Gmail **Harshini Senthilkumar <senthilkumar.harshini123@gmail.com>**

Fwd: MADLAB CODES

1 message

Harshini S <harshini18058@cse.ssn.edu.in> 28 October 2021 at 21:11
To: senthilkumar.harshini123@gmail.com

---------- Forwarded message ---------
From: **Harshini S** <harshini18058@cse.ssn.edu.in>
Date: Thu, Oct 28, 2021 at 9:11 PM
Subject: MADLAB CODES
To: Likhithaverma A <likhithaverma18084@cse.ssn.edu.in>

Graphical primitives (img.animate().rotation(360).setDuration(1000);)

```
public class AnimateCar extends AppCompatActivity {
    Bitmap bg;
    ImageView img;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_animate_car);
        //Creating a bitmap
        bg = Bitmap.createBitmap(720, 1280, Bitmap.Config.RGB_565);
        img = findViewById(R.id.imageView);
        img.setImageBitmap(bg);
        //Create canvas object
        Canvas canvas = new Canvas(bg);
        //Create paint object
        Paint paint = new Paint();
        Path path=new Path();
        paint.setColor(Color.BLUE);
        paint.setTextSize(50);
        canvas.drawRect(50, 700, 650, 900, paint);
        paint.setColor(Color.GRAY);
        canvas.drawCircle(200, 900, 75, paint);
        canvas.drawCircle(500, 900, 75, paint);
        canvas.drawRect(200, 550, 550, 700, paint);
        findViewById(R.id.zoomIn).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                img.startAnimation(AnimationUtils.loadAnimation(
                        getApplicationContext(),R.anim.zoom
                ));
            }
        });

        findViewById(R.id.buttonRotate).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                img.startAnimation(AnimationUtils.loadAnimation(
                        getApplicationContext(),R.anim.rotate
                ));
            }
        });
}
}
```

DATABASE

```
public class MainActivity extends AppCompatActivity {
    EditText name, contact, dob, email;
    Button insert, update, delete, view;
    DBHelper DB;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        name = findViewById(R.id.name);
        contact = findViewById(R.id.contact);
        dob = findViewById(R.id.dob);
        email = findViewById(R.id.email);

        insert = findViewById(R.id.btnInsert);
        update = findViewById(R.id.btnUpdate);
        delete = findViewById(R.id.btnDelete);
        view = findViewById(R.id.btnView);
        DB = new DBHelper(this);
        insert.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String nameTXT = name.getText().toString();
                String contactTXT = contact.getText().toString();
                String dobTXT = dob.getText().toString();
                String emailTXT = email.getText().toString();

                Boolean checkinsertdata = DB.insertuserdata(nameTXT, contactTXT, dobTXT,
emailTXT);
                if(checkinsertdata==true)
                    Toast.makeText(MainActivity.this, "New Entry Inserted",
Toast.LENGTH_SHORT).show();
                else
                    Toast.makeText(MainActivity.this, "New Entry Not Inserted",
Toast.LENGTH_SHORT).show();
            }        });
        update.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String nameTXT = name.getText().toString();
                String contactTXT = contact.getText().toString();
                String dobTXT = dob.getText().toString();
                String emailTXT = email.getText().toString();

                Boolean checkupdatedata = DB.updateuserdata(nameTXT, contactTXT, dobTXT,
emailTXT);
                if(checkupdatedata==true)
                    Toast.makeText(MainActivity.this, "Entry Updated",
Toast.LENGTH_SHORT).show();
                else
                    Toast.makeText(MainActivity.this, "New Entry Not Updated",
Toast.LENGTH_SHORT).show();
            }        });
        delete.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String nameTXT = name.getText().toString();
                Boolean checkudeletedata = DB.deletedata(nameTXT);
                if(checkudeletedata==true)
                    Toast.makeText(MainActivity.this, "Entry Deleted",
Toast.LENGTH_SHORT).show();
                else
                    Toast.makeText(MainActivity.this, "Entry Not Deleted",
Toast.LENGTH_SHORT).show();
            }        });

        view.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
```

```
                Cursor res = DB.getdata();
                if(res.getCount()==0){
                    Toast.makeText(MainActivity.this, "No Entry Exists",
Toast.LENGTH_SHORT).show();
                    return;
                }
                StringBuffer buffer = new StringBuffer();
                while(res.moveToNext()){
                    buffer.append("Name :"+res.getString(0)+"\n");
                    buffer.append("Contact :"+res.getString(1)+"\n");
                    buffer.append("Address :"+res.getString(2)+"\n");
                    buffer.append("Email :"+res.getString(3)+"\n\n");
                }

                AlertDialog.Builder builder = new AlertDialog.Builder(MainActivity.this);
                builder.setCancelable(true);
                builder.setTitle("User Entries");
                builder.setMessage(buffer.toString());
                builder.show();
            }        });
    }}
```

DBHELPER

```
public class DBHelper extends SQLiteOpenHelper {
    public DBHelper(Context context) {
        super(context, "Userdata.db", null, 1);
    }

    @Override
    public void onCreate(SQLiteDatabase DB) {
        DB.execSQL("create Table Userdetails(name TEXT primary key, contact TEXT, dob TEXT,
email TEXT)");
    }

    @Override
    public void onUpgrade(SQLiteDatabase DB, int i, int i1) {
        DB.execSQL("drop Table if exists Userdetails");
    }

    public Boolean insertuserdata(String name, String contact, String dob, String email)
    {
        SQLiteDatabase DB = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put("name", name);
        contentValues.put("contact", contact);
        contentValues.put("dob", dob);
        contentValues.put("email", email);

        long result=DB.insert("Userdetails", null, contentValues);
        if(result==-1){
            return false;
        }else{
            return true;
        }
    }


    public Boolean updateuserdata(String name, String contact, String dob, String email) {
        SQLiteDatabase DB = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put("contact", contact);
        contentValues.put("dob", dob);
        contentValues.put("email", email);

        Cursor cursor = DB.rawQuery("Select * from Userdetails where name = ?", new String[]
{name});
        if (cursor.getCount() > 0) {
            long result = DB.update("Userdetails", contentValues, "name=?", new String[]
```

```
{name});
            if (result == -1) {
                return false;
            } else {
                return true;
            }
        } else {
            return false;
        }}


    public Boolean deletedata (String name)
    {
        SQLiteDatabase DB = this.getWritableDatabase();
        Cursor cursor = DB.rawQuery("Select * from Userdetails where name = ?", new String[]
{name});
        if (cursor.getCount() > 0) {
            long result = DB.delete("Userdetails", "name=?", new String[]{name});
            if (result == -1) {
                return false;
            } else {
                return true;
            }
        } else {
            return false;
        }

    }

    public Cursor getdata ()
    {
        SQLiteDatabase DB = this.getWritableDatabase();
        Cursor cursor = DB.rawQuery("Select * from Userdetails", null);
        return cursor;

    }
}
```

PROGRESS BAR

```
public class MainActivity extends AppCompatActivity {
    private ProgressBar pb;
    private TextView tv;
    private Button btn;
    Handler h;
    int counter=0;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        pb = (ProgressBar)findViewById(R.id.pb);
        tv = findViewById(R.id.tv);
        btn=findViewById(R.id.btn);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Thread t=new Thread(new Runnable() {
                    @Override
                    public void run() {
                        check();
                    }
                });
                t.start();
            }
        });

    }
    public void check()
```

```
        {
            pb.setProgressTintList(ColorStateList.valueOf(Color.GRAY));
            for(int c=0;c<101;c++)
            {
                pb.setProgress(c);
                if(c==24)
                {
                    pb.setProgressTintList(ColorStateList.valueOf(Color.YELLOW));
                }
                if(c==49)
                {
                    pb.setProgressTintList(ColorStateList.valueOf(Color.RED));
                }
                if(c==74)
                {
                    pb.setProgressTintList(ColorStateList.valueOf(Color.WHITE));
                }
                int finalC = c;
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
                        tv.setText(String.valueOf(finalC)+"%");
                    }
                });
                try {
                    Thread.sleep(250);
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
            }
        }
    }
```

MULTITHREADING SONG PLAYING

```
public class MainActivity extends AppCompatActivity {
    private Button button;
    private EditText time;
    private TextView finalResult;
    MediaPlayer player;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        time = (EditText) findViewById(R.id.in_time);
        button = (Button) findViewById(R.id.btn_run);
        finalResult = (TextView) findViewById(R.id.tv_result);
        if (player == null) {
            player = MediaPlayer.create(this, R.raw.song);
            player.setOnCompletionListener(new MediaPlayer.OnCompletionListener() {
                @Override
                public void onCompletion(MediaPlayer mp) {
                    player.release();
                }
            });
        }

        player.start();
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                AsyncTaskRunner runner = new AsyncTaskRunner();
                String sleepTime = time.getText().toString();
                runner.execute(sleepTime);
            }
        });
    }

    private class AsyncTaskRunner extends AsyncTask<String, String, String> {
```

```
        private String resp;
        ProgressDialog progressDialog;

        @Override
        protected String doInBackground(String... params) {
            publishProgress("Sleeping..."); // Calls onProgressUpdate()
            try {
                int time = Integer.parseInt(params[0])*1000;

                Thread.sleep(time);
                resp = "Slept for " + params[0] + " seconds";
            } catch (InterruptedException e) {
                e.printStackTrace();
                resp = e.getMessage();
            } catch (Exception e) {
                e.printStackTrace();
                resp = e.getMessage();
            }
            return resp;
        }


        @Override
        protected void onPostExecute(String result) {
            // execution of result of Long time consuming operation
            progressDialog.dismiss();
            finalResult.setText(result);
            player.start();
        }


        @Override
        protected void onPreExecute() {
            player.pause();
            progressDialog = ProgressDialog.show(MainActivity.this,
                    "ProgressDialog",
                    "Wait for "+time.getText().toString()+ " seconds");
        }


        @Override
        protected void onProgressUpdate(String... text) {
            finalResult.setText(text[0]);

        }
    }
}
```

LOCATION TRACKING

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"

<uses-permission android:name="android.permission.INTERNET" />

public class MainActivity extends AppCompatActivity implements LocationListener {
    Button btn;
    LocationManager lmanager;
    String provider;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        btn = findViewById(R.id.button);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                lmanager=(LocationManager)getSystemService(Context.LOCATION_SERVICE);
                Criteria criteria=new Criteria();
```

```
                provider=lmanager.getBestProvider(criteria,false);
                if(Build.VERSION.SDK_INT>=23 && ContextCompat.checkSelfPermiss
ion(MainActivity.this, Manifest.permission.ACCESS_FINE_LOCATION)!= PackageManager.PERMISSION_
GRANTED && ContextCompat.checkSelfPermission(MainActivity.this, Manifest.permission.ACCESS_
COARSE_LOCATION)!= PackageManager.PERMISSION_GRANTED)
                {
                    Toast.makeText(MainActivity.this, "Error", Toast.LENGTH_SHORT).show();
                    return;
                }
                if(provider!=null && !provider.equals(""))
                {
                    Location location=lmanager.getLastKnownLocation(provider);
                    lmanager.requestLocationUpdates(provider,20000,1,MainActivity.this);
                    if(location!=null)
                    {
                        onLocationChanged(location);
                    }
                    else
                    {
                        Toast.makeText(MainActivity.this, "Location not Available",
Toast.LENGTH_SHORT).show();
                    }
                }
                else
                {
                    Toast.makeText(MainActivity.this, "No Provider Found",
Toast.LENGTH_SHORT).show();
                }
            }
        });
    }

    @Override
    public void onLocationChanged(@NonNull Location location) {

        TextView lat=(TextView)findViewById(R.id.lat);
        TextView lon=(TextView)findViewById(R.id.lon);
        lat.setText("Latitude : "+location.getLatitude());
        lon.setText("Longitude : "+location.getLongitude());
    }


    @Override
    public void onStatusChanged(String provider, int status, Bundle extras) {

    }

    @Override
    public void onProviderEnabled(@NonNull String provider) {

    }

    @Override
    public void onProviderDisabled(@NonNull String provider) {

    }
}
```

LOCATION OF A PLACE

```
public class MainActivity extends AppCompatActivity {
    double latitude,longitude;
    EditText locationName;
    Button okbtn;
    TextView lat;
    TextView lon;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        locationName=(EditText)findViewById(R.id.location);
        okbtn=(Button)findViewById(R.id.button);
        lat=(TextView)findViewById(R.id.lat);
        lon=(TextView)findViewById(R.id.lon);
        okbtn.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View v) {
                // TODO Auto-generated method stub
                Geocoder geocoder = new Geocoder(getBaseContext(), Locale.getDefault());
                try
                {
                    List<Address> address;

                    address = geocoder.getFromLocationName(locationName.getText().
toString(),1);

                    if(address.size() >= 0)
                    {

                            latitude= address.get(0).getLatitude();
                            longitude= address.get(0).getLongitude();
                            lat.setText("Latitude : "+latitude);
                            lon.setText("Longitude : "+longitude);
                       }

                } catch (IOException e)
                {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
                finally
                {

                }
            }
        });
    }

}
```

SDCARD

```
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.WRITE_INTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.READ_INTERNAL_STORAGE"/>

public class MainActivity extends AppCompatActivity {
    EditText fname,cont,rfname;
    Button btn1,btn2;
    String fn,text_con;
    TextView tv;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        cont=findViewById(R.id.cont);
        fname=findViewById(R.id.fname);
        btn1=findViewById(R.id.btn1);
        btn2=findViewById(R.id.btn2);
        rfname=findViewById(R.id.rfname);
        tv=findViewById(R.id.tv);
        btn1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                fn = fname.getText().toString();
```

```
                text_con = cont.getText().toString();
                try {
                    File f=new File("/sdcard/"+fn);
                    f.createNewFile();
                    FileOutputStream fos;
                    fos = new FileOutputStream(f);
                    OutputStreamWriter mout= new OutputStreamWriter(fos);
                    mout.append(text_con);
                    mout.close();
                    fos.close();
                    Toast.makeText(MainActivity.this, "Data written to SD
Card",Toast.LENGTH_LONG).show();
                } catch (Exception e) {
                    Toast.makeText(MainActivity.this,e.getMessage(),Toast.LENGTH_LONG).show();
                }
            }
        });
        btn2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                File f=new File("/sdcard/"+rfname.getText().toString());
                try {
                    FileInputStream fin = new FileInputStream(f);
                    BufferedReader bf=new BufferedReader(new InputStreamReader(fin));
                    String drow="";
                    String dbuf="";
                    while((drow=bf.readLine())!=null)
                    {
                        dbuf+=drow+"\n";

                    }
                    tv.setText(dbuf);
                    bf.close();
                    fin.close();
                } catch (Exception e) {
                    e.printStackTrace();
                }

            }
        });
    }
}
```

SEND SMS

```
<uses-permission android:name="android.permission.SEND_SMS"/>
<uses-permission android:name="android.permission.RECEIVE_SMS"/>

public class MainActivity extends AppCompatActivity {
    private static final int MY_PERMISSIONS_REQUEST_SEND_SMS = 0;
    Button sendBtn;
    EditText txtphoneNo;
    EditText txtMessage;
    String phoneNo;
    String message;


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

        sendBtn = (Button) findViewById(R.id.sendbtn);
        txtphoneNo = (EditText) findViewById(R.id.etPhone);
        txtMessage = (EditText) findViewById(R.id.content);

        sendBtn.setOnClickListener(new View.OnClickListener() {
            public void onClick(View view) {
                if (ActivityCompat.checkSelfPermission(MainActivity.this,
Manifest.permission.SEND_SMS) == PackageManager.PERMISSION_GRANTED) {
```

```
                    try {
                        SmsManager smsManager = SmsManager.getDefault();
                        smsManager.sendTextMessage(txtphoneNo.getText().toString(), null,
txtMessage.getText().toString(), null, null);
                        Toast.makeText(MainActivity.this, "SMS Sent!",
Toast.LENGTH_LONG).show();
                    } catch (Exception e) {
                        Toast.makeText(MainActivity.this, "Error!", Toast.LENGTH_LONG).show();
                    }
                } else {
                    ActivityCompat.requestPermissions(MainActivity.this, new String[]
{Manifest.permission.SEND_SMS}, 0);
                }
            }
        });
    }

}
```

HYBRID APP

```
public class MainActivity extends AppCompatActivity implements View.OnClickListener{
    WebView simpleWebView;
    Button loadWebPage;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        loadWebPage = (Button) findViewById(R.id.loadWebPage);
        loadWebPage.setOnClickListener(this);
        simpleWebView = (WebView) findViewById(R.id.simpleWebView);
    }
    @Override
    public void onClick(View v) {
        String customHtml = "<html><body><h1>Welcome</h1>"+"<h3>This is a webpage made of
HTML</h3>"+"</body></html>";
        simpleWebView.loadData(customHtml, "text/html", "UTF-8");
    }
    private class MyWebViewClient extends WebViewClient
    {
        @Override
        public boolean shouldOverrideUrlLoading(WebView view, String url) {
            view.loadUrl(url);
            return true;
        }
    }
}
```

KEYBOARD SAMPLE

```
public class MainActivity extends AppCompatActivity {
    TextView tv;
    Button btn;
    LinearLayout allc,alls;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        tv=findViewById(R.id.tv);
        btn=findViewById(R.id.t);
        allc=findViewById(R.id.allcaps);
        alls=findViewById(R.id.allsmall);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if (allc.getVisibility()==View.VISIBLE) {
                    alls.setVisibility(View.VISIBLE);
                    allc.setVisibility(View.INVISIBLE);
                }
                else
```

```
                {
                    alls.setVisibility(View.INVISIBLE);
                    allc.setVisibility(View.VISIBLE);
                }

            }
        });
    }

    public void func(View v)
    {
        tv.setText(tv.getText().toString()+""+v.getTag().toString());
    }
}
```

XML

```
<LinearLayout 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:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/tv"
        android:text="Hello"
        tools:ignore="InvalidId" />
    <RelativeLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">
        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:gravity="center"
            android:id="@+id/allcaps">
            <LinearLayout
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:orientation="horizontal">

                <Button
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:textAllCaps="true"
                    android:onClick="func"
                    android:tag="A"
                    android:text="A" />

                <Button
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:textAllCaps="true"
                    android:onClick="func"
                    android:tag="B"
                    android:text="B"/>

                <Button
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:textAllCaps="true"
                    android:onClick="func"
                    android:tag="C"
                    android:text="C"/>
```

```
        </LinearLayout>

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

            <Button
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:textAllCaps="true"
                android:onClick="func"
                android:tag="D"
                android:text="D"/>

            <Button
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:textAllCaps="true"
                android:onClick="func"
                android:tag="E"
                android:text="E" />

            <Button
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:textAllCaps="true"
                android:onClick="func"
                android:tag="F"
                android:text="F" />
        </LinearLayout>

    </LinearLayout>



    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:gravity="center"
        android:visibility="invisible"
        android:id="@+id/allsmall">
        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="horizontal">

            <Button
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:textAllCaps="false"
                android:onClick="func"
                android:tag="a"
                android:text="a" />

            <Button
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:textAllCaps="false"
                android:onClick="func"
                android:tag="b"
                android:text="b"/>

            <Button
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:textAllCaps="false"
                android:onClick="func"
```

```
                android:tag="c"
                android:text="c"/>

        </LinearLayout>

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

            <Button
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:textAllCaps="false"
                android:onClick="func"
                android:tag="d"
                android:text="d"/>

            <Button
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:textAllCaps="false"
                android:onClick="func"
                android:tag="e"
                android:text="e" />

            <Button
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:textAllCaps="false"
                android:onClick="func"
                android:tag="f"
                android:text="f" />
        </LinearLayout>

    </LinearLayout>
</RelativeLayout>

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="toggle"
    android:id="@+id/t"
    />
</LinearLayout>
```