**APPENDIX A**

**SAMPLE CODES**

**ACTIVITY\_MAIN**

<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"

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

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

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

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

<TextView

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

android:text="@string/welcome" />

<TextView

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

android:layout\_gravity="center"

android:text="@string/name" />

<EditText

android:id="@+id/getname"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

android:layout\_gravity="center"

android:hint="@string/hint1"

/>

<Button

android:id="@+id/button1"

android:layout\_width="250dp"

android:layout\_height="wrap\_content"

android:gravity="center"

android:layout\_gravity="center"

android:textSize="20dp"

android:text="@string/but1name"

/>

</LinearLayout>

**//**refer the fig no b1 for the use of get the user name.

**ACTIVITY\_MAIN\_SCREEN**

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

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

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:orientation="vertical"

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

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

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

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

<TextView

android:id="@+id/question"

android:layout\_width="wrap\_content"

android:layout\_height="match\_parent"

android:layout\_weight="1" />

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="match\_parent"

android:layout\_gravity="center\_horizontal"

android:layout\_weight="1"

android:orientation="vertical" >

<RadioGroup

android:id="@+id/rg1"

android:layout\_width="wrap\_content"

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

<RadioButton

android:id="@+id/radio0"

android:layout\_width="wrap\_content"

android:layout\_height="match\_parent"

android:checked="true" />

<RadioButton

android:id="@+id/radio1"

android:layout\_width="wrap\_content"

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

<RadioButton

android:id="@+id/radio2"

android:layout\_width="wrap\_content"

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

<RadioButton

android:id="@+id/radio3"

android:layout\_width="wrap\_content"

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

</RadioGroup>

<Button

android:id="@+id/button1"

style="?android:attr/buttonStyleSmall"

android:layout\_width="wrap\_content"

android:layout\_height="match\_parent"

android:text="Next Question" />

<TextView

android:id="@+id/explanation"

android:layout\_width="wrap\_content"

android:layout\_height="match\_parent"

/>

</LinearLayout>

</LinearLayout>

//refer the fig no:b5 for the use of radio group.

**WELWITHNAME**

<RelativeLayout 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:paddingBottom="@dimen/activity\_vertical\_margin"

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

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

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

>

<TextView

android:id="@+id/textView1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

/>

<Button

android:id="@+id/sq"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/textView1"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="92dp"

android:text="Start Quiz" />

<Button

android:id="@+id/exit"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:layout\_alignRight="@+id/sq"

android:layout\_below="@+id/sq"

android:layout\_marginTop="14dp"

android:text="Exit" />

</RelativeLayout>

//refer the fig no:b2 welcome screen with the user name.

**MAIN SCREEN**

package com.example.sampleapp;

import java.io.IOException;

import java.io.InputStream;

import java.io.OutputStream;

import android.app.Activity;

import android.content.Context;

import android.content.Intent;

import android.database.Cursor;

import android.database.SQLException;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteException;

import android.database.sqlite.SQLiteOpenHelper;

import android.os.Bundle;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.RadioButton;

import android.widget.RadioGroup;

import android.widget.TextView;

import android.widget.Toast;

public class MainScreen extends Activity

{

TextView myques;

RadioGroup myrg;

RadioButton myc1,myc2,myc3,myc4;

Button NxtQues;

int counter=0;

String checked,correct;

/\*\* Called when the activity is first created. \*/

Cursor c=null;

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main\_screen);

DatabaseHelper myDbHelper = new DatabaseHelper(MainScreen.this);

try {

myDbHelper.createDataBase();

}

catch (IOException ioe)

{

throw new Error("Unable to create database");

}

try

{

myDbHelper.openDataBase();

}

catch(SQLException sqle)

{

throw sqle;

}

c=myDbHelper.query("myDbHelper.MainTable",null, null, null, null,null, null);

c.moveToFirst();

myques=(TextView)findViewById(R.id.question);

myrg=(RadioGroup)findViewById(R.id.rg1);

myc1=(RadioButton)findViewById(R.id.radio0);

myc2=(RadioButton)findViewById(R.id.radio1);

myc3=(RadioButton)findViewById(R.id.radio2);

myc4=(RadioButton)findViewById(R.id.radio3);

NxtQues=(Button)findViewById(R.id.button1);

myques.setText(c.getString(1));

myc1.setText(c.getString(2));

myc2.setText(c.getString(3));

myc3.setText(c.getString(4));

myc4.setText(c.getString(5));

correct=c.getString(6);

RadioButton.OnClickListener mop =

new RadioButton.OnClickListener()

{

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

counter++;

if( myc1.isChecked() && c.getString(1).equals(correct))

{

Toast.makeText(MainScreen.this,"You are a Genius!!!" ,

Toast.LENGTH\_LONG).show();

}

else if( myc2.isChecked() && c.getString(2).equals(correct))

{

Toast.makeText(MainScreen.this,"You are a Genius!!!" ,

Toast.LENGTH\_LONG).show();

}

else if( myc3.isChecked() && c.getString(3).equals(correct))

{

Toast.makeText(MainScreen.this,"You are a Genius!!!" ,

Toast.LENGTH\_LONG).show();

}

else if( myc4.isChecked() && c.getString(4).equals(correct))

{

Toast.makeText(MainScreen.this,"You are a Genius!!!" ,

Toast.LENGTH\_LONG).show();

}

else

{

Toast.makeText(MainScreen.this,"Wrong, make another Guess!!!" ,

Toast.LENGTH\_LONG).show();

}

}

};

myc1.setOnClickListener(mop);

myc2.setOnClickListener(mop);

myc3.setOnClickListener(mop);

myc4.setOnClickListener(mop);

NxtQues.setOnClickListener(new OnClickListener()

{

@Override

public void onClick(View V)

{

if(c.moveToNext())

{

myques.setText(c.getString(1));

myc1.setText(c.getString(2));

myc2.setText(c.getString(3));

myc3.setText(c.getString(4));

myc4.setText(c.getString(5));

correct=c.getString(6);

RadioButton.OnClickListener mop =

new RadioButton.OnClickListener()

{

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

counter++;

if( myc1.isChecked() && c.getString(1).equals(correct))

{

Toast.makeText(MainScreen.this,"You are a Genius!!!" ,

Toast.LENGTH\_LONG).show();

}

else if( myc2.isChecked() && c.getString(2).equals(correct))

{

Toast.makeText(MainScreen.this,"You are a Genius!!!" ,

Toast.LENGTH\_LONG).show();

}

else if( myc3.isChecked() && c.getString(3).equals(correct))

{

Toast.makeText(MainScreen.this,"You are a Genius!!!" ,

Toast.LENGTH\_LONG).show();

}

else if( myc4.isChecked() && c.getString(4).equals(correct))

{

Toast.makeText(MainScreen.this,"You are a Genius!!!" ,

Toast.LENGTH\_LONG).show();

}

else

{

Toast.makeText(MainScreen.this,"Wrong, make another Guess!!!" ,

Toast.LENGTH\_LONG).show();

}

}

};

myc1.setOnClickListener(mop);

myc2.setOnClickListener(mop);

myc3.setOnClickListener(mop);

myc4.setOnClickListener(mop);

}

}

});

**//**refer fig no:b6 selecting the right one among the four options

**WELCOME WITH NAME**

package com.example.sampleapp;

import android.app.Activity;

import android.content.Intent;

import android.os.Bundle;

import android.view.Menu;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.TextView;

public class Welwithname extends Activity

{

@Override

protected void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState);

setContentView(R.layout.welwithname);

Intent intent = getIntent();

String message = intent.getStringExtra(MainActivity.EXTRA\_MESSAGE);

// Create the text view

TextView textView = (TextView)findViewById(R.id.textView1);

textView.setTextSize(40);

textView.setText("Welcome" + message + "!!!" );

Button startQuiz = (Button)findViewById(R.id.sq);

// Button exit = (Button)findViewById(R.id.exit);

final Intent open = new Intent(this,MainScreen.class);

startQuiz.setOnClickListener(new OnClickListener()

{

@Override

public void onClick(View V)

{

startActivity(open);

}

});

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.welwithname, menu);

return true;

}

}

**DATABASEHELPER**

package com.example.sampleapp;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.InputStream;

import java.io.OutputStream;

import android.content.Context;

import android.database.Cursor;

import android.database.SQLException;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteException;

import android.database.sqlite.SQLiteOpenHelper;

public class DatabaseHelper extends SQLiteOpenHelper

{

//The Android's default system path of your application database.

String DB\_PATH =null;

private static String DB\_NAME = "vocablearner";

private SQLiteDatabase myDataBase;

private final Context myContext;

/\*\*

\* Constructor

\* Takes and keeps a reference of the passed context in order to access to the

application assets and resources.

\* @param context

\*/

public DatabaseHelper(Context context)

{

super(context, DB\_NAME, null, 1);

this.myContext = context;

DB\_PATH="/data/data/"+context.getPackageName()+"/"+"databases/";

}

/\*\*

\* Creates a empty database on the system and rewrites it with your own

database.

\* \*/

public void createDataBase() throws IOException

{

boolean dbExist = checkDataBase();

if(dbExist)

{

//do nothing - database already exist

}

Else

{

//By calling this method and empty database will be created into the default

system path

//of your application so we are gonna be able to overwrite that database with our

database.

this.getReadableDatabase();

try

{

copyDataBase();

}

catch (IOException e)

**40**

{

throw new Error("Error copying database");

}

}

}

/\*\*

\* Check if the database already exist to avoid re-copying the file each time you

open the application.

\* @return true if it exists, false if it doesn't

\*/

private boolean checkDataBase()

{

SQLiteDatabase checkDB = null;

Try

{

String myPath = DB\_PATH + DB\_NAME;

checkDB = SQLiteDatabase.openDatabase(myPath, null,

SQLiteDatabase.OPEN\_READONLY);

}

catch(SQLiteException e)

{

//database does't exist yet.

}

if(checkDB != null)

{

checkDB.close();

}

return checkDB != null ? true : false;

}

/\*\*

\* Copies your database from your local assets-folder to the just created empty

database in the

\* system folder, from where it can be accessed and handled.

\* This is done by transfering bytestream.

\* \*/

private void copyDataBase() throws IOException

{

//Open your local db as the input stream

InputStream myInput = myContext.getAssets().open(DB\_NAME);

// Path to the just created empty db

String outFileName = DB\_PATH + DB\_NAME;

//Open the empty db as the output stream

OutputStream myOutput = new FileOutputStream(outFileName);

//transfer bytes from the inputfile to the outputfile

byte[] buffer = new byte[1024];

int length;

while ((length = myInput.read(buffer))>0)

{

myOutput.write(buffer, 0, length);

}

//Close the streams

myOutput.flush();

myOutput.close();

myInput.close();

}

public void openDataBase() throws SQLException{

//Open the database

String myPath = DB\_PATH + DB\_NAME;

myDataBase = SQLiteDatabase.openDatabase(myPath, null,

SQLiteDatabase.OPEN\_READONLY);

}

@Override

public synchronized void close() {

if(myDataBase != null)

myDataBase.close();

super.close();

}

@Override

public void onCreate(SQLiteDatabase db) {

}

@Override

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

}

//return cursor

public Cursor query(String table,String[] columns, String selection,String[]

selectionArgs,String groupBy,String having,String orderBy){

return myDataBase.query("MainTable",null, null, null, null, null, null);

}

}