

# Linguagem e Técnica de Programação Mobile

## **AULA 10 - Usando Banco de Dados no Android – SQLite – Parte 2**

Prof. João Paulo Pimentel  
[joao.pimentel@projecao.br](mailto:joao.pimentel@projecao.br)

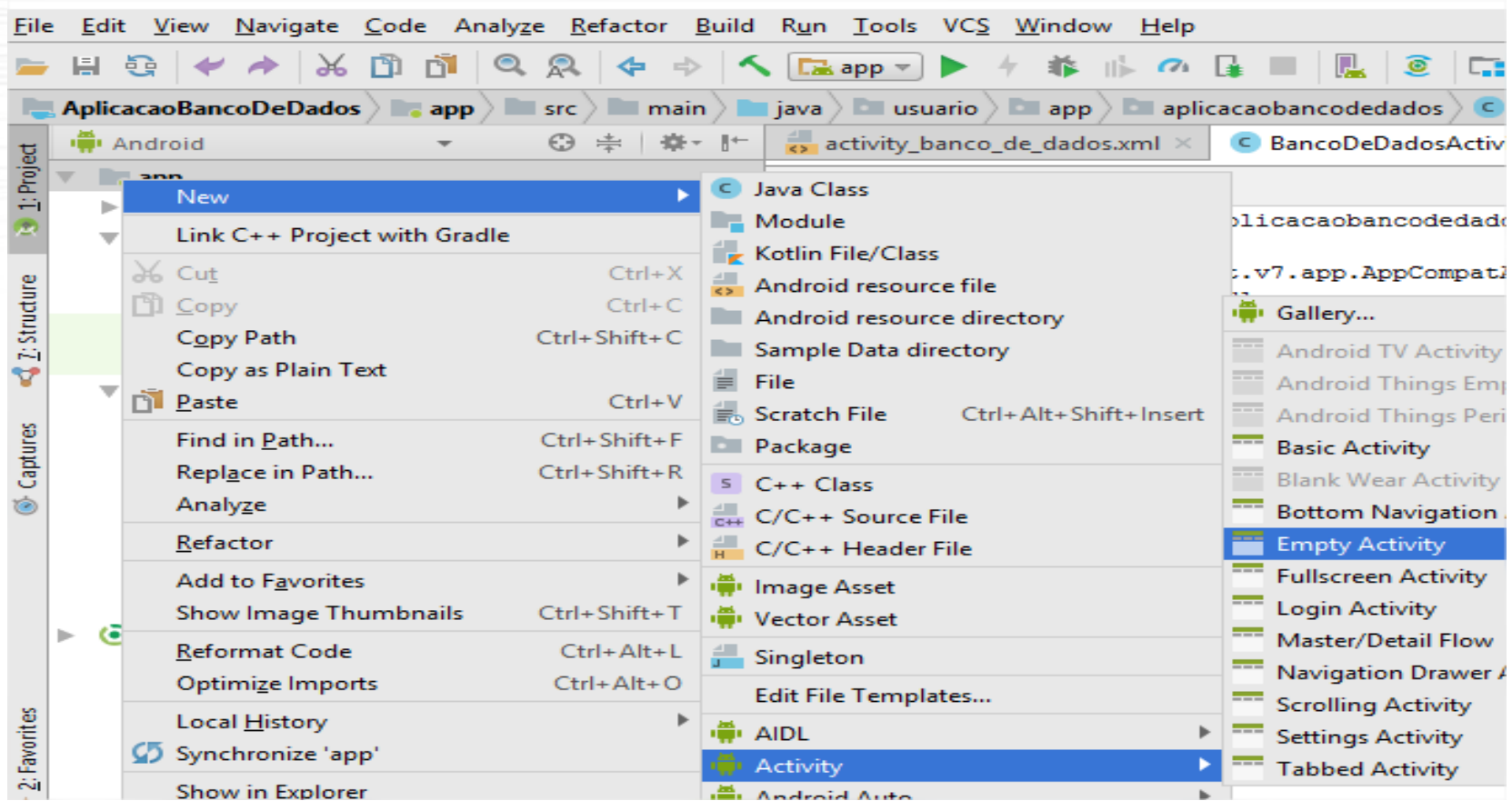
---

# SQLite do Android

- Registrando informações no Banco (Segunda Versão)
  - Na versão anterior (**aula passada**) que desenvolvemos da **Activity**, o registro das informações na tabela “**usuarios**” eram realizados através do uso da instrução SQL “**insert**”, expressa dentro do método **execSQL**.
  - Agora vamos realizar a inserção dos dados através do método **insert** presente dentro da classe **SQLiteDatabase**.
-

# Aplicação Android com Banco de Dados

- Relembrando como se cria uma nova Activity ...
- Clique com o botão direito do mouse sobre o diretório “app” do nosso projeto e selecione “New / Activity / Empty Activity”. Veja na figura seguinte :



# Aplicação Android com Banco de Dados

- Vamos agora criar uma nova “**Activity**” (Atividade) , no mesmo projeto da aula passada, com as seguintes informações abaixo :
  - Activity Name : GravaRegistros2Activity
  - Layout Name : activity\_grava\_registros2
  - Title : Gravar Registros (2ª Versão)
  - Menu Resource Name : menu\_grava\_registros2
-

# Aplicação Android com Banco de Dados

- Após criar a Activity , vamos dentro do arquivo “activity\_grava\_registros2.xml” para escrevermos **O MESMO CÓDIGO XML** do arquivo “activity\_grava\_registros.xml” (**não muda nada**). Depois de copiar o código para o arquivo, salve as alterações.
  - Depois disso vamos abrir o arquivo “**GravaRegistros2Activity.java**” para digitarmos o seguinte código apresentado nos próximo slides:
-

# Aplicação Android com Banco de Dados

- Pacotes e import's

```
package usuario.app.aplicacaobancodedados;

import android.app.Activity;
import android.os.Bundle;
import android.app.AlertDialog;
import android.content.ContentValues;
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
```

---

# Aplicação Android com Banco de Dados

- Classe java e declaração de atributos

```
public class GravaRegistros2Activity extends Activity {  
  
    Button btcadastrar;  
  
    EditText ednome;  
    EditText edtelefone;  
    EditText edemail;  
  
    SQLiteDatabase db;
```

---



# Aplicação Android com Banco de Dados

@Override

```
protected void onCreate(Bundle savedInstanceState) {  
  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_grava_registros2);  
  
    btcadastrar = (Button) findViewById(R.id.btcadastrar);  
  
    ednome = (EditText) findViewById(R.id.ednome);  
    edtelefone = (EditText) findViewById(R.id.edtelefone);  
    edemail = (EditText) findViewById(R.id.edemail);  
  
    try {  
  
        db = openOrCreateDatabase("banco_dados",  
            Context.MODE_PRIVATE, null);  
    }  
    catch (Exception e)  
    {  
        MostraMensagem("Erro : " + e.toString());  
    }  
}
```



# Aplicação Android com Banco de Dados

```
btcadastrar.setOnClickListener(new View.OnClickListener() {  
  
    @Override  
    public void onClick(View arg0) {  
  
        String nome = ednome.getText().toString();  
        String telefone = edtelefone.getText().toString();  
        String email = edemail.getText().toString();  
  
        ContentValues valor = new ContentValues();  
  
        valor.put("nome", nome);  
        valor.put("telefone", telefone);  
        valor.put("email", email);  
    }  
});
```

---

# Aplicação Android com Banco de Dados

```
try {  
    db.insert("usuarios", null, valor);  
    MostraMensagem("Dados cadastrados com sucesso");  
}  
catch(Exception e)  
{  
    MostraMensagem("Erro : " + e.toString());  
}  
});  
}
```

---

# Aplicação Android com Banco de Dados

```
public void MostraMensagem(String str)
{
    AlertDialog.Builder dialogo = new
    AlertDialog.Builder(GravaRegistros2Activity.this);

    dialogo.setTitle("Aviso");
    dialogo.setMessage(str);
    dialogo.setNeutralButton("OK", null);
    dialogo.show();
}
}
```

---

# Aplicação Android com Banco de Dados

- Segue os comentários de algumas linhas do nosso programa de cadastro. O código presente dentro do **botão cadastrar** da aplicação possui o seguinte conjunto de instruções :

```
ContentValues valor = new ContentValues();

valor.put("nome", nome);
valor.put("telefone", telefone);
valor.put("email", email);

try {

    db.insert("usuarios", null, valor);
    MostraMensagem("Dados cadastrados com sucesso");
}
catch(Exception e)
{
    MostraMensagem("Erro : " + e.toString());
}
```

---

# Aplicação Android com Banco de Dados

```
ContentValues valor = new ContentValues();

valor.put("nome", nome);
valor.put("telefone", telefone);
valor.put("email", email);

try {

    db.insert("usuarios", null, valor);
    MostraMensagem("Dados cadastrados com sucesso");
}
catch(Exception e)
{
    MostraMensagem("Erro : " + e.toString());
}
```

- Esse conjunto de instruções mostrados acima, funciona de forma similar ao método **execSQL** (utilizado na versão do programa anterior), executando a instrução INSERT da linguagem SQL.
-

# Aplicação Android com Banco de Dados

- O primeiro bloco de instruções:

```
ContentValues valor = new ContentValues();  
  
valor.put("nome", nome);  
valor.put("telefone", telefone);  
valor.put("email", email);
```

- Cria o objeto do tipo **ContentValues** chamado *valor*. Dentro desse objeto armazenamos os valores a serem adicionados no Banco de Dados através do método **put**. Esse método possui dois argumentos: No primeiro argumento do método é informado o **nome do campo presente dentro da tabela** (onde iremos armazenar as informações) e no segundo argumento informamos o **dado que aquele campo irá assumir**.
-

# Aplicação Android com Banco de Dados

- Na instrução seguinte:

```
db.insert("usuarios", null, valor);
```

- Executamos o método **insert** presente dentro do objeto *db* (do tipo **SQLiteDataBase**) que é responsável por inserir as informações dentro da tabela “usuarios” (funcionamento similar ao da instrução INSERT da linguagem SQL).
-



# Aplicação Android com Banco de Dados

- Agora vamos abrir o arquivo “**activity\_main.xml**” e dentro da tela da aplicação, vamos adicionar um componente do tipo **Button** (abaixo do botão “Cadastrar Dados”), de acordo com a tabela abaixo:

Propriedade	Valor
id	btcadastrardados2
text	Cadastrar Dados (2ª Edição)
layout:width	200dp

- Ou através dos **XML's** apresentado no próximo slide:
-

# Aplicação Android com Banco de Dados

- Ou através do XML abaixo:

```
<Button
    android:id="@+id/btcadastrardados2"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
    android:text="@string/cadastrar_dados2" />
```

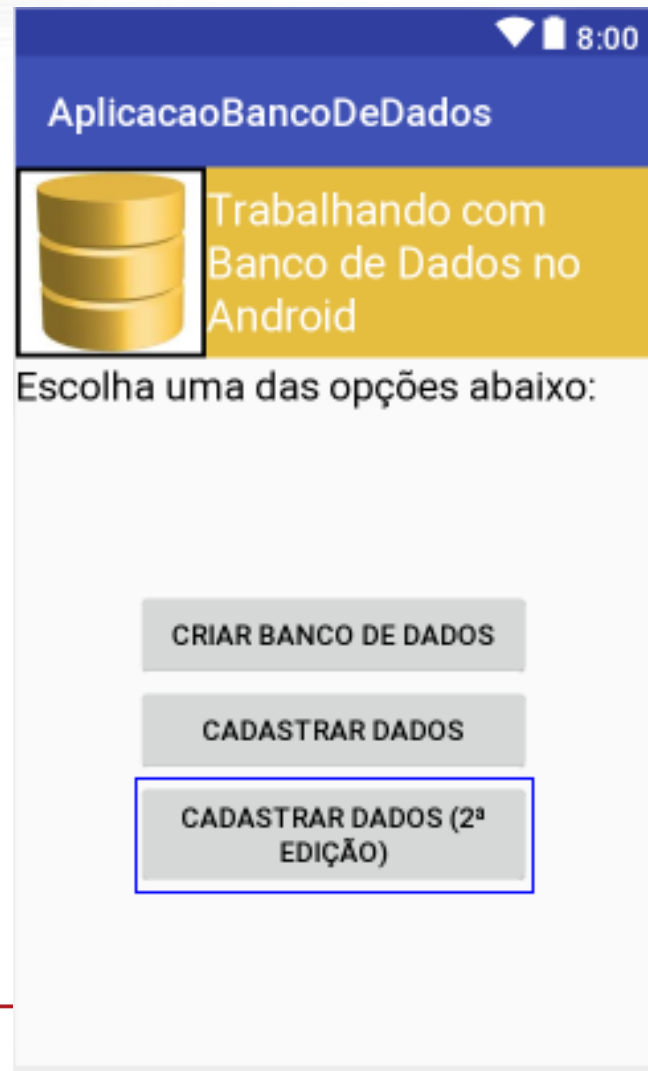
- Acrescentar também no strings.xml o “cadastrar\_dados2” apresentado abaixo:

```
<string name="cadastrar_dados">Cadastrar Dados</string>
💡 <string name="cadastrar_dados2">Cadastrar Dados (2ª Edição)</string>
</resources>
```

---

# Aplicação Android com Banco de Dados

- Feito isso teremos o seguinte resultado:



# Aplicação Android com Banco de Dados

- Agora dentro do arquivo “**MainActivity.java**” vamos escrever as seguintes instruções destacadas em **azul abaixo**:

```
public class MainActivity extends Activity {  
  
    Button btcriabanco;  
    Button btcadastrardados;  
    Button btcadastrardados2;  
    SQLiteDatabase db;
```

---

# Aplicação Android com Banco de Dados

- //Continuação do código...

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_banco_de_dados);

    btcriabanco = (Button) findViewById(R.id.btcriarbanco);
    :
    Btcadastrardados2 = (Button)
    findViewById(R.id.btcadastrardados2);
```

# Aplicação Android com Banco de Dados

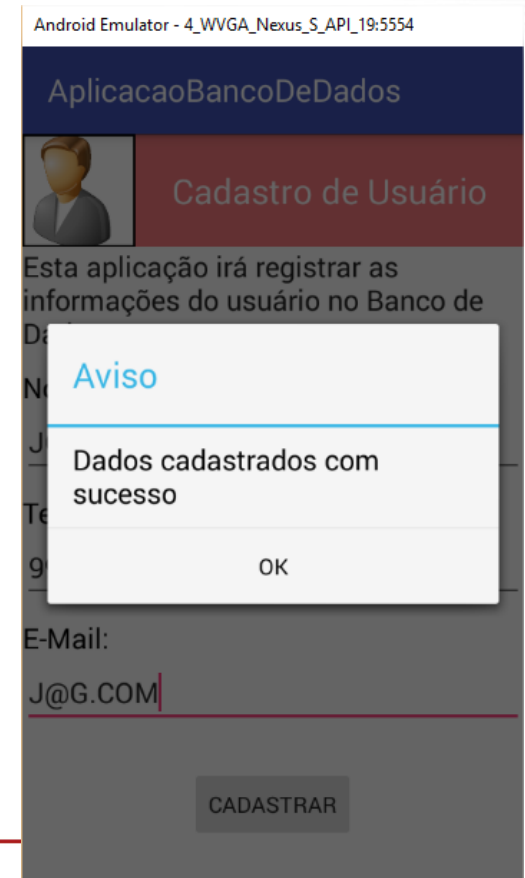
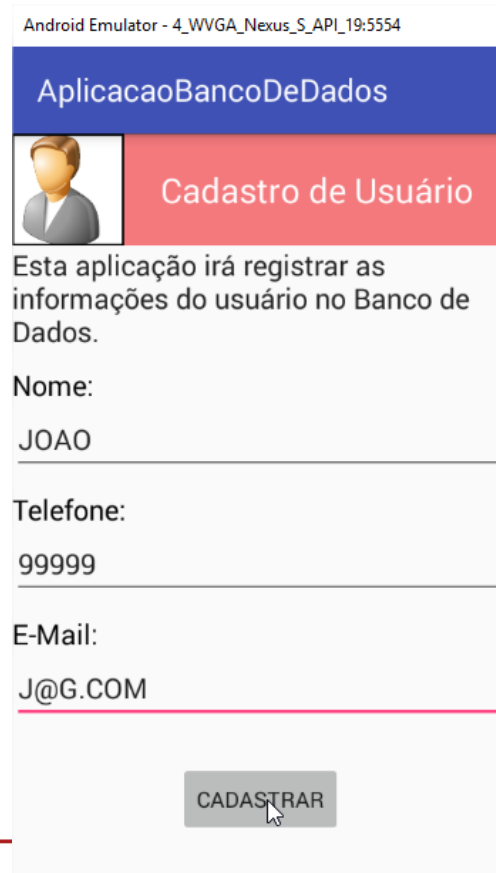
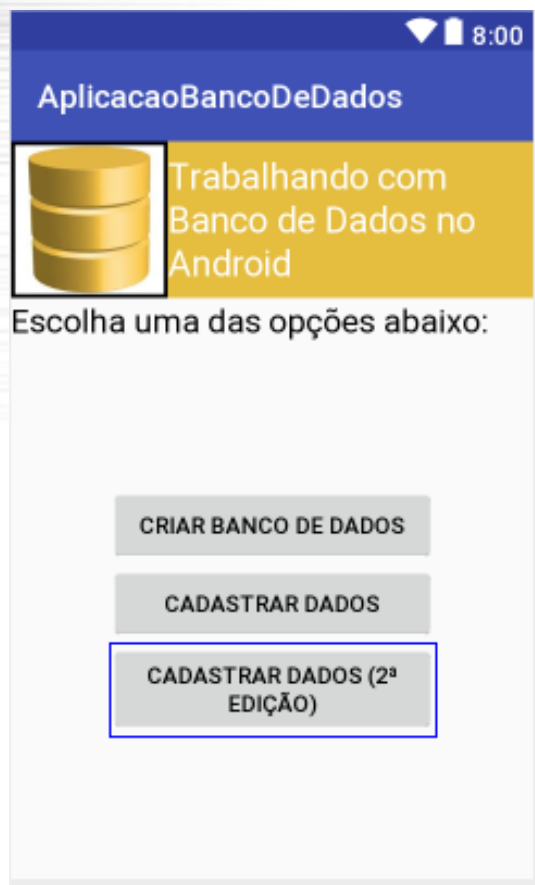
- // Intent **gravaRegistroActivity2**

```
btcadastrardados2.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View args0){  
        Intent gravaRegistroActivity2 = new Intent ( packageContext: MainActivity.this,  
            GravaRegistros2Activity.class);  
        MainActivity.this.startActivity(gravaRegistroActivity2);  
    }  
});
```

---

# Aplicação Android com Banco de Dados

- Após acrescentar o código no `MainActivity.java`, vamos executar a nossa aplicação. Veja o resultado nas figuras seguintes. Experimente agora um novo cadastro:







# Aplicação Android com Banco de Dados

## Consultando as informações da tabela

- Agora vamos desenvolver uma **Activity** que irá realizar a consulta das informações que foram gravadas na tabela através das aplicações desenvolvidas anteriormente.
-

# Aplicação Android com Banco de Dados

- Crie uma nova “**Activity**” (Atividade) , conforme já foi mostrado, de acordo com as informações a seguir (**ConsultaDadosActivity**):

**Creates a new empty activity**

Activity Name:

☒ Generate Layout File

Layout Name:

☐ Launcher Activity

Package name:

Source Language:

Target Source Set:

# Aplicação Android com Banco de Dados

- Vamos colocar dentro do diretório “**drawable**” (diretório de imagens, presente dentro da pasta “res” do projeto) as imagens “botao\_anterior.png”, botao\_primeiro .png, botao\_proximo.png, botao\_ultimo.png e icone\_consulta.png que estão disponibilizadas no Blog.



botao\_anterior.p  
ng



botao\_primeiro.p  
ng



botao\_proximo.p  
ng



botao\_ultimo.pn  
g



icone\_consulta.p  
ng

# Aplicação Android com Banco de Dados

- Depois de criada a nossa **Activity** e incluído as imagens no projeto, vamos adicionar dentro do arquivo “**activity\_consulta\_dados.xml**” o seguinte código XML:

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".ConsultaDadosActivity" >

    <LinearLayout
        android:id="@+id/LinearLayout1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#3ba0e2" >

        <ImageView
            android:id="@+id/imageView1"
            android:layout_width="72dp"
            android:layout_height="72dp"
            android:src="@drawable/icone_consulta" />
```

# Aplicação Android com Banco de Dados

- Continuação do XML...

```
<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance=
        "?android:attr/textAppearanceLarge" />
```

```
<LinearLayout
    android:id="@+id/LayoutCorFundo"
    android:layout_width="fill_parent"
    android:layout_height="match_parent"
    android:background="#749ff5"
    android:gravity="center" >
```

```
<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Consultar Dados"
```

# Aplicação Android com Banco de Dados

- Continuação do XML...

```
        android:textAppearance=
            "?android:attr/textAppearanceLarge"
        android:textColor="#ffffff" />
    </LinearLayout>
</LinearLayout>

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingTop="20dp"
    android:text="Nome:"
    android:textAppearance="?android:attr/textAppearanceSmall"
    android:textColor="#000000"
    android:textSize="18sp" />
```

---

# Aplicação Android com Banco de Dados

- Continuação do XML...

```
<TextView
```

```
    android:id="@+id/txtnome"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:paddingTop="5dp"  
    android:text="[Nome]"  
    android:textAppearance="?android:attr/textAppearanceSmall"  
    android:textColor="#2a58b4"  
    android:textSize="18sp" />
```

```
<TextView
```

```
    android:id="@+id/textView4"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:paddingTop="20dp"  
    android:text="Telefone:"  
    android:textAppearance="?android:attr/textAppearanceSmall"  
    android:textColor="#000000"  
    android:textSize="18sp" />
```



# Aplicação Android com Banco de Dados

- Continuação do XML...

```
<TextView
    android:id="@+id/txttelefone"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingTop="5dp"
    android:text="[Telefone]"
    android:textAppearance="?android:attr/textAppearanceSmall"
    android:textColor="#2a58b4"
    android:textSize="18sp" />
```

```
<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingTop="20dp"
    android:text="E-Mail:"
    android:textAppearance="?android:attr/textAppearanceSmall"
    android:textColor="#000000"
    android:textSize="18sp" />
```

# Aplicação Android com Banco de Dados

- Continuação do XML...

```
<TextView
    android:id="@+id/txtemail"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingTop="5dp"
    android:text="[EMail]"
    android:textAppearance="?android:attr/textAppearanceSmall"
    android:textColor="#2a58b4"
    android:textSize="18sp" />
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:paddingTop="15dp" >
```

# Aplicação Android com Banco de Dados

- Continuação do XML...

```
<ImageView
    android:id="@+id/imgprimeiro"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:scaleType="fitXY"
    android:src="@drawable/botao_primeiro" />
```

```
<ImageView
    android:id="@+id/imganterior"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:scaleType="fitXY"
    android:src="@drawable/botao_anterior" />
```

```
<ImageView
    android:id="@+id/imgproximo"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:scaleType="fitXY"
    android:src="@drawable/botao_proximo" />
```

# Aplicação Android com Banco de Dados

- Continuação do XML...

```
<ImageView  
    android:id="@+id/imgultimo"  
    android:layout_width="50dp"  
    android:layout_height="50dp"  
    android:scaleType="fitXY"  
    android:src="@drawable/botao_ultimo" />
```

```
</LinearLayout>
```

```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:gravity="center"  
    android:paddingTop="15dp" >
```



# Aplicação Android com Banco de Dados

- Continuação do XML...

```
<TextView
    android:id="@+id/txtstatus_registro"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="#status"
    android:textAppearance="?android:attr/textAppearanceLarge"
/>
```

```
</LinearLayout>
```

```
</LinearLayout>
```

---

# Aplicação Android com Banco de Dados

- Vejamos o resultado:



# Aplicação Android com Banco de Dados

- Agora vamos abrir o arquivo “ConsultaDadosActivity.java” para digitarmos o seguinte código abaixo:

```
package usuario.app.aplicacaobancodedados;

import android.os.Bundle;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;
```



# Aplicação Android com Banco de Dados

```
public class ConsultaDadosActivity extends Activity {

    TextView txtnome, txttelefone, txtemail,txtstatus_registro;

    SQLiteDatabase db;

    ImageView imgprimeiro, imganterior, imgproximo, imgultimo;

    int indice;

    Cursor c;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_consulta_dados);

        txtnome = (TextView) findViewById(R.id.txtnome);
        txttelefone = (TextView) findViewById(R.id.txttelefone);
        txtemail = (TextView) findViewById(R.id.txtemail);
        txtstatus_registro = (TextView)
            findViewById(R.id.txtstatus_registro);
    }
}
```

# Aplicação Android com Banco de Dados

```
txtnome.setText("");
txttelefone.setText("");
txtemail.setText("");

imgprimeiro = (ImageView) findViewById(R.id.imgprimeiro);
imganterior = (ImageView) findViewById(R.id.imganterior);
imgproximo = (ImageView) findViewById(R.id.imgproximo);
imgultimo = (ImageView) findViewById(R.id.imgultimo);

try {

    //Abre o banco de dados
    db = openOrCreateDatabase
    ("banco_dados",Context.MODE_PRIVATE, null);

    c = db.query("usuarios",new String []
    {"nome","telefone","email"},
    null,null,null,null,null);
```

---

# Aplicação Android com Banco de Dados

```
if(c.getCount() > 0) {  
  
    //Move para o primeiro registro  
    c.moveToFirst();  
    indice = 1;  
  
    txtnome.setText(c.getString(0)); //Obtem o nome  
    txttelefone.setText(c.getString(1)); //Obtém o telefone  
    txtemail.setText(c.getString(2)); //Obtém o e-mail  
  
    txtstatus_registro.setText(indice + " / " + c.getCount());  
  
}  
else {  
  
    txtstatus_registro.setText("Nenhum Registro");  
  
}
```

# Aplicação Android com Banco de Dados

```
imgprimeiro.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View view) {

        if(c.getCount() > 0)
        {
            //Move para o primeiro registro
            c.moveToFirst();
            indice = 1;
            //Obtem o nome
            txtnome.setText(c.getString(0));
            //Obtem o telefone
            txttelefone.setText(c.getString(1));
            //Obtém o e-mail
            txtemail.setText(c.getString(2));

            txtstatus_registro.setText(indice + " / " +
            c.getCount());
        }
    }
});
```

# Aplicação Android com Banco de Dados

```
imganterior.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {

        if(c.getCount() > 0)
        {
            if(indice > 1) {

                indice--;
                //Move para o registro anterior
                c.moveToPrevious();

                txtnome.setText(c.getString(0)); //Obtem o nome
                txttelefone.setText(c.getString(1)); //Obtém o telefone
                txtemail.setText(c.getString(2)); //Obtém o e-mail

                txtstatus_registro.setText(indice + " / " +
                    c.getCount());
            }
        }
    }
});
```

# Aplicação Android com Banco de Dados

```
imgproximo.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View arg0) {

        if(c.getCount() > 0)
        {
            if(indice != c.getCount()) {

                indice++;
                //Move para o próximo registro
                c.moveToNext();

                txtnome.setText(c.getString(0)); //Obtem o nome
                txttelefone.setText
                (c.getString(1)); //Obtém o telefone
                txtemail.setText
                (c.getString(2)); //Obtém o e-mail

                txtstatus_registro.setText(indice + " / " +
                c.getCount());

            }

        }

    }

});
```

# Aplicação Android com Banco de Dados

```
imgultimo.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {

        if(c.getCount() > 0)
        {
            //Move para o último registro
            c.moveToLast();
            indice = c.getCount();
            //Obtem o nome
            txtnome.setText(c.getString(0));
            //Obtém o telefone
            txttelefone.setText(c.getString(1));
            //Obtém o e-mail
            txtemail.setText(c.getString(2));

            txtstatus_registro.setText(indice + " / " +
            c.getCount());
        }

    }

});
```



# Aplicação Android com Banco de Dados

```
        }catch(Exception e)
        {
            MostraMensagem("Erro : " + e.toString());
        }
    }

    public void MostraMensagem(String str)
    {
        AlertDialog.Builder dialogo = new
        AlertDialog.Builder(ConsultaDadosActivity.this);

        dialogo.setTitle("Aviso");
        dialogo.setMessage(str);
        dialogo.setNeutralButton("OK", null);
        dialogo.show();
    }
}
```

# Aplicação Android com Banco de Dados

- Vamos explicar agora algumas linhas de código do nosso programa. Ao carregarmos o nosso programa , é executado o seguinte trecho de código:

```
//Abre o banco de dados
db = openOrCreateDatabase
("banco_dados",Context.MODE_PRIVATE, null);

c = db.query("usuarios",new String []
{"nome","telefone","email"},
null,null,null,null,null);
```

---

# Aplicação Android com Banco de Dados

- A primeira instrução realiza a abertura do nosso Banco de Dados por meio da instrução:

**openOrCreateDataBase.**

- A segunda instrução realiza a consulta de todas as informações presentes dentro da tabela “**usuarios**” através do método **query** do objeto *db* . O resultado é retornado para o objeto *c* (como uma instância da classe **Cursor**)
-

# Aplicação Android com Banco de Dados

- A navegação entre as informações é feita através dos botões presentes na aplicação. Quando navegamos para o primeiro registro da nossa tabela (através do botão primeiro registro), é executado o seguinte trecho de comandos:

```
if(c.getCount() > 0)
{
    //Move para o primeiro registro
    c.moveToFirst();
    indice = 1;
    //Obtem o nome
    txtnome.setText(c.getString(0));
    //Obtem o telefone
    txttelefone.setText(c.getString(1));
    //Obtém o e-mail
    txtemail.setText(c.getString(2));

    txtstatus_registro.setText(indice + " / " +
    c.getCount());
}
```

---

# Aplicação Android com Banco de Dados

- No trecho de comandos mostrados no slide anterior, temos a seguinte instrução:

```
c.moveToFirst();
```

- Que move o cursor para o primeiro registro da tabela **“usuarios”**. Nas instruções seguintes:

```
//Obtem o nome  
txtnome.setText(c.getString(0));  
//Obtem o telefone  
txttelefone.setText(c.getString(1));  
//Obtém o e-mail  
txtemail.setText(c.getString(2));
```

- Que exibe as informações do primeiro registro na tela da nossa aplicação.
-

# Aplicação Android com Banco de Dados

- A instrução seguinte:

```
txtstatus_registro.setText(indice + " / " +  
c.getCount());
```

- Atualiza o status de navegação, onde são exibidos na tela a posição corrente do cursor na tabela de dados e o total de registros presentes na tabela.
  - Todas as atividades descritas desse botão, é similar para os demais botões de navegação.
-

# Aplicação Android com Banco de Dados

- Agora vamos abrir o arquivo “**activity\_main.xml**” e dentro da tela da nossa aplicação, vamos adicionar um componente do tipo **Button** (abaixo do botão “**Cadastrar Dados (2ª Versão)**”, conforme a tabela seguinte:

Propriedade	Valor
id	btconsultardados
text	Consultar Dados
layout:width	200dp

- Ou com o código **XML** no próximo slide.
-



# Aplicação Android com Banco de Dados

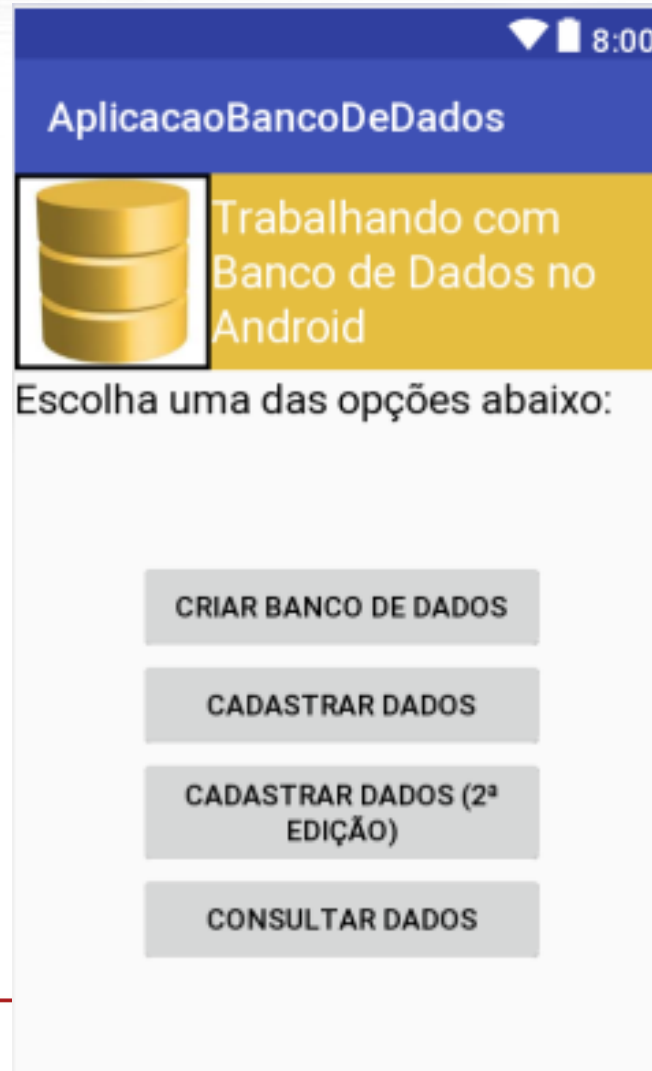
- Código XML abaixo referente ao Botão “Consultar Dados”:

```
<Button  
    android:id="@+id/btconsultardados"  
    android:layout_width="200dp"  
    android:layout_height="wrap_content"  
    android:text="@string/consultar_dados" />
```

---

# Aplicação Android com Banco de Dados

- Feito isso teremos o seguinte resultado:



# Aplicação Android com Banco de Dados

- Agora vamos abrir o arquivo “MainActivity.java” para digitarmos os seguintes blocos de comandos destacados em azul abaixo:

```
public class MainActivity extends Activity {  
  
    Button btcriabanco;  
    Button btcadastrardados;  
    Button btcadastrardados2;  
    Button btconsultardados;  
    SQLiteDatabase db;
```

---

# Aplicação Android com Banco de Dados

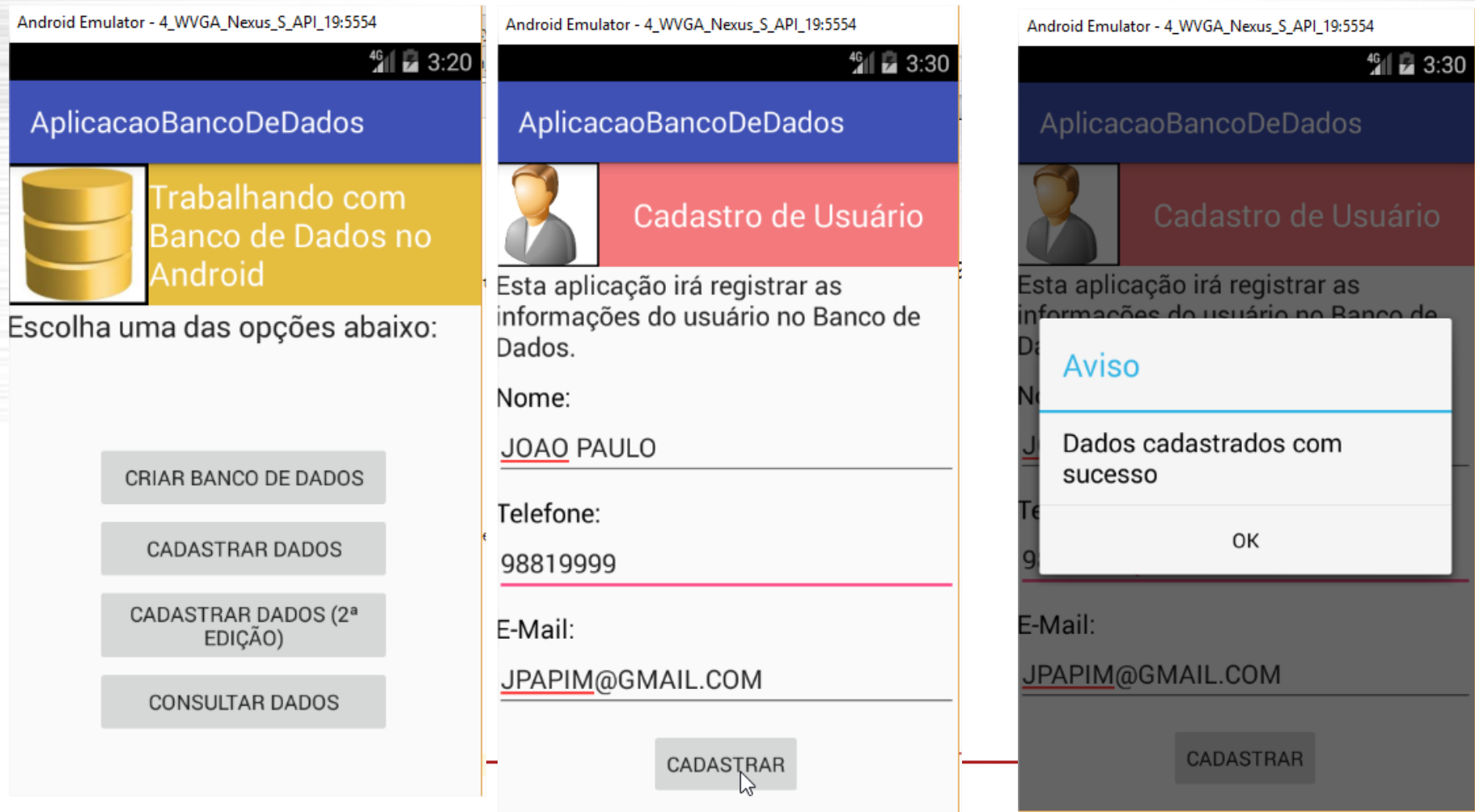
```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    btcriabanco = (Button) findViewById(R.id.btcriarbanco);
    :
    btconsultardados = (Button)
    findViewById(R.id.btconsultardados);

    btconsultardados.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View args0) {
            Intent consultaDadosActivity = new Intent( packageContext: MainActivity.this,
                ConsultaDadosActivity.class);
            MainActivity.this.startActivity(consultaDadosActivity);
        }
    });
});
```

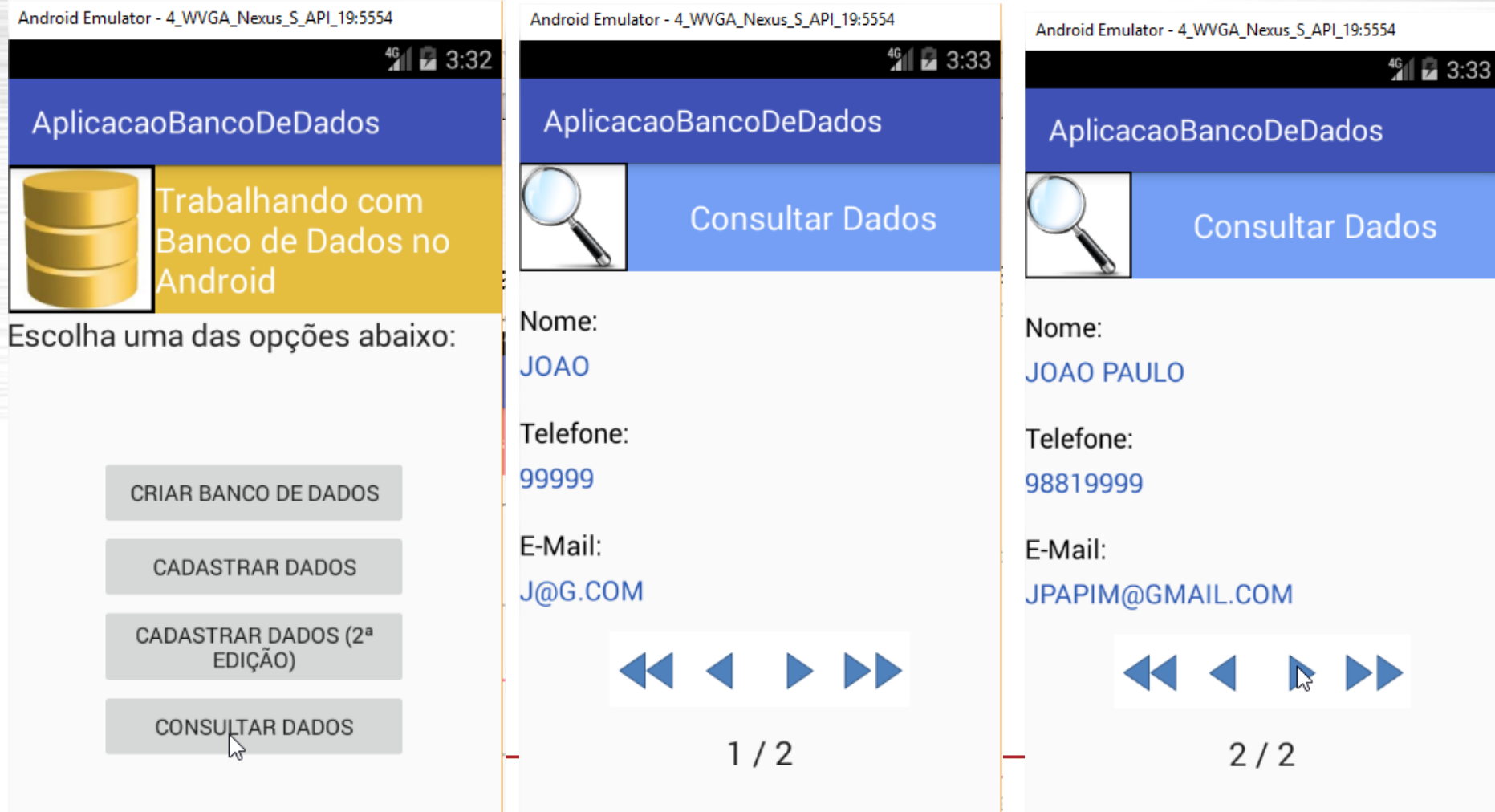
# Aplicação Android com Banco de Dados

- Depois disso vamos executar a nossa aplicação. O resultado você confere nas figuras seguintes:



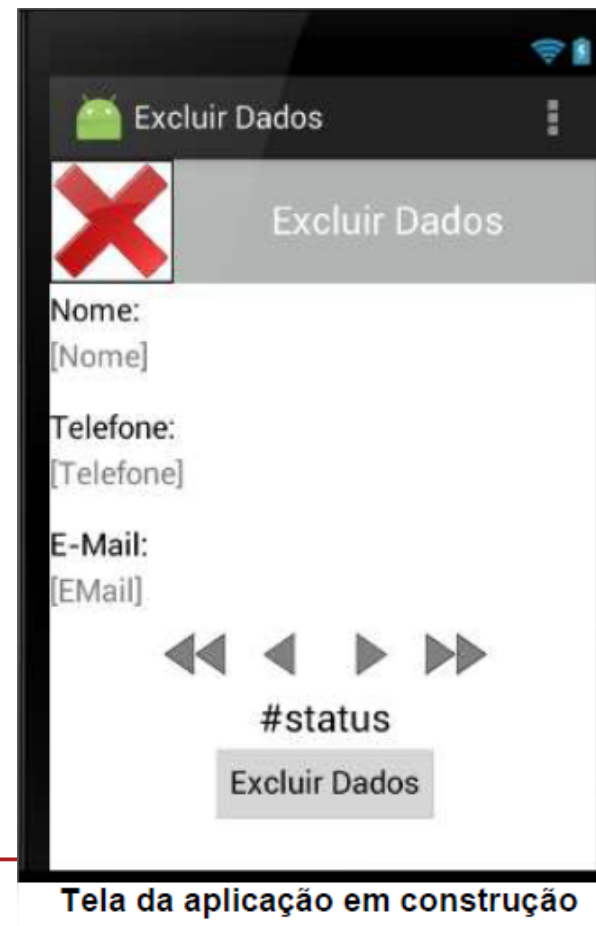
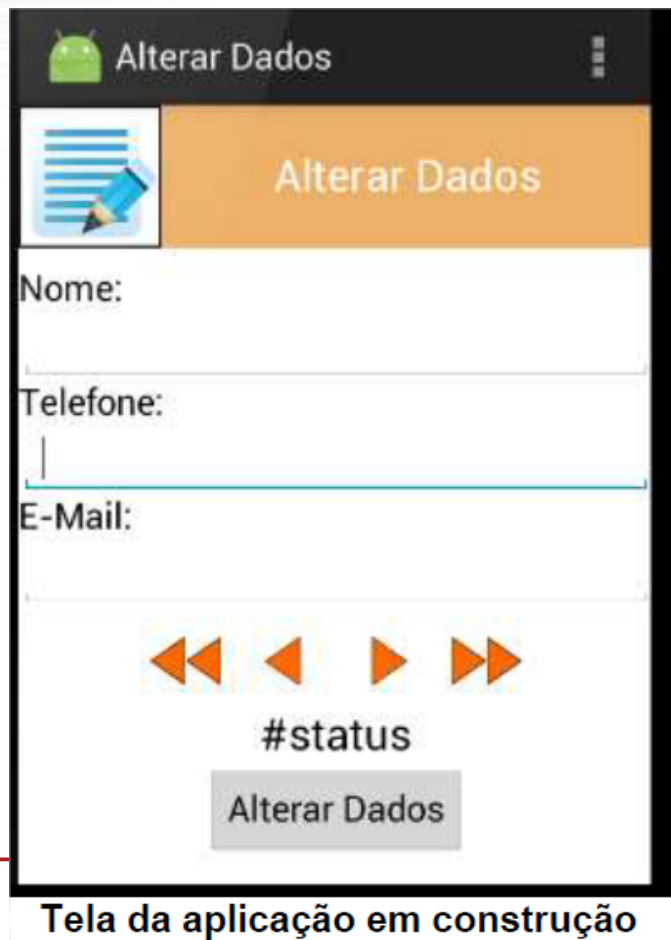
# Aplicação Android com Banco de Dados

- Depois disso vamos executar a nossa aplicação. O resultado você confere na figura seguinte :



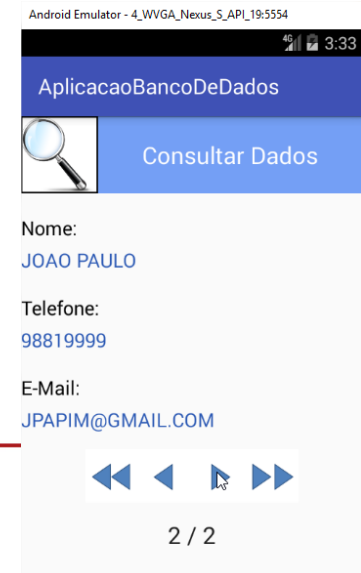
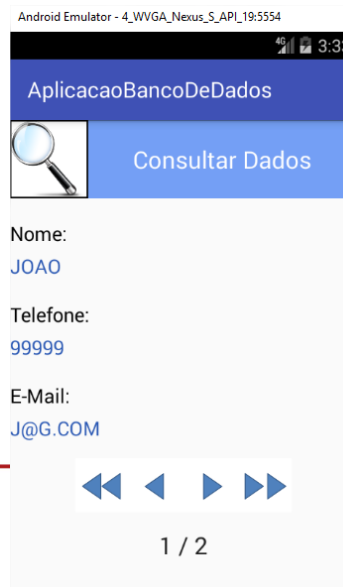
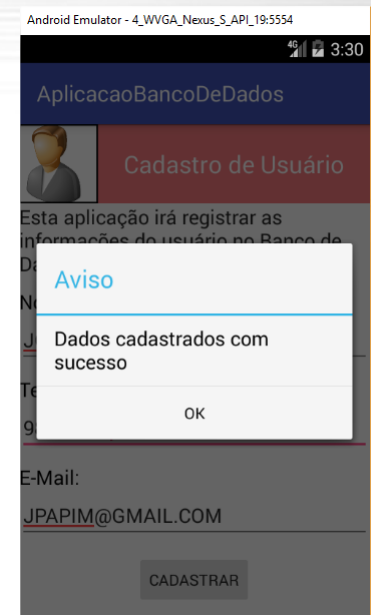
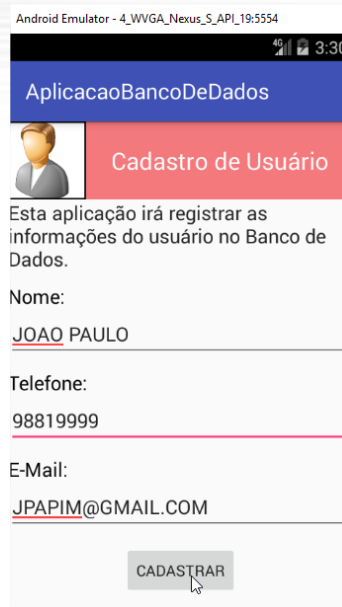
# Aplicação Android com Banco de Dados

- Por hoje é só, na próxima aula iremos aprender a **alterar** e **excluir** os dados cadastrados na nossa aplicação.





# App funcionando até a consulta





**Por hoje é só !!!**

**Até a próxima aula...**

---