Versão 1 29/06/2018

**============================================================================**

**Índice**

============================================================================

# [Room - banco de dados - Exemplo](#Room_banco_de_dados_Exemplo)

# [RecyclerView - como usar essa lista](#RecyclerView_como_usar_essa_lista)

# [Passando Dados entre Activities](#Passando_Dados_entre_Activities)

**============================================================================**

**============================================================================**

Room - banco de dados - Exemplo

============================================================================

// ------------------CLASS POJO DA TABELA DO BANCO ---------------------

import android.arch.persistence.room.ColumnInfo;

import android.arch.persistence.room.Entity;

import android.arch.persistence.room.PrimaryKey;

// Classe POJO que serve como uma entidade "tabela" do banco de dados

@Entity(tableName = "texto")

public class Texto

{

@PrimaryKey(autoGenerate = true)

private int id;

@ColumnInfo(name = "tipo\_configuracao")

private int tipoConfiguracao;

@ColumnInfo(name = "texto\_escrito")

private String textoEscrito;

public Texto(int tipoConfiguracao, String textoEscrito)

{

this.tipoConfiguracao = tipoConfiguracao;

this.textoEscrito = textoEscrito;

}

public Texto()

{

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public int getTipoConfiguracao() {

return tipoConfiguracao;

}

public void setTipoConfiguracao(int tipoConfiguracao) {

this.tipoConfiguracao = tipoConfiguracao;

}

public String getTextoEscrito() {

return textoEscrito;

}

public void setTextoEscrito(String textoEscrito) {

this.textoEscrito = textoEscrito;

}

}

// --------------INTERFACE DAO PARA REQUISIÇÕES DA TABELA-----------------

import android.arch.persistence.room.Dao;

import android.arch.persistence.room.Insert;

import android.arch.persistence.room.Query;

import java.util.List;

import br.com.rafaelflsoftware.diradministrativo.banco.entidades.Texto;

// Interface Dao, que serve para fazer as requisições e as querys

@Dao

public interface TextoDaoInterface

{

// Seleciona todos as linha

@Query("SELECT \* FROM TEXTO")

List<Texto> getAllTextos();

// com um objeto criado dentro da função

@Insert

void inserirAll(Texto... textos);

// Insere com uma lista de objetos do tipo do POJO

@Insert

void insertTextoList(List<Texto> textos);

}

// --------CLASS QUE CRIA O BANCO DE DADOS E AS TABELAS-------------------

import android.arch.persistence.room.Database;

import android.arch.persistence.room.Room;

import android.arch.persistence.room.RoomDatabase;

import android.content.Context;

import br.com.rafaelflsoftware.diradministrativo.banco.dao.TextoDaoInterface;

import br.com.rafaelflsoftware.diradministrativo.banco.entidades.Texto;

// Classe que cria o banco de dados. {Texto.class} é o nome das classes que POJO.

@Database(entities = {Texto.class},version = 1)

public abstract class AppDataBase extends RoomDatabase

{

// Interface Dao da Classe da tabela, uma dessa para cada tabela.

public abstract TextoDaoInterface textoDao();

// nome do banco de dados

public static String DATABASE\_NAME = "adm-db";

// Método Construtor

public AppDataBase()

{

}

}

// --------CLASS ACTIVITY DE EXEMPLO DE USO DO ROOM-------------------

import android.arch.lifecycle.LiveData;

import android.arch.persistence.room.Room;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.util.Log;

import java.util.ArrayList;

import java.util.List;

import br.com.rafaelflsoftware.diradministrativo.banco.AppDataBase;

import br.com.rafaelflsoftware.diradministrativo.banco.entidades.Texto;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

// cria o banco ou requisita o banco de dados

AppDataBase dataBase = Room.databaseBuilder(

getApplicationContext(),

AppDataBase.class,

AppDataBase.DATABASE\_NAME)

.allowMainThreadQueries()

.build();

// uso para inserção de um novo registro

dataBase.textoDao().inserirAll(new Texto(4,"Manoel freitas lima"));

// Requisição de select

List<Texto> textos = dataBase.textoDao().getAllTextos();

// forach para testar o banco

int i = 0;

for (Texto t: textos) {

Log.i("Teste ++++++++++", textos.get(i).getTextoEscrito());

i++;

}

}

}

[Voltar para Índice](#indice)

**============================================================================**

################################################################################

**============================================================================**

RecyclerView - como usar essa lista

============================================================================

// ----------------------CLASSE MAIN -------------------------------------

public class MainActivity extends AppCompatActivity

{

// cria a variável do tipo RecyclerView

RecyclerView recyclerView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

\\ faz a atribuição da variável com a view

recyclerView = findViewById(R.id.recycleViewMain);

// Criação de um objeto do tipo Texto criado por mim para exemplo.

Texto texto = new Texto();

texto.setTipoConfiguracao(“1”);

texto.setTextoEscrito(“Rafael Freitas”);

Texto texto2 = new Texto();

texto1.setTipoConfiguracao(“6”);

texto2.setTextoEscrito(“Antônio Freitas”);

// Criação de uma lista para ser o conteúdo do recycleView

List<Texto> textos = new ArrayList<>();

textos.add(texto);

textos.add(texto);

// Criação de um objeto do RecyclerView.Adapter customizado para a recycleView.

MeuViewHolderAdapterTexto adapterTexto = new MeuViewHolderAdapterTexto(getApplicationContext(), textos);

// Adiciona a recycleView o RecyclerView.Adapter customizado

recyclerView.setAdapter(adapterTexto);

// Criação de um objeto do RecyclerView.LayoutManager que tem o papel de gerenciar o modo de apresentação da recycleView.

RecyclerView.LayoutManager layoutManager = new LinearLayoutManager(getApplicationContext(), LinearLayoutManager.VERTICAL,false);

// Configura o modo de comportamento da recycleView

recyclerView.setLayoutManager(layoutManager);

} Fim - protected void onCreate

} // Fim - public class MainActivity extends AppCompatActivity

// ----------------CLASSE RECYCLE.VIEW.ADAPTER------------------------------

// Está classe gerência o comportamento das view que serviram para criar uma das linha da recycleView.

public class MeuViewHolderAdapterTexto extends RecyclerView.Adapter

{

// Cria da variáveis

Context context;

List<Texto> textos;

// Método construtor

public MeuViewHolderAdapterTexto(Context context, List<Texto> textos)

{

this.context = context;

this.textos = textos;

}

// Este método é que cria as linha,

@Override

public RecyclerView.ViewHolder onCreateViewHolder(ViewGroup parent, int viewType)

{

// Está linha tem a função de inflar o layout criado para ser a linha da recycleView.

View view = LayoutInflater.from(this.context).inflate(R.layout.linha\_recycle\_view\_texto, parent,false);

// Objeto criado RecyclerView.ViewHolder customizado, que retorna o Layout da linhas.

MeuViewHolderTexto meuViewHolderTexto = new MeuViewHolderTexto(view);

// Retorna as linhas da recycleView

return meuViewHolderTexto;

}

// Este método serve para fazer configurações nas view que estão no Layout, como mudar a fonte, cor, alinhamento.

@Override

public void onBindViewHolder(RecyclerView.ViewHolder holder, int position)

{

// Instância da classe MeuViewHolderTexto e atribuição do holder do método.

MeuViewHolderTexto meuViewHolderTexto = (MeuViewHolderTexto) holder;

// exemplo de configuração de uma view que está dentro do Layout

switch (textos.get(position).getTipoConfiguracao())

{

case "1":

meuViewHolderTexto.tv1\_linha\_recycle\_view.setTextSize(16.0f);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setTypeface(defaultFromStyle(BOLD));

meuViewHolderTexto.tv1\_linha\_recycle\_view.setGravity(1);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setPadding(8,28,0,28);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setText(textos.get(position).getTextoEscrito());

break;

case "2":

meuViewHolderTexto.tv1\_linha\_recycle\_view.setTextSize(16.0f);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setTypeface(Typeface.defaultFromStyle(1));

meuViewHolderTexto.tv1\_linha\_recycle\_view.setGravity(1);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setPadding(8,28,8,28);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setText(textos.get(position).getTextoEscrito());

break;

case "3":

meuViewHolderTexto.tv1\_linha\_recycle\_view.setTextSize(16.0f);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setTypeface(Typeface.defaultFromStyle(0));

meuViewHolderTexto.tv1\_linha\_recycle\_view.setGravity(0);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setPadding(24,8,8,16);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setText(textos.get(position).getTextoEscrito());

break;

case "4":

meuViewHolderTexto.tv1\_linha\_recycle\_view.setTextSize(16.0f);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setTypeface(Typeface.defaultFromStyle(0));

meuViewHolderTexto.tv1\_linha\_recycle\_view.setGravity(0);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setPadding(48,8,8,8);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setText(textos.get(position).getTextoEscrito());

break;

case "5":

meuViewHolderTexto.tv1\_linha\_recycle\_view.setTextSize(16.0f);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setTypeface(Typeface.defaultFromStyle(0));

meuViewHolderTexto.tv1\_linha\_recycle\_view.setGravity(0);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setPadding(96,8,8,8);

meuViewHolderTexto.tv1\_linha\_recycle\_view.setText(textos.get(position).getTextoEscrito());

break;

}

\\ Este método serve para contar quantas linha tem o recycleView.

@Override

public int getItemCount() {

return this.textos.size();

}

} Fim - public class MeuViewHolderAdapterTexto

//----------------CLASSE RECYCLEVIEW.VIEWHOLDER-----------------------------

// Classe que tem o objetivo de ligar o Layout criado para ser gerenciado pelo adapter.

public class MeuViewHolderTexto extends RecyclerView.ViewHolder

{

// Aqui tem que ser declarados todas as Views que serão alteradas, neste caso só temos um TextView, mas pode-se colocar mais.

TextView tv1\_linha\_recycle\_view;

public MeuViewHolderTexto(View itemView)

{

super(itemView);

// atribuição da variável com o componente View, neste caso só temos um TextView, mas pode-se colocar mais.

tv1\_linha\_recycle\_view = itemView.findViewById(R.id.tv1\_linha\_recycle\_view);

}

} // Fim - public class MeuViewHolderTexto

//-----------------------LAYOUT DA LINHA------------------------------------

// LAYOUT DA LINHA / Layout linha\_recycle\_view\_texto

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

<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:background="@android:color/holo\_green\_light">

<TextView

android:id="@+id/tv1\_linha\_recycle\_view"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_marginEnd="8dp"

android:layout\_marginStart="8dp"

android:padding="8dp"

android:text="as instruções que fazem parte deste Edital."

android:textAppearance="@style/TextAppearance.AppCompat.Body1"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>

//---------------LAYOUT QUE ESTÁ O RECYCLEVIEW-------------------------------

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

<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context="br.com.rafaelflsoftware.diradministrativo.MainActivity">

<android.support.v7.widget.RecyclerView

android:id="@+id/recycleViewMain"

android:layout\_width="0dp"

android:layout\_height="0dp"

android:layout\_marginBottom="8dp"

android:layout\_marginEnd="8dp"

android:layout\_marginStart="8dp"

android:layout\_marginTop="8dp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>

//--------------------------CLASSE POJO -------------------------------

// NÃO FAZ PARTE DO RECYCLEVIEW, TEM APENAS A FUNÇÃO DE CRIAR UM OBJETO QUE VAI // PREENCHER ELA.

public class Texto

{

private int id;

private String tipoConfiguracao;

private String textoEscrito;

//GETTES e SETTES

public Texto(String tipoConfiguracao, String textoEscrito)

{

this.tipoConfiguracao = tipoConfiguracao;

this.textoEscrito = textoEscrito;

}

public Texto()

{

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getTipoConfiguracao() {

return tipoConfiguracao;

}

public void setTipoConfiguracao(String tipoConfiguracao) {

this.tipoConfiguracao = tipoConfiguracao;

}

public String getTextoEscrito() {

return textoEscrito;

}

public void setTextoEscrito(String textoEscrito) {

this.textoEscrito = textoEscrito;

}

}

[Voltar para Índice](#indice)

**============================================================================**

################################################################################

**============================================================================**

Passando Dados entre Activities

============================================================================

// ----------------------CLASSE MAIN -------------------------------------

import android.app.Activity;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.view.Window;

import android.view.WindowManager;

import android.webkit.WebView;

import android.widget.Button;

import android.widget.EditText;

public class MainActivity extends Activity {

private EditText edtTxtNome;

private Button btnOK;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

requestWindowFeature(Window.FEATURE\_NO\_TITLE);

getWindow().addFlags(WindowManager.LayoutParams.FLAG\_FULLSCREEN);

setContentView(R.layout.activity\_main);

edtTxtNome = (EditText) findViewById(R.id.edtTxtNome);

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

btnOK.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

// Inícia a Intent que vai fazer a mudança de uma Activity para outra.

Intent intent = new Intent(MainActivity.this, Transferida.class);

// Inicialização de uma variável String.

String texto = "";

// Variável texto recebe o valor da caixa de texto

texto = edtTxtNome.getText().toString();

// Inicializa o Bundle

Bundle bundle = new Bundle();

// Coloca no bundle a String da caixa de texto.

// "testoPassado" é a chave do Bundle

// texto é de onde vem a String

bundle.putString("testoPassado", texto);

// Coloca na Intent o valor da bundle que será passada para a próxima Intent.

intent.putExtras(bundle);

// Começa a Intent, que abrirá a Activity Transferida

startActivity(intent);

}

});

}

}

// ----------------A interface da MainActivity----------------------------

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

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

xmlns:tools="http://schemas.android.com/tools"

android:id="@+id/activity\_main"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:paddingBottom="@dimen/activity\_vertical\_margin"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

tools:context="rafaelflsoftware.webteste.MainActivity"

android:padding="0dp">

<LinearLayout

android:orientation="vertical"

android:layout\_width="match\_parent"

android:layout\_height="150dp">

<EditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:inputType="textPersonName"

android:text="Name"

android:ems="10"

android:id="@+id/edtTxtNome"

android:layout\_weight="0.99" />

<Button

android:text="Ok"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/btnOK"

android:layout\_weight="1" />

</LinearLayout>

</LinearLayout>

// ------------ Classe que os dados foram passados-------------------------

import android.content.Intent;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.widget.TextView;

public class Transferida extends AppCompatActivity {

private TextView textView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_transferida);

textView = (TextView) findViewById(R.id.textView);

// pegar os dados da Intent

Intent intent = getIntent();

// Pega os valores que estão na intent e atribuem a bundle.

Bundle bundle = intent.getExtras();

// Cria uma variável String e pega o valor que está guardado no bundle de chave "testoPassado".

String textoRecebido = bundle.getString("testoPassado");

// Atribui o valor ao textView.

textView.setText(textoRecebido);

}

}

// --------------A interface da class Transferida------------------------

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

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

xmlns:tools="http://schemas.android.com/tools"

android:id="@+id/activity\_transferida"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:paddingBottom="@dimen/activity\_vertical\_margin"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

tools:context="rafaelflsoftware.webteste.Transferida">

<TextView

android:text="TextView"

android:layout\_height="wrap\_content"

android:layout\_alignParentTop="true"

android:layout\_marginTop="62dp"

android:id="@+id/textView"

android:layout\_width="150dp" />

</RelativeLayout>

[Voltar para Índice](#indice)

**============================================================================**

################################################################################