**Código da classe Database.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using Android.App;

using Android.Content;

using Android.OS;

using Android.Runtime;

using Android.Views;

using Android.Widget;

using SQLite;

using crudSQLite.Resources.Model;

using Android.Util;

namespace crudSQLite.Resources.DataBaseHelper

{

class DataBase

{

string pasta = System.Environment.GetFolderPath(System.Environment.SpecialFolder.Personal);

public bool CriarBancoDeDados()

{

try

{

using (var conexao = new SQLiteConnection(System.IO.Path.Combine(pasta, "Alunos.db")))

{

conexao.CreateTable<Aluno>();

return true;

}

}

catch (SQLiteException ex)

{

Log.Info("SQLiteEx", ex.Message);

return false;

}

}

public bool InserirAluno(Aluno aluno)

{

try

{

using (var conexao = new SQLiteConnection(System.IO.Path.Combine(pasta, "Alunos.db")))

{

conexao.Insert(aluno);

return true;

}

}

catch (SQLiteException ex)

{

Log.Info("SQLiteEx", ex.Message);

return false;

}

}

public List<Aluno> GetAlunos()

{

try

{

using (var conexao = new SQLiteConnection(System.IO.Path.Combine(pasta, "Alunos.db")))

{

return conexao.Table<Aluno>().ToList();

}

}

catch (SQLiteException ex)

{

Log.Info("SQLiteEx", ex.Message);

return null;

}

}

public bool AtualizarAluno(Aluno aluno)

{

try

{

using (var conexao = new SQLiteConnection(System.IO.Path.Combine(pasta, "Alunos.db")))

{

conexao.Query<Aluno>("UPDATE Aluno set Nome=?,Idade=?, Email=? Where Id=?", aluno.Nome, aluno.Idade, aluno.Email, aluno.Id);

return true;

}

}

catch (SQLiteException ex)

{

Log.Info("SQLiteEx", ex.Message);

return false;

}

}

public bool DeletarAluno(Aluno aluno)

{

try

{

using (var conexao = new SQLiteConnection(System.IO.Path.Combine(pasta, "Alunos.db")))

{

conexao.Delete(aluno);

return true;

}

}

catch (SQLiteException ex)

{

Log.Info("SQLiteEx", ex.Message);

return false;

}

}

public bool GetAluno(int Id)

{

try

{

using (var conexao = new SQLiteConnection(System.IO.Path.Combine(pasta, "Alunos.db")))

{

conexao.Query<Aluno>("SELECT \* FROM Aluno Where Id=?", Id);

//conexao.Update(aluno);

return true;

}

}

catch (SQLiteException ex)

{

Log.Info("SQLiteEx", ex.Message);

return false;

}

}

}

}

**Código do arquivo activity\_main**

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

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

android:orientation="vertical"

android:background="#720707"

android:layout\_width="match\_parent"

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

<EditText

android:textColorHint="#E0FFFF"

android:textColor="#E0FFFF"

android:hint="Informe o nome"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/txtNome" />

<EditText

android:textColorHint="#E0FFFF"

android:textColor="#E0FFFF"

android:hint="Informe a Idade"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/txtIdade" />

<EditText

android:textColorHint="#E0FFFF"

android:textColor="#E0FFFF"

android:hint="Informe o Email"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/txtEmail" />

<LinearLayout

android:orientation="horizontal"

android:layout\_width="match\_parent"

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

<Button

android:id="@+id/btnIncluir"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Incluir"

android:layout\_weight="1" />

<Button

android:id="@+id/btnEditar"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Editar"

android:layout\_weight="1" />

<Button

android:id="@+id/btnDeletar"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Deletar"

android:layout\_weight="1" />

</LinearLayout>

<ListView

android:minWidth="25px"

android:minHeight="25px"

android:background="#8b7355"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:id="@+id/lvDados" />

</LinearLayout>

**Código da Classe ListAdapter.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using Android.App;

using Android.Content;

using Android.OS;

using Android.Runtime;

using Android.Views;

using Android.Widget;

using crudSQLite.Resources.Model;

namespace crudSQLite

{

class ListAdapter : BaseAdapter<Aluno>

{

Activity context;

private List<Aluno> alunos;

public ListAdapter(Activity \_context, List<Aluno> \_alunos)

{

this.context = \_context;

this.alunos = \_alunos;

}

public override int Count

{

get

{

return alunos.Count;

}

}

public override long GetItemId(int position)

{

return alunos[position].Id;

//return position;

}

public override View GetView(int position, View convertView, ViewGroup parent)

{

var item = alunos[position];

var view = convertView;

if (view == null)

view = context.LayoutInflater.Inflate(Resource.Layout.listViewLayout, null);

var lvtxtNome = view.FindViewById<TextView>(Resource.Id.txtvNome);

var lvtxtIdade = view.FindViewById<TextView>(Resource.Id.txtvIdade);

var lvtxtEmail = view.FindViewById<TextView>(Resource.Id.txtvEmail);

lvtxtNome.Text = alunos[position].Nome;

lvtxtIdade.Text = "" + alunos[position].Idade;

lvtxtEmail.Text = alunos[position].Email;

return view;

}

public override Aluno this[int position]

{

get

{

return alunos[position];

}

}

public override Java.Lang.Object GetItem(int position)

{

return null;

}

}

}

**Código da classe MainActivity**

using Android.App;

using Android.OS;

using Android.Support.V7.App;

using Android.Runtime;

using Android.Widget;

using System.Collections.Generic;

using crudSQLite.Resources.Model;

using crudSQLite.Resources.DataBaseHelper;

using Android.Content;

using AlertDialog = Android.App.AlertDialog;

namespace crudSQLite

{

[Activity(Label = "@string/app\_name", Theme = "@style/AppTheme", MainLauncher = true)]

public class MainActivity : AppCompatActivity

{

ListView lvDados;

List<Aluno> listaAlunos = new List<Aluno>();

DataBase db = new DataBase();

protected override void OnCreate(Bundle savedInstanceState)

{

base.OnCreate(savedInstanceState);

Xamarin.Essentials.Platform.Init(this, savedInstanceState);

// Set our view from the "main" layout resource

SetContentView(Resource.Layout.activity\_main);

//criar banco de dados

CriarBancoDados();

lvDados = FindViewById<ListView>(Resource.Id.lvDados);

var txtNome = FindViewById<EditText>(Resource.Id.txtNome);

var txtIdade = FindViewById<EditText>(Resource.Id.txtIdade);

var txtEmail = FindViewById<EditText>(Resource.Id.txtEmail);

var btnIncluir = FindViewById<Button>(Resource.Id.btnIncluir);

var btnEditar = FindViewById<Button>(Resource.Id.btnEditar);

var btnDeletar = FindViewById<Button>(Resource.Id.btnDeletar);

//carregar Dados

CarregarDados();

//botão Incluir

btnIncluir.Click += delegate

{

Aluno aluno = new Aluno()

{

Nome = txtNome.Text,

Idade = int.Parse(txtIdade.Text),

Email = txtEmail.Text

};

db.InserirAluno(aluno);

CarregarDados();

};

//botão editar

btnEditar.Click += delegate

{

Aluno aluno = new Aluno()

{

Id = int.Parse(txtNome.Tag.ToString()),

Nome = txtNome.Text,

Idade = int.Parse(txtIdade.Text),

Email = txtEmail.Text

};

db.AtualizarAluno(aluno);

CarregarDados();

};

//botão deletar

btnDeletar.Click += delegate

{

//define o alerta para executar a tarefa

AlertDialog.Builder alerta = new AlertDialog.Builder(this);

//define o titulo e o ícone a exibir no dialogo

alerta.SetTitle("Atenção");

alerta.SetIcon(Android.Resource.Drawable.IcDelete);

//define a mensagem

alerta.SetMessage("Deseja realmente excluir?");

//define o botão positivo

alerta.SetPositiveButton("Sim", (senderAlert, args) =>

{

Aluno aluno = new Aluno()

{

Id = int.Parse(txtNome.Tag.ToString()),

Nome = txtNome.Text,

Idade = int.Parse(txtIdade.Text),

Email = txtEmail.Text

};

db.DeletarAluno(aluno);

CarregarDados();

});

alerta.SetNegativeButton("Não", (senderAlert, args) =>

{

Toast.MakeText(this, "Dados Preservados", ToastLength.Short).Show();

});

//cria o alerta e exibe

Dialog dialog = alerta.Create();

dialog.Show();

};

//evento itemClick do ListView

lvDados.ItemClick += (s, e) =>

{

for (int i = 0; i < lvDados.Count; i++)

{

if (e.Position == i)

lvDados.GetChildAt(i).SetBackgroundColor(Android.Graphics.Color.Chocolate);

else

lvDados.GetChildAt(i).SetBackgroundColor(Android.Graphics.Color.Transparent);

}

//vinculando dados do listview

var lvtxtNome = e.View.FindViewById<TextView>(Resource.Id.txtvNome);

var lvtxtIdade = e.View.FindViewById<TextView>(Resource.Id.txtvIdade);

var lvtxtEmail = e.View.FindViewById<TextView>(Resource.Id.txtvEmail);

txtNome.Text = lvtxtNome.Text;

txtNome.Tag = e.Id;

txtIdade.Text = lvtxtIdade.Text;

txtEmail.Text = lvtxtEmail.Text;

};

}

//rotina para criar o banco de dados

private void CriarBancoDados()

{

db = new DataBase();

db.CriarBancoDeDados();

}

//Obtem todos os alunos da tabela Aluno e exibe no ListView

private void CarregarDados()

{

listaAlunos = db.GetAlunos();

var adapter = new ListAdapter(this, listaAlunos);

lvDados.Adapter = adapter;

}

public override void OnRequestPermissionsResult(int requestCode, string[] permissions, [GeneratedEnum] Android.Content.PM.Permission[] grantResults)

{

Xamarin.Essentials.Platform.OnRequestPermissionsResult(requestCode, permissions, grantResults);

base.OnRequestPermissionsResult(requestCode, permissions, grantResults);

}

}

}

**Código do arquivo listViewLayout**

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

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

android:orientation="vertical"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:padding="8dp">

<TextView

android:id="@+id/txtvNome"

android:text="Nome"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textColor="#000000"

android:textSize="20dp"

android:textStyle="bold"

android:paddingLeft="5dp" />

<TextView

android:id="@+id/txtvIdade"

android:text="Idade"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textColor="#ffff00"

android:paddingLeft="5dp" />

<TextView

android:id="@+id/txtvEmail"

android:text="Email"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textColor="#ffffff"

android:paddingLeft="5dp" />

</LinearLayout>