Demo13-RecyclerView最基本的Demo

1. 在app模块的build.gradle文件添加recyclerview的依赖

implementation **'com.android.support:recyclerview-v7:28.0.0'**

1. activity\_main.xml

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

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

**android:layout\_width="match\_parent"**

**android:layout\_height="match\_parent">**

**<android.support.v7.widget.RecyclerView**

**android:id="@+id/recycler\_view"**

**android:layout\_width="match\_parent"**

**android:layout\_height="match\_parent" />**

**</LinearLayout>**

3、fruit\_item.xml

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

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

**android:orientation="vertical"**

**android:layout\_width="match\_parent"**

**android:layout\_height="wrap\_content"**

**android:layout\_margin="5dp" >**

**<ImageView**

**android:id="@+id/fruit\_image"**

**android:layout\_width="wrap\_content"**

**android:layout\_height="wrap\_content"**

**android:layout\_gravity="center\_horizontal" />**

**<TextView**

**android:id="@+id/fruit\_name"**

**android:layout\_width="wrap\_content"**

**android:layout\_height="wrap\_content"**

**android:layout\_gravity="left"**

**android:layout\_marginTop="10dp" />**

**</LinearLayout>**

4、bean: Fruit.java 解析后存放数据

//属性+Getter+Setter

public class Fruit {

private String name;

private int imageId;

public Fruit(String name, int imageId) {

this.name = name;

this.imageId = imageId;

}

public String getName() {

return name;

}

public int getImageId() {

return imageId;

}

}

5、适配器： 数据整理类，将数据整理到API，setAdapter传入适配器对象，给系统自己回调，我们只要负责构造就行

import android.support.v7.widget.RecyclerView;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.ImageView;

import android.widget.TextView;

import android.widget.Toast;

import java.util.List;

public class FruitAdapter extends RecyclerView.Adapter<FruitAdapter.ViewHolder>{

//适配器要整理的数据

private List<Fruit> mFruitList;

public FruitAdapter(List<Fruit> fruitList) {

mFruitList = fruitList;

}

@Override

public ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {

View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.fruit\_item, parent, false);

final ViewHolder holder = new ViewHolder(view);

//灵活设置RecyclerView每个view的点击事件，这是比ListView的优势

holder.fruitView.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

int position = holder.getAdapterPosition();

Fruit fruit = mFruitList.get(position);

Toast.makeText(v.getContext(), "you clicked view " + fruit.getName(), Toast.LENGTH\_SHORT).show();

}

});

holder.fruitImage.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

int position = holder.getAdapterPosition();

Fruit fruit = mFruitList.get(position);

Toast.makeText(v.getContext(), "you clicked image " + fruit.getName(), Toast.LENGTH\_SHORT).show();

}

});

return holder;

}

@Override

public void onBindViewHolder(ViewHolder holder, int position) {

Fruit fruit = mFruitList.get(position);

holder.fruitImage.setImageResource(fruit.getImageId());

holder.fruitName.setText(fruit.getName());

}

@Override

public int getItemCount() {

return mFruitList.size();

}

//内部类ViewHolder的作用就是初始化itemView

static class ViewHolder extends RecyclerView.ViewHolder {

View fruitView;

ImageView fruitImage;

TextView fruitName;

public ViewHolder(View view) {

super(view);

fruitView = view;

fruitImage = (ImageView) view.findViewById(R.id.fruit\_image);

fruitName = (TextView) view.findViewById(R.id.fruit\_name);

}

}

}

6、MainActivity

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.support.v7.widget.LinearLayoutManager;

import android.support.v7.widget.RecyclerView;

import android.support.v7.widget.StaggeredGridLayoutManager;

import java.util.ArrayList;

import java.util.List;

import java.util.Random;

public class MainActivity extends AppCompatActivity {

private List<Fruit> fruitList = new ArrayList<Fruit>();

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

//initView

RecyclerView recyclerView = (RecyclerView) findViewById(R.id.recycler\_view);

// StaggeredGridLayoutManager layoutManager = new

// StaggeredGridLayoutManager(3, StaggeredGridLayoutManager.VERTICAL);

// recyclerView.setLayoutManager(layoutManager);

recyclerView.setLayoutManager(new LinearLayoutManager(getApplicationContext()));

//initData

initFruits();

FruitAdapter adapter = new FruitAdapter(fruitList);

recyclerView.setAdapter(adapter);

}

private void initFruits() {

for (int i = 0; i < 2; i++) {

Fruit apple = new Fruit(getRandomLengthName("Apple"), R.drawable.apple\_pic);

fruitList.add(apple);

Fruit banana = new Fruit(getRandomLengthName("Banana"), R.drawable.banana\_pic);

fruitList.add(banana);

Fruit orange = new Fruit(getRandomLengthName("Orange"), R.drawable.orange\_pic);

fruitList.add(orange);

Fruit watermelon = new Fruit(getRandomLengthName("Watermelon"), R.drawable.watermelon\_pic);

fruitList.add(watermelon);

Fruit pear = new Fruit(getRandomLengthName("Pear"), R.drawable.pear\_pic);

fruitList.add(pear);

Fruit grape = new Fruit(getRandomLengthName("Grape"), R.drawable.grape\_pic);

fruitList.add(grape);

Fruit pineapple = new Fruit(getRandomLengthName("Pineapple"), R.drawable.pineapple\_pic);

fruitList.add(pineapple);

Fruit strawberry = new Fruit(getRandomLengthName("Strawberry"), R.drawable.strawberry\_pic);

fruitList.add(strawberry);

Fruit cherry = new Fruit(getRandomLengthName("Cherry"), R.drawable.cherry\_pic);

fruitList.add(cherry);

Fruit mango = new Fruit(getRandomLengthName("Mango"), R.drawable.mango\_pic);

fruitList.add(mango);

}

}

private String getRandomLengthName(String name) {

Random random = new Random();

int length = random.nextInt(20) + 1;

StringBuilder builder = new StringBuilder();

for (int i = 0; i < length; i++) {

builder.append(name);

}

return builder.toString();

}

}