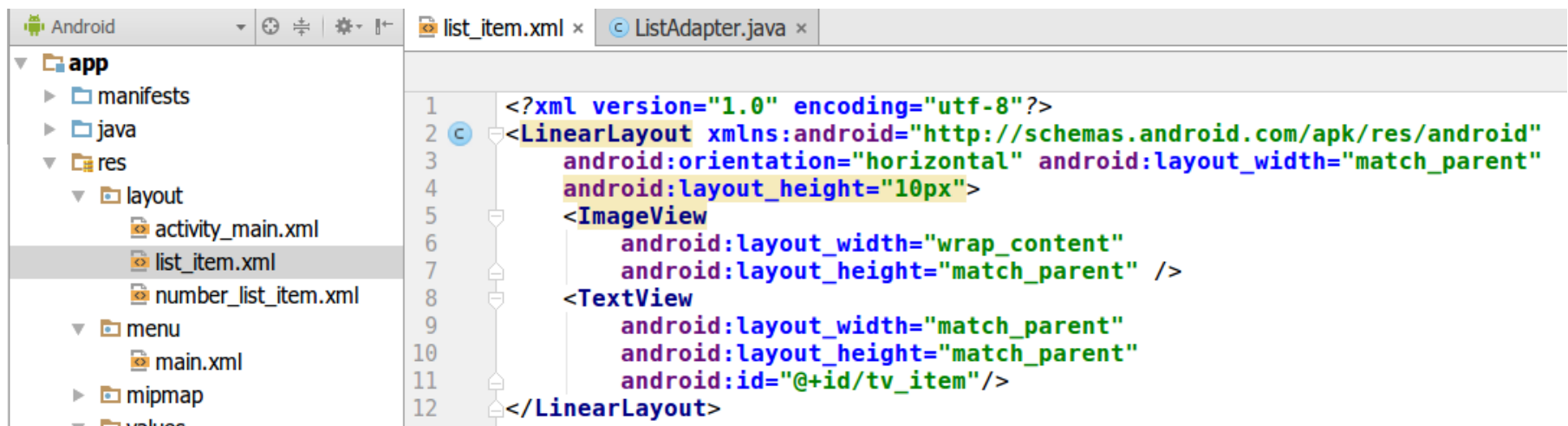
1. Создать RecyclerView в файле с layout, добавить зависимость в app/build.gradle

```
<android.support.v7.widget.RecyclerView  
    android:id="@+id/rv_items"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"/>
```

```
dependencies {  
    ...  
    compile 'com.android.support:recyclerview-v7:25.1.0'  
}
```

Применение RecyclerView

2. Создать файл layout и класс ViewHolder, задающие содержимое одного элемента объекта RecyclerView



```
class ElemViewHolder extends RecyclerView.ViewHolder {
    TextView listItemView;
    int index;

    public ElemViewHolder(View itemView) {
        super(itemView);
        listItemView = (TextView) itemView.findViewById(R.id.tv_item);
    }

    public void setData(String data) { listItemView.setText(data); }
}
```

Применение RecyclerView

3. Создать адаптер

```
public class ListAdapter extends RecyclerView.Adapter<ListAdapter.ElemViewHolder> {  
  
    @Override  
    public ElemViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {  
        return null;  
    }  
  
    @Override  
    public void onBindViewHolder(ElemViewHolder holder, int position) {  
  
    }  
  
    @Override  
    public int getItemCount() { return 0; }  
  
    class ElemViewHolder extends RecyclerView.ViewHolder {  
  
        public ElemViewHolder(View itemView) {  
            super(itemView);  
        }  
    }  
}
```

Создает новый объект ViewHolder

Заполняет объект данными

Возвращает количество элементов

Применение RecyclerView

```
public class ListAdapter extends RecyclerView.Adapter<ListAdapter.ElemViewHolder> {

    private static int countItems;

    public ListAdapter() { countItems = 0; }

    @Override
    public ElemViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
        Context context = parent.getContext();
        int layoutIdForListItem = R.layout.list_item;
        LayoutInflater inflater = LayoutInflater.from(context);
        boolean shouldAttachToParentImmediately = false;

        View view = inflater.inflate(layoutIdForListItem, parent, shouldAttachToParentImmediately);
        ListAdapter.ElemViewHolder viewHolder = new ListAdapter.ElemViewHolder(view);

        viewHolder.setData("ViewHolder index: " + countItems);
        countItems++;

        return viewHolder;
    }

    @Override
    public void onBindViewHolder(ElemViewHolder holder, int position) {
        holder.setData("item index changed:" + position);
    }

    @Override
    public int getItemCount() { return 50; }

    class ElemViewHolder extends RecyclerView.ViewHolder {
        TextView listItemView;
```


Применение RecyclerView

4. Добавить LayoutManager и объединить реализованные части

```
private RecyclerView mItemsList;  
private ListAdapter mListAdapter;  
  
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
  
    mItemsList = (RecyclerView) findViewById(R.id.rv_items);  
  
    LinearLayoutManager layoutManager =  
        new LinearLayoutManager(this);  
    mItemsList.setLayoutManager(layoutManager);  
  
    mListAdapter = new ListAdapter();  
    mItemsList.setAdapter(mListAdapter);  
  
    mItemsList.setHasFixedSize(true);
```



Лабораторная работа 3.

Выгруженные из интернета данные (во второй лабораторной) представить в виде списка элементов, которые обрабатываются с помощью RecyclerView

* добавить действие при нажатии на элемент в RecyclerView