



www.apostilaandroid.net



Apostila de Android - Programando Passo a Passo
Programação com Banco de Dados (2ª Edição)
(Versão Android Studio)

Desenvolvida por : Luciano Alves da Silva (lucianopascal@yahoo.com.br)



/ApostilaAndroid

[VENDA E DISTRIBUIÇÃO PROIBIDA]

Outubro/2015



Aviso sobre esta apostila

Antes de iniciar a leitura deste material, veja esses avisos importantes:

Esse material **NÃO PODERÁ SER DISTRIBUÍDO**, em hipótese alguma, em outros sites da Internet ou através outros processos/meios .

Essa material , em hipótese alguma, **NÃO PODE SER COMERCIALIZADO** tanto pela Internet ou de forma impressa.

Se por acaso você ver este material sendo distribuído em outro site ou sendo comercializado (sem ser pelo site oficial da apostila), por favor, entre em contato com o autor (ver e-mail na primeira página).



Sobre o Autor da Apostila

Luciano Alves da Silva é Bacharelado em Ciência da Computação pela UNISUAM e Pós-Graduado em Docência do Ensino Superior pelo Instituto A Vez do Mestre (Universidade Cândido Mendes - UCAM). Possui conhecimento e domínio das linguagens de programação Pascal, Java, C/C++, C#, Visual Basic, Delphi, PHP e HTML. Já criou Ambientes de Desenvolvimento Integrado (conhecidos como IDE) como o MakeWare (que trabalha com as linguagens Pascal, C++ e Java) e o AlgoWare (interpretador de algoritmos).

É autor também dos seguintes livros, pela editora AGBOOK

- Aprenda Passo a Passo a Programar em Android – Guia Essencial para Desenvolvedores
- Desenvolvendo Jogos com a Plataforma XNA 2ª Edição – Guia para Desenvolvedores.
- Desenvolvendo Jogos com o Framework MONOGAME – Guia para Desenvolvedores.
- Desenvolvendo Jogos 2D com Plataforma Java – Guia para Desenvolvedores.



Apresentação

Este material é dedicado para aqueles que desejam trabalhar com banco de dados usando o SQLite, presente na plataforma Android para o armazenamento e gerenciamento de dados.

Caso você, que esteja lendo este material, seja iniciante em programação com Android, recomendo antes adquirir no site oficial (www.apostilaandroid.net) a “Apostila de Android : Programação Básica”, pois esse material nada mais é do que um “complemento” da apostila citada.

Se você por acaso NUNCA VIU PROGRAMAÇÃO NA VIDA, e deseja construir aplicações que manipulem informações através da Web, recomendo antes a aquisição e leitura da “Apostila de Android : Programando DO ZERO”.



Índice analítico

Capítulo 1 – Fundamentos sobre Banco de Dados	6
1.1) A linguagem SQL	6
1.1.1) O comando CREATE TABLE	7
1.1.2) O comando INSERT	7
1.1.3) O comando UPDATE	8
1.1.4) O comando DELETE	8
1.1.5) O comando SELECT	8
Capítulo 2 – Usando Banco de Dados no Android (SQLite)	10
2.1) Trabalhando com o SQLite do Android.....	10
2.1.1) O método query	10
2.1.2) O método insert.....	10
2.1.3) O método update	11
2.1.4) O método delete.....	11
2.1.5) O método execSQL.....	12
2.2) Exemplos de SQL usando as funções do Android	12
2.2.1) Fazendo uma consulta a tabela (usando SELECT)	12
2.2.2) Inserindo dados (usando INSERT)	13
2.2.3) Atualizando dados (usando UPDATE)	13
2.2.4) Removendo dados (usando DELETE)	14
2.2.5) Criando uma tabela (usando CREATE TABLE)	14
Capítulo 3 – Programação com Banco de Dados na prática	18
3.1 – Construindo uma aplicação que irá criar o Banco de Dados	18
3.2 – Registrando informações no Banco (Primeira Versão)	31
3.3 – Registrando informações no Banco (Segunda Versão)	41
3.4 – Consultando as informações da tabela.....	46
3.5 – Alterando as informações da tabela (Primeira Versão).....	59
3.6 – Alterando as informações da tabela (Segunda Versão).....	71
3.7 – Excluindo registros da tabela (Primeira Versão)	82
3.8 – Excluindo registros da tabela (Segunda Versão)	95
Capítulo 4 – Desenvolvendo uma aplicação completa	103
Conclusão a respeito do material	158



Capítulo 1 – Fundamentos sobre Banco de Dados

Atualmente as aplicações Android (seja ela uma aplicação comercial, jogo e etc.) sempre estão guardando informações (como por exemplo, o nome do cliente, endereço e etc.). Essas informações são armazenadas normalmente em um banco de dados, podendo esse banco ser on-line (situado em um servidor para essa finalidade) ou local (situado no próprio dispositivo).

A plataforma Android já oferece (presente em sua arquitetura) um conjunto de recursos (bibliotecas e classes) que permite que nós possamos trabalhar com banco de dados (baseado no SQLite, um sistema gerenciador de banco de dados (SGBD)).

Para trabalharmos com banco de dados no Android, naturalmente, é preciso ter o conhecimento e o domínio da linguagem SQL (Structured Query Language), que é uma linguagem “padrão” voltada para manipulação de informações em um BD (seja ele uma consulta, uma atualização e etc). Antes de começarmos a trabalhar com BD no Android, precisamos conhecer e entender sobre esta linguagem SQL.

1.1) A linguagem SQL

A linguagem SQL (Structured Query Language) é uma linguagem de consulta estruturada que realiza consultas à banco de dados. A SQL é dividida em duas sub-linguagens: LMD (Linguagem de Manipulação de Dados ou Data Manipulation Language – DML) e LDD (Linguagem de Definição de Dados ou Data Definition Language – DDL) . A LDD é um conjunto de comandos dentro da SQL usada para a definição das estruturas de dados, fornecendo as instruções que permitem a criação, modificação e remoção das tabelas, assim como criação de índices. Já a LMD é o grupo de comandos dentro da linguagem SQL utilizado para a recuperação, inclusão, remoção e modificação de informações em bancos de dados.

Vamos conhecer agora alguns comandos básicos muito utilizados da linguagem SQL.



1.1.1) O comando CREATE TABLE

Esse comando serve para criar tabelas em banco de dados. Veja a sua sintaxe abaixo:

```
create table <tabela>(<campo1> <tipo> [<constraints>] [,<campo2> <tipo> [<constraints>],...,<campon> <tipo> [<constraints> ]])
```

Exemplo: Criar uma tabela chamada “cadastro” com os seguintes campos:

- codusuario: Do tipo inteiro e com propriedades de chave primária e auto numeração.
- nome : Do tipo texto e requerido
- idade : Do tipo inteiro e requerido

```
create table cadastro(codusuario integer primary key autoincrement, nome text not null, idade integer not null)
```

Outro exemplo : Criar uma tabela chamada “cadastro” (com os mesmos campos do exemplo acima) no banco de dados, se ela não existir.

```
create table if not exists cadastro(codusuario integer primary key autoincrement, nome text not null, idade integer not null)
```

1.1.2) O comando INSERT

Esse comando serve para inserir dados em uma tabela já existente. Veja a sua sintaxe abaixo:

```
insert into <tabela>(<campo1>,...<campon>)  
values(<valor1>,...,<valorn>)
```

Exemplo: Adicionar dados na tabela cadastro (que foi criada com o comando **CREATE TABLE**), conforme é mostrado abaixo:

nome : Luciano
idade : 30



OBS: Como o campo `codusuario` é do tipo auto-numeração, o valor dele já é definido automaticamente.

```
insert into cadastro(nome,idade)
values('Luciano',23)
```

1.1.3) O comando UPDATE

Esse comando serve para atualizar dados de uma tabela já existente. Veja a sua sintaxe abaixo:

```
update <tabela> set <campo1> = <valor1>[ ,..., <campon> = <valorn>]
where <condição>
```

Exemplo: Atualizar a idade do primeiro registro (Luciano) para 27 anos.

```
update cadastro set idade = 27
where codusuario=1
```

1.1.4) O comando DELETE

Esse comando serve para remover dados de uma tabela já existente. Veja a sua sintaxe abaixo:

```
delete from <tabela> where <condição>
```

Exemplo: Remover o primeiro registro da tabela (Luciano)

```
delete from cadastro where codusuario=1
```

1.1.5) O comando SELECT

Esse comando serve para fazer consultas em uma tabela já existente. Veja a sua sintaxe abaixo:

```
select <campo1>[,<campo2>,...,<campon>]
from <tabela>
where <condição>
```




Exemplos :

Mostrar todos os dados cadastrados na tabela

```
select codusuario,nome,idade  
from cadastro
```

Mostrar todos os nomes cuja idade seja maior que 25.

```
select nome  
from cadastro  
where (idade >= 25)
```

Mostrar todos os nomes cujo prefixo seja 'David', ou melhor, todos que se chamam David independente do sobrenome.

```
select nome  
from cadastro  
where (nome like 'David%')
```

A partir de agora vamos ver abaixo algumas funções do Android que trabalham com banco de dados.



Capítulo 2 – Usando Banco de Dados no Android (SQLite)

2.1) Trabalhando com o SQLite do Android

No Android existe um pacote chamado “android.sqlite” que possui várias classes, muitas delas para o tratamento de banco de dados. Uma dessas classes (que é a que vamos trabalhar neste material) a classe chamada **SQLiteDataBase**, que possui os seguintes métodos, conforme é mostrado em seguida:

2.1.1) O método query

O método **query** realiza uma consulta SQL no banco de dados (equivalente ao comando **SELECT**). Essa função retorna uma instância (objeto) do tipo **Cursor**. Vejamos a sintaxe do método em seguida:

```
Cursor query(String <tabela>, String[] <colunas>, String <condicao_de_busca>,  
String[] <selecionArgs>, String <groupby>, String <having>, String <orderby>)
```

Vamos conhecer os parâmetros que iremos utilizar dessa função:

<tabela> : Neste parâmetro você informa o nome da tabela.

<coluna> : Neste parâmetro a(s) coluna(s) que o banco possui.

<condição de busca>: Aqui você informa a condição de busca de dados. Este parâmetro funciona como se fosse a cláusula where do SQL, onde é informada a condição de busca.

O restante dos parâmetros iremos trabalhar em seu valor **null**.

2.1.2) O método insert

O método **insert** realiza uma inserção de dados na tabela (equivalente ao comando **INSERT**). Vejamos a sintaxe deste método:



```
long insert(String <tabela>, String <nullColumnHack>, ContentValues <values>)
```

Essa função possui três parâmetros:

<tabela> : Neste parâmetro você informa o nome da tabela.

<nullColumnHack> : O SQL não permite a inserção de linhas em branco, logo, se o valor de uma coluna for vazio, ele será iniciado com o valor **null**.

<values> : Este parâmetro possui os valores a serem adicionados na tabela.

2.1.3) O método update

O método **update** realiza uma atualização de dados na tabela (equivale ao comando **UPDATE** do SQL). Vejamos agora a sintaxe deste método :

```
int update(String <tabela>, ContentValues <values>, String <condição>, String  
<argumentos_condicao>)
```

Essa função possui três parâmetros:

<tabela> : Neste parâmetro você informa o nome da tabela.

<values> : Este parâmetro possui os valores a serem adicionados na tabela.

<condição>: Aqui você informa a condição para a realização da atualização dos dados.

<argumentos_condicao>: Aqui você informa os argumentos relativos à condição informada.

2.1.4) O método delete

O método **delete** realiza a remoção de dados na tabela (equivale ao comando **DELETE** do SQL). Vejamos a sintaxe deste método:

```
int delete(String <tabela>, String <condição>, String <argumentos_condicao>)
```



Essa função possui três parâmetros:

<tabela> : Neste parâmetro você informa o nome da tabela.

<condição>: Aqui você informa a condição para a realização da remoção dos dados da tabela.

<argumentos_condicao>: Aqui você informa os argumentos relativos à condição informada.

2.1.5) O método execSQL

O método **execSQL** executa uma consulta SQL (como **CREATE TABLE**, **INSERT INTO**, **UPDATE**, **DELETE** e etc.). Não é possível usar a cláusula **SELECT** nesta função. Para esse tipo de situação, use o método **query**. Vejamos a sintaxe deste método:

```
void execSQL(String sql)
```

Vamos fazer algumas comparações da função “execSQL”, que permite sintaxes de comando SQL com as demais funções, como “update”, “insert” e “delete” para a realização de uma consulta SQL.

2.2) Exemplos de SQL usando as funções do Android

2.2.1) Fazendo uma consulta a tabela (usando SELECT)

- Comando SQL :

```
select codusuario, nome, idade  
from cadastro
```

- Usando a função query :

```
query("cadastro", (new String[]  
{ "codusuario", "nome", "idade" }), null, null, null,  
null, null);
```



- Comando SQL :

```
select nome  
from cadastro  
where idade > 24
```

- Usando a função query :

```
query("cadastro", (new String[] {"nome"}), "idade > 24"  
    , null, null, null, null);
```

2.2.2) Inserindo dados (usando INSERT)

- Comando SQL :

```
insert into cadastro(nome, idade)  
values('Luciano', 23)
```

- Usando a função execSQL

```
execSQL("insert into cadastro(nome, idade) values('Luciano', 23); ");
```

Usando a função insert :

```
ContentValues valor = new ContentValues();
```

```
valor.put("nome", "Luciano" );  
valor.put("idade", "23");
```

```
insert("cadastro", null, valor);
```

2.2.3) Atualizando dados (usando UPDATE)

- Comando SQL :

```
update cadastro set idade = 27  
where codusuario=1
```




- Usando a função execSQL

```
execSQL("update cadastro set idade = 27 where (codusuario=1); ");
```

Usando a função update :

```
ContentValues valor = new ContentValues();  
valor.put("idade", "27");  
update("cadastro", valor, "codusuario=1", null);
```

2.2.4) Removendo dados (usando DELETE)

- Comando SQL :

```
delete from cadastro where codusuario=1
```

- Usando a função execSQL

```
execSQL("delete from cadastro where (codusuario=1); ");
```

Usando a função delete :

```
delete("cadastro", "codusuario=1", null);
```

2.2.5) Criando uma tabela (usando CREATE TABLE)

- Comando SQL :

```
create table cadastro(codusuario integer primary key autoincrement, nome text  
not null, idade integer not null)
```

- Usando a função execSQL

```
execSQL("create table cadastro(codusuario integer primary key autoincrement,  
nome text not null, idade integer not null); ");
```



Agora vamos conhecer as funções responsáveis por criar e abrir banco de dados no Android.

```
SQLiteDatabase openOrCreateDatabase(String nome_do_banco,int mode,
CursorFactory cf);
```

Essa função abre ou cria um novo banco de dados. Você deve especificar o nome do banco e o modo de abertura (somente leitura ; somente escrita e etc.) e um terceiro parâmetro, que normalmente é **null**. Veja um exemplo abaixo:

```
SQLiteDatabase db;
```

```
db = openOrCreateDatabase("dbbanco", Context.MODE_PRIVATE, null);
```

O comando acima abre ou cria o banco de dados chamado “dbbanco”.

Quando realizamos uma consulta do tipo **SELECT** usando a função **query**, ela retorna um objeto do tipo **Cursor**, onde nela estão armazenados os registros solicitados pela consulta. Vamos ver abaixo os métodos da classe **Cursor**:

Método	Descrição
boolean moveToFirst()	Move o cursor para o primeiro registro da tabela.
boolean moveToPrevious()	Move o cursor para o registro anterior da tabela.
boolean moveToNext()	Move o cursor para o próximo registro da tabela.
boolean moveToLast()	Move o cursor para o último registro da tabela.
int getCount()	Retorna o número de registros da tabela.
int getColumnIndex(String columnName)	Retorna o índice da coluna na tabela, através do seu nome, que é passado como parâmetro.
String getColumnName(int columnIndex)	Retorna o nome da coluna na tabela, através do seu índice, que é passado como parâmetro.
int getInt(int columnIndex)	Retorna o valor do campo, tendo como seu parâmetro o seu índice, convertido em int. Lembre-se : o índice do primeiro campo é 0, o índice do segundo campo é 1 e assim por diante.



Método	Descrição
float getFloat(int columnIndex)	Retorna o valor do campo, tendo como seu parâmetro o seu índice, convertido em float. Lembre-se : o índice do primeiro campo é 0, o índice do segundo campo é 1 e assim por diante.
double getDouble(int columnIndex)	Retorna o valor do campo, tendo como seu parâmetro o seu índice, convertido em double. Lembre-se : o índice do primeiro campo é 0, o índice do segundo campo é 1 e assim por diante.
short getShort(int columnIndex)	Retorna o valor do campo, tendo como seu parâmetro o seu índice, convertido em short. Lembre-se : o índice do primeiro campo é 0, o índice do segundo campo é 1 e assim por diante.

Vou mostrar agora um exemplo para que você tenha um melhor entendimento do mecanismo de banco de dados do Android.

Imagine uma tabela chamada “cadastro” com os seguintes dados abaixo:

nome	idade
Amanda	32
Bianca	30
Bruna	23
Carla	20

Agora, observe a linha de código abaixo:

```
SQLiteDatabase db;
```

```
db = openOrCreateDatabase("dbbanco",  
Context.MODE_PRIVATE, null);
```

```
Cursor c = db.query("cadastro", (new String[] {"nome","idade"}), "idade <  
32", null, null, null, null);
```

Observe que a linha acima cria um objeto do tipo **Cursor** que vai receber o resultado da consulta da função query, que retorna uma instância do mesmo tipo. Logo, a instância retornada pela função query na verdade, retorna uma tabela resultante da consulta. Veja esse resultado abaixo:



nome	idade
Bruna	23
Carla	20

Continuando a codificação. Veja a linha abaixo:

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

A linha acima coloca o ponteiro no primeiro registro da tabela. A linha:

```
String nome = c.getString(0);
```

Retorna o valor do campo “nome” do primeiro registro, no caso, “Bruna”. Veja agora a próxima linha:

```
int idade = c.getInt(1);
```

Retorna o valor do campo “idade” do primeiro registro, no formato int. Neste caso, o valor retornado é 23. A linha:

```
c.moveToNext();
```

A linha acima avança para o próximo registro. A linha:

```
nome = c.getString(0);
```

Retorna o valor do campo “nome” do segundo registro, no caso, “Carla”. Veja agora a próxima linha:

```
int idade = c.getInt(1);
```

Retorna o valor do campo “idade” do segundo registro, no formato int. Neste caso, o valor retornado é 20.

Bom, com certeza você entendeu como funciona o mecanismo de manipulação de banco de dados no Android. Agora vamos por esse aprendizado na prática, começando desenvolvendo algumas aplicações Android básicas.



Capítulo 3 – Programação com Banco de Dados na prática

No capítulo anterior vimos os comandos de SQL presentes na classe **SQLiteDataBase** para a manipulação de informações. Agora iremos ver na prática como usar os comandos abordados no capítulo anterior através da construção de pequenas aplicações básicas, passo a passo.

A ferramenta que utilizaremos aqui nessa apostila para criarmos as nossas aplicações em Android é a IDE Android Studio. Caso você não tenha instalado o Android Studio no seu computador e não sabe os procedimentos de instalação, visite o link : <https://goo.gl/MgImX8>

3.1 – Construindo uma aplicação que irá criar o Banco de Dados

A nossa primeira aplicação que iremos construir irá criar o nosso banco , como também a nossa tabela que iremos utilizar para inserir e guardar as nossas informações.

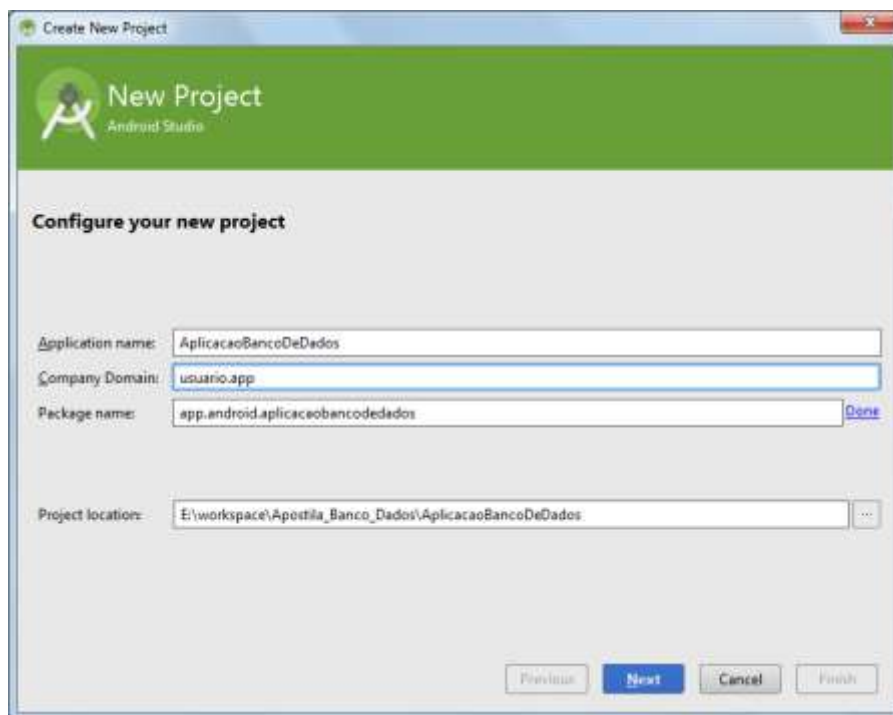
Para isso, com o Android Studio aberto, vamos criar um novo projeto de acordo com as informações abaixo :

Application Name : AplicacaoBancoDeDados

Company Domain : <fica a sua escolha>

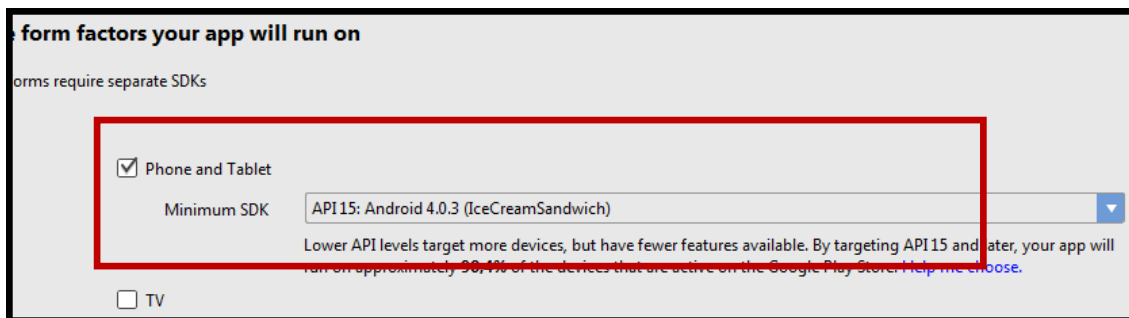
Packgame name : app.android. aplicacaobancodedados

Veja como ficou na figura a seguir :



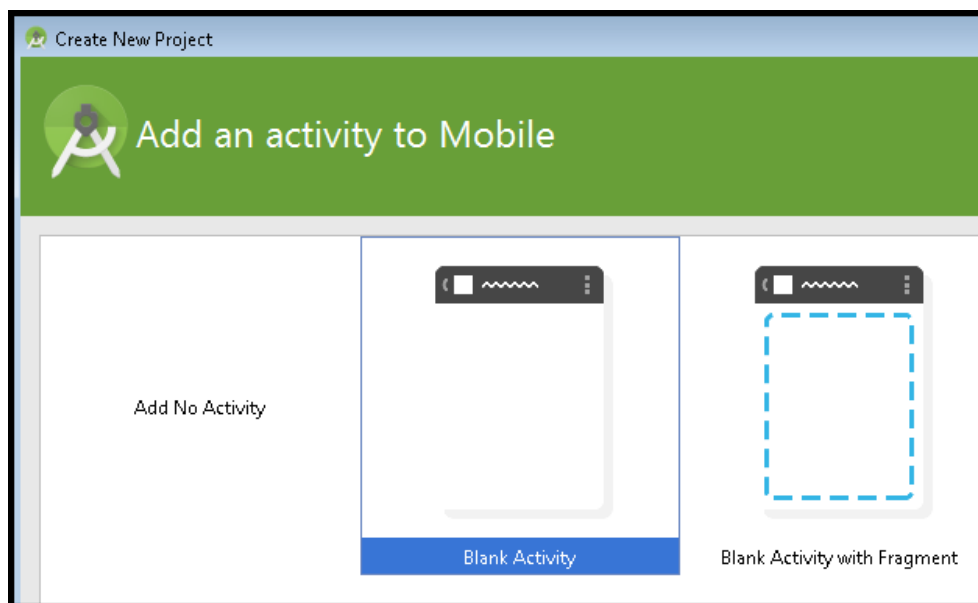
Criando um novo projeto

Depois basta clicarmos no botão “Next” para prosseguirmos. Na tela seguinte deixe marcada a opção “Phone and Tablet”, conforme é indicado a seguir :



Deixar marcada esta opção

Feito isso clique em “Next”. Na opção seguinte da próxima tela deixe marcada a opção “Blank Activity”. Veja na figura abaixo :



Deixar marcada esta opção

Clique em “Next” novamente para prosseguirmos. No formulário que surge preencha as informações a seguir:

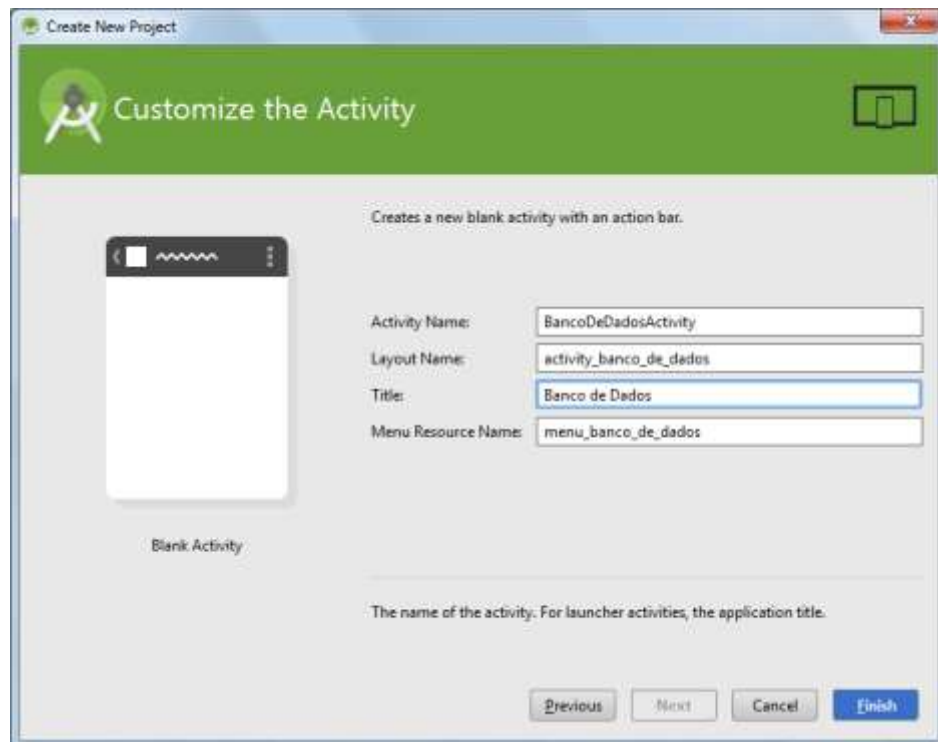
Activity Name : BancoDeDadosActivity

Layout Name : activity_banco_de_dados

Title : Banco de Dados

Menu Resource Name : menu_banco_de_dados

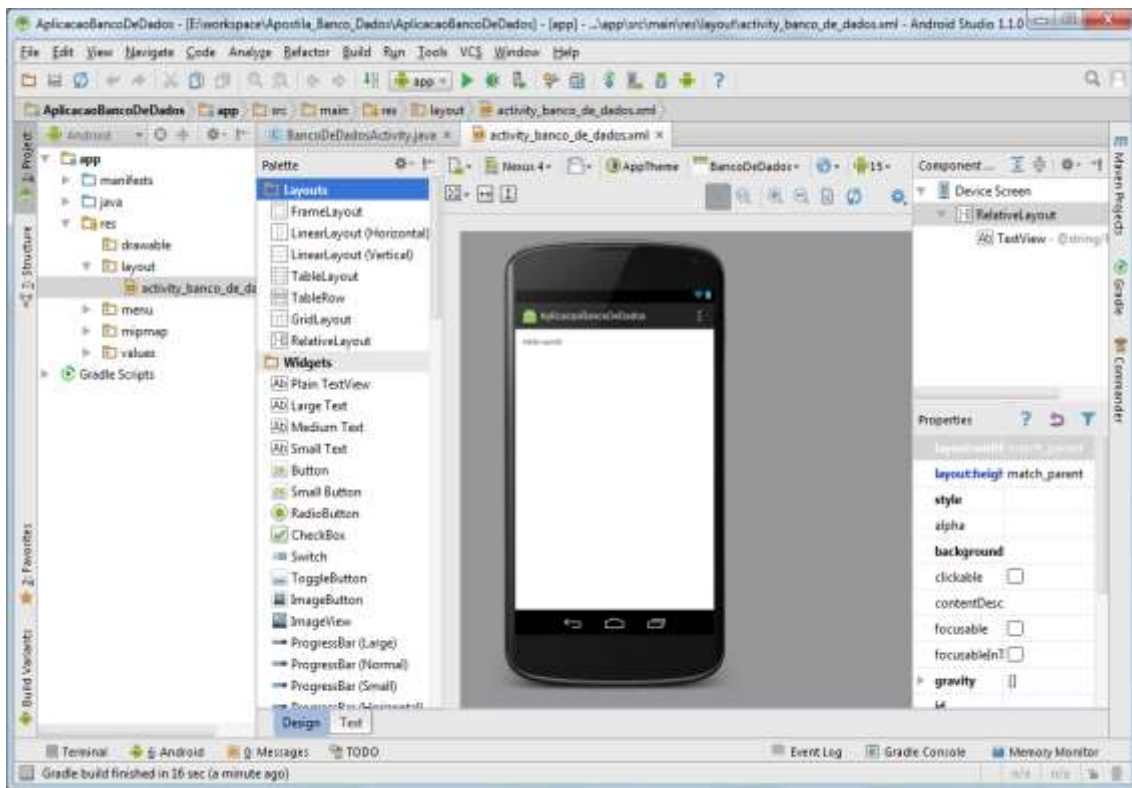
Veja como ficou na figura a seguir :



Dados preenchidos

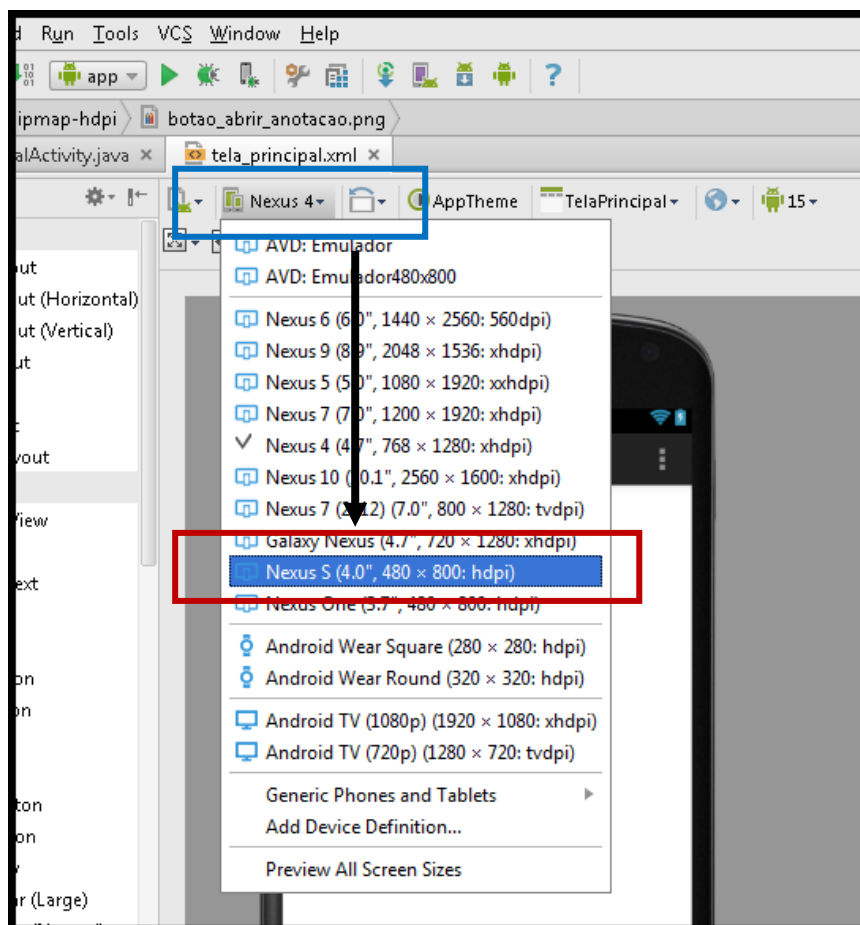
Depois disso basta clicarmos no botão “Finish” para que o projeto possa ser criado.

Vejamos o resultado:



Projeto Criado com Sucesso

Para o desenvolvimento deste projeto irei trabalhar com a Skin “Nexus S” (cuja resolução é 480x800 hdpi). Para alterar a Skin basta seguir o procedimento a figura a seguir:



Mudando a Skin

Depois de criar o nosso projeto no Android Studio vamos colocar dentro do diretório “drawable” (diretório de imagens, presente dentro da pasta “res” do projeto) todas as imagens da pasta “capítulo 3”, que acompanham este material.

Feito o que foi solicitado acima, vamos escrever o seguinte código XML para a tela da nossa aplicação:

```
<?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:orientation="vertical" >

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#3ba0e2" >
        <ImageView
            android:id="@+id/imageView1"
```




```

        android:layout_width="96dp"
        android:layout_height="96dp"
        android:src="@drawable/icone_bd" />

```

```

<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="#E5BE40"
    android:gravity="center" >

```

```

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Trabalhando com Banco de
        Dados no Android"
        android:textAppearance=
            "?android:attr/textAppearanceLarge"
        android:textColor="#ffffff" />

```

```

</LinearLayout>

```

```

</LinearLayout>

```

```

<LinearLayout
    android:id="@+id/LayoutImagemFundo"
    android:layout_width="match_parent"
    android:layout_height="fill_parent"

```

```

    android:orientation="vertical" >

```

```

    <TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Escolha uma das opções abaixo:"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:textSize="20sp" />

```

```

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="fill_parent"
        android:gravity="center"
        android:orientation="vertical" >

```



```

        <Button
            android:id="@+id/btcriarbanco"
            android:layout_width="200dp"
            android:layout_height="wrap_content"
            android:text="Criar Banco de Dados" />
    </LinearLayout>
</LinearLayout>
</LinearLayout>

```

Feito isso teremos o seguinte resultado abaixo:



Tela da nossa aplicação

Agora vamos abrir o arquivo "BancoDeDadosActivity.java" (presente dentro do pacote "app.android.aplicacaobancodedados") para digitarmos o seguinte código seguinte:

```

package app.android.aplicacaobancodedados;

import android.os.Bundle;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;

```



```
import android.view.Menu;
import android.view.View;
import android.widget.*;
import android.view.*;

public class BancoDeDadosActivity extends Activity {

    Button btcriarbanco;
    SQLiteDatabase db;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_cria_banco_de_dados);
        btcriarbanco = (Button) findViewById(R.id.btcriarbanco);
        btcriarbanco.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View view) {

                try {

                    db = openOrCreateDatabase("banco_dados",
                        Context.MODE_PRIVATE, null);

                    db.execSQL("create table if not exists " +
                        " usuarios(numreg integer primary key " +
                        " autoincrement, nome text not null, telefone text " +
                        + "not null," + "email text not null)");

                    AlertDialog.Builder dialog = new
                    AlertDialog.Builder(BancoDeDadosActivity.this);
                    dialog.setTitle("Aviso")
                    .setMessage("Banco de dados criado com sucesso")
                    .setNeutralButton("OK", null)
                    .show();
                } catch (Exception e){

                }

            }
        });
    }
}
```

Irei comentar algumas linhas de código do programa. A instrução :

```
db = openOrCreateDatabase("banco_dados",
    Context.MODE_PRIVATE, null);
```

Cria (ou abre, caso já exista) o nosso banco de dados, cujo nome se chama "banco_dados" através do método **openOrCreateDataBase** do objeto **db** .

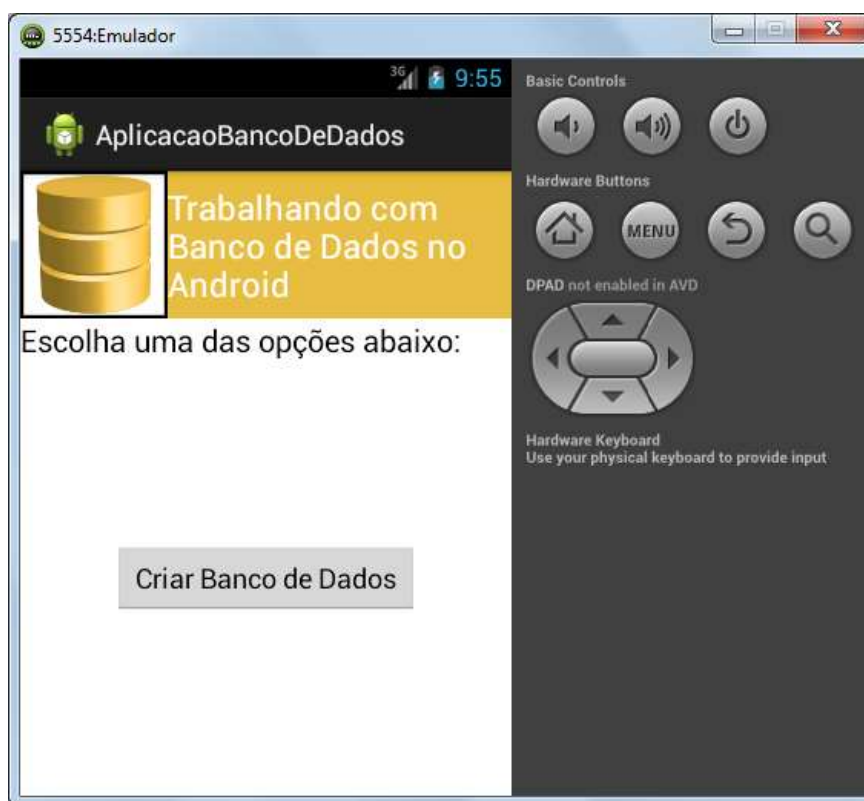


A próxima instrução :

```
db.execSQL("create table if not exists " +
" usuarios(numreg integer primary key " +
" autoincrement, nome text not null, telefone text " +
" not null," + "email text not null)");
```

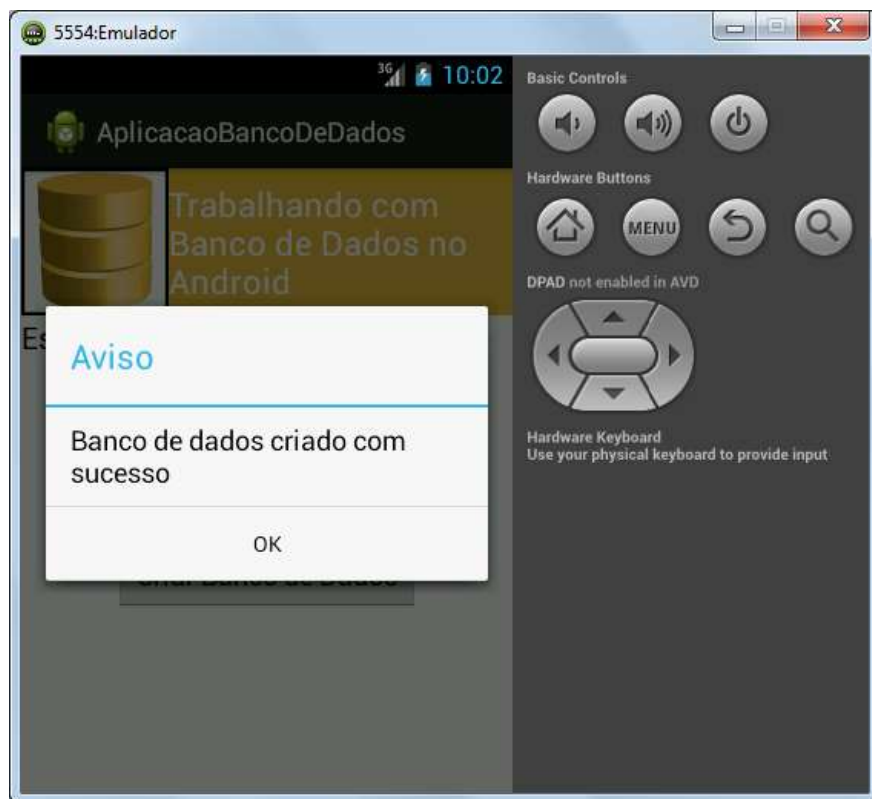
Cria uma tabela (caso ela não exista) dentro do banco de dados chamada “usuarios”, com os seus devidos campos, através do método **execSQL**. No parâmetro do método passamos uma “string” que contém uma instrução SQL do tipo **create table**, que irá criar a tabela que iremos trabalhar em nossa aplicação.

Vamos executar a nossa aplicação, o resultado você confere na figura seguinte:



Aplicação em execução

Para criarmos o nosso banco basta clicarmos no botão presente em nossa aplicação. Feito isso será exibida a seguinte mensagem abaixo:

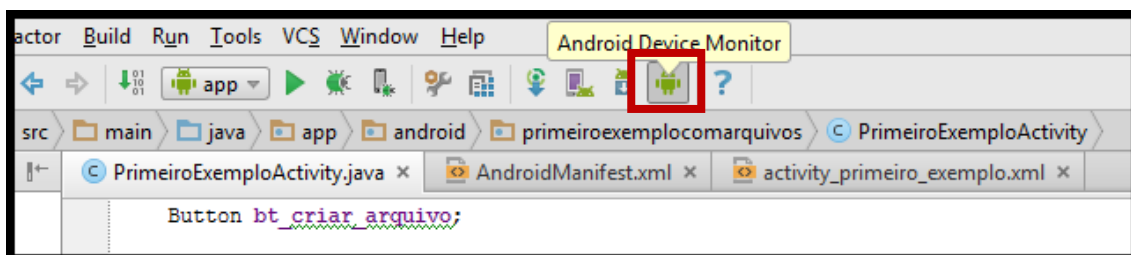


Aplicação em execução – Banco de dados criado com sucesso

Como confirmar se o banco de dados realmente foi criado ?

Para confirmarmos se o banco de dados foi realmente criado, podemos utilizar o recurso **File Explorer** (que se encontra dentro do Android Device Monitor).

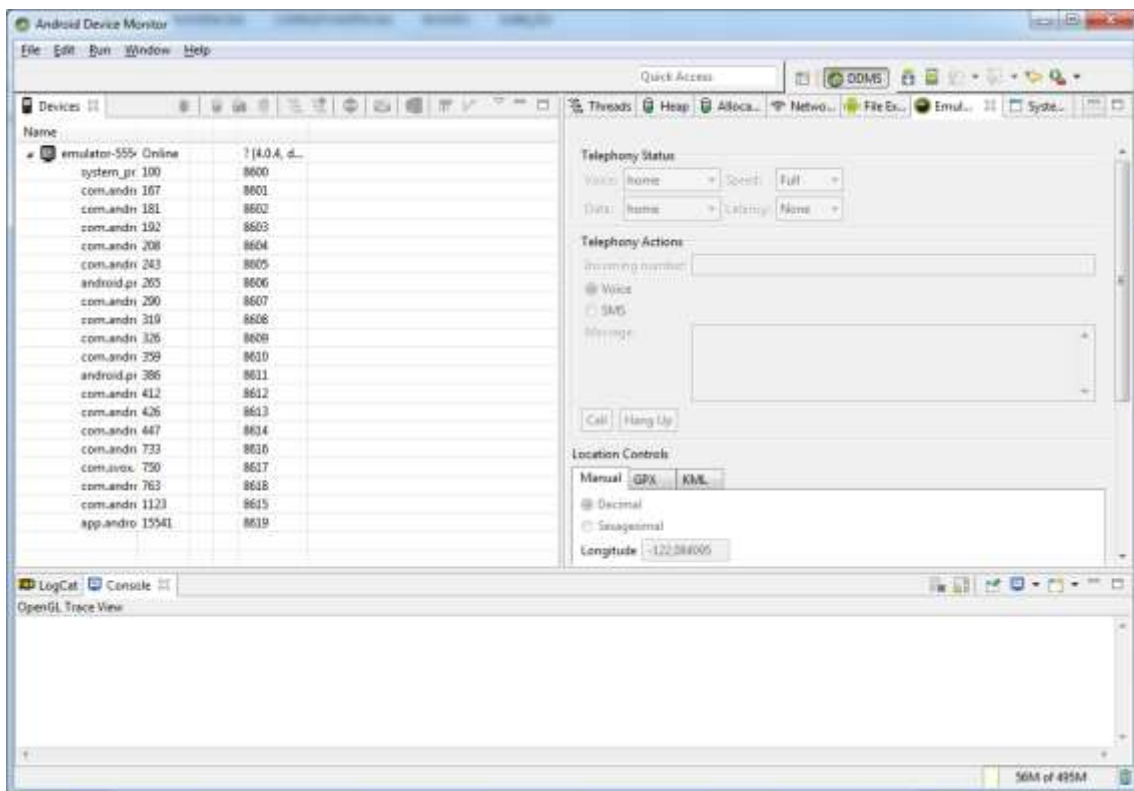
Para abrirmos o “File Explorer” clique no botão do “Android Device Monitor” situado na barra de menu do Android Studio, conforme mostra a figura a seguir:



Chamando o Android Device Monitor

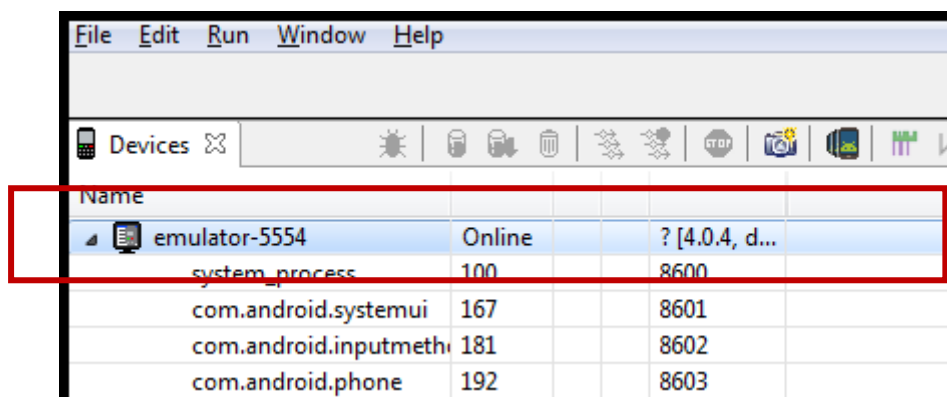


Feito isso será aberta o seguinte painel a seguir :



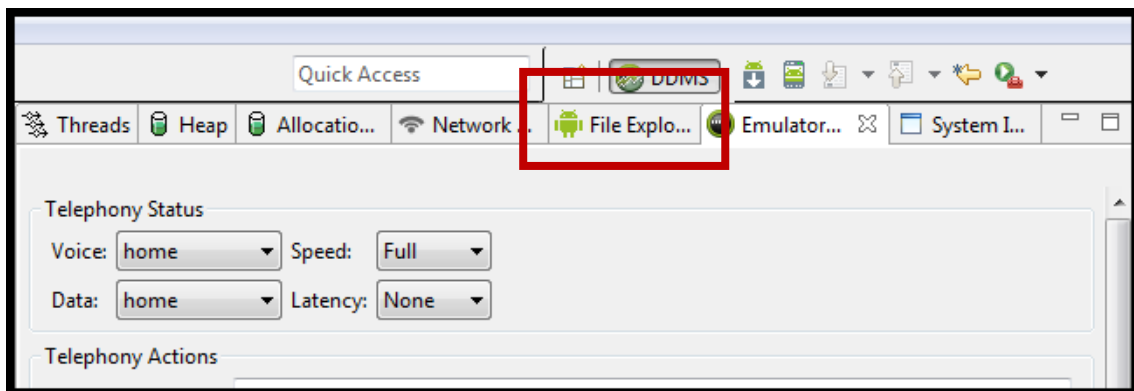
Caixa de diálogo – Android Device Monitor

Para visualizarmos a estrutura de diretórios selecione primeiramente o dispositivo em execução na guia “Devices”, conforme acompanhamos logo na imagem a seguir :



Selecionando o dispositivo em execução

Feito isso, clique na guia “File Explorer” para visualizarmos a estrutura de arquivos do dispositivo selecionado :



Selecionando a guia “File Explorer”

Vejamos na figura seguinte a estrutura diretórios apresentada abaixo :

Name	Size	Date	Time	Permissions	Info
▶ folder acct		2015-09-01	16:58	drwxr-xr-x	
▶ folder cache		2015-08-12	17:05	drwxrwx---	
▶ folder config		2015-09-01	16:58	dr-x-----	
file d		2015-09-01	16:58	lrwxrwxrwx	-> /sys/ker...
▶ folder data		2015-07-02	23:34	drwxrwx--x	
file default.prop	116	1970-01-01	00:00	-rw-r--r--	
▶ folder dev		2015-09-01	16:58	drwxr-xr-x	
folder etc		2015-09-01	16:58	lrwxrwxrwx	-> /system...
file init	98676	1970-01-01	00:00	-rwxr-x---	
file init.goldfish.rc	2344	1970-01-01	00:00	-rwxr-x---	
file init.rc	17105	1970-01-01	00:00	-rwxr-x---	
▶ folder mnt		2015-09-01	16:58	drwxrwxr-x	
▶ folder proc		1970-01-01	00:00	dr-xr-xr-x	
▶ folder root		2011-11-14	19:00	drwx-----	
▶ folder sbin		1970-01-01	00:00	drwxr-x---	
folder sdcard		2015-09-01	16:58	lrwxrwxrwx	-> /mnt/s...
▶ folder sys		1970-01-01	00:00	drwxr-xr-x	
▶ folder system		2012-03-27	23:21	drwxr-xr-x	
file ueventd.goldfish.rc	272	1970-01-01	00:00	-rw-r--r--	
file ueventd.rc	3825	1970-01-01	00:00	-rw-r--r--	
file vendor		2015-09-01	16:58	lrwxrwxrwx	-> /system...

Estrutura de diretórios do “File Explorer”

O diretório onde o nosso banco de dados foi criado dentro do Android é o “data/data/app.android.aplicacaobancodedados/databases/”. Confira na figura seguinte :



Name	Size	Date	Time
▶ backup		2015-09-29	21:33
▶ dalvik-cache		2015-09-29	21:39
▲ data		2015-09-29	21:36
▶ android.usuario.helloandroid		2015-04-15	16:30
▲ app.android.aplicacaobancodedados		2015-09-29	21:39
▶ cache		2015-09-29	21:36
▲ databases		2015-09-29	21:39
banco_dados	5120	2015-09-29	21:39
banco_dados-journal	0	2015-09-29	21:39
▶ lib		2015-09-29	21:36
▶ app.android.notingnet		2015-06-27	17:28
▶ app.android.primeiroexemplocomarquivos		2015-09-01	16:20
▶ app.usuario.conversortemperatura		2015-07-27	21:13
▶ app.usuario.conversortemperatura2		2015-07-27	22:46
▶ app.usuario.conversortemperatura3		2015-08-01	14:25

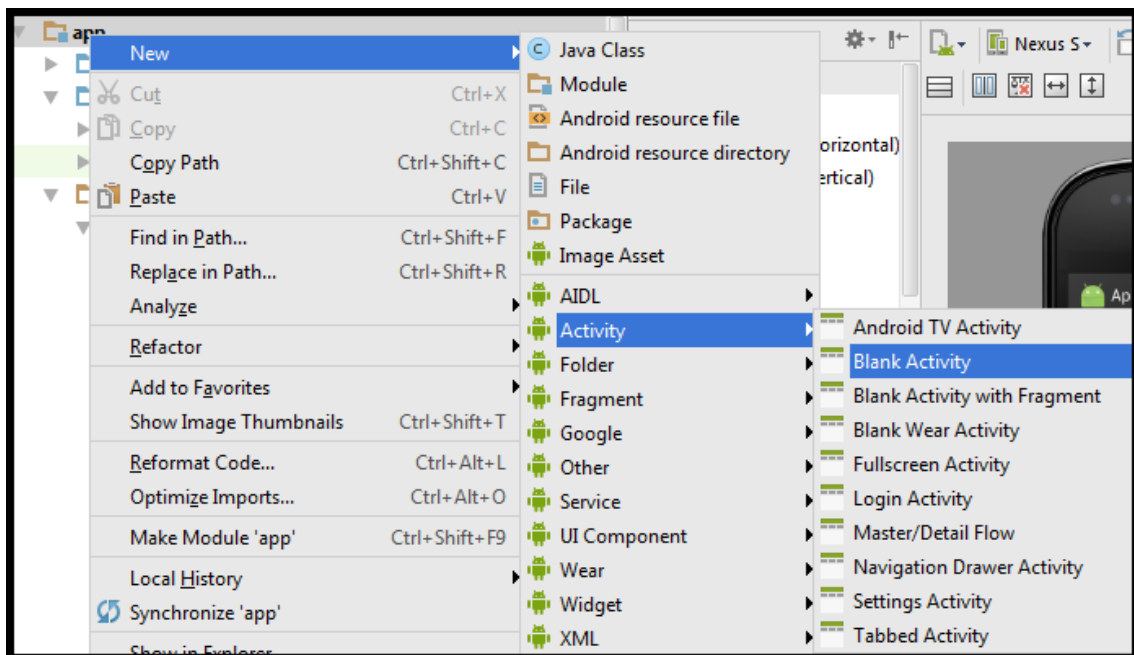
Visualizando o banco de dados

3.2 – Registrando informações no Banco (Primeira Versão)

Vamos construir agora, dentro do mesmo projeto, uma Activity que irá registrar as informações dentro da nossa tabela dentro do Banco de Dados (a tabela “usuarios”). Esse processo estará dividido em duas versões. Essa será a primeira versão, onde iremos trabalhar com o método **execSQL** para a inserção das informações na tabela (através de comando SQL).

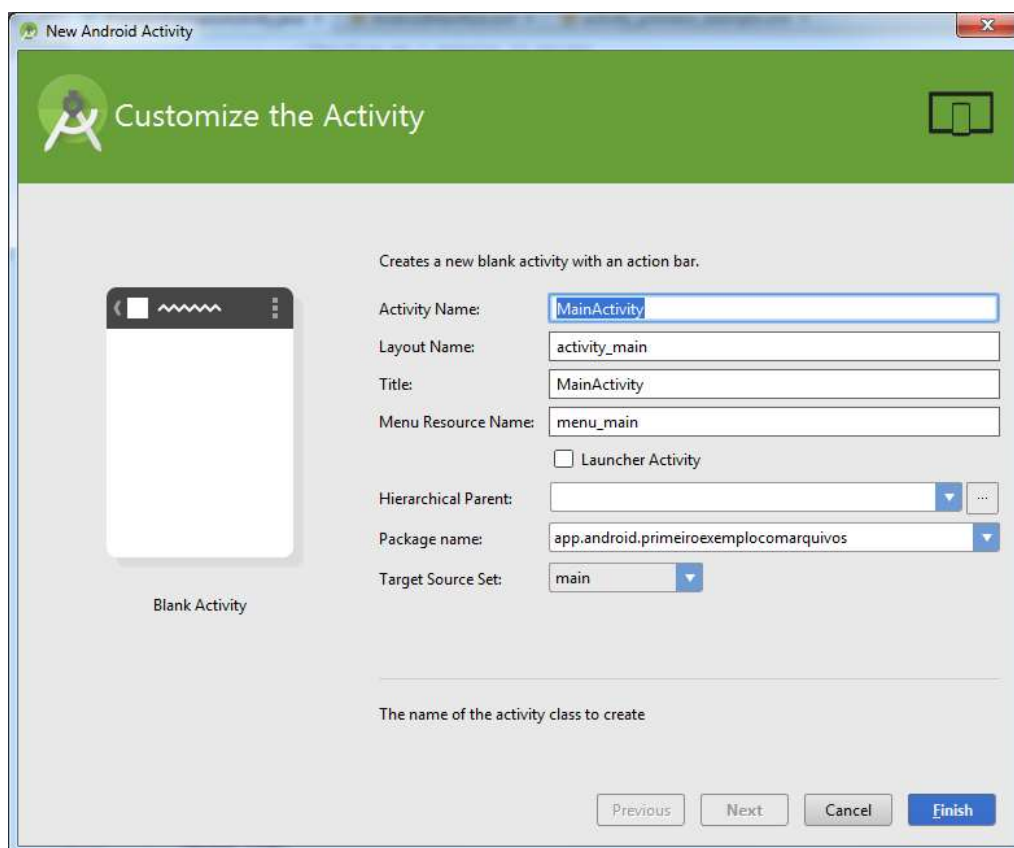
Vamos criar agora um nova Activity, seguindo os passos a seguir:

Clique com o botão direito do mouse sobre o diretório “app” do nosso projeto e selecione “New / Activity / Blank Activity”. Veja na figura seguinte :



Criando uma nova Activity

Feito isso será aberta a seguinte caixa de diálogo a seguir :



Caixa de diálogo – New Android Activity



Agora na caixa de diálogo que aparece, digite as seguintes informações da Activity a ser criada abaixo:

Activity Name : GravaRegistrosActivity

Layout Name : activity_grava_registros

Title : Gravar Registros

Menu Resource Name : menu_grava_registros

Clique em “Finish” para que a nossa Activity possa ser criada.

Depois de criada a Activity, abra o arquivo “activity_grava_registros.xml” para digitarmos o seguinte código XML abaixo:

```
<?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:orientation="vertical" >

    <LinearLayout
        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_cadastro" />

        <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="#f4797d"
        android:gravity="center" >

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




```

        android:text="Cadastro de Usuário"
        android:textAppearance=
            "?android:attr/textAppearanceLarge"
        android:textColor="#ffffff" />

```

```
</LinearLayout>
```

```
</LinearLayout>
```

```
<LinearLayout
```

```

    android:id="@+id/LayoutImagemFundo"
    android:layout_width="match_parent"
    android:layout_height="fill_parent"

```

```
    android:orientation="vertical" >
```

```
<TextView
```

```

    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Esta aplicação irá registrar as
    informações do usuário no Banco de Dados."
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:textSize="18sp" />

```

```
<TextView
```

```

    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingTop="10dp"
    android:text="Nome:"
    android:textColor="#000000"
    android:textSize="18sp" />

```

```
<EditText
```

```

    android:id="@+id/ednome"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" >

```

```
        <requestFocus />
```

```
</EditText>
```

```
<TextView
```

```

    android:id="@+id/TextView01"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingTop="10dp"
    android:text="Telefone:"
    android:textColor="#000000"
    android:textSize="18sp" />

```



```
<EditText
    android:id="@+id/edtelefone"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" />
<TextView
    android:id="@+id/TextView02"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingTop="10dp"
    android:text="E-Mail:"
    android:textColor="#000000"
    android:textSize="18sp" />

<EditText
    android:id="@+id/edemail"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="fill_parent"
    android:gravity="center" >

    <Button
        android:id="@+id/btcadastrar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Cadastrar" />

</LinearLayout>

</LinearLayout>

</LinearLayout>
```

Feito o que foi solicitado, salve o arquivo "activity_grava_registros.xml".
Veamos na figura seguinte o resultado:



Tela da nossa aplicação

Agora dentro do arquivo “GravaRegistrosActivity.java” vamos digitar o seguinte código abaixo:

```
package app.android.aplicacaobancodedados;

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

public class GravaRegistrosActivity extends Activity {

    Button btcadastrar;
    EditText ednome, edtelefone, edemail;

    SQLiteDatabase db;
```



```

@Override
protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_grava_registros);
    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());
    }

    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();

            try {

                db.execSQL("insert into usuarios(nome," +
                    "telefone, email) values('" + nome + "','" +
                    + telefone + "','" + email + "')");

                MostraMensagem("Dados cadastrados com
                    sucesso");

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

        }

    });
}

```



```

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

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

```

Vamos a explicação de algumas linhas de código. A instrução:

```

String nome = ednome.getText().toString();
String telefone = edtelefone.getText().toString();
String email = edemail.getText().toString();

```

Obtém as informações digitadas nos campos “nome”, “telefone” e “email” (representado respectivamente pelos componentes/objetos *ednome*, *edtelefone* e *edmail*) e atribui os seus valores para as variáveis de mesmo nome : *nome*, *telefone* e *email*.

Na instrução seguinte :

```

db.execSQL("insert into usuarios(nome,telefone, email) values('" +
nome + "', '" + telefone + "', '" + email + "')");

```

Realiza a inserção dos dados na tabela “usuários” por meio da instrução “insert” da linguagem SQL, através do método **execSQL**.

Vamos abrir agora o arquivo “activity_banco_de_dados.xml” (no “modo Design”) para inserirmos o seguinte componente abaixo, abaixo do botão rotulado “Criar Banco de Dados”):

Button

Propriedade	Valor
id	btcadastrardados
text	Cadastrar Dados
layout:width	200dp

Veja o resultado na figura seguinte :



Tela da aplicação em construção

Agora vamos abrir novamente o arquivo “BancoDeDadosActivity.java” para digitarmos as seguintes instruções **destacadas em azul** abaixo:

```
public class BancoDeDadosActivity extends Activity {

    Button btcriabanco;
    Button btcadastrardados;
    SQLiteDatabase db;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_banco_de_dados);
        btcriabanco = (Button) findViewById(R.id.btcriarbanco);
        btcadastrardados = (Button)
        findViewById(R.id.btcadastrardados);
        btcadastrardados.setOnClickListener(new
        View.OnClickListener(){
            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method stub
                Intent gravaRegistroActivity = new
                Intent(BancoDeDadosActivity.this,
                GravaRegistrosActivity.class);
                BancoDeDadosActivity.this.
                startActivity(gravaRegistroActivity);
            }
        });
    }
}
```



```

        btcriabanco.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View view) {

                :

            }

        });
    }

}

```

Depois de escrever o código acima, vamos executar a nossa aplicação. Veja o resultado na figura seguinte:



Aplicação em execução

Experimente agora realizar um cadastro de um usuário e em seguida, clique no botão cadastrar para armazenar as informações dentro da tabela “usuarios” presente dentro do banco de dados. Veja o resultado na figura seguinte:



3.3 – Registrando informações no Banco (Segunda Versão)

Na versão anterior 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**.

Vamos agora criar uma nova “Activity” (Atividade) , conforme já foi mostrado, com as seguintes informações abaixo :

Activity Name : GravaRegistros2Activity

Layout Name : activity_grava_registros2

Title : Gravar Registros (2ª Versão)

Menu Resource Name : menu_grava_registros2

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 abaixo:

```
package app.android.aplicacaobancodedados;

import android.os.Bundle;
import android.app.Activity;
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;

public class GravaRegistros2Activity extends Activity {

    Button btcadastrar;

    EditText ednome;
    EditText edtelefone;
    EditText edemail;

    SQLiteDatabase db;
```



```

@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());
    }

    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);

            try {

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

        }
    });
}

```



```

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();
}
}

```

Irei comentar aqui agora 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());
}

```

Esse conjunto de instruções mostrados anteriormente, funciona de forma similar ao método **execSQL** (utilizado na versão do programa anterior), executando a instrução INSERT da linguagem SQL. Explicarei passo a passo as etapas de execução. 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.

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).

Agora vamos abrir o arquivo “activity_banco_dados.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:

Button

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

Feito isso teremos o seguinte resultado:



Tela da aplicação em execução

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

```
public class BancoDeDadosActivity extends Activity {

    Button btcriabanco;
    Button btcadastrardados;
    Button btcadastrardados2;
    SQLiteDatabase db;

    @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);

        btcadastrardados.setOnClickListener(new
            View.OnClickListener(){
                @Override
                public void onClick(View arg0) {
                    // TODO Auto-generated method stub
```

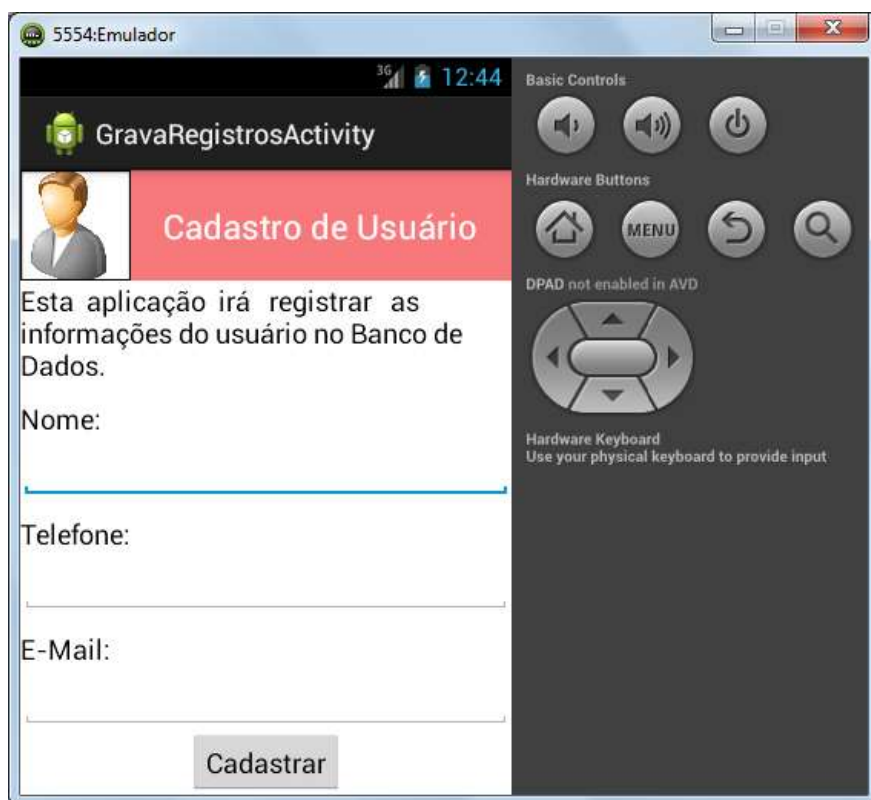


```

        Intent gravaRegistroActivity2 = new
        Intent(BancoDeDadosActivity.this,
        GravaRegistrosActivity2.class);
        BancoDeDadosActivity.this.
        startActivity(gravaRegistroActivity2);
    }
});

```

Depois de escrever o código acima, vamos executar a nossa aplicação. Veja o resultado na figura seguinte:



Aplicação em execução

Experimente agora realizar um segundo cadastro utilizando a segunda versão do nosso programa.

3.4 – 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.



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

Activity Name : ConsultaDadosActivity

Layout Name : activity_consulta_dados

Title : Consulta Dados

Menu Resource Name : menu_consulta_dados

Depois de criada a nossa Activity, 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" />

        <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"
```



```

        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" />

<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/textView3"
    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" />

<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/textView3"
    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" />

```




```
<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" >
```

```
<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" />
```

```
<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" >
```

```
<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>

Vejamos o resultado na figura seguinte :



Tela da aplicação em construção



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

```
package app.android.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.Menu;
import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;

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);

        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);

    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)); //Obtem o telefone
        txtemail.setText(c.getString(2)); //Obtem o e-mail

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

    }
    else {

        txtstatus_registro.setText("Nenhum Registro");

    }

    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));
                //Obtem o e-mail
                txtemail.setText(c.getString(2));

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

            }

        }

    });

}
```



```

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());
            }
        }
    }
});

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());
            }
        }
    }
});

```




```
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());
        }

    }

});

}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();
}
}
```

Irei 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);
```



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**)

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());
}
```

No trecho de comandos mostrados acima, 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. 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.

Agora vamos abrir o arquivo “activity_banco_dados.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:

Button

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

Feito isso teremos o seguinte resultado:



Tela da aplicação em construção



Depois disso vamos abrir o arquivo “BancoDeDadosActivity.java” para digitarmos os seguintes blocos de comandos destacados em azul abaixo:

```
public class BancoDeDadosActivity extends Activity {

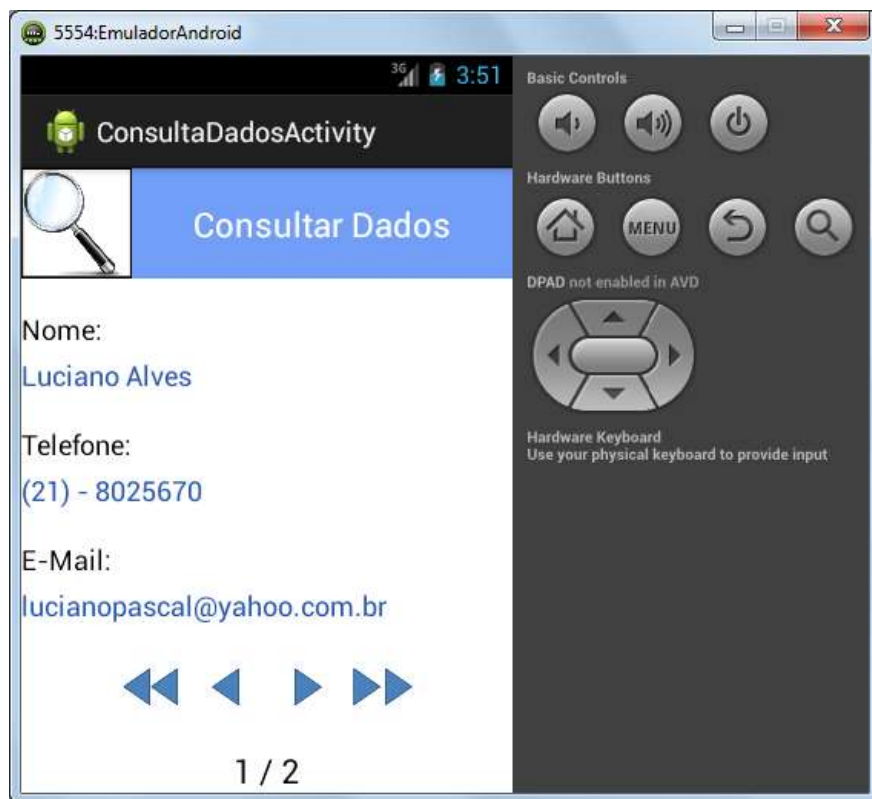
    Button btcriabanco;
    Button btcadastrardados;
    Button btcadastrardados2;
    Button btconsultardados;
    SQLiteDatabase db;

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

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

        btconsultardados.setOnClickListener(new
        View.OnClickListener(){
            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method stub
                Intent consultaDadosActivity = new
                Intent(BancoDeDadosActivity.this,
                ConsultaDadosActivity.class);
                BancoDeDadosActivity.this.
                startActivity(consultaDadosActivity);
            }
        });
    }
}
```

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



Aplicação em execução



3.5 – Alterando as informações da tabela (Primeira Versão)

Vamos construir agora, dentro do mesmo projeto, uma Activity que irá alterar as informações dentro da nossa tabela dentro do Banco de Dados. Esse processo estará dividido em duas versões. Essa será a primeira versão da Activity, onde iremos trabalhar com o método **execSQL** para a inserção das informações na tabela (através de comando SQL).

Crie uma nova atividade Android , conforme já foi mostrado, de acordo com as informações a seguir:

Activity Name : AlterarDadosActivity

Layout Name : activity_alterar_dados

Title : Alterar Dados

Menu Resource Name : menu_alterar_dados

Depois de criado a Activity, vamos adicionar dentro do arquivo “activity_alterar_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_alterar_dados" />

        <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="#edb068"
    android:gravity="center" >

```

<TextView

```

    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Alterar Dados"
    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="10dp"
    android:text="Nome: "
    android:textAppearance="?android:attr/textAppearanceSmall"
    android:textColor="#000000"
    android:textSize="18sp" />

```

<EditText

```

    android:id="@+id/txtnome"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" />

```

<TextView

```

    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Telefone: "
    android:textAppearance="?android:attr/textAppearanceSmall"
    android:textColor="#000000"
    android:textSize="18sp" />

```

<EditText

```

    android:id="@+id/txttelefone"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" >

```

<requestFocus />

</EditText>

<TextView

```

    android:id="@+id/textView3"
    android:layout_width="wrap_content"

```



```

        android:layout_height="wrap_content"
        android:text="E-Mail:"
        android:textAppearance="?android:attr/textAppearanceSmall"
        android:textColor="#000000"
        android:textSize="18sp" />
<EditText
    android:id="@+id/txtemail"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" />

```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:paddingTop="0dp" >

```

```

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

```

```

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

```

```

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

```

```

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

```

```

</LinearLayout>

```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:paddingTop="0dp" >

```



```
<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
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center" >
    <Button
        android:id="@+id/btalterardados"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Alterar Dados" />
</LinearLayout>
```

```
</LinearLayout>
```

Feito isso teremos o seguinte resultado:



Tela da aplicação em construção



Depois disso vamos abrir o arquivo “AlterarDadosActivity.java” para digitarmos o seguinte código abaixo:

```
package app.android.aplicacaobancodedados;

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

public class AlterarDadosActivity extends Activity {

    EditText txtnome, txttelefone, txtemail;

    TextView txtstatus_registro;

    SQLiteDatabase db;

    ImageView imgprimeiro, imganterior, imgproximo, imgultimo;

    Button balterardados;

    int indice;

    int numreg;

    Cursor c;

    DialogInterface.OnClickListener diAlteraInformacoes;

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

        txtnome = (EditText) findViewById(R.id.txtnome);
        txttelefone = (EditText) findViewById(R.id.txttelefone);
        txtemail = (EditText) findViewById(R.id.txtemail);

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

        imgprimeiro = (ImageView) findViewById(R.id.imgprimeiro);
```




```

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

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

try {

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

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

    if(c.getCount() > 0) {

        //Move para o primeiro registro

        c.moveToFirst();
        indice = 1;
        numreg = c.getInt(0); //Obtem o número de registro
        txtnome.setText(c.getString(1)); //Obtem o nome
        txttelefone.setText(c.getString(2)); //Obtem o telefone
        txtemail.setText(c.getString(3)); //Obtem o e-mail

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

    }
    else {

        txtstatus_registro.setText("Nenhum Registro");

    }
}
catch(Exception e)
{

}

imgprimeiro.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        if(c.getCount() > 0)
        {
            //Move para o primeiro registro
            c.moveToFirst();
            indice = 1;

```



```

        //Obtem o número de registro
        numreg = c.getInt(0);
        //Obtem o nome
        txtnome.setText(c.getString(1));
        //Obtém o telefone
        txttelefone.setText(c.getString(2));
        //Obtém o e-mail
        txtemail.setText(c.getString(3));

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

    });

imganterior.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        if(c.getCount() > 0)
        {
            if(indice > 1) {

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

                numreg = c.getInt(0); //Obtem o número de
                //registro
                txtnome.setText(c.getString(1)); //Obtem o
                //nome
                txttelefone.setText(c.getString(2)); //Obtém o
                //telefone
                txtemail.setText(c.getString(3)); //Obtém o
                //e-mail
                txtstatus_registro.setText(indice + " / " +
                c.getCount());
            }
        }
    });

imgproximo.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        if(c.getCount() > 0)
        {
            if(indice != c.getCount()) {

                indice++;
            }
        }
    });
}

```



```

//Move para o proximo registro
c.moveToNext();

numreg = c.getInt(0); //Obtem o número de
//registro
txtnome.setText(c.getString(1)); //Obtem o
//nome
txttelefone.setText(c.getString(2)); //Obtém o
//telefone

//Obtém o e-mail
txtemail.setText(c.getString(3));

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

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();

        numreg = c.getInt(0); //Obtem o número de registro
        txtnome.setText(c.getString(1)); //Obtem o nome
        txttelefone.setText(c.getString(2)); //Obtém o
        //telefone
        txtemail.setText(c.getString(3)); //Obtém o e-mail
        txtstatus_registro.setText(indice + " / " +
        c.getCount());
    }
}
});

diAlterarInformacoes = new DialogInterface.OnClickListener() {

@Override
public void onClick(DialogInterface dialog, int which) {

    //Altera as informações do registro na tabela
    String nome = txtnome.getText().toString();
    String telefone = txttelefone.getText().toString();
    String email = txtemail.getText().toString();

```



```

try {

    db.execSQL("update usuarios set nome = '" + nome + "', "
        + "telefone = '" + telefone + "', email = '" + email + " "
        + "' where numreg = " + numreg);

    MostraMensagem("Dados alterados com sucesso.");
}
catch(Exception e) {
    MostraMensagem("Erro: " + e.toString());
}
}

btalterardados.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {

        AlertDialog.Builder dialogo = new
        AlertDialog.Builder(AlterarDadosActivity.this);
        dialogo.setTitle("Confirma");
        dialogo.setMessage("Deseja alterar as informações");
        dialogo.setNegativeButton("Não", null);
        dialogo.setPositiveButton("Sim", diAlterarInformacoes);
        dialogo.show();

    }
});

public void MostraMensagem(String str)
{
    AlertDialog.Builder dialogo = new
    AlertDialog.Builder(AlterarDadosActivity.this);
    dialogo.setTitle("Aviso");
    dialogo.setMessage(str);
    dialogo.setNeutralButton("OK", null);
    dialogo.show();
}
}

```

Irei explicar aqui algumas linhas de código do nosso programa. Dentro do botão “Alterar Dados” temos o seguinte código abaixo:

```

AlertDialog.Builder dialogo = new
AlertDialog.Builder(AlterarDadosActivity.this);
dialogo.setTitle("Confirma");
dialogo.setMessage("Deseja alterar as informações");
dialogo.setNegativeButton("Não", null);
dialogo.setPositiveButton("Sim", diAlterarInformacoes);
dialogo.show();

```



Que exibe uma mensagem confirmando se o usuário deseja alterar as informações do registro selecionado (e modificado). Se observarmos acima, o método **setPositiveButton**, que exibe o botão “Sim” da caixa de mensagem, chama a instância *diAlterarInformacoes* que nada mais é do que uma estrutura do tipo **DialogInterface**, onde dentro da mesma existe um evento do tipo “click”, que será executado quando o botão “Sim” for clicado. Vejamos o conjunto de instruções do evento abaixo:

```
diAlterarInformacoes = new DialogInterface.OnClickListener() {

    @Override
    public void onClick(DialogInterface dialog, int which) {

        //Altera as informações do registro na tabela
        String nome = txtnome.getText().toString();
        String telefone = txttelefone.getText().toString();
        String email = txtemail.getText().toString();

        try {

            db.execSQL("update usuarios set nome = '" + nome + "', "
                + "telefone = '" + telefone + "', email = '" + email + " "
                + "' where numreg = " + numreg);

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

Como podemos observar na estrutura mostrada, existe uma instrução que executa o método **execSQL** presente dentro da instância *db*, que realiza a instrução UPDATE da linguagem SQL para alterar as informações presentes dentro da tabela “usuarios” do nosso Banco de Dados.

Agora vamos voltar para o arquivo “activity_banco_de_dados.xml” e dentro da tela da nossa aplicação vamos inserir um componente **Button** (abaixo do botão “Consultar Dados”), de acordo com a tabela abaixo:

Button

Propriedade	Valor
id	balterardados
text	Alterar Dados
layout:width	200dp



Feito isso teremos o seguinte resultado:



Tela da aplicação em construção

Agora dentro do arquivo “BancoDeDadosActivity.java” vamos escrever o seguinte código destacado em azul abaixo:

```
public class BancoDeDadosActivity extends Activity {

    Button btcriabanco;
    Button btcadastrardados;
    Button btcadastrardados2;
    Button btconsultardados;
    Button btalterardados;
    SQLiteDatabase db;

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



```

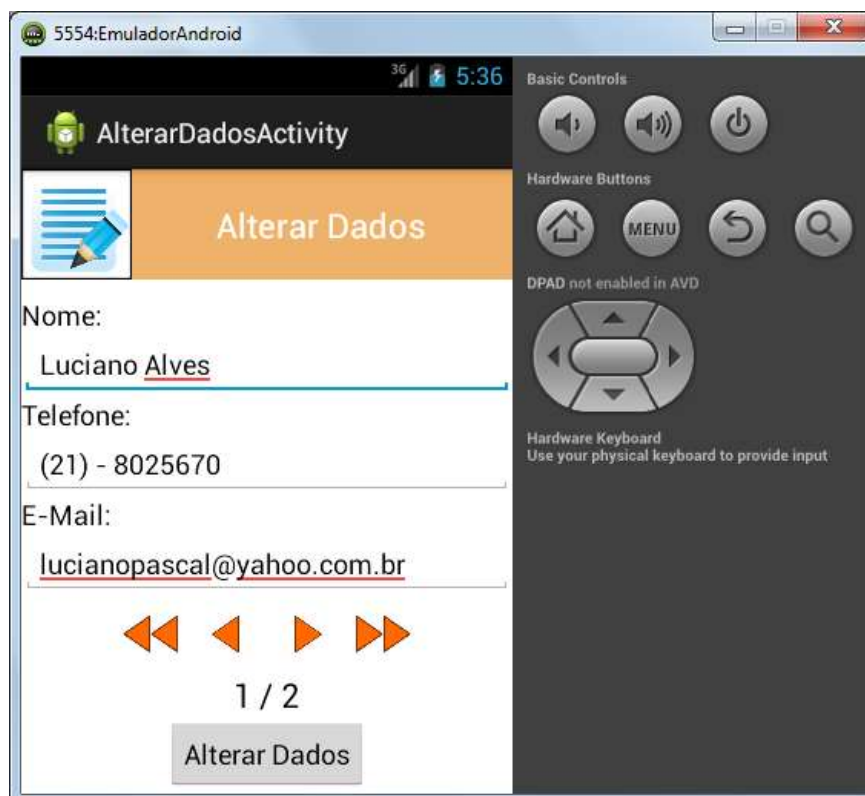
btcriabanco = (Button) findViewById(R.id.btcriarbanco);
:
btalterardados = (Button)
findViewById(R.id.btconsultardados);

btalterardados.setOnClickListener(new
View.OnClickListener(){
    @Override
    public void onClick(View arg0) {
        // TODO Auto-generated method stub
        Intent alterarDadosActivity = new
        Intent(BancoDeDadosActivity.this,
        AlterarDadosActivity.class);

        BancoDeDadosActivity.this.
        startActivity(alterarDadosActivity);
    }
});

```

Depois de escrever os códigos solicitados vamos executar a nossa aplicação. O resultado você confere na figura seguinte:



Aplicação em Execução

Experimente realizar uma alteração de dados em um dos registros da nossa tabela.



3.6 – Alterando as informações da tabela (Segunda Versão)

Agora vamos desenvolver uma segunda versão da Activity desenvolvida anteriormente, onde nela vamos utilizar como meio de alteração das informações o método **update** da classe **SQLiteDataBase** que possui o mesmo comportamento da instrução UPDATE da linguagem SQL.

Vamos criar uma nova Activity de acordo com as informações abaixo:

Activity Name : AlterarDados2Activity

Layout Name : activity_alterar_dados2

Title : Alterar Dados (2ª Versão)

Menu Resource Name : menu_alterar_dados2

A tela da segunda versão da nossa Activity de alteração de dados é igual ao da primeira versão, logo, copie o código XML do arquivo “activity_alterar_dados.xml” para o arquivo “activity_alterar_dados2.xml”. Feito isso teremos o seguinte resultado:



Tela da aplicação em construção

Agora dentro do arquivo “AlterarDados2Activity.java” vamos escrever o seguinte código em seguida:



```
package app.android.aplicacaobancodedados;

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

public class AlterarDados2Activity extends Activity {

    EditText txtnome, txttelefone, txtemail;

    TextView txtstatus_registro;

    SQLiteDatabase db;

    ImageView imgprimeiro, imganterior, imgproximo, imgultimo;

    Button btalterardados;

    int indice;

    int numreg;

    Cursor c;

    DialogInterface.OnClickListener diAlterarInformacoes;

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

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

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



```

imgultimo = (ImageView) findViewById(R.id.imgultimo);

btalterardados = (Button)
findViewById(R.id.balterardados);

try {

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

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

    if(c.getCount() > 0) {

        //Move para o primeiro registro
        c.moveToFirst();
        indice = 1;
        numreg = c.getInt(0); //Obtem o número de registro
        txtnome.setText(c.getString(1)); //Obtem o nome

        //Obtem o telefone
        txttelefone.setText(c.getString(2));

        //Obtem o e-mail
        txtemail.setText(c.getString(3));

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

    }
    else {
        txtstatus_registro.setText("Nenhum Registro");
    }
}
catch(Exception e)
{

}

imgprimeiro.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        if(c.getCount() > 0)
        {
            //Move para o primeiro registro
            c.moveToFirst();
            indice = 1;
            //Obtem o número de registro
            numreg = c.getInt(0);

```




```

        //Obtem o nome
        txtnome.setText(c.getString(1));
        //Obtém o telefone
        txttelefone.setText(c.getString(2));
        //Obtém o e-mail
        txtemail.setText(c.getString(3));

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

imganterior.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        if(c.getCount() > 0)
        {
            if(indice > 1) {
                indice--;
                //Move para o registro anterior
                c.moveToPrevious();

                numreg = c.getInt(0); //Obtem o número de
                //registro

                //Obtem o nome
                txtnome.setText(c.getString(1));

                //Obtém o telefone
                txttelefone.setText(c.getString(2));

                //Obtém o e-mail
                txtemail.setText(c.getString(3));

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

imgproximo.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        if(c.getCount() > 0)
        {
            if(indice != c.getCount()) {

```



```

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

        //Obtem o número de registro
        numreg = c.getInt(0);
        txtnome.setText(c.getString(1)); //Obtem o nome

        //Obtém o telefone
        txttelefone.setText(c.getString(2));

        //Obtém o e-mail
        txtemail.setText(c.getString(3));

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

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();

            numreg = c.getInt(0); //Obtem o número de registro
            txtnome.setText(c.getString(1)); //Obtem o nome

            //Obtém o telefone
            txttelefone.setText(c.getString(2));
            txtemail.setText(c.getString(3)); //Obtém o e-mail

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

diAlterarInformacoes = new DialogInterface.OnClickListener() {

    @Override
    public void onClick(DialogInterface dialog, int which) {

        //Altera as informações do registro na tabela
        String nome = txtnome.getText().toString();
        String telefone = txttelefone.getText().toString();
        String email = txtemail.getText().toString();
    }
}

```



```

try {

    ContentValues valor = new ContentValues();

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

    db.update("usuarios", valor, "numreg=" + numreg,
    null);

    MostraMensagem("Dados alterados com sucesso.");
}

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

btalterardados.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {

        AlertDialog.Builder dialogo = new
        AlertDialog.Builder(AlterarDados2Activity.this);

        dialogo.setTitle("Confirma");
        dialogo.setMessage("Deseja alterar as informações ?");
        dialogo.setNegativeButton("Não", null);
        dialogo.setPositiveButton("Sim",
        diAlteraInformacoes);
        dialogo.show();

    }

});

public void MostraMensagem(String str)
{
    AlertDialog.Builder dialogo = new
    AlertDialog.Builder(AlterarDados2Activity.this);
    dialogo.setTitle("Aviso");
    dialogo.setMessage(str);
    dialogo.setNeutralButton("OK", null);
    dialogo.show();
}
}

```



Irei comentar agora o código responsável por realizar as alterações na tabela do banco de dados. Em relação ao programa anterior, agora usamos o método **update**, presente dentro da instância *db*, responsável por alterar as informações dentro da tabela "usuarios". Veja o código abaixo:

```

ContentValues valor = new ContentValues();

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

db.update("usuarios", valor, "numreg=" + numreg,
null);

```

Dentro da instância *valor* (do tipo **ContentValues**) adicionamos todas informações que foram alteradas na aplicação, e por último realizamos essas alterações na tabela por meio do método **update**, passando como argumento a tabela que queremos alterar as informações (a tabela "usuarios"), a instância que possui as informações a serem alteradas (o argumento *valor*) e por último, a condição (where) para que as informações sejam alteradas (o argumento "numreg = " + numreg).

Agora vamos abrir o arquivo "activity_banco_de_dados.xml" e vamos substituir o código XML já presente no arquivo pelo novo código XML abaixo:

```

<?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:orientation="vertical" >

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#3ba0e2" >

        <ImageView
            android:id="@+id/imageView1"
            android:layout_width="96dp"
            android:layout_height="96dp"
            android:src="@drawable/icone_bd" />

        <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="#E5BE40"
android:gravity="center" >

```

```

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Trabalhando com Banco de Dados no
    Android"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:textColor="#ffffff" />

```

```

</LinearLayout>

```

```

</LinearLayout>

```

```

<LinearLayout
    android:id="@+id/LayoutImagemFundo"
    android:layout_width="match_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

```

```

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Escolha uma das opções abaixo:"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:textSize="20sp" />

```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="fill_parent"
    android:gravity="center"
    android:orientation="vertical" >

```

```

<Button
    android:id="@+id/btcriarbanco"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Criar Banco de Dados"
    android:textSize="15sp" />

```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content" >

```

```

<LinearLayout
    android:layout_width="160dp"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical" >

```




<Button

```
    android:id="@+id/btcadastrardados"
    android:layout_width="160dp"
    android:layout_height="wrap_content"
    android:text="Cadastrar Dados"
    android:textSize="15sp" />
```

<Button

```
    android:id="@+id/btcadastrardados2"
    android:layout_width="160dp"
    android:layout_height="wrap_content"
    android:text="Cadastrar Dados (2ª Versão)"
    android:textSize="15sp" />
```

<Button

```
    android:id="@+id/btconsultardados"
    android:layout_width="160dp"
    android:layout_height="wrap_content"
    android:text="Consultar Dados"
    android:textSize="15sp" />
```

<Button

```
    android:id="@+id/btalterardados"
    android:layout_width="160dp"
    android:layout_height="wrap_content"
    android:text="Alterar Dados"
    android:textSize="15sp" />
```

</LinearLayout>

<LinearLayout

```
    android:layout_width="160dp"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical" >
```

<Button

```
    android:id="@+id/btalterardados2"
    android:layout_width="160dp"
    android:layout_height="wrap_content"
    android:text="Alterar Dados (2ª Versão)"
    android:textSize="15sp" />
```

</LinearLayout>

</LinearLayout>

</LinearLayout>

</LinearLayout>

</LinearLayout>



Feito isso teremos o seguinte resultado:



Tela da aplicação em construção

Agora dentro do arquivo “BancoDeDadosActivity.java” vamos digitar o seguinte código destacado em azul abaixo:

```
public class BancoDeDadosActivity extends Activity {

    Button btcriabanco;
    Button btcadastrardados;
    Button btcadastrardados2;
    Button btconsultardados;
    Button btalterardados;
    Button balterardados2;

    SQLiteDatabase db;

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



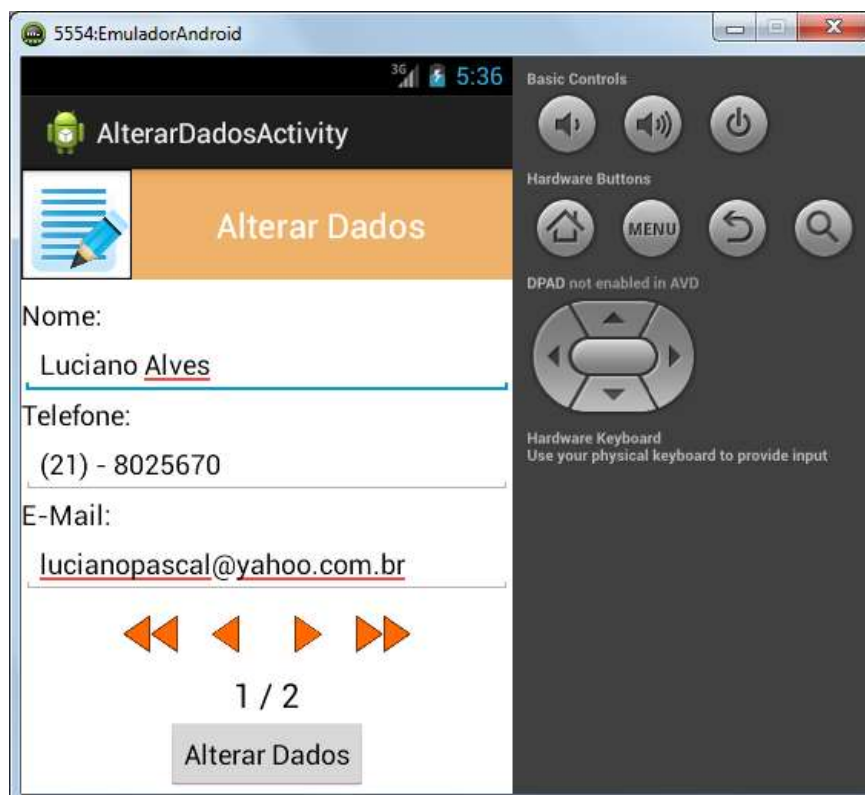
```

btcriabanco = (Button) findViewById(R.id.btcriarbanco);
:
btalterardados2 = (Button)
findViewById(R.id.btconsultardados2);

btalterardados2.setOnClickListener(new
View.OnClickListener(){
    @Override
    public void onClick(View arg0) {
        // TODO Auto-generated method stub
        Intent alterarDados2Activity = new
        Intent(BancoDeDadosActivity.this,
        AlterarDados2Activity.class);
        BancoDeDadosActivity.this.
        startActivity(alterarDados2Activity);
    }
});

```

Depois de digitar os códigos solicitados vamos executar a nossa aplicação. O resultado você confere na figura seguinte:



Aplicação em Execução

Experimente agora alterar algumas informações de algum registro presente na tabela.



3.7 – Excluindo registros da tabela (Primeira Versão)

Vamos construir agora, dentro do mesmo projeto, uma Activity que irá excluir os registros da nossa tabela dentro do Banco de Dados. Esse processo estará dividido em duas versões. Essa será a primeira versão, onde iremos trabalhar com o método **execSQL** para a inserção das informações na tabela (através de comando SQL).

Crie uma nova atividade Android , conforme já foi mostrado, de acordo com as informações abaixo:

Activity Name : ExcluirDadosActivity

Layout Name : activity_excluir_dados

Title : Excluir Dados

Menu Resource Name : menu_excluir_dados

Depois de criado o projeto, vamos adicionar dentro do arquivo “activity_excluir_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_excluir_dados" />

        <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="#b0b5b3"
        android:gravity="center" >

```

```

        <TextView
            android:id="@+id/textView2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Excluir Dados"
            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="5dp"
    android:text="Nome: "
    android:textAppearance="?android:attr/textAppearanceSmall"
    android:textColor="#000000"
    android:textSize="18sp" />

```

```

<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="#777a79"
    android:textSize="18sp" />

```

```

<TextView
    android:id="@+id/textView3"
    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" />

```

```

<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="#777a79"
android:textSize="18sp" />
```

<TextView

```
android:id="@+id/textView3"
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" />
```

<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="#777a79"
android:textSize="18sp" />
```

<LinearLayout

```
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:gravity="center"
android:paddingTop="0dp" >
```

<ImageView

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

<ImageView

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

<ImageView

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

<ImageView

```
android:id="@+id/imgultimo"
android:layout_width="50dp"
android:layout_height="50dp"
```



```

        android:scaleType="fitXY"
        android:src="@drawable/botao_del_ultimo" />

```

```

</LinearLayout>

```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:paddingTop="0dp" >

    <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
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center" >

    <Button
        android:id="@+id/btexcluidados"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Excluir Dados" />

```

```

</LinearLayout>

```

```

</LinearLayout>

```

Depois de colocar o código no arquivo salve as alterações. O resultado você confere na figura seguinte:



Tela da aplicação em construção

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

```
package app.andrid.aplicacaobancodedados;

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

public class ExcluirDadosActivity extends Activity {

    TextView txtnome, txttelefone, txtemail,txtstatus_registro;

    SQLiteDatabase db;
```



```

ImageView imgprimeiro, imganterior, imgproximo, imgultimo;

int indice;

int numreg;

Cursor c;

Button btexcluidados;

DialogInterface.OnClickListener diExcluiRegistro;

@Override
protected void onCreate(Bundle savedInstanceState) {

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

    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);

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

    try {

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

        CarregarDados();

        imgprimeiro.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View view) {

                if(c.getCount() > 0)
                {
                    //Move para o primeiro registro
                    c.moveToFirst();

```



```

        indice = 1;
        numreg= c.getInt(0); //Obtem o número do registro
        txtnome.setText(c.getString(1)); //Obtem o nome
        txttelefone.setText(c.getString(2)); //Obtém o
        //telefone
        txtemail.setText(c.getString(3)); //Obtém o e-mail

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

});

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();
                numreg= c.getInt(0); //Obtem o número do registro
                txtnome.setText(c.getString(1)); //Obtem o nome
                txttelefone.setText(c.getString(2)); //Obtém o
                //telefone
                txtemail.setText(c.getString(3)); //Obtém o e-mail

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

});

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();

                numreg= c.getInt(0); //Obtem o número do registro
                txtnome.setText(c.getString(1)); //Obtem o nome
                txttelefone.setText(c.getString(2)); //Obtém o telefone

```




```

        txtemail.setText(c.getString(3)); // Obtém o e-mail

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

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();

            numreg = c.getInt(0); // Obtém o número do registro
            txtnome.setText(c.getString(1)); // Obtém o nome
            txttelefone.setText(c.getString(2)); // Obtém o telefone
            txtemail.setText(c.getString(3)); // Obtém o e-mail

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

diExcluiRegistro = new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int which) {
        db.execSQL("delete from usuarios where numreg = " +
        numreg);
        CarregarDados();
        MostraMensagem("Dados excluidos com sucesso");
    }
};

btexcluidados.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(c.getCount() > 0) {
            AlertDialog.Builder dialogo = new
            AlertDialog.Builder(ExcluirDadosActivity.this);
            dialogo.setTitle("Confirma");
            dialogo.setMessage("Deseja excluir esse registro ?");

```



```

        dialogo.setNegativeButton("Não", null);
        dialogo.setPositiveButton("Sim", diExcluiRegistro);
        dialogo.show();
    }
    else {
        MostraMensagem("não existem registros para excluir");
    }
}
});

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

}

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

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

    if(c.getCount() > 0) {

        //Move para o primeiro registro
        c.moveToFirst();
        indice = 1;
        numreg= c.getInt(0); //Obtem o número do registro
        txtnome.setText(c.getString(1)); //Obtem o nome
        txttelefone.setText(c.getString(2)); //Obtém o telefone
        txtemail.setText(c.getString(3)); //Obtém o e-mail

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

    }
    else {
        txtstatus_registro.setText("Nenhum Registro");
    }
}

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

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

```



```
        dialogo.show();
    }
}
```

Irei explicar aqui algumas linhas de código do nosso programa. Dentro do botão “Excluir Dados” temos o seguinte código abaixo:

```
AlertDialog.Builder dialogo = new
AlertDialog.Builder(ExcluirDadosActivity.this);
dialogo.setTitle("Confirma");
dialogo.setMessage("Deseja excluir esse registro ?");
dialogo.setNegativeButton("Não", null);
dialogo.setPositiveButton("Sim", diExcluiRegistro);
dialogo.show();
```

Que exibe uma mensagem confirmando se o usuário deseja excluir o registro selecionado. Se observarmos acima, o método **setPositiveButton**, que exibe o botão “Sim” da caixa de mensagem, chama a instância *diExcluiRegistro* que nada mais é do que uma estrutura do tipo **DialogInterface**, onde dentro da mesma existe um evento do tipo “click”, que será executado quando o botão “Sim” for clicado. Vejamos o conjunto de instruções do evento abaixo:

```
diExcluiRegistro = new DialogInterface.OnClickListener() {

    @Override
    public void onClick(DialogInterface dialog, int which) {

        db.execSQL("delete from usuarios where numreg = " +
numreg);
        CarregarDados();
        MostraMensagem("Dados excluidos com sucesso");
    }
};
```

Como podemos observar na estrutura mostrada acima, existe uma instrução que executa o método **execSQL** presente dentro da instância *db*, que realiza a execução instrução DELETE da linguagem SQL, cuja finalidade é excluir o registro selecionado na tabela “usuarios” do nosso Banco de Dados.

Agora vamos voltar para o arquivo “activity_banco_de_dados.xml” e dentro da tela da nossa aplicação vamos inserir um componente **Button** (abaixo do botão “Alterar Dados (2ª Versão)”), de acordo com a tabela em seguida:



Button

Propriedade	Valor
id	btexcluidados
text	Excluir Dados
layout:width	160dp

Feito isso teremos o seguinte resultado em seguida:



Tela da aplicação em construção

Agora vamos abrir o arquivo “BancoDeDadosActivity.java” para digitarmos as seguintes instruções destacadas em azul abaixo:

```
public class BancoDeDadosActivity extends Activity {

    Button btcriabanco;
    Button btcadastrardados;
    Button btcadastrardados2;
    Button btconsultardados;
    Button btalterardados;
    Button btalterardados2;
    Button btexcluidados;

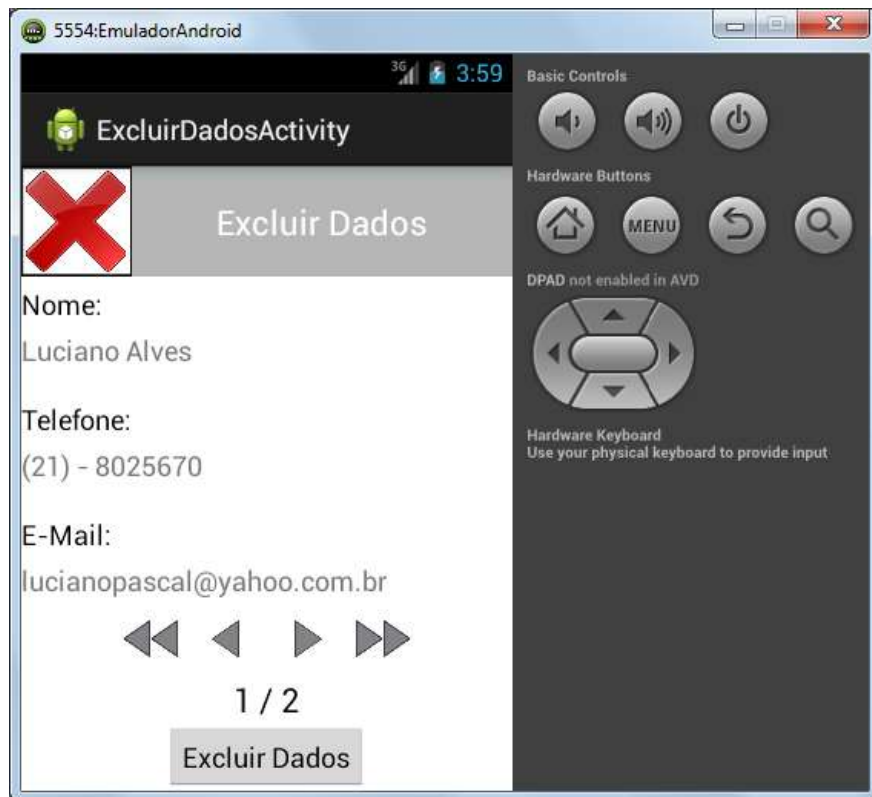
    SQLiteDatabase db;
```



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

    btcriabanco = (Button) findViewById(R.id.btcriarbanco);
    :
    btexcluidados = (Button)
    findViewById(R.id.btexcluidados);
    btexcluidados.setOnClickListener(new
    View.OnClickListener(){
        @Override
        public void onClick(View arg0) {
            // TODO Auto-generated method stub
            Intent excluirDadosActivity = new
            Intent(BancoDeDadosActivity.this,
            ExcluirDadosActivity.class);
            BancoDeDadosActivity.this.
            startActivity(excluirDadosActivity);
        }
    });
}
```

Depois de escrever as instruções solicitadas, vamos executar a nossa aplicação. O resultado você confere na figura seguinte:



Aplicação em execução

Experimente excluir um registro da tabela do nosso Banco de Dados.



3.8 – Excluindo registros da tabela (Segunda Versão)

Agora vamos desenvolver uma segunda versão da Activity desenvolvida anteriormente, onde nela vamos utilizar como meio para excluir os registros o método **delete** da classe **SQLiteDataBase** que possui o mesmo comportamento da instrução DELETE da linguagem SQL.

Vamos criar uma nova Activity de acordo com as informações abaixo:

Activity Name : ExcluirDados2Activity

Layout Name : activity_excluir_dados2

Title : Excluir Dados (2ª Versão)

Menu Resource Name : menu_excluir_dados2

A tela da segunda versão da nossa aplicação de alteração de dados é igual ao da primeira versão, logo, copie o código XML do arquivo “activity_excluir_dados.xml” para o arquivo “activity_excluir_dados2.xml”. Feito isso teremos o seguinte resultado:



Tela da aplicação em construção



Depois de copiar o código XML para o arquivo , salve as alterações. Feito isso vamos abrir o arquivo “ExcluirDados2Activity.java” para digitarmos o seguinte código abaixo:

```
package app.android.aplicacaobancodedados;

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

public class ExcluirDados2Activity extends Activity {

    TextView txtnome, txttelefone, txtemail,txtstatus_registro;

    SQLiteDatabase db;

    ImageView imgprimeiro, imganterior, imgproximo, imgultimo;

    int indice;

    int numreg;

    Cursor c;

    Button btexcluidados;

    DialogInterface.OnClickListener diExcluiRegistro;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

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

        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);

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

try {

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

    CarregarDados();

    imgprimeiro.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View view) {

            if(c.getCount() > 0)
            {
                //Move para o primeiro registro
                c.moveToFirst();
                indice = 1;
                numreg= c.getInt(0); //Obtem o número do registro
                txtnome.setText(c.getString(1)); //Obtem o nome
                txttelefone.setText(c.getString(2));
                //Obtém o telefone
                txtemail.setText(c.getString(3)); //Obtém o e-mail

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

    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();
                    numreg= c.getInt(0); //Obtem o número do registro
                    txtnome.setText(c.getString(1)); //Obtem o nome
                    //Obtém o telefone
                    txttelefone.setText(c.getString(2));
                }
            }
        }
    });
}

```



```

        txtemail.setText(c.getString(3)); //Obtém o e-mail

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

```

```

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();

                numreg= c.getInt(0); //Obtem o número do registro
                txtnome.setText(c.getString(1)); //Obtem o nome
                txttelefone.setText(c.getString(2)); //Obtém o telefone
                txtemail.setText(c.getString(3)); //Obtém o e-mail

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

```

```

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();

            numreg= c.getInt(0); //Obtem o número do registro
            txtnome.setText(c.getString(1)); //Obtem o nome
            txttelefone.setText(c.getString(2)); //Obtém o telefone
            txtemail.setText(c.getString(3)); //Obtém o e-mail

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

```




```

    }
    });

    diExcluiRegistro = new DialogInterface.OnClickListener() {

        @Override
        public void onClick(DialogInterface dialog, int which) {

            db.delete("usuarios", "numreg=" + numreg, null);
            CarregarDados();
            MostraMensagem("Dados excluidos com sucesso");
        }
    };

    btexcluidados.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View v) {

            if(c.getCount() > 0) {

                AlertDialog.Builder dialogo = new
                AlertDialog.Builder(ExcluirDados2Activity.this);
                dialogo.setTitle("Confirma");
                dialogo.setMessage("Deseja excluir esse registro ?");
                dialogo.setNegativeButton("Não", null);
                dialogo.setPositiveButton("Sim", diExcluiRegistro);
                dialogo.show();
            }
            else {
                MostraMensagem("não existem registros para excluir");
            }
        }
    });

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

}

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

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

```



```

if(c.getCount() > 0) {

    //Move para o primeiro registro
    c.moveToFirst();
    indice = 1;
    numreg= c.getInt(0); //Obtem o número do registro
    txtnome.setText(c.getString(1)); //Obtem o nome
    txttelefone.setText(c.getString(2)); //Obtem o telefone
    txtemail.setText(c.getString(3)); //Obtem o e-mail

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

}
else {

    txtstatus_registro.setText("Nenhum Registro");

}

}

public void MostraMensagem(String str)
{
    AlertDialog.Builder dialogo = new
    AlertDialog.Builder(ExcluirDados2Activity.this);

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

```

Irei comentar agora o código responsável por realizar a exclusão do registro na tabela do banco de dados. Em relação ao programa anterior, agora usamos o método **delete** ,presente dentro da instância *db* , responsável por excluir o registro dentro da tabela “usuarios”. Veja o código abaixo:

```
db.delete("usuarios", "numreg=" + numreg, null);
```

Dentro do método **delete** passamos como argumento a tabela na qual queremos excluir as informações (a tabela “usuarios”) e a condição (where) que determinará a exclusão do registro (o argumento “numreg = ” + numreg).

Agora vamos voltar para o arquivo “activity_banco_de_dados.xml” e dentro da tela da nossa aplicação vamos inserir um componente **Button** (abaixo do botão “Excluir Dados”), de acordo com a tabela abaixo:



Button

Propriedade	Valor
id	btexcluidados2
text	Excluir Dados (2ª Versão)
layout:width	160dp

Feito isso teremos o seguinte resultado:



Tela da aplicação em construção

Agora vamos abrir o arquivo “BancoDeDadosActivity.java” e em seguida vamos digitar o seguinte código em azul abaixo:

```
public class BancoDeDadosActivity extends Activity {

    Button btcriabanco;
    Button btcadastrardados;
    Button btcadastrardados2;
    Button btconsultardados;
    Button btalterardados;
    Button btalterardados2;
    Button btexcluidados;
    Button btexcluidados2;
```



```
SQLiteDatabase db;
```

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

    btcriabanco = (Button) findViewById(R.id.btcriarbanco);
    :
    btexcluidados2 = (Button)
    findViewById(R.id.btexcluidados2);
    btexcluidados2.setOnClickListener(new
    View.OnClickListener(){
        @Override
        public void onClick(View arg0) {
            // TODO Auto-generated method stub
            Intent excluirDados2Activity = new
            Intent(BancoDeDadosActivity.this,
            ExcluirDados2Activity.class);

            BancoDeDadosActivity.this.
            startActivity(excluirDados2Activity);
        }
    });
};
```

Depois de escrever os códigos solicitados vamos executar a nossa aplicação. Com a aplicação aberta experimente excluir um registro para conferir o resultado.



Capítulo 4 – Desenvolvendo uma aplicação completa

Neste capítulo iremos desenvolver uma aplicação completa, onde iremos utilizar todos os conceitos aprendidos até agora sobre programação com Banco de Dados na plataforma Android.

4.1) Sobre a aplicação a ser desenvolvida

A aplicação que iremos desenvolver agora vai consistir em um sistema onde iremos gerenciar peças e ferramentas para material de construção (como furadeiras, serras, chaves de fendas em geral e etc.). Nessa aplicação teremos o módulo de cadastro de peças e ferramentas, consulta de peças e ferramentas, alteração de informações das peças e ferramentas e exclusão das informações.

Nossa aplicação irá conter um banco de dados com uma tabela chamada “ferramentas”, e nessa tabela iremos adicionar os seguintes campos :

Nome do Campo	Descrição
numreg	Campo chave primária, que será o identificador dos registros da tabela.
nome_ferramenta	Neste campo será armazenado o nome da ferramenta.
fabricante	Nesta campo será armazenado o nome da fabricante da ferramenta.
preco	Neste campo será armazenado o preço da ferramenta.
cor	Neste campo armazenaremos a cor da ferramenta.
referencia	Neste campo armazenaremos o código de referencia da ferramenta.



4.2) Desenvolvendo a nossa aplicação

Com as informações do Banco de Dados e da tabela a ser criada, vamos desenvolver a nossa aplicação. Para começarmos, vamos criar um novo projeto no Android Studio, de acordo com as informações abaixo:

Application Name: SistemaFerramentas

Company Domain : <FICA A SEU CRITÉRIO>

Package Name : app.android.sistemaFerramentas

Activity Name: MenuPrincipalActivity

Layout Name : activity_menu_principal

Title: Sistema Ferramentas

Menu Resource Name : menu_menu_principal

Depois de criarmos o nosso projeto, vamos copiar para dentro do diretório "drawable" o arquivo : "icone_aplicacao_construcao.png" (que acompanha este material), que será a imagem que irá representar o ícone da nossa aplicação:

Depois de adicionar o arquivo solicitado acima dentro da pasta de imagens do nosso projeto, copie o seguinte código XML para dentro do arquivo "activity_menu_principal.xml":

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/LinearLayout1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MenuPrincipalActivity" >
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#3ba0e2" >

        <ImageView
            android:id="@+id/imageView1"
            android:layout_width="96dp"
```



```

        android:layout_height="96dp"
        android:scaleType="fitXY"
        android:src="@drawable/icone_aplicacao_construcao" />

```

```

<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="#efa95e"
    android:gravity="center" >

```

```

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Sistema Gerenciador de Ferramentas"
        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:text="Escolha uma das opções abaixo:"
    android:textAppearance="?android:attr/textAppearanceLarge" />

```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="fill_parent"
    android:gravity="center"
    android:orientation="vertical" >

```

```

    <Button
        android:id="@+id/btcadastrar_ferramenta"
        android:layout_width="250dp"
        android:layout_height="wrap_content"
        android:text="Cadastrar Ferramenta" />

```

```

    <Button
        android:id="@+id/btalterar_dados"
        android:layout_width="250dp"
        android:layout_height="wrap_content"
        android:text="Alterar Dados" />

```



<Button

```
android:id="@+id/btconsultar_ferramenta"  
android:layout_width="250dp"  
android:layout_height="wrap_content"  
android:text="Consultar Ferramenta" />
```

<Button

```
android:id="@+id/btexcluir_ferramenta"  
android:layout_width="250dp"  
android:layout_height="wrap_content"  
android:text="Excluir Ferramenta" />
```

</LinearLayout>

</LinearLayout>

Feito isso teremos o seguinte resultado :



Tela da aplicação em construção



Depois de copiar o código XML para o nosso arquivo vamos construir mais uma tela para a nossa aplicação, criando um novo “Android Activity”, com os seguintes dados abaixo:

Activity Name: CadastrarFerramentasActivity

Layout Name : activity_cadastrar_ferramentas

Title: Cadastrar Ferramentas

Menu Resource Name : menu_cadastrar_ferramentas

Depois de criar a Activity acima, vamos abrir o arquivo “activity_cadastrar_ferramentas.xml” para digitarmos o seguinte código XML abaixo:

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/LinearLayout1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".CadastrarFerramentasActivity" >

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#3ba0e2" >

        <ImageView
            android:id="@+id/imageView1"
            android:layout_width="96dp"
            android:layout_height="96dp"
            android:scaleType="fitXY"
            android:src="@drawable/icone_aplicacao_construcao" />

        <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="#efa95e"
android:gravity="center"
android:orientation="vertical" >

```

```

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Sistema Gerenciador de Ferramentas
        [CADASTRO]"
        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:paddingBottom="10dp"
    android:text="Digite as informações abaixo:"
    android:textAppearance="?android:attr/textAppearanceMedium" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Nome da Ferramenta:"
    android:textColor="#000000"
    android:textColorHint="#000000" />

<EditText
    android:id="@+id/ednome_ferramenta"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" >

    <requestFocus />

</EditText>

<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Fabricante:"
    android:textColor="#000000"
    android:textColorHint="#000000" />

```




```

<EditText
    android:id="@+id/edfabricante"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" >
</EditText>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal" >

    <LinearLayout
        android:layout_width="160dp"
        android:layout_height="match_parent"
        android:orientation="vertical" >
        <TextView
            android:id="@+id/TextView01"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Preço:"
            android:textColor="#000000"
            android:textColorHint="#000000" />

        <EditText
            android:id="@+id/edpreco"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="numberDecimal" />

        </LinearLayout>

        <LinearLayout
            android:layout_width="160dp"
            android:layout_height="match_parent"
            android:orientation="vertical" >
            <TextView
                android:id="@+id/TextView02"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Cor:"
                android:textColor="#000000"
                android:textColorHint="#000000" />

            <EditText
                android:id="@+id/edcor"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:ems="10" />

            </LinearLayout>

        </LinearLayout>
    </LinearLayout>

```



```

<TextView
    android:id="@+id/textView6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Referência:"
    android:textColor="#000000" />

<EditText
    android:id="@+id/edreferencia"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="fill_parent"
    android:layout_gravity="center"
    android:gravity="center" >

    <Button
        android:id="@+id/btcadastrar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Cadastrar" />

    <Button
        android:id="@+id/btfechar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Fechar" />

</LinearLayout>

</LinearLayout>

```

Feito isso teremos o seguinte resultado em seguida:



Tela da aplicação em construção

Depois de copiar o código para o nosso arquivo, salve as suas alterações.

Agora vamos construir mais uma tela para a nossa aplicação, criando uma nova Activity, com os seguintes dados abaixo:

Activity Name: AlterarDadosActivity

Layout Name : activity_alterar_ferramentas

Title: Alterar Dados

Menu Resource Name : menu_alterar_dados

Depois de criar a nossa Activity, abra o arquivo "activity_alterar_dados.xml" para digitarmos o seguinte código XML abaixo:



```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/LinearLayout1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".CadastrarFerramentasActivity" >

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#3ba0e2" >

        <ImageView
            android:id="@+id/imageView1"
            android:layout_width="96dp"
            android:layout_height="96dp"
            android:scaleType="fitXY"
            android:src="@drawable/icone_aplicacao_construcao" />

        <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="#efa95e"
            android:gravity="center"
            android:orientation="vertical" >

            <TextView
                android:id="@+id/textView2"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Sistema Gerenciador de Ferramentas
                [ALTERAR DADOS]"
                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:paddingBottom="10dp"
            android:text="Digite as informações abaixo:"
            android:textAppearance="?android:attr/textAppearanceMedium" />

        <TextView
            android:id="@+id/textView4"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"

```



```

        android:text="Nome da Ferramenta:"
        android:textColor="#000000"
        android:textColorHint="#000000" />
<EditText
    android:id="@+id/ednome_ferramenta"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" >

    <requestFocus />
</EditText>

<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Fabricante:"
    android:textColor="#000000"
    android:textColorHint="#000000" />
<EditText
    android:id="@+id/edfabricante"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" >
</EditText>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal" >

    <LinearLayout
        android:layout_width="160dp"
        android:layout_height="match_parent"
        android:orientation="vertical" >
        <TextView
            android:id="@+id/TextView01"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Preço:"
            android:textColor="#000000"
            android:textColorHint="#000000" />
        <EditText
            android:id="@+id/edpreco"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="numberDecimal" />
        </LinearLayout>

    <LinearLayout
        android:layout_width="160dp"
        android:layout_height="match_parent"

```




```

        android:orientation="vertical" >
<TextView
    android:id="@+id/TextView02"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Cor: "
    android:textColor="#000000"
    android:textColorHint="#000000" />

<EditText
    android:id="@+id/edcor"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" />

</LinearLayout>

</LinearLayout>

<TextView
    android:id="@+id/textView6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Referência: "
    android:textColor="#000000" />

<EditText
    android:id="@+id/edreferencia"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" />
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="fill_parent"
    android:layout_gravity="center"
    android:gravity="center" >

    <Button
        android:id="@+id/btalterar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Alterar Dados" />

    <Button
        android:id="@+id/btfechar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Fechar" />

</LinearLayout>

</LinearLayout>

```



Feito isso teremos o seguinte resultado abaixo:



Tela da aplicação em construção

Depois de copiar o código para o nosso arquivo, salve as suas alterações.

Agora vamos construir mais uma tela para a nossa aplicação, criando um novo “Android Activity”, com os seguintes dados abaixo:

Activity Name: ConsultarFerramentaActivity

Layout Name : activity_consultar_ferramenta

Title: Consultar Ferramenta

Menu Resource Name : menu_consultar_ferramenta

Depois de criar a nossa Activity, abra o arquivo “activity_consultar_ferramenta.xml” para digitarmos o seguinte código XML em seguida:



```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/LinearLayout1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".CadastrarFerramentasActivity" >

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#3ba0e2" >

        <ImageView
            android:id="@+id/imageView1"
            android:layout_width="96dp"
            android:layout_height="96dp"
            android:scaleType="fitXY"
            android:src="@drawable/icone_aplicacao_construcao" />

        <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="#efa95e"
            android:gravity="center"
            android:orientation="vertical" >

            <TextView
                android:id="@+id/textView2"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Sistema Gerenciador de Ferramentas -
                [CONSULTAR DADOS]"
                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:paddingBottom="10dp"
            android:text="Visualize as informações abaixo:"
            android:textAppearance="?android:attr/textAppearanceMedium" />

```



```

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingBottom="5dp"
    android:text="Nome da Ferramenta:"
    android:textColor="#000000"
    android:textColorHint="#000000" />
<TextView
    android:id="@+id/txtnome_ferramenta"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:paddingBottom="5dp"
    android:text="[Nome da Ferramenta]"
    android:textColor="#c38a4c"
    android:textSize="15sp" >
    <requestFocus />
</TextView>
<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingBottom="5dp"
    android:text="Fabricante:"
    android:textColor="#000000"
    android:textColorHint="#000000" />
<TextView
    android:id="@+id/txtfabricante"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:paddingBottom="5dp"
    android:text="[Fabricante]"
    android:textColor="#c38a4c"
    android:textSize="15sp" >
</TextView>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal" >

    <LinearLayout
        android:layout_width="160dp"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:paddingBottom="5dp" >

        <TextView
            android:id="@+id/TextView01"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:paddingBottom="5dp"

```



```

        android:text="Preço:"
        android:textColor="#000000"
        android:textColorHint="#000000" />

```

```

<TextView
    android:id="@+id/txtpreco"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="numberDecimal"
    android:text="[Preço]"
    android:textColor="#c38a4c"
    android:textSize="15sp" />

</LinearLayout>

<LinearLayout
    android:layout_width="160dp"
    android:layout_height="match_parent"
    android:orientation="vertical" >

```

```

<TextView
    android:id="@+id/TextView02"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingBottom="5dp"
    android:text="Cor:"
    android:textColor="#000000"
    android:textColorHint="#000000" />

```

```

<TextView
    android:id="@+id/txtcor"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:text="[Cor]"
    android:textColor="#c38a4c"
    android:textSize="15sp" />

```

```

</LinearLayout>

```

```

</LinearLayout>

```

```

<TextView
    android:id="@+id/textView6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingBottom="5dp"
    android:text="Referência:"
    android:textColor="#000000" />

```

```

<TextView
    android:id="@+id/txtreferencia"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"

```




```

        android:ems="10"
        android:paddingBottom="5dp"
        android:text="[Referência]"
        android:textColor="#c38a4c"
        android:textSize="15sp" />
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="fill_parent"
        android:layout_gravity="center"
        android:gravity="center" >

        <Button
            android:id="@+id/btfechar"
            android:layout_width="200dp"
            android:layout_height="wrap_content"
            android:text="Fechar Consulta" />
    </LinearLayout>
</LinearLayout>

```

Feito isso teremos o seguinte resultado abaixo:



Tela da aplicação em construção



Agora vamos construir mais uma tela para a nossa aplicação, criando uma nova Activity, com os seguintes dados abaixo:

Activity Name: ExcluirFerramentaActivity

Layout Name : activity_excluir_ferramenta

Title: Excluir Ferramenta

Menu Resource Name : menu_excluir_ferramenta

Depois de criar a nossa Activity, abra o arquivo "activity_excluir_ferramenta.xml" para digitarmos o seguinte código XML abaixo:

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/LinearLayout1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".CadastrarFerramentasActivity" >
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#3ba0e2" >

        <ImageView
            android:id="@+id/imageView1"
            android:layout_width="96dp"
            android:layout_height="96dp"
            android:scaleType="fitXY"
            android:src="@drawable/icone_aplicacao_construcao" />

        <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="#efa95e"
            android:gravity="center"
            android:orientation="vertical" >
```



```

        <TextView
            android:id="@+id/textView2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Sistema Gerenciador de Ferramentas [EXCLUIR DADOS]"
            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:paddingBottom="10dp"
    android:text="Visualize as informações abaixo:"
    android:textAppearance="?android:attr/textAppearanceMedium" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingBottom="5dp"
    android:text="Nome da Ferramenta:"
    android:textColor="#000000"
    android:textColorHint="#000000" />

<TextView
    android:id="@+id/txtnome_ferramenta"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:paddingBottom="5dp"
    android:text="[Nome da Ferramenta]"
    android:textColor="#c38a4c"
    android:textSize="15sp" >

    <requestFocus />
</TextView>

<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingBottom="5dp"
    android:text="Fabricante:"
    android:textColor="#000000"
    android:textColorHint="#000000" />

<TextView
    android:id="@+id/txtfabricante"
    android:layout_width="match_parent"

```



```

android:layout_height="wrap_content"
android:ems="10"
android:paddingBottom="5dp"
android:text="[Fabricante]"
android:textColor="#c38a4c"
android:textSize="15sp" >

```

</TextView>

<LinearLayout

```

android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="horizontal" >

```

<LinearLayout

```

android:layout_width="160dp"
android:layout_height="match_parent"
android:orientation="vertical"
android:paddingBottom="5dp" >

```

<TextView

```

android:id="@+id/TextView01"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:paddingBottom="5dp"
android:text="Preço:"
android:textColor="#000000"
android:textColorHint="#000000" />

```

<TextView

```

android:id="@+id/txtpreco"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:ems="10"
android:inputType="numberDecimal"
android:text="[Preço]"
android:textColor="#c38a4c"
android:textSize="15sp" />

```

</LinearLayout>

<LinearLayout

```

android:layout_width="160dp"
android:layout_height="match_parent"
android:orientation="vertical" >

```

<TextView

```

android:id="@+id/TextView02"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:paddingBottom="5dp"
android:text="Cor:"
android:textColor="#000000"
android:textColorHint="#000000" />

```



```

        <TextView
            android:id="@+id/txtcor"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:text="[Cor]"
            android:textColor="#c38a4c"
            android:textSize="15sp" />
    </LinearLayout>
</LinearLayout>

<TextView
    android:id="@+id/textView6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingBottom="5dp"
    android:text="Referência:"
    android:textColor="#000000" />
<TextView
    android:id="@+id/txtreferencia"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:paddingBottom="5dp"
    android:text="[Referência]"
    android:textColor="#c38a4c"
    android:textSize="15sp" />
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="fill_parent"
    android:layout_gravity="center"
    android:gravity="center" >

    <Button
        android:id="@+id/btexcluir_ferramenta"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:text="Excluir Ferramenta" />

    <Button
        android:id="@+id/btfechar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Fechar" />

</LinearLayout>
</LinearLayout>

```

Feito isso teremos o seguinte resultado em seguida:



Tela da aplicação em construção

Agora vamos construir mais uma tela para a nossa aplicação, criando um novo “Android Activity”, com os seguintes dados abaixo:

Activity Name: BuscaFerramentasActivity

Layout Name : activity_busca_ferramentas

Title: Busca Ferramentas

Menu Resource Name : menu_busca_ferramentas

Depois de criar a nossa Activity, abra o arquivo “activity_busca_ferramentas.xml” para digitarmos o seguinte código XML em seguida:



```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/LinearLayout1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".BuscaFerramentasActivity" >
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#3ba0e2" >

        <ImageView
            android:id="@+id/imageView1"
            android:layout_width="96dp"
            android:layout_height="96dp"
            android:scaleType="fitXY"
            android:src="@drawable/icone_aplicacao_construcao" />

        <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="#efa95e"
            android:gravity="center" >

            <TextView
                android:id="@+id/textView2"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Sistema Gerenciador de Ferramentas [BUSCAR]"
                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:text="Opções de Busca:"
            android:textAppearance="?android:attr/textAppearanceMedium" />

        <Spinner
            android:id="@+id/spnopcoes"
            android:layout_width="match_parent"

```



```

        android:layout_height="wrap_content"
        android:entries="@array/opcoes_busca" />

```

<LinearLayout

```

        android:id="@+id/layout_campo_busca"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:visibility="invisible" >

```

<EditText

```

        android:id="@+id/edpalavrachave"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:layout_weight="1" >

```

```

        <requestFocus />

```

```

    </EditText>

```

<Button

```

        android:id="@+id/btbuscar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Buscar" />

```

```

</LinearLayout>

```

<ScrollView

```

        android:id="@+id/scrollView1"
        android:layout_width="match_parent"
        android:layout_height="175dp" >

```

<LinearLayout

```

        android:layout_width="match_parent"
        android:layout_height="match_parent" >

```

<ListView

```

        android:id="@+id/lstresultado_busca"
        android:layout_width="match_parent"
        android:layout_height="175dp"
        android:layout_weight="1" >

```

```

    </ListView>

```

```

    </LinearLayout>

```

```

</ScrollView>

```

```

</LinearLayout>

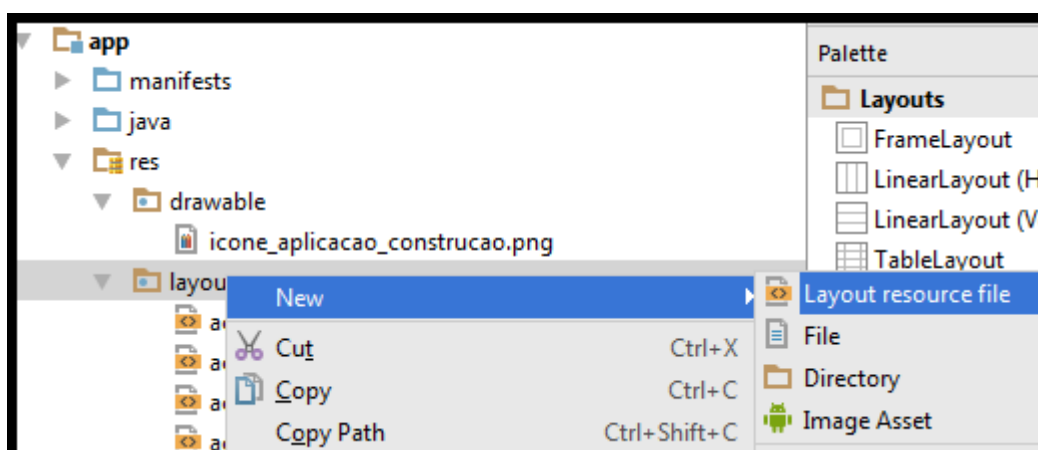
```

Feito isso teremos o seguinte resultado abaixo:



Tela da aplicação em construção

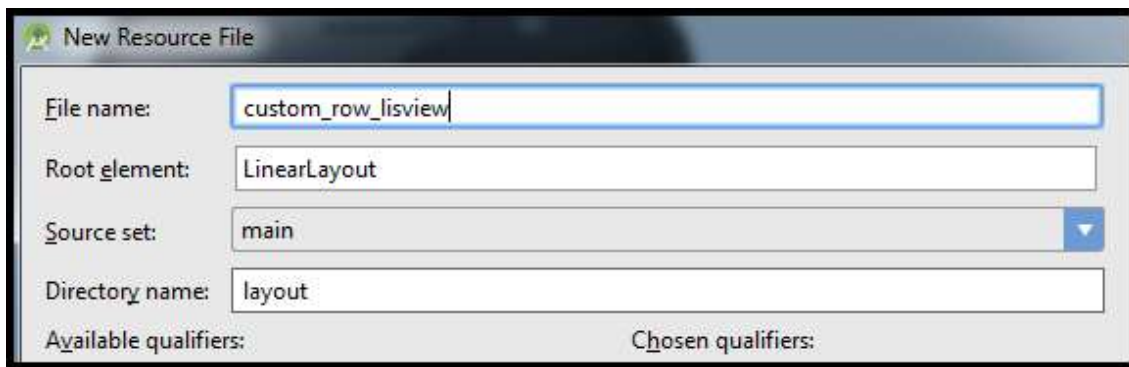
Agora vamos criar um novo arquivo XML (sem Activity) clicando com o botão direito sobre o diretório “layout” (presente dentro da pasta “res”) e em seguida selecionando “New” / “Layout resource file”. Vejamos na figura abaixo :



Criando um novo arquivo XML



Na caixa de diálogo que surge, digite no campo “File name” o nome do nosso arquivo, que irá se chamar “custom_row_listview” (não precisa informar a extensão “.xml”). Veja na figura a seguir :



Definindo o nome do arquivo

Para finalizar, clique em “OK” para gerar o nosso arquivo.

Agora vamos dentro do arquivo “custom_row_listview.xml” digitar o seguinte código abaixo:

```
<?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:orientation="vertical" >
    <TextView
        android:id="@+id/txtnome_ferramenta"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="[Nome da Ferramenta]"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:textColor="#c38a4c"
        android:textSize="17sp" />
    <TextView
        android:id="@+id/txtfabricante"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="[Fabricante]"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textSize="15sp" />
    <TextView
        android:id="@+id/txtreferencia"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="[Referência]"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textSize="15sp" />
</LinearLayout>
```

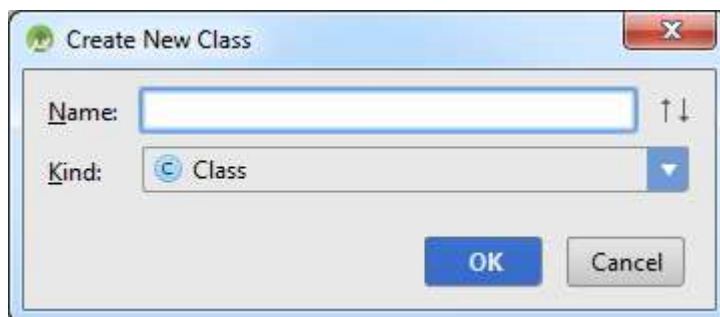



O resultado conferimos na figura seguinte :

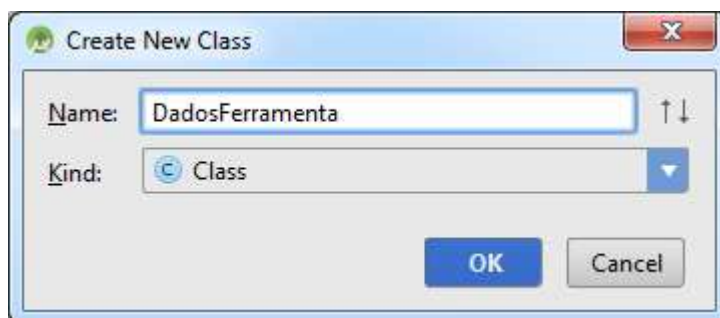


Layout de como será a exibida os resultados da busca na ListView

Agora vamos criar uma classe chamada **DadosFerramenta** dentro no nosso projeto (dentro do diretório “app.android.sistemaFerramentas”). Para isso basta clicar com o botão direito do mouse sobre o nome do nosso projeto e em seguida selecionar “New” / “Class”. Feito irá surgir a seguinte caixa de diálogo:



Dentro do campo “Name” digite o nome da nossa classe, que vai se chamar **DadosFerramenta**. Veja o resultado na figura seguinte:





Depois de preencher os dados da nossa classe, clique no botão “OK” para que a classe seja gerada. Feito isso vamos digitar o seu código abaixo:

```
package app.android.sistemaferamentas;

public class DadosFerramenta {

    String nome_ferramenta;
    String fabricante;
    String referencia;

    public void setNomeFerramenta(String n)
    {
        nome_ferramenta = n;
    }

    public String getNomeFerramenta()
    {
        return nome_ferramenta;
    }

    public void setFabricante(String f)
    {
        fabricante = f;
    }

    public String getFabricante()
    {
        return fabricante;
    }

    public void setReferencia(String ref)
    {
        referencia = ref;
    }

    public String getReferencia()
    {
        return referencia;
    }

}
```



Agora dentro do nosso projeto vamos criar uma classe chamada **MyCustomRowBaseAdapter**, com o seguinte código abaixo:

```
package app.android.sistemaferamentas;

import java.util.ArrayList;

import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.TextView;

public class MyCustomRowBaseAdapter extends BaseAdapter{

    private static ArrayList<DadosFerramenta>
        dadosFerramentaArrayList;

    private LayoutInflater mInflater;

    public MyCustomRowBaseAdapter(Context context,
        ArrayList<DadosFerramenta> dadosFerramenta)
    {
        dadosFerramentaArrayList = dadosFerramenta;
        mInflater = LayoutInflater.from(context);
    }

    @Override
    public int getCount() {

        return dadosFerramentaArrayList.size();
    }

    @Override
    public Object getItem(int position) {

        return dadosFerramentaArrayList.get(position);
    }

    @Override
    public long getItemId(int position) {

        return position;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup
        parent) {

        ViewHolder holder;
```



```

if(convertView == null)
{
    convertView = mInflater.inflate
        (R.layout.custom_row_listview, null);
    holder = new ViewHolder();

    holder.txtnomeferramenta = (TextView)
        convertView.findViewById(R.id.txtnome_ferramenta);
    holder.txtfabricante = (TextView)
        convertView.findViewById(R.id.txtfabricante);
    holder.txtreferencia = (TextView)
        convertView.findViewById(R.id.txtreferencia);
    convertView.setTag(holder);

}
else
{
    holder = (ViewHolder) convertView.getTag();

    holder.txtnomeferramenta.setText(dadosFerramentaArrayList.get
        (position).nome_ferramenta);
    holder.txtfabricante.setText(dadosFerramentaArrayList.get
        (position).fabricante);
    holder.txtreferencia.setText(dadosFerramentaArrayList.get
        (position).referencia);

    return convertView;
}

static class ViewHolder
{
    TextView txtnomeferramenta;
    TextView txtfabricante;
    TextView txtreferencia;
}
}

```

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

```

package app.android.sistemaferamentas;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.Context;
import android.content.Intent;

```



```
import android.database.sqlite.SQLiteDatabase;

public class MenuPrincipalActivity extends Activity {

    SQLiteDatabase db;

    Button btcadaststrar_ferramenta;
    Button btconsultar_ferramenta;
    Button btalterar_dados;
    Button btexcluir_ferramenta;

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

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

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

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

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

        btcadaststrar_ferramenta.setOnClickListener(new
            View.OnClickListener() {

                @Override
                public void onClick(View arg0) {

                    Intent cadastrarFerramentaActivity = new
                        Intent(MenuPrincipalActivity.this,
                            CadastrarFerramentasActivity.class);

                    MenuPrincipalActivity.this.startActivity
                        (cadastrarFerramentaActivity);

                }
            });

        btalterar_dados.setOnClickListener(new View.OnClickListener() {

                @Override
                public void onClick(View arg0) {

                    Intent consultaFerramentaActivity = new
                        Intent(MenuPrincipalActivity.this,
                            BuscaFerramentasActivity.class);
```




```

        consultaFerramentaActivity.putExtra
        ("opcao_dados", 1);

        MenuPrincipalActivity.this.startActivity
        (consultaFerramentaActivity);
    }
});

btconsultar_ferramenta.setOnClickListener(new
View.OnClickListener() {

    @Override
    public void onClick(View arg0) {

        Intent consultaFerramentaActivity = new
        Intent(MenuPrincipalActivity.this,
        BuscaFerramentasActivity.class);

        consultaFerramentaActivity.putExtra
        ("opcao_dados", 2);

        MenuPrincipalActivity.this.startActivity
        (consultaFerramentaActivity);
    }
});

btexcluir_ferramenta.setOnClickListener(new
View.OnClickListener() {

    @Override
    public void onClick(View arg0) {

        Intent consultaFerramentaActivity = new
        Intent(MenuPrincipalActivity.this,
        BuscaFerramentasActivity.class);
        consultaFerramentaActivity.putExtra
        ("opcao_dados", 3);

        MenuPrincipalActivity.this.startActivity
        (consultaFerramentaActivity);
    }
});

try {

    db = openOrCreateDatabase("banco_dados",
    Context.MODE_PRIVATE, null);

    db.execSQL("create table if not exists " +
    "ferramentas(numreg integer primary key " +
    "autoincrement, nome_ferramenta text not null, " +
    "fabricante text not null," + "preco float not null, " +
    "cor text not null, referencia text not null)");
}

```



```

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

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

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

```

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

```

package app.android.sistemaFerramentas;

import android.os.Bundle;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.ContentValues;
import android.content.Context;
import android.content.DialogInterface;
import android.database.sqlite.SQLiteDatabase;
import android.widget.*;
import android.view.*;

public class CadastrarFerramentasActivity extends Activity {

    EditText ednome_ferramenta, edfabricante, edpreco, edcor,
    edreferencia;

    Button btcadastrar, btfechar;

    SQLiteDatabase db;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

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

        ednome_ferramenta = (EditText)

```



```

findViewById(R.id.ednome_ferramenta);
edfabricante = (EditText) findViewById(R.id.edfabricante);
edpreco = (EditText) findViewById(R.id.edpreco);
edcor = (EditText) findViewById(R.id.edcor);
edreferencia = (EditText) findViewById(R.id.edreferencia);

btcadastrar = (Button) findViewById(R.id.btcadastrar);
btcadastrar.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View view) {

        String nome_ferramenta =
            ednome_ferramenta.getText().toString();
        String fabricante = edfabricante.getText().toString();
        String preco = edpreco.getText().toString();
        String cor = edcor.getText().toString();
        String referencia = edreferencia.getText().toString();

        ContentValues valor = new ContentValues();

        valor.put("nome_ferramenta", nome_ferramenta);
        valor.put("fabricante", fabricante);
        valor.put("preco", Float.parseFloat(preco));
        valor.put("cor", cor);
        valor.put("referencia", referencia);

        db.insert("ferramentas", null, valor);
        AlertDialog.Builder dialogo = new

        AlertDialog.Builder(CadastrarFerramentasActivity.this);
        dialogo.setTitle("Aviso");
        dialogo.setMessage("Dados cadastrados com sucesso!");
        dialogo.setNeutralButton("OK", new

        DialogInterface.OnClickListener() {

            @Override
            public void onClick(DialogInterface arg0, int arg1) {

                CadastrarFerramentasActivity.this.finish();
            }
        });

        dialogo.show();

    }
});

```



```

        btfechar = (Button) findViewById(R.id.btfechar);
        btfechar.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View arg0) {

                CadastrarFerramentasActivity.this.finish();

            }
        });
    try {

        db = openOrCreateDatabase("banco_dados",
            Context.MODE_PRIVATE, null);

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

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

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

```

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

```

package app.android.sistemaferamentas;

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

```



```
public class AlterarDadosActivity extends Activity {

    EditText ednome_ferramenta, edfabricante, edpreco,
    edcor, edreferencia;

    Button btalterar, btfechar;

    SQLiteDatabase db;

    int numreg;

    Cursor c;

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

        Bundle b = getIntent().getExtras();

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

        edfabricante = (EditText) findViewById(R.id.edfabricante);
        edpreco = (EditText) findViewById(R.id.edpreco);
        edcor = (EditText) findViewById(R.id.edcor);
        edreferencia = (EditText) findViewById(R.id.edreferencia);

        btalterar = (Button) findViewById(R.id.btalterar);
        btfechar = (Button) findViewById(R.id.btfechar);

        numreg = b.getInt("numreg");

        db = openOrCreateDatabase("banco_dados",
        Context.MODE_PRIVATE, null);

        c = db.query("ferramentas", new String[] {
            "nome_ferramenta", "fabricante", "preco", "cor",
            "referencia" }, "numreg = " + numreg, null, null, null, null);

        c.moveToFirst();

        ednome_ferramenta.setText(c.getString(0));
        edfabricante.setText(c.getString(1));
        edpreco.setText(c.getString(2));
        edcor.setText(c.getString(3));
        edreferencia.setText(c.getString(4));
    }
}
```




```

        btalterar.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View view) {

                String nome_ferramenta =
                    ednome_ferramenta.getText().toString();
                String fabricante =
                    edfabricante.getText().toString();
                String preco = edpreco.getText().toString();
                String cor = edcor.getText().toString();
                String referencia =
                    edreferencia.getText().toString();

                ContentValues valor = new ContentValues();
                valor.put("nome_ferramenta", nome_ferramenta);
                valor.put("fabricante", fabricante);
                valor.put("preco", Float.parseFloat(preco));
                valor.put("cor", cor);
                valor.put("referencia", referencia);

                db.update("ferramentas", valor, "numreg=" +
                    numreg, null);

                AlertDialog.Builder dialogo = new
                    AlertDialog.Builder(AlterarDadosActivity.this);
                dialogo.setTitle("Aviso");
                dialogo.setMessage("Dados atualizados com sucesso!");

                dialogo.setNeutralButton("OK", new
                    DialogInterface.OnClickListener() {

                        @Override
                        public void onClick(DialogInterface dialog, int
                            which) {

                            AlterarDadosActivity.this.finish();

                        }
                    });

                dialogo.show();
            }
        });

        btfechar.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                AlterarDadosActivity.this.finish();
            }
        });
    }
}

```



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

```
package app.android.sistemaferamentas;

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

public class ConsultarFerramentaActivity extends Activity {

    TextView txtnome_ferramenta, txtfabricante, txtpreco, txtcor,
    txtreferencia;

    Button btfechar;

    SQLiteDatabase db;

    Cursor c;

    int numreg;

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

        Bundle b = getIntent().getExtras();

        txtnome_ferramenta = (TextView)
        findViewById(R.id.txtnome_ferramenta);
        txtfabricante = (TextView)
        findViewById(R.id.txtfabricante);
        txtpreco = (TextView) findViewById(R.id.txtpreco);
        txtcor = (TextView) findViewById(R.id.txtcor);
        txtreferencia = (TextView)
        findViewById(R.id.txtreferencia);

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

        numreg = b.getInt("numreg");

        db = openOrCreateDatabase("banco_dados",
        Context.MODE_PRIVATE, null);
```



```

c = db.query("ferramentas", new String[] {
    "nome_ferramenta", "fabricante", "preco", "cor",
    "referencia" }, "numreg = " + numreg, null, null, null, null);

c.moveToFirst();

txtnome_ferramenta.setText(c.getString(0));
txtfabricante.setText(c.getString(1));
txtpreco.setText(c.getString(2));
txtcor.setText(c.getString(3));
txtreferencia.setText(c.getString(4));

btfechar.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View arg0) {
        ConsultarFerramentaActivity.this.finish();
    }
});
}
}

```

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

```

package app.android.sistemaferamentas;

import android.os.Bundle;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.ContentValues;
import android.content.Context;
import android.content.DialogInterface;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.view.Menu;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

public class ExcluirFerramentaActivity extends Activity {

    TextView txtnome_ferramenta, txtfabricante, txtpreco, txtcor,
    txtreferencia;

    Button btfechar;

    Button btexcluir_ferramenta;

    SQLiteDatabase db;

    Cursor c;

```



```

int numreg;

@Override
protected void onCreate(Bundle savedInstanceState) {

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

    Bundle b = getIntent().getExtras();

    txtnome_ferramenta = (TextView)
    findViewById(R.id.txtnome_ferramenta);
    txtfabricante = (TextView)
    findViewById(R.id.txtfabricante);
    txtpreco = (TextView) findViewById(R.id.txtpreco);
    txtcor = (TextView) findViewById(R.id.txtcor);
    txtreferencia = (TextView)
    findViewById(R.id.txtreferencia);

    btexcluir_ferramenta = (Button)
    findViewById(R.id.btexcluir_ferramenta);
    btfechar = (Button) findViewById(R.id.btfechar);

    numreg = b.getInt("numreg");

    db = openOrCreateDatabase("banco_dados",
    Context.MODE_PRIVATE, null);

    c = db.query("ferramentas", new String[] {
    "nome_ferramenta", "fabricante", "preco", "cor",
    "referencia" }, "numreg = " + numreg, null, null, null, null);

    c.moveToFirst();

    txtnome_ferramenta.setText(c.getString(0));
    txtfabricante.setText(c.getString(1));
    txtpreco.setText(c.getString(2));
    txtcor.setText(c.getString(3));
    txtreferencia.setText(c.getString(4));

    btexcluir_ferramenta.setOnClickListener(new
    View.OnClickListener() {

        @Override
        public void onClick(View view) {

            db.delete("ferramentas", "numreg=" + numreg, null);
            AlertDialog.Builder dialogo = new AlertDialog.Builder
            (ExcluirFerramentaActivity.this);
            dialogo.setTitle("Aviso");
            dialogo.setMessage("Dados excluidos com sucesso!");
        }
    });
}

```



```

        dialogo.setNeutralButton("OK", new
        DialogInterface.OnClickListener() {

            @Override
            public void onClick(DialogInterface dialog, int
            which) {

                ExcluirFerramentaActivity.this.finish();
            }
        });
        dialogo.show();
    }
});

btfechar.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View view) {
        ExcluirFerramentaActivity.this.finish();
    }
});
}

}
}

```

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

```

package app.android.sistemaferamentas;

import java.util.ArrayList;

import android.os.Bundle;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.EditText;
import android.widget.LinearLayout;
import android.widget.ListView;
import android.widget.Spinner;

public class BuscaFerramentasActivity extends Activity {

    Spinner spnopcoes;

```




```

LinearLayout layout_campo_busca;
EditText edpalavrachave;
ListView lstresultado_busca;

Button btbuscar;

int opcao_busca;

int opcao_dados;

int numreg;

SQLiteDatabase db;

Cursor c;

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

    layout_campo_busca = (LinearLayout)
        findViewById(R.id.layout_campo_busca);

    spnopcoes = (Spinner) findViewById(R.id.spnopcoes);

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

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

    lstresultado_busca = (ListView)
        findViewById(R.id.lstresultado_busca);

    Bundle b = getIntent().getExtras();

    opcao_dados = b.getInt("opcao_dados");

    spnopcoes.setOnItemClickListener(new
        AdapterView.OnItemClickListener() {

            @Override
            public void onItemClick(AdapterView<?> parent, View
                view, int position, long id) {

                if(position > 0)
                    layout_campo_busca.setVisibility(View.VISIBLE);
                else {
                    layout_campo_busca.setVisibility(View.INVISIBLE);
                    BuscarTudo();
                }
                opcao_busca = position;
            }
        }
    );

```



```

        @Override
        public void onNothingSelected(AdapterView<?> parent)
        {

        }

    });

    btbuscar.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View arg0) {

            String busca = edpalavrachave.
                getText().toString();

            switch (opcao_busca) {

                case 1: BuscaPorNome(busca); break;
                case 2: BuscaPorFabricante(busca); break;
                case 3: BuscaPorReferencia(busca); break;

            }

        }

    });

    lstresultado_busca.setOnItemClickListener(new

        AdapterView.OnItemClickListener() {

            @Override
            public void onItemClick(AdapterView<?> parent, View
                view, int position, long id) {

                Intent telaDados = null;
                switch (opcao_dados) {

                    case 1: telaDados = new
                        Intent(BuscaFerramentasActivity.this,
                            AlterarDadosActivity.class);
                        break;

                    case 2: telaDados = new
                        Intent(BuscaFerramentasActivity.this,
                            ConsultarFerramentaActivity.class);
                        break;

                    case 3: telaDados = new
                        Intent(BuscaFerramentasActivity.this,
                            ExcluirFerramentaActivity.class);
                        break;

                }

            }

        }

    });

```



```

        c.moveToPosition(position);
        telaDados.putExtra("numreg", c.getInt(0));
        BuscaFerramentasActivity.this.startActivity(telaDados);
        BuscaFerramentasActivity.this.finish();
    }
});

try {

    db = openOrCreateDatabase("banco_dados",
        Context.MODE_PRIVATE, null);

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

public void BuscarTudo()
{
    c = db.query("ferramentas",new String []
    {"numreg","nome_ferramenta","fabricante",
    "referencia"},null,null,null,null,null);

    c.moveToFirst();

    ArrayList<DadosFerramenta> dadosFerramentaArray = new
    ArrayList<DadosFerramenta>();

    for(int x = 0; x < c.getCount() ;x++)
    {
        DadosFerramenta dadosFerramenta = new DadosFerramenta();
        dadosFerramenta.setNomeFerramenta(c.getString(1));
        dadosFerramenta.setFabricante(c.getString(2));
        dadosFerramenta.setReferencia(c.getString(3));
        dadosFerramentaArray.add(dadosFerramenta);
        c.moveToNext();
    }

    lstresultado_busca.setAdapter(new MyCustomRowBaseAdapter(this,
    dadosFerramentaArray));
}

```



```

public void BuscaPorNome(String palavra_chave)
{
    try {

        c = db.query("ferramentas", new String[] { "numreg",
            "nome_ferramenta", "fabricante", "referencia" },
            "nome_ferramenta like '%" + palavra_chave + "%' ", null,
            null, null, null);

        c.moveToFirst();

        ArrayList<DadosFerramenta> dadosFerramentaArray =
            new ArrayList<DadosFerramenta>();

        if (c.getCount() > 0) {

            for (int x = 0; x < c.getCount(); x++) {
                DadosFerramenta dadosFerramenta = new
                    DadosFerramenta();

                dadosFerramenta.setNomeFerramenta
                    (c.getString(1));
                dadosFerramenta.setFabricante
                    (c.getString(2));
                dadosFerramenta.setReferencia
                    (c.getString(3));
                dadosFerramentaArray.add
                    (dadosFerramenta);
                c.moveToNext();
            }

            lstresultado_busca.setAdapter(new
                MyCustomRowBaseAdapter(this, dadosFerramentaArray));

        } else {
            MostraMensagem("Nenhum registro foi encontrado!");
        }

    } catch (Exception e) {

        MostraMensagem("Erro " + e.toString());

    }
}

public void BuscaPorFabricante(String palavra_chave)
{
    try {

        c = db.query("ferramentas", new String[] { "numreg",
            "nome_ferramenta", "fabricante", "referencia" },
            "fabricante like '%" + palavra_chave + "%' ", null,
            null, null, null);
    }
}

```



```

        c.moveToFirst();

        ArrayList<DadosFerramenta> dadosFerramentaArray = new
        ArrayList<DadosFerramenta>();

        if (c.getCount() > 0) {

            for (int x = 0; x < c.getCount(); x++) {

                DadosFerramenta dadosFerramenta = new
                DadosFerramenta();
                dadosFerramenta.setNomeFerramenta
                (c.getString(1));
                dadosFerramenta.setFabricante
                (c.getString(2));
                dadosFerramenta.setReferencia
                (c.getString(3));
                dadosFerramentaArray.add(dadosFerramenta);
                c.moveToNext();
            }

            lstresultado_busca.setAdapter(new
            MyCustomRowBaseAdapter(this, dadosFerramentaArray));
        } else {
            MostraMensagem("Nenhum registro foi encontrado!");
        }

    } catch (Exception e) {

        MostraMensagem("Erro " + e.toString());

    }
}

public void BuscaPorReferencia(String palavra_chave)
{

    try {

        c = db.query("ferramentas", new String[] { "numreg",
        "nome_ferramenta", "fabricante", "referencia" },
        "referencia = \"" + palavra_chave + "\"", null, null,
        null, null);

        c.moveToFirst();

        ArrayList<DadosFerramenta> dadosFerramentaArray = new
        ArrayList<DadosFerramenta>();
    
```




```

        if (c.getCount() > 0) {

            for (int x = 0; x < c.getCount(); x++) {

                DadosFerramenta dadosFerramenta = new
                DadosFerramenta();

                dadosFerramenta.setNomeFerramenta(c.getString(1));
                dadosFerramenta.setFabricante(c.getString(2));
                dadosFerramenta.setReferencia(c.getString(3));

                dadosFerramentaArray.add(dadosFerramenta);

                c.moveToNext();
            }

            lstresultado_busca.setAdapter(new
            MyCustomRowBaseAdapter(this, dadosFerramentaArray));
        } else {
            MostraMensagem("Nenhum registro foi encontrado!");
        }

    } catch (Exception e) {

        MostraMensagem("Erro :" + e.toString());
    }

}

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

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

```

Depois de escrever todos os código solicitados para os arquivos do projeto informados anteriormente, vamos agora executar a nossa aplicação. O resultado você confere na figura seguinte:



Aplicação em execução – Tela principal

Vamos experimentar agora efetuar um cadastro de uma ferramenta, clicando no botão “Cadastra Ferramenta”. Feito isso será aberta a seguinte tela em seguida:

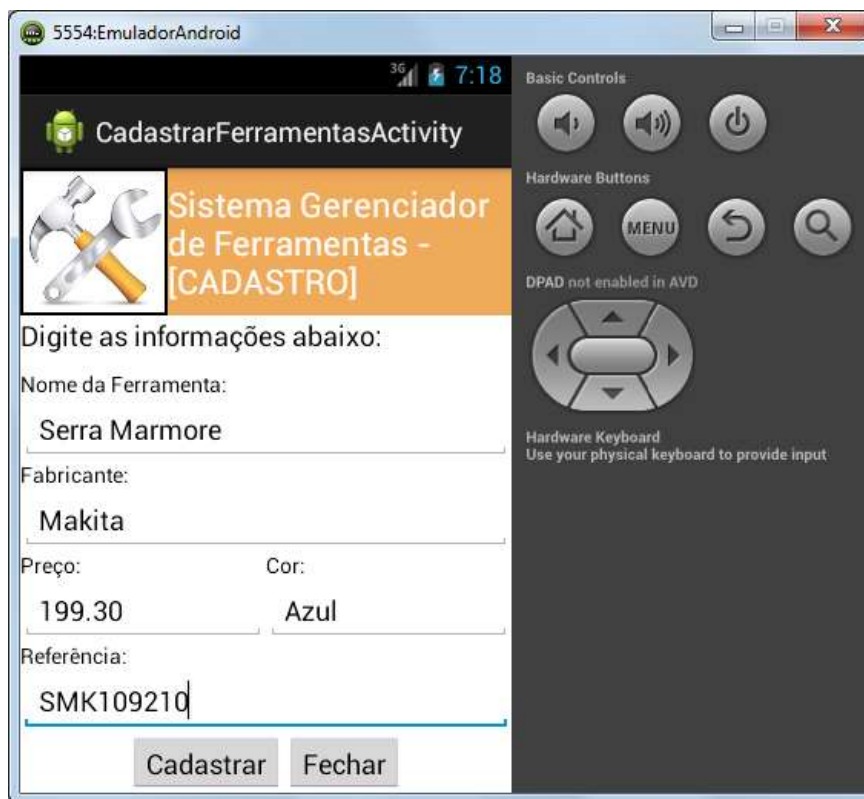


Tela da aplicação – Cadastro de Ferramenta

Para demonstrarmos o uso do cadastro de ferramentas, vamos digitar os seguintes dados abaixo:

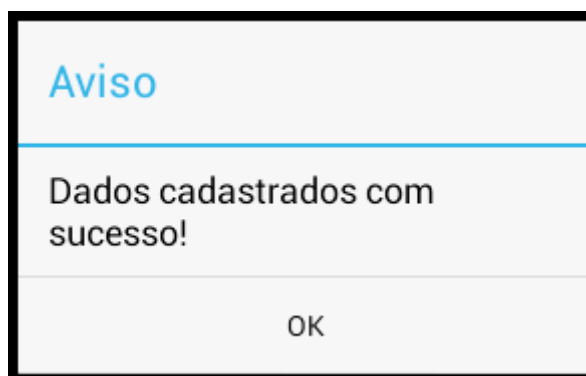
Nome da Ferramenta : Serra Marmore
Fabricante : Makita
Preço : 199,30
Cor : Azul
Referência: SMK109210

Veja o resultado em seguida :

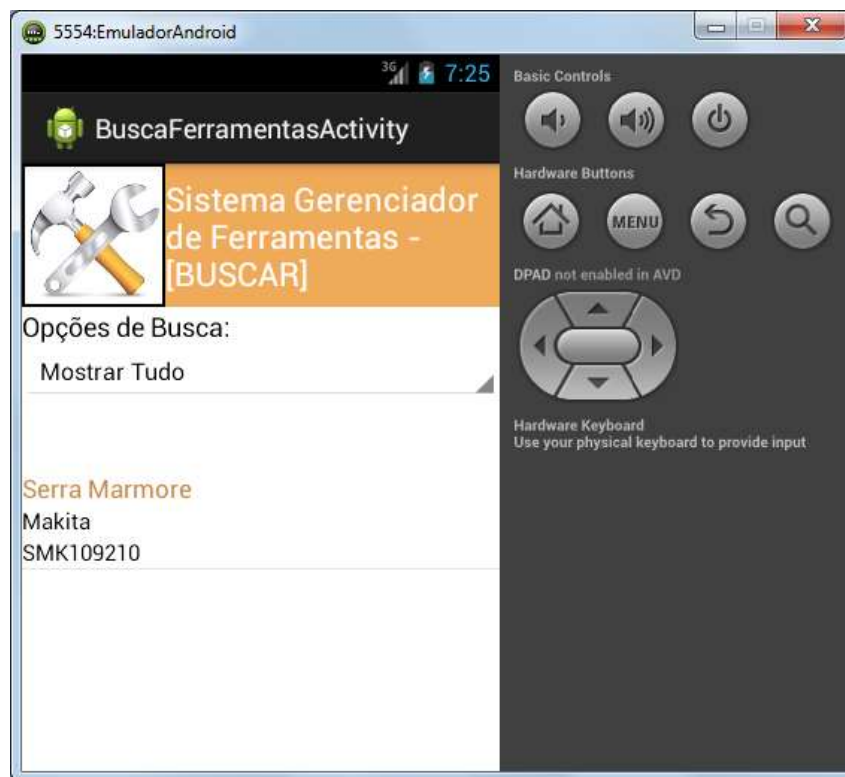


Aplicação em execução – Efetuando cadastro

Depois de preencher os dados, clique no botão “Cadastrar” para armazenar as informações no banco de dados:



Para visualizarmos a ferramenta cadastrada vamos clicar no botão “Consultar Ferramenta” na tela principal da nossa aplicação. Feito isso será aberta a seguinte tela abaixo:



Aplicação em execução – Busca por Ferramentas

Se observarmos na tela da aplicação, são exibidas , ou melhor, a exibida a ferramenta atualmente cadastrada. Vamos clicar (tocar) na ferramenta cadastrar para visualizarmos todas as suas informações, conforme podemos na figura seguinte:



Aplicação em execução – Consultando Dados

Clique no botão “Fechar Consulta” para encerrarmos a consulta. Vamos agora efetuar os seguintes cadastrados conforme é mostrado abaixo:

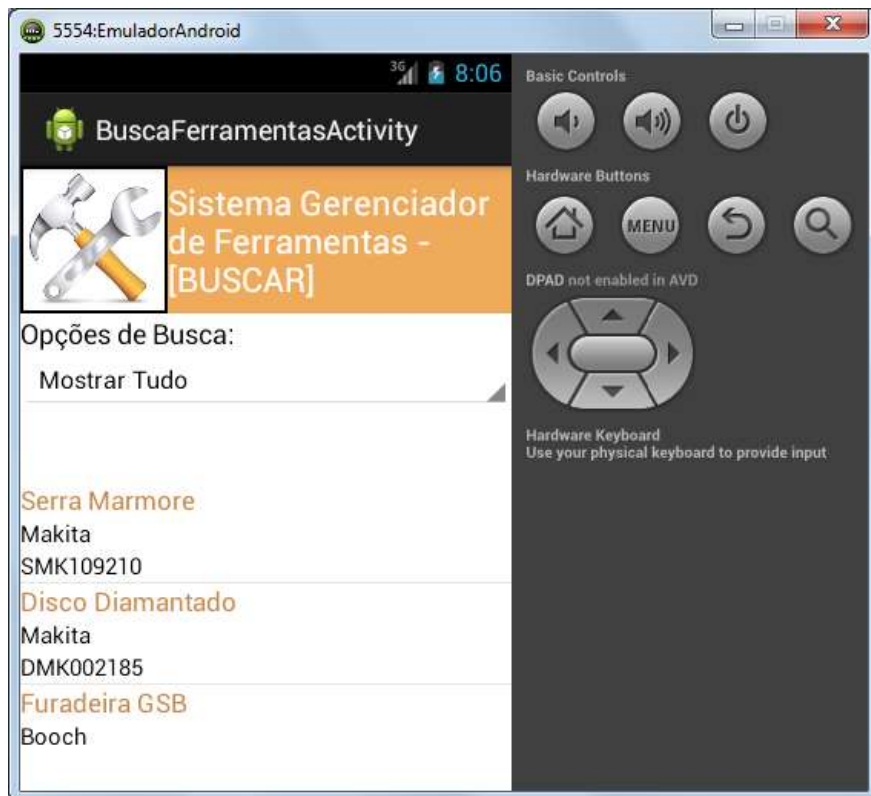
Nome da Ferramenta : Disco Diamantado
 Fabricante : Makita
 Preço : 19,00
 Cor : Azul
 Referência: DMK002185

Nome da Ferramenta : Furadeira GSB
 Fabricante : Booch
 Preço : 278,45
 Cor : Verde
 Referência: FGB190213

Nome da Ferramenta : Furadeira de Impacto
 Fabricante : Decker
 Preço : 239,90
 Cor : Amarelo
 Referência: FID003189

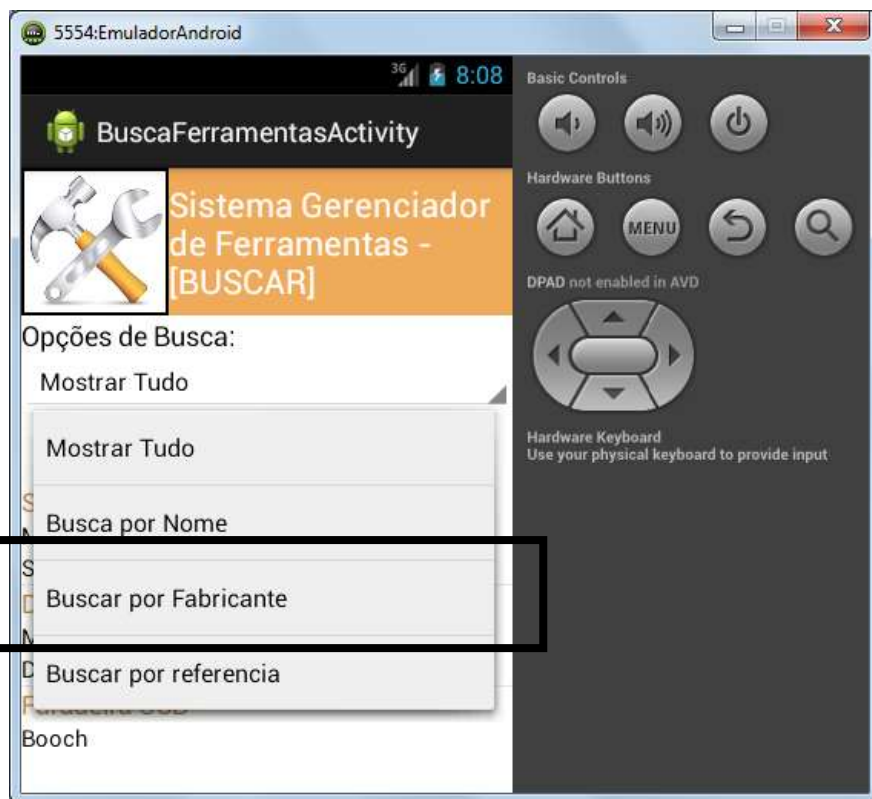


Vamos explorar agora alguns recursos da tela de busca de ferramentas. Após efetuarmos os cadastrados solicitados acima, vamos consultar uma ferramenta, clicando no botão “Consultar Ferramenta”, na tela principal da aplicação.



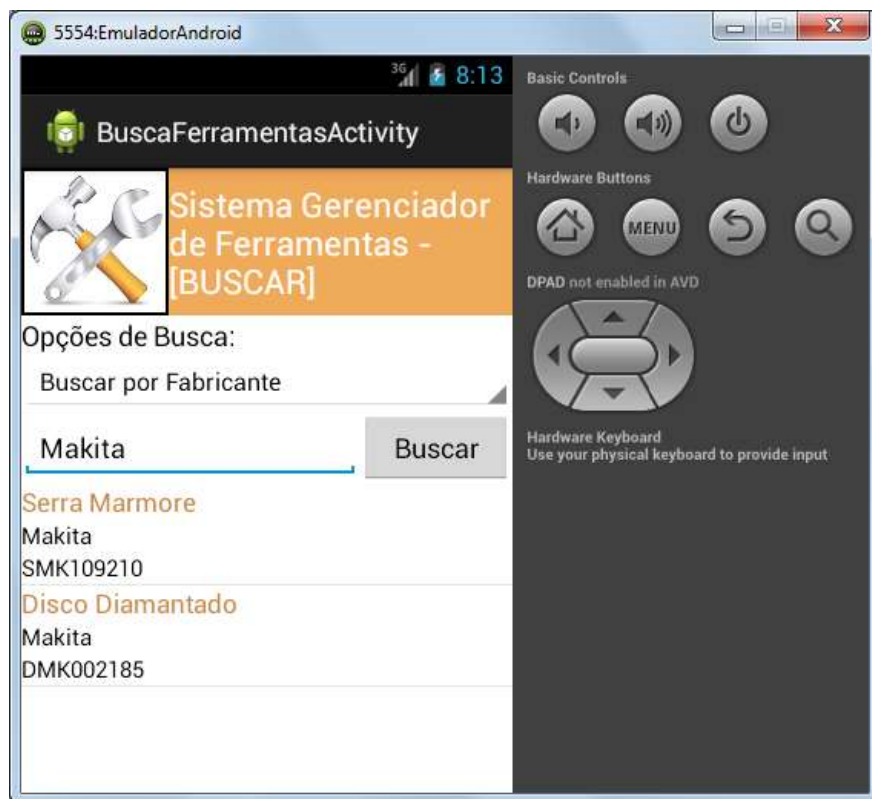
Aplicação em execução – Listando todas as ferramentas cadastradas

Vamos realizar agora uma busca personalizada pelas ferramentas cadastrados no banco de dados. Em “Opções de Busca” vamos alterar a opção de busca, buscando as ferramentas “por fabricante”, simplesmente clicando na opção já selecionada “Mostrar Tudo” e em seguida, selecionando a opção “Buscar por Fabricante”, na lista de opções apresentadas a seguir. Veja a figura a seguir:



Aplicação em execução – Selecionando a opção destacada acima

Observe que surgiu uma caixa de texto. Na vamos digitar o nome do fabricante que queremos buscar. Experimente digitar o nome “Makita” e em seguida clique no botão “Buscar”. Feito isso teremos o seguinte resultado:



Aplicação em execução – Busca personalizada

Experimente agora realizar as busca de ferramentas por “Nome” e por “Referência”, e confira os resultados mostrados pela ferramenta.



Conclusão a respeito do material

Nessa nova edição do material sobre Banco de Dados da Apostila de Android (voltada para o Android Studio) aprendemos a desenvolver aplicações voltadas para Banco de Dados. Neste material aprendemos o que vem a ser a linguagem SQL e suas principais características. Aprendemos os comandos básicos da linguagem (para a realização de consultas, atualizações de dados e etc.) . Em seguida, aprendemos os comandos presentes dentro do pacote do Android que trabalham com Banco de Dados (o pacote “android.database” e “android.database.sqlite”) para realizarmos consultas e entre outros comandos da linguagem SQL.

No capítulo seguinte aprendemos passo a passo como aplicar os comandos aprendidos no capítulo anterior na prática, através da construção de vários programas, e no último capítulo, aprendemos a construir uma aplicação completa, que trabalha com todos os recursos aprendidos sobre Banco de Dados.

Espero que esse material lhe tenha sido útil.

Abraços