**UNIT: MOBILE APPLICATIONS DEVELOPMENT**

**SUBJECT: ALL ASSIGNMENTS**

**Assignment TWO: CONTENT PROVIDERS IN ANDROID**

Step 1: Create a new project

Step 2: Modify the strings.xml file

Step3: **create a database class**

Step 4: Creating the Content Provider class

- Click on File, then New => Other ContentProvider.

- -Create content URI

-create UriMatcher

-Create a database helper

Step5:Design the layout

Step 6: Modify the MainActivity file to make buttons respond

Activity

Cursor

Content values

Content provider

Content resolver

Content: sqlite, file, url

**1.Activity\_main.xml**

*<?*xml version="1.0" encoding="utf-8"*?>  
<*androidx.constraintlayout.widget.ConstraintLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="#168BC34A"  
 tools:context=".MainActivity"*>  
  
 <*LinearLayout  
 android:id="@+id/linearLayout"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerVertical="true"  
 android:orientation="vertical"  
 app:layout\_constraintBottom\_toTopOf="@+id/imageView"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.13"  
 tools:ignore="MissingConstraints"*>  
  
 <*TextView  
 android:id="@+id/textView1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="40dp"  
 android:layout\_marginBottom="70dp"  
 android:text="@string/heading"  
 android:textAlignment="center"  
 android:textAppearance="@style/TextAppearance.AppCompat.Large"  
 android:textColor="@android:color/holo\_green\_dark"  
 android:textSize="36sp"  
 android:textStyle="bold" */>  
  
 <*EditText  
 android:id="@+id/textName"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="20dp"  
 android:layout\_marginEnd="20dp"  
 android:layout\_marginBottom="40dp"  
 android:hint="@string/hintText" */>  
  
 <*Button  
 android:id="@+id/insertButton"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_marginStart="20dp"  
 android:layout\_marginTop="10dp"  
 android:layout\_marginEnd="20dp"  
 android:layout\_marginBottom="20dp"  
 android:background="#4CAF50"  
 android:onClick="onClickAddDetails"  
 android:text="@string/insertButtontext"  
 android:textAlignment="center"  
 android:textAppearance="@style/TextAppearance.AppCompat.Display1"  
 android:textColor="#FFFFFF"  
 android:textStyle="bold" */>  
  
 <*Button  
 android:id="@+id/loadButton"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_marginStart="20dp"  
 android:layout\_marginTop="10dp"  
 android:layout\_marginEnd="20dp"  
 android:layout\_marginBottom="20dp"  
 android:background="#4CAF50"  
 android:onClick="onClickShowDetails"  
 android:text="@string/loadButtonText"  
 android:textAlignment="center"  
 android:textAppearance="@style/TextAppearance.AppCompat.Display1"  
 android:textColor="#FFFFFF"  
 android:textStyle="bold" */>  
  
 <*TextView  
 android:id="@+id/res"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="20dp"  
 android:layout\_marginEnd="20dp"  
 android:clickable="false"  
 android:ems="10"  
 android:textColor="@android:color/holo\_green\_dark"  
 android:textSize="18sp"  
 android:textStyle="bold" */>  
  
 </*LinearLayout*>  
  
 <*ImageView  
 android:id="@+id/imageView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:srcCompat="@mipmap/ic\_launcher" */>  
  
</*androidx.constraintlayout.widget.ConstraintLayout*>*

**2.Main\_activity.java**

package com.example.reminder;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.ContentValues;  
import android.content.Context;  
import android.database.Cursor;  
import android.net.Uri;  
import android.os.Bundle;  
import android.view.MotionEvent;  
import android.view.View;  
import android.view.inputmethod.InputMethodManager;  
import android.widget.EditText;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import com.example.reminder.MyStudentsProvider;  
import com.example.reminder.R;  
  
public class MainActivity extends AppCompatActivity *{* @Override  
 protected void onCreate*(*Bundle savedInstanceState*) {* super.onCreate*(*savedInstanceState*)*;  
 setContentView*(*R.layout.*activity\_main)*;  
 *}* @Override  
 public boolean onTouchEvent*(*MotionEvent event*) {* InputMethodManager imm = *(*InputMethodManager*)*getSystemService*(*Context.*INPUT\_METHOD\_SERVICE)*;  
 imm.hideSoftInputFromWindow*(*getCurrentFocus*()*.getWindowToken*()*, 0*)*;  
 return true;  
 *}* public void onClickAddDetails*(*View view*) {  
  
 // class to add values in the database* ContentValues values = new ContentValues*()*;  
  
 *// fetching text from user* values.put*(*MyStudentsProvider.*name*, *((*EditText*)* findViewById*(*R.id.*textName))*.getText*()*.toString*())*;  
  
 *// inserting into database through content URI* getContentResolver*()*.insert*(*MyStudentsProvider.*CONTENT\_URI*, values*)*;  
  
 *// displaying a toast message* Toast.*makeText(*getBaseContext*()*, "New Record Inserted", Toast.*LENGTH\_LONG)*.show*()*;  
 *}* public void onClickShowDetails*(*View view*) {  
 // inserting complete table details in this text field* TextView resultView= *(*TextView*)* findViewById*(*R.id.*res)*;  
  
 *// creating a cursor object of the  
 // content URI* Cursor cursor = getContentResolver*()*.query*(*Uri.*parse(*"content://com.example.reminder.MyStudentsProvider/users"*)*, null, null, null, null*)*;  
  
 *// iteration of the cursor  
 // to print whole table* if*(*cursor.moveToFirst*()) {* StringBuilder strBuild=new StringBuilder*()*;  
 while *(*!cursor.isAfterLast*()) {* strBuild.append*(*"\n"+cursor.getString*(*cursor.getColumnIndex*(*"id"*))*+ "-"+ cursor.getString*(*cursor.getColumnIndex*(*"name"*)))*;  
 cursor.moveToNext*()*;  
 *}* resultView.setText*(*strBuild*)*;  
 *}* else *{* resultView.setText*(*"No Records Found"*)*;  
 *}  
 }  
}*

**3.Database\_helper.java**

package com.example.reminder; *// creating a database*import android.content.Context;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
  
public class DatabaseHelper extends SQLiteOpenHelper *{  
 // defining a constructor* DatabaseHelper*(*Context context*) {* super*(*context, *DATABASE\_NAME*, null, *DATABASE\_VERSION)*;  
 *}  
 // creating object of database  
 // to perform query  
 //private SQLiteDatabase db;  
  
 // declaring name of the database* static final String *DATABASE\_NAME* = "UserDB";  
  
 *// declaring table name of the database* static final String *TABLE\_NAME* = "Users";  
  
 *// declaring version of the database* static final int *DATABASE\_VERSION* = 1;  
  
 *// sql query to create the table* static final String *CREATE\_DB\_TABLE* = " CREATE TABLE " + *TABLE\_NAME* + " *(*id INTEGER PRIMARY KEY AUTOINCREMENT, "  
 + " name TEXT NOT NULL*)*;";  
  
  
  
 *// creating a table in the database* @Override  
 public void onCreate*(*SQLiteDatabase db*) {* db.execSQL*(CREATE\_DB\_TABLE)*;  
 *}* @Override  
 public void onUpgrade*(*SQLiteDatabase db, int oldVersion, int newVersion*) {  
  
 // sql query to drop a table  
 // having similar name* db.execSQL*(*"DROP TABLE IF EXISTS " + *TABLE\_NAME)*;  
 onCreate*(*db*)*;  
 *}  
}*

**4.Android\_manifest.xml**

*<?*xml version="1.0" encoding="utf-8"*?>  
<*manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.example.reminder"*>  
  
 <*application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/Theme.Reminder"*>  
 <*provider  
 android:name=".MyStudentsProvider"  
 android:authorities="com.example.reminder.MyStudentsProvider"  
 android:enabled="true"  
 android:exported="true"*>  
 </*provider*>  
  
 <*activity android:name=".MainActivity"*>  
 <*intent-filter*>  
 <*action android:name="android.intent.action.MAIN" */>  
  
 <*category android:name="android.intent.category.LAUNCHER" */>  
 </*intent-filter*>  
 </*activity*>  
 </*application*>  
  
</*manifest*>*

**5.MyStudentProvider.java**

package com.example.reminder;  
  
import android.content.ContentProvider;  
import android.content.ContentUris;  
import android.content.ContentValues;  
import android.content.Context;  
import android.content.UriMatcher;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteException;  
import android.database.sqlite.SQLiteQueryBuilder;  
import android.net.Uri;  
  
import java.util.HashMap;  
  
public class MyStudentsProvider extends ContentProvider *{* public MyStudentsProvider*() {  
 }  
  
 // defining authority so that other application can access it* static final String *PROVIDER\_NAME* = "com.example.reminder.MyStudentsProvider";  
  
 *// defining content URI* static final String *URL* = "content://" + *PROVIDER\_NAME* + "/users";  
  
 *// parsing the content URI* public static final Uri *CONTENT\_URI* = Uri.*parse(URL)*;  
  
 static final String *id* = "id";  
 static final String *name* = "name";  
 static final int *uriCode* = 1;  
 static final UriMatcher *uriMatcher*;  
 static HashMap*<*String, String*> values*;  
  
 static *{  
  
 // to match the content URI  
 // every time user access table under content provider  
 uriMatcher* = new UriMatcher*(*UriMatcher.*NO\_MATCH)*;  
  
 *// to access whole table  
 uriMatcher*.addURI*(PROVIDER\_NAME*, "users", *uriCode)*;  
  
 *// to access a particular row  
 // of the table  
 uriMatcher*.addURI*(PROVIDER\_NAME*, "users/\*", *uriCode)*;  
 *}* @Override  
 public String getType*(*Uri uri*) {* switch *(uriMatcher*.match*(*uri*)) {* case *uriCode*:  
 return "vnd.android.cursor.dir/users";  
 default:  
 throw new IllegalArgumentException*(*"Unsupported URI: " + uri*)*;  
 *} }  
 // creating the database* @Override  
 public boolean onCreate*() {* Context context = getContext*()*;  
 DatabaseHelper dbHelper = new DatabaseHelper*(*context*)*;  
 db = dbHelper.getWritableDatabase*()*;  
 if *(*db != null*) {* return true;  
 *}* return false;  
 *}*

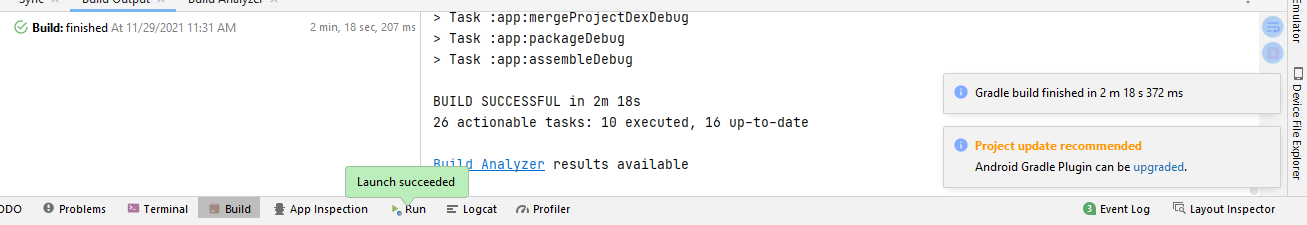
private SQLiteDatabase db;  
  
 *// declaring name of the database* static final String *DATABASE\_NAME* = "UserDB";  
  
 *// declaring table name of the database* static final String *TABLE\_NAME* = "Users";  
  
 *// declaring version of the database* static final int *DATABASE\_VERSION* = 1;  
  
 *// sql query to create the table* static final String *CREATE\_DB\_TABLE* = " CREATE TABLE " + *TABLE\_NAME* + " (id INTEGER PRIMARY KEY AUTOINCREMENT, "  
 + " name TEXT NOT NULL);";

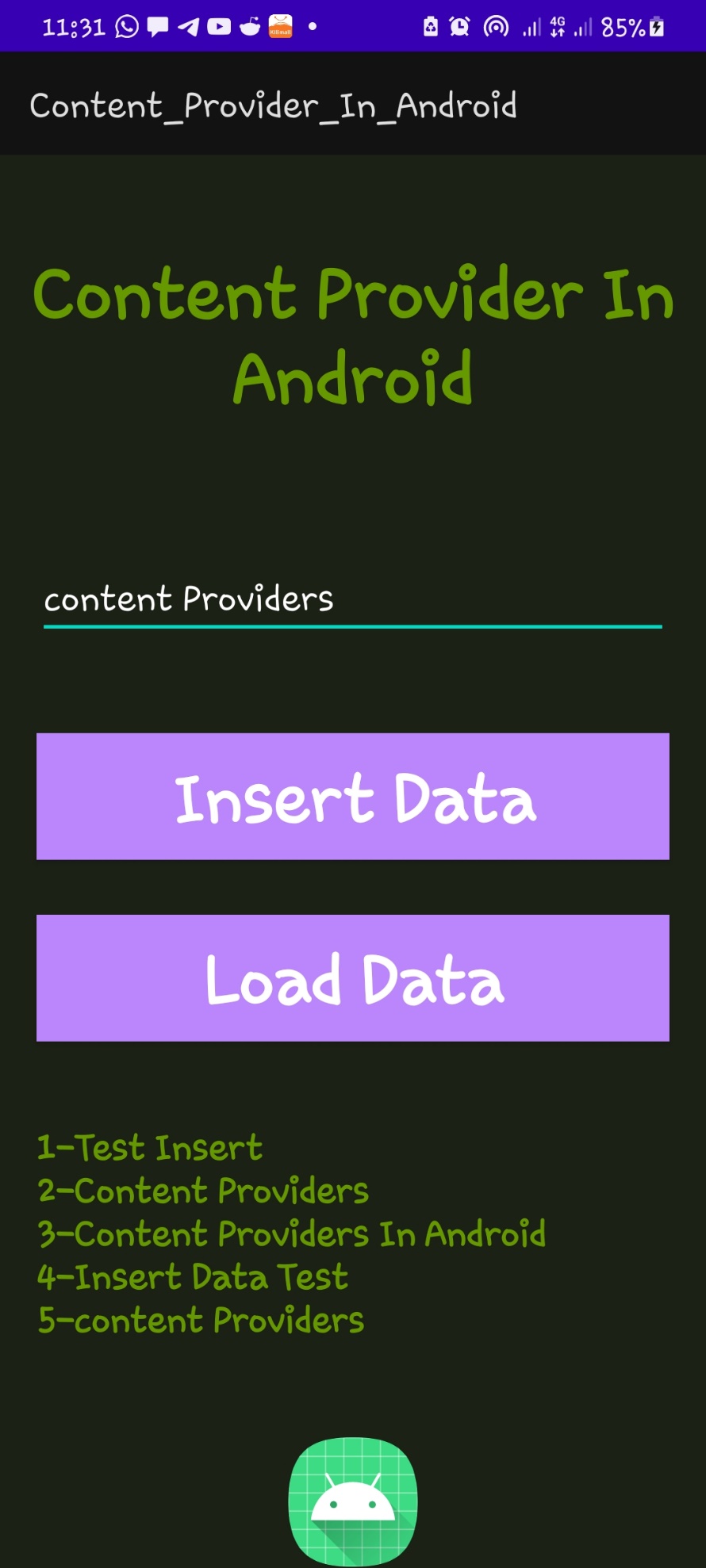
@Override  
 public Cursor query*(*Uri uri, String*[]* projection, String selection,  
 String*[]* selectionArgs, String sortOrder*) {* SQLiteQueryBuilder qb = new SQLiteQueryBuilder*()*;  
 qb.setTables*(TABLE\_NAME)*;  
 switch *(uriMatcher*.match*(*uri*)) {* case *uriCode*:  
 qb.setProjectionMap*(values)*;  
 break;  
 default:  
 throw new IllegalArgumentException*(*"Unknown URI " + uri*)*;  
 *}* if *(*sortOrder == null || sortOrder == ""*) {* sortOrder = *id*;  
 *}* Cursor c = qb.query*(*db, projection, selection, selectionArgs, null,  
 null, sortOrder*)*;  
 c.setNotificationUri*(*getContext*()*.getContentResolver*()*, uri*)*;  
 return c;  
 *}  
  
 // adding data to the database* @Override  
 public Uri insert*(*Uri uri, ContentValues values*) {* long rowID = db.insert*(TABLE\_NAME*, "", values*)*;  
 if *(*rowID > 0*) {* Uri \_uri = ContentUris.*withAppendedId(CONTENT\_URI*, rowID*)*;  
 getContext*()*.getContentResolver*()*.notifyChange*(*\_uri, null*)*;  
 return \_uri;  
 *}* throw new SQLiteException*(*"Failed to add a record into " + uri*)*;  
 *}* @Override  
 public int update*(*Uri uri, ContentValues values, String selection,  
 String*[]* selectionArgs*) {* int count = 0;  
 switch *(uriMatcher*.match*(*uri*)) {* case *uriCode*:  
 count = db.update*(TABLE\_NAME*, values, selection, selectionArgs*)*;  
 break;  
 default:  
 throw new IllegalArgumentException*(*"Unknown URI " + uri*)*;  
 *}* getContext*()*.getContentResolver*()*.notifyChange*(*uri, null*)*;  
 return count;  
 *}* @Override  
 public int delete*(*Uri uri, String selection, String*[]* selectionArgs*) {* int count = 0;  
 switch *(uriMatcher*.match*(*uri*)) {* case *uriCode*:  
 count = db.delete*(TABLE\_NAME*, selection, selectionArgs*)*;  
 break;  
 default:  
 throw new IllegalArgumentException*(*"Unknown URI " + uri*)*;  
 *}* getContext*()*.getContentResolver*()*.notifyChange*(*uri, null*)*;  
 return count;  
 *}*   
  
*}*

**6.Strings.xml**

*<*resources*>  
 <*string name="app\_name"*>*Content\_Provider\_In\_Android*</*string*>  
 <*string name="hintText"*>*Enter User Name*</*string*>  
 <*string name="heading"*>*Content Provider In Android*</*string*>  
 <*string name="insertButtontext"*>*Insert Data*</*string*>  
 <*string name="loadButtonText"*>*Load Data*</*string*>  
</*resources*>*

**7.User Interface screenshots**

****

****

**Assignment THREE: OPENING URL IN BROWSER**

**To open URL in a web browser and phone dial from an Activity**

This session shows how a mobile Application can open a web page. A web page is one of the common content. The following are five components (highlighted in bold) used to compose the application ans how they interact.

**Activity** will send **ACTION\_WEB\_SEARCH Intent** to the Android **Intent Resolver** to open given **URL** in the **web browser**

Activity

URL

Intent Resolver

The Intent Resolver parses through a list of Activities and chooses the one that would best match your Intent, in this case, the Web Browser Activity. The Intent Resolver then passes your web page to the web browser and starts the Web Browser Activity

Carry out the following tasks as assignment and practice. Submit source code by copy paste in word and screen shots for user interface, compile results and run results

**Task Example /Assignment4**

1.Create a new project with empty activity

2. using Designer layout place a button on linear layout- vertical

3. name the button Browse and label it as Browse

4.On its Onclick property write the method named *onBrowseClick*

5.in the activity include the method

**public void** onBrowseClick(View v) { // details see following code

**}**

**1.Activity\_main.xml**

*<?*xml version="1.0" encoding="utf-8"*?>  
<*LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="fill\_parent"  
 android:orientation="vertical" *>  
  
 <*Button android:id="@+id/start\_browser"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="@string/start\_browser"*/>  
  
 <*Button android:id="@+id/start\_phone"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="@string/start\_phone" */>  
  
</*LinearLayout*>*

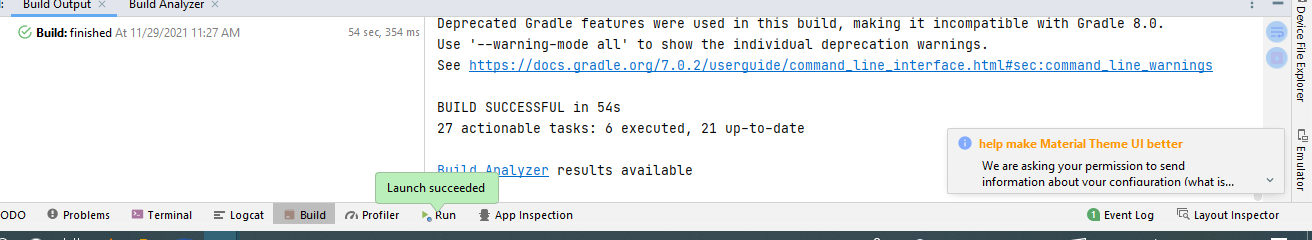
**2.Main\_activity.java**

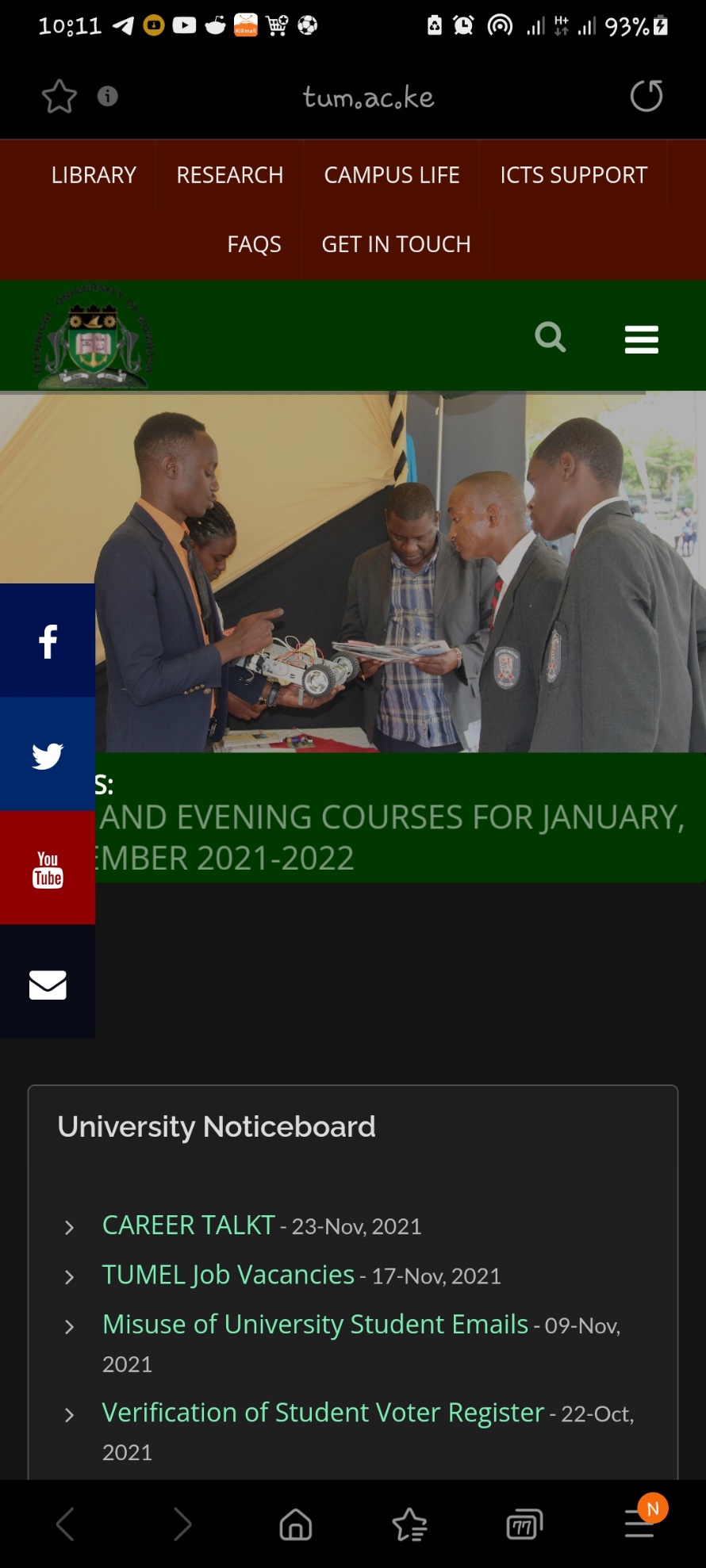
package com.example.open\_url;  
import android.net.Uri;  
import android.os.Bundle;  
import android.app.Activity;  
import android.content.Intent;  
import android.view.View;  
import android.widget.Button;  
public class MainActivity extends Activity *{* @Override protected void onCreate*(*Bundle savedInstanceState*) {* super.onCreate*(*savedInstanceState*)*;  
 setContentView*(*R.layout.*activity\_main)*;  
  
 Button startBrowser = *(*Button*)* findViewById*(*R.id.*start\_browser)*;  
 startBrowser.setOnClickListener*(*new View.OnClickListener*() {* public void onClick*(*View view*) {* Intent i = new Intent*(*android.content.Intent.*ACTION\_VIEW*, Uri.*parse(*"https://jkuat.ac,ke"*))*;  
 startActivity*(*i*)*; *}  
 })*;  
  
 Button startPhone = *(*Button*)* findViewById*(*R.id.*start\_phone)*;  
 startPhone.setOnClickListener*(*new View.OnClickListener*() {* public void onClick*(*View view*) {* Intent i = new Intent*(*android.content.Intent.*ACTION\_DIAL*, Uri.*parse(*"0702051060"*))*;  
 startActivity*(*i*)*; *}  
 })*;  
 *}  
}*

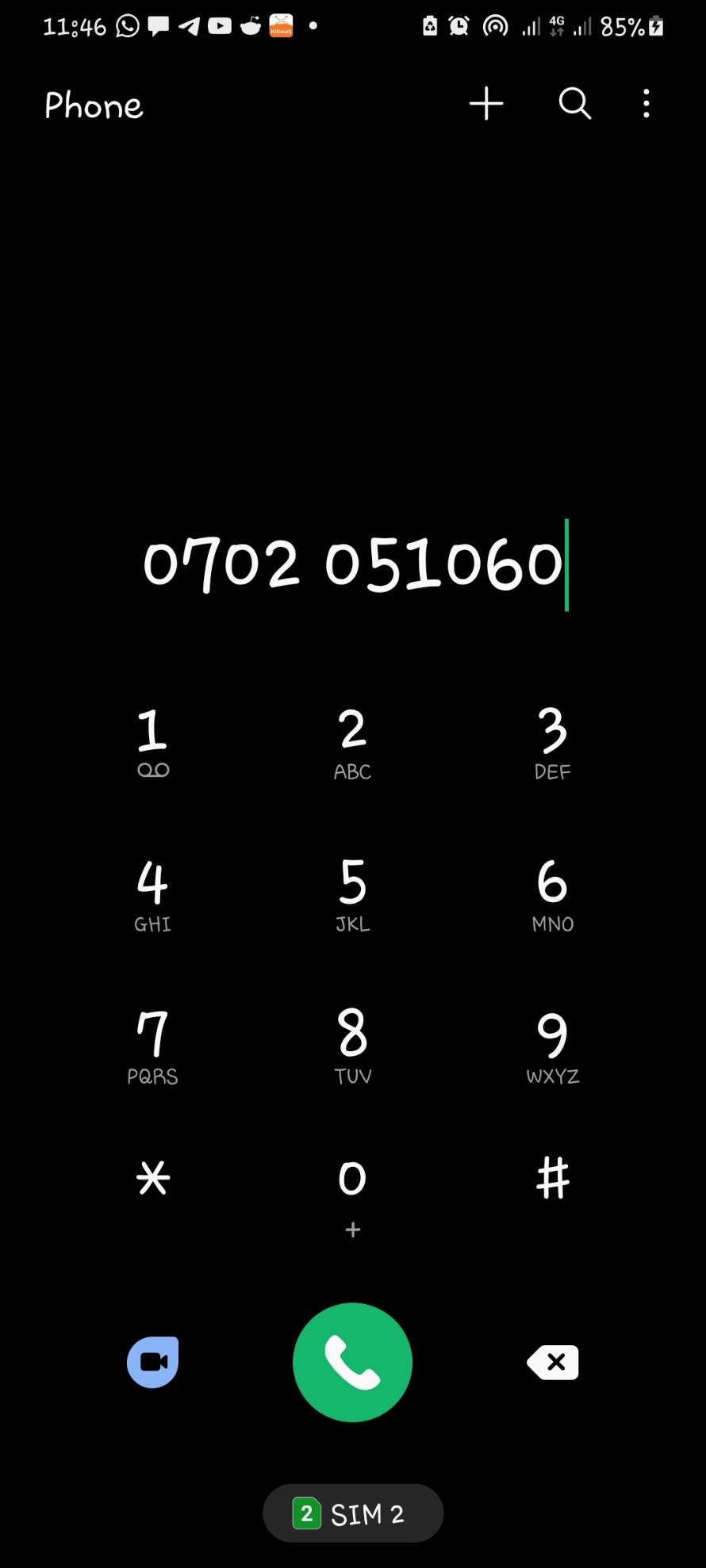
**3.Strings.xml**

*<*resources*>  
 <*string name="app\_name"*>*open\_url*</*string*>  
 <*string name="action\_settings"*>*Settings*</*string*>  
 <*string name="hello\_world"*>*Hello world!*</*string*>  
 <*string name="start\_browser"*>*Start Browser*</*string*>  
 <*string name="start\_phone"*>*Start Phone*</*string*>  
</*resources*>*

**4.User Interface screenshots**



****

****

**Assignment FOUR: ANDROID SENSORS**

Task : create the code files provided at the end of the explanation this document

**Accessing sensor information**

Include classes and interfaces as follows:

1. SensorManager class: Android.hardware.SensorManager

Methods

To get sensor instance

SensorManager sm= (SensorManager) getSystemService(SENSOR\_SERVICE);

To access and list sensors

Sensor=sm.getDefaultSensor(Sensor.TYPE\_LIGHT);

List<Sensor>deviceSensors>=sm.getSensorList(Sensor.TYPE\_ALL):

To register and unregister sensor listeners

Sm.registerListener( this, Sensor,SensorManger.SENSOR\_DELAY\_NORMAL);

1. Sensor class : Android.hardware.Sensor

Methods used to get information of the sensor

Sensor name, type, resolution

1. SensorEvent class:

its instance is created by the system. It provides information about the sensor

1. SensorEventListener interface:

provides information when sensor values or sensor (x,y,z)accuracy change

ACTIVITY

SENSOR MANAGER

SENSOR

**1.Activity\_main.xml**

*<?*xml version="1.0" encoding="utf-8"*?>  
<*RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity"*>  
  
 <*TextView  
 android:id="@+id/textView"  
 android:text="TextVieeeww"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentTop="true"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="92dp"  
 android:layout\_marginTop="114dp"*/>  
  
</*RelativeLayout*>*

**2.Main\_activity.java**

package com.example.android\_sensor;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.hardware.Sensor;  
import android.hardware.SensorEvent;  
import android.hardware.SensorEventListener;  
import android.hardware.SensorManager;  
import android.os.Bundle;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import java.util.List;  
  
public class MainActivity extends AppCompatActivity *{* SensorManager sm = null;  
 TextView textView1 = null;  
 List list;  
  
 SensorEventListener sel = new SensorEventListener*() {* @Override  
 public void onSensorChanged*(*SensorEvent event*) {* float*[]* values = event.values;  
 textView1.setText*(*"x: "+values*[*0*]*+"\ny: "+values*[*1*]*+"\nz: "+values*[*2*])*;  
 *}* @Override  
 public void onAccuracyChanged*(*Sensor sensor, int accuracy*) {  
  
 }  
 }*;  
  
 @Override  
 protected void onCreate*(*Bundle savedInstanceState*) {* super.onCreate*(*savedInstanceState*)*;  
 setContentView*(*R.layout.*activity\_main)*;  
  
 sm =*(*SensorManager*)* getSystemService*(SENSOR\_SERVICE)*;  
 textView1 = *(*TextView*)*findViewById*(*R.id.*textView)*;  
 list = sm.getSensorList*(*Sensor.*TYPE\_ACCELEROMETER)*;  
  
 if*(*list.size*()* > 0*){* sm.registerListener*(*sel, *(*Sensor*)* list.get*(*0*)*, SensorManager.*SENSOR\_DELAY\_NORMAL)*;  
 *}* else*{* Toast.*makeText(*getBaseContext*()*, "Error: No Accelerometer found", Toast.*LENGTH\_LONG)*.show*()*;  
 *}  
 }* @Override  
 protected void onStop*() {* if*(*list.size*()* > 0*){* sm.unregisterListener*(*sel*)*;  
 *}* super.onStop*()*;  
 *}  
}*

**3.User Interface screenshots**

