Demo14-RecyclerView实现聊天界面

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:orientation="vertical"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="#d8e0e8" >

<android.support.v7.widget.RecyclerView

android:id="@+id/msg\_recycler\_view"

android:layout\_width="match\_parent"

android:layout\_height="0dp"

android:layout\_weight="1" />

<LinearLayout

android:layout\_width="match\_parent"

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

<EditText

android:id="@+id/input\_text"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:hint="Type something here"

android:maxLines="2" />

<Button

android:id="@+id/send"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Send" />

</LinearLayout>

</LinearLayout>

1. recyclerview的item布局： msg\_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:padding="10dp" >

<LinearLayout

android:id="@+id/left\_layout"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="left"

android:background="@drawable/message\_left" >

<TextView

android:id="@+id/left\_msg"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:layout\_margin="10dp"

android:textColor="#fff" />

</LinearLayout>

<LinearLayout

android:id="@+id/right\_layout"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="right"

android:background="@drawable/message\_right" >

<TextView

android:id="@+id/right\_msg"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:layout\_margin="10dp" />

</LinearLayout>

</LinearLayout>

1. bean: 存储解析后的数据，是一个实体类

public class Msg {

public static final int TYPE\_RECEIVED = 0;

public static final int TYPE\_SENT = 1;

private String content;

private int type;

public Msg(String content, int type) {

this.content = content;

this.type = type;

}

public String getContent() {

return content;

}

public int getType() {

return type;

}

}

1. 适配器： 数据整理类，将数据整理到API中，setAdapter将适配器对象传入系统，让系统自己根据监听结果来回调适配器的API

import android.support.v7.widget.RecyclerView;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.LinearLayout;

import android.widget.TextView;

import java.util.List;

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

private List<Msg> mMsgList;

public MsgAdapter(List<Msg> msgList) {

mMsgList = msgList;

}

@Override

public ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {

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

return new ViewHolder(view);

}

@Override

public void onBindViewHolder(ViewHolder holder, int position) {

Msg msg = mMsgList.get(position);

if (msg.getType() == Msg.TYPE\_RECEIVED) {

// 如果是收到的消息，则显示左边的消息布局，将右边的消息布局隐藏

holder.leftLayout.setVisibility(View.VISIBLE);

holder.rightLayout.setVisibility(View.GONE);

holder.leftMsg.setText(msg.getContent());

} else if(msg.getType() == Msg.TYPE\_SENT) {

// 如果是发出的消息，则显示右边的消息布局，将左边的消息布局隐藏

holder.rightLayout.setVisibility(View.VISIBLE);

holder.leftLayout.setVisibility(View.GONE);

holder.rightMsg.setText(msg.getContent());

}

}

@Override

public int getItemCount() {

return mMsgList.size();

}

//初始化RecyclerView的itemView

static class ViewHolder extends RecyclerView.ViewHolder {

LinearLayout leftLayout;

LinearLayout rightLayout;

TextView leftMsg;

TextView rightMsg;

public ViewHolder(View view) {

super(view);

leftLayout = (LinearLayout) view.findViewById(R.id.left\_layout);

rightLayout = (LinearLayout) view.findViewById(R.id.right\_layout);

leftMsg = (TextView) view.findViewById(R.id.left\_msg);

rightMsg = (TextView) view.findViewById(R.id.right\_msg);

}

}

}

1. MainActivity

import android.os.Bundle;

import android.support.v7.app.AppCompatActivity;

import android.support.v7.widget.LinearLayoutManager;

import android.support.v7.widget.RecyclerView;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import java.util.ArrayList;

import java.util.List;

public class MainActivity extends AppCompatActivity {

private List<Msg> msgList = new ArrayList<Msg>();

private EditText inputText;

private Button send;

private RecyclerView msgRecyclerView;

private MsgAdapter adapter;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

//initView

inputText = (EditText) findViewById(R.id.input\_text);

send = (Button) findViewById(R.id.send);

msgRecyclerView = (RecyclerView) findViewById(R.id.msg\_recycler\_view);

LinearLayoutManager layoutManager = new LinearLayoutManager(this);

msgRecyclerView.setLayoutManager(layoutManager);

//initData

initMsgs(); // 初始化消息数据

adapter = new MsgAdapter(msgList);

msgRecyclerView.setAdapter(adapter);

send.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String content = inputText.getText().toString();

if (!"".equals(content)) {

Msg msg = new Msg(content, Msg.TYPE\_SENT);

msgList.add(msg);

adapter.notifyItemInserted(msgList.size() - 1); // 当有新消息时，刷新ListView中的显示

msgRecyclerView.scrollToPosition(msgList.size() - 1); // 将ListView定位到最后一行

inputText.setText(""); // 清空输入框中的内容

}

}

});

}

private void initMsgs() {

Msg msg1 = new Msg("Hello guy.", Msg.TYPE\_RECEIVED);

msgList.add(msg1);

Msg msg2 = new Msg("Hello. Who is that?", Msg.TYPE\_SENT);

msgList.add(msg2);

Msg msg3 = new Msg("This is Tom. Nice talking to you. ", Msg.TYPE\_RECEIVED);

msgList.add(msg3);

}

}

