

Mobile Application Development Practical

1. Design an application representing a simple calculator.

XML Code:

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="wrap_content"
    tools:context="ty.practical1.MainActivity">

    <TextView
        android:id="@+id/txtDisplay"
        android:layout_width="match_parent"
        android:layout_height="90dp"
        android:maxLength="15"
        android:paddingLeft="10sp"
        android:textAppearance="@android:style/TextAppearance.DeviceDefault.Large"
        android:textSize="40sp" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:layout_below="@+id/txtDisplay"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true">

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

            <Button
                android:id="@+id/btnSeven"
                android:layout_width="90dp"
                android:layout_height="90dp"
                android:text="7"
                android:textSize="30dp" />

            <Button
                android:id="@+id/btnEight"
                android:layout_width="90dp"
                android:layout_height="90dp"
                android:text="8"
                android:textSize="30dp" />

            <Button
                android:id="@+id/btnNine"
                android:layout_width="90dp"
                android:layout_height="90dp"
                android:text="9"
                android:textSize="30dp" />

            <Button
                android:id="@+id/btnDivide"
                android:layout_width="90dp"
                android:layout_height="90dp"
                android:text="/"
                android:textSize="30dp" />

        </LinearLayout>

    </LinearLayout>

</RelativeLayout>
```

```

        android:layout_width="90dp"
        android:layout_height="90dp"
        android:text="/"
        android:textSize="30dp" />

</LinearLayout>

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

    <Button
        android:id="@+id/btnFour"
        android:layout_width="90dp"
        android:layout_height="90dp"
        android:text="4"
        android:textSize="30dp" />

    <Button
        android:id="@+id/btnFive"
        android:layout_width="90dp"
        android:layout_height="90dp"
        android:text="5"
        android:textSize="30dp" />

    <Button
        android:id="@+id/btnSix"
        android:layout_width="90dp"
        android:layout_height="90dp"
        android:text="6"
        android:textSize="30dp" />

    <Button
        android:id="@+id/btnMultiply"
        android:layout_width="90dp"
        android:layout_height="90dp"
        android:text="*"
        android:textSize="30dp" />

</LinearLayout>

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

    <Button
        android:id="@+id/btnOne"
        android:layout_width="90dp"
        android:layout_height="90dp"
        android:text="1"
        android:textSize="30dp" />

    <Button
        android:id="@+id/btnTwo"
        android:layout_width="90dp"
        android:layout_height="90dp"
        android:text="2"
        android:textSize="30dp" />

    <Button
        android:id="@+id/btnThree"
        android:layout_width="90dp"
        android:layout_height="90dp"
        android:text="3"
        android:textSize="30dp" />

```

```

        <Button
            android:id="@+id/btnSub"
            android:layout_width="90dp"
            android:layout_height="90dp"
            android:text="-"
            android:textSize="30dp"/>

    </LinearLayout>

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

        <Button
            android:id="@+id/btnZero"
            android:layout_width="90dp"
            android:layout_height="90dp"
            android:text="0"
            android:textSize="30dp"/>

        <Button
            android:id="@+id/btnDot"
            android:layout_width="90dp"
            android:layout_height="90dp"
            android:text="."
            android:textSize="30dp"/>

        <Button
            android:id="@+id/btnEqual"
            android:layout_width="90dp"
            android:layout_height="90dp"
            android:text="="
            android:textSize="30dp"/>

        <Button
            android:id="@+id/btnAdd"
            android:layout_width="90dp"
            android:layout_height="90dp"
            android:text="+"
            android:textSize="30dp"/>

    </LinearLayout>

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

        <Button
            android:id="@+id/btnClear"
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:text="C"
            android:textSize="20sp"/>

    </LinearLayout>

</LinearLayout>

</RelativeLayout>

```

Source Code:

MainActivity.java

```
package ty.practical1;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import java.text.DecimalFormat;

public class MainActivity extends AppCompatActivity {

    private double num1, num2, answer;
    private char op;
    private boolean hasDot; //Variable to know whether Dot(.) is pressed.

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

        final Button btnOne = (Button) findViewById(R.id.btnOne);
        final Button btnTwo = (Button) findViewById(R.id.btnTwo);
        final Button btnThree = (Button) findViewById(R.id.btnThree);
        final Button btnFour = (Button) findViewById(R.id.btnFour);
        final Button btnFive = (Button) findViewById(R.id.btnFive);
        final Button btnSix = (Button) findViewById(R.id.btnSix);
        final Button btnSeven = (Button) findViewById(R.id.btnSeven);
        final Button btnEight = (Button) findViewById(R.id.btnEight);
        final Button btnNine = (Button) findViewById(R.id.btnNine);
        final Button btnZero = (Button) findViewById(R.id.btnZero);
        final Button btnAdd = (Button) findViewById(R.id.btnAdd);
        final Button btnSub = (Button) findViewById(R.id.btnSub);
        final Button btnMultiply = (Button) findViewById(R.id.btnMultiply);
        final Button btnDivide = (Button) findViewById(R.id.btnDivide);
        final Button btnDot = (Button) findViewById(R.id.btnDot);
        final Button btnEqual = (Button) findViewById(R.id.btnEqual);
        final Button btnClear = (Button) findViewById(R.id.btnClear);
        final TextView txtDisplay = (TextView) findViewById(R.id.txtDisplay);

        btnOne.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                txtDisplay.append("1");
            }
        });

        btnTwo.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                txtDisplay.append("2");
            }
        });

        btnThree.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                txtDisplay.append("3");
            }
        });

        btnFour.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                txtDisplay.append("4");
            }
        });
    }
}
```

```

});

btnFive.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v){
        txtDisplay.append("5");
    }
});

btnSix.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v){
        txtDisplay.append("6");
    }
});

btnSeven.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v){
        txtDisplay.append("7");
    }
});

btnEight.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v){
        txtDisplay.append("8");
    }
});

btnNine.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v){
        txtDisplay.append("9");
    }
});

btnZero.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v){
        txtDisplay.append("0");
    }
});

btnDot.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v){
        //if Dot(.) is pressed then set hasDot to true to restrict
        if(hasDot==false) {
            txtDisplay.append(".");
            hasDot = true;
        }
    }
});

btnAdd.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v){
        num1 = Double.parseDouble(txtDisplay.getText().toString());
        op = '+';
        txtDisplay.setText("");
        hasDot = false; //set hasDot to false to use in the next operand.
    }
});

btnSub.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v){
        num1 = Double.parseDouble(txtDisplay.getText().toString());
        op = '-';
        txtDisplay.setText("");
        hasDot = false; //set hasDot to false to use in the next operand.
    }
});

btnMultiply.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v){

```

```

        num1 = Double.parseDouble(txtDisplay.getText().toString());
        op = '*';
        txtDisplay.setText("");
        hasDot = false; //set hasDot to false to use in the next operand.
    }
});

btnDivide.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v){
        num1 = Double.parseDouble(txtDisplay.getText().toString());
        op = '/';
        txtDisplay.setText("");
        hasDot = false; //set hasDot to false to use in the next operand.
    }
});

btnEqual.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v){

        num2 = Double.parseDouble(txtDisplay.getText().toString());

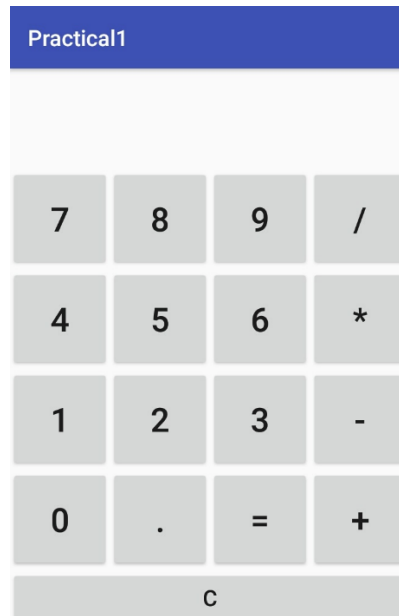
        switch (op)
        {
            case '+':
                answer = num1 + num2;
                break;
            case '-':
                answer = num1 - num2;
                break;
            case '*':
                answer = num1 * num2;
                break;
            case '/':
                answer = num1 / num2;
                break;
            default:
                break;
        }

        DecimalFormat d = new DecimalFormat();
        String ans = d.format(answer);
        txtDisplay.setText(ans);
        hasDot = false; //set hasDot to false to use in new calculation.
    }
});

btnClear.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v){
        txtDisplay.setText("");
        hasDot = false; //set hasDot to false to use in new calculation.
    }
});
}
}

```

Output:



2. Develop an application for working with Menus and Screen Navigation.

XML Code:

[Create a menu - Android Resource Directory and create a main_menu.xml – Android Resource File in it.]

main_menu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:app="http://schemas.android.com/apk/res-auto"
      xmlns:android="http://schemas.android.com/apk/res/android">

    <item
        android:id="@+id/item1"
        android:title="FYCS" />

    <item
        android:id="@+id/item2"
        android:title="SYCS" />

    <item
        android:id="@+id/item3"
        android:title="TYCS" />
</menu>
```

Source Code:

[Using the MenuInflater link the main_menu.xml file in the MainActivity file]

MainActivity.java

```
package ty.practical2;

import android.content.Intent;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;

import ty.practical5.R;

public class MainActivity extends AppCompatActivity {

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

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        //return super.onCreateOptionsMenu(menu);

        MenuInflater menuInflater = getMenuInflater();
        menuInflater.inflate(R.menu.main_menu, menu);

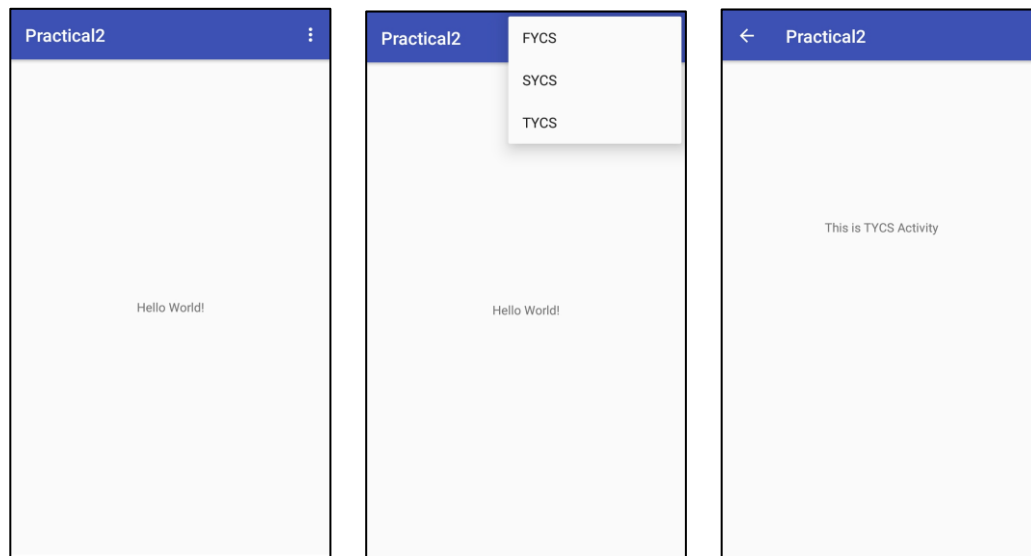
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {

        switch(item.getItemId())
        {
            case R.id.item1:
                startActivity(new Intent(MainActivity.this, FYCS.class));
                return true;
            case R.id.item2:
                startActivity(new Intent(MainActivity.this, SYCS.class));
                return true;
            case R.id.item3:
                startActivity(new Intent(MainActivity.this, TYCS.class));
                return true;
            default:
                return super.onOptionsItemSelected(item);
        }
    }
}
```


[Create 3 new activities to open when the menu items in the options menu is clicked/selected named as FYCS, SYCS, TYCS]

Output:



3. Develop an application for working with Notifications.

XML Code:

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="ty.practical3.MainActivity">

    <Button
        android:id="@+id/btnCreate"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="199dp"
        android:onClick="CreateNotification"
        android:text="Create Notification" />
</RelativeLayout>
```

activity_dummy.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="ty.practical3.Dummy">
```

```

<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="207dp"
    android:text="This is Dummy Activity" />
</RelativeLayout>

```

Source Code:

MainActivity.java

```

package ty.practical3;

import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Intent;
import android.os.Bundle;
import android.support.v4.app.NotificationCompat;
import android.support.v7.app.AppCompatActivity;
import android.view.View;

public class MainActivity extends AppCompatActivity {

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

    int notifyID = 1;

    int numMessages = 0;

    public void CreateNotification(View v) {

        numMessages++;

        NotificationManager notificationManager = (NotificationManager)
        getSystemService(NOTIFICATION_SERVICE);

        Intent intent = new Intent(this, Dummy.class);

        // use System.currentTimeMillis() to have a unique ID for the pending
        intent

        PendingIntent pIntent = PendingIntent.getActivity(this, (int)
        System.currentTimeMillis(), intent, 0);

        NotificationCompat.Builder n = new NotificationCompat.Builder(this)
            .setContentTitle("Hello")
            .setContentText("Hello World Notification")
            .setContentIntent(pIntent)
            .setSmallIcon(R.mipmap.ic_launcher)
            .setNumber(numMessages)
            .setAutoCancel(true);

        notificationManager.notify(notifyID, n.build());
    }
}

```

DummyActivity.java

```
package ty.practical3;

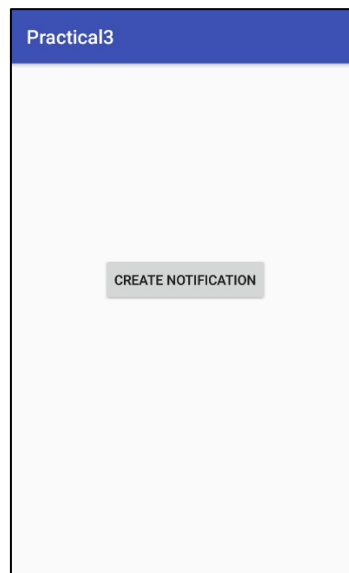
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;

public class Dummy extends AppCompatActivity {

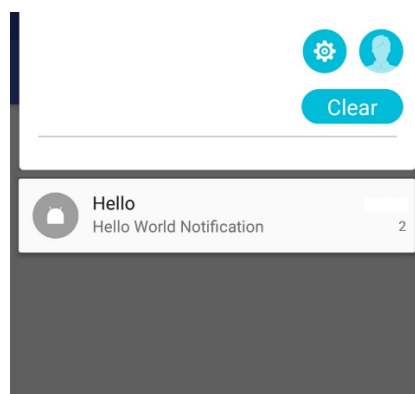
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_dummy);
    }
}
```

Output:

This practical displays a notification bar similar a message on the device.



When the user clicks the Create Notification Button a notification appears as below with a count as 1 on clicking the button again the notification count increases.



When the notification on the notification bar is clicked a dummy activity screen is to be displayed.



4. Develop an application demonstrating Internal Storage to store private data on the device memory.

XML Code:

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="ty.practical4.MainActivity">

    <EditText
        android:id="@+id/txtData"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="55dp"
        android:ems="10"
        android:inputType="textMultiLine"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true" />

    <Button
        android:id="@+id/txtWrite"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignBaseline="@+id/btnRead"
        android:layout_alignBottom="@+id/btnRead"
        android:layout_alignParentStart="true"
        android:layout_marginStart="34dp"
        android:onClick="writeData"
        android:text="Write" />

    <Button
        android:id="@+id/btnRead"
```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_below="@+id/txtData"
        android:layout_marginEnd="38dp"
        android:layout_marginTop="86dp"
        android:onClick="readData"
        android:text="Read" />
</RelativeLayout>

```

Source Code:

MainActivity.java

```

package ty.practical4;

import android.content.Context;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;

public class MainActivity extends AppCompatActivity {

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

    public void writeData(View v){
        try {
            String FILENAME = "demo.txt";
            String data = "";

            EditText txtData = (EditText) findViewById(R.id.txtData);
            String string = txtData.getText().toString();
            FileOutputStream fos = openFileOutput(FILENAME,
Context.MODE_PRIVATE);
            fos.write(data.getBytes());
            fos.close();
            txtData.setText("");
            Toast.makeText(MainActivity.this, "File
Created", Toast.LENGTH_LONG).show();

        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    public void readData(View v){
        try {
            String FILENAME = "demo.txt";
            FileInputStream fin = openFileInput(FILENAME);
            int c;
            String temp="";
            while( (c = fin.read()) != -1){
                temp = temp + Character.toString((char)c);
            }

```

```

        fin.close();
        //string temp contains all the data of the file.
        Toast.makeText(MainActivity.this,temp,Toast.LENGTH_LONG).show();

    } catch (IOException e) {
        e.printStackTrace();
    }
}
}

```

Output:

Here, the demo.txt file created using File I/O objects has a private mode. Thus, demo.txt is not accessible directly from other apps not even File Explorer it can be accessed within the same app.

To verify that the file is successfully created follow the below steps to see the file in adb shell window. [The mobile device or emulator where the app is connected should be running].

1. Open Android terminal or windows command prompt (Run -> cmd)

Run command:

adb shell

2. To obtain permission to file system:

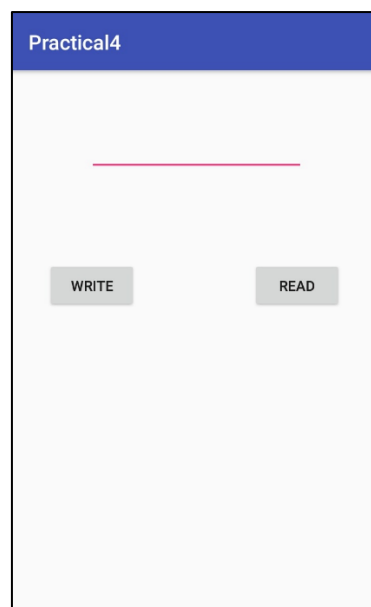
run-as ty.practical4

3. Change to internal storage:

cd files

4. View your file:

cat demo.txt



5. Design a simple to-do list application using SQLite

XML Code:

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="ty.practical5.MainActivity">

    <ListView
        android:id="@+id/lvData"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_above="@+id/btnAdd"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_alignParentTop="true" />

    <EditText
        android:id="@+id/txtItem"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_alignTop="@+id/btnAdd"
        android:layout_toLeftOf="@+id/btnAdd"
        android:layout_toStartOf="@+id/btnAdd"
        android:hint="Enter a New Item"
        android:inputType="textMultiLine"
        />

    <Button
        android:id="@+id/btnAdd"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_alignParentEnd="true"
        android:layout_alignParentRight="true"
        android:onClick="AddItem"
        android:text="Add Item" />

</RelativeLayout>
```

activity_task_details.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="ty.practical5.TaskDetails">

    <Button
        android:id="@+id/btnUpdate"
```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_below="@+id/txtData"
        android:layout_marginStart="48dp"
        android:layout_marginTop="50dp"
        android:onClick="Update"
        android:text="Update" />

<Button
    android:id="@+id/btnDelete"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignEnd="@+id/txtData"
    android:layout_alignTop="@+id/btnUpdate"
    android:layout_marginEnd="13dp"
    android:onClick="Delete"
    android:text="Delete" />

<EditText
    android:id="@+id/txtData"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_alignStart="@+id/btnUpdate"
    android:layout_marginStart="19dp"
    android:layout_marginTop="57dp"
    android:ems="10"
    android:inputType="textMultiLine" />

</RelativeLayout>

```

Source Code:

ToDoDatabaseHelper.java

```

package ty.practical5;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import java.util.ArrayList;

public class ToDoDatabaseHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME = "todoList.db";
    private static final int DATABASE_VERSION = 1;

    public ToDoDatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String query = "CREATE TABLE ToDo (task TEXT)";
        db.execSQL(query);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS ToDo");
        onCreate(db);
    }
}

```



```

public void addTask(String item){
    ContentValues values = new ContentValues();
    values.put("task", item);
    SQLiteDatabase db = getWritableDatabase();
    db.insert("ToDo", null, values);
    db.close();
}

//Delete a product from the database
public void deleteTask(String item){
    SQLiteDatabase db = this.getWritableDatabase();
    db.execSQL("DELETE FROM ToDo where task='"+item+"'");
}

public void updateTask(String oldvalue, String newvalue){

    try {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put("task", newvalue);
        db.update("ToDo", contentValues, "task='"+oldvalue + "'", null);
        db.close();
    } catch (Exception e) {
        e.printStackTrace();
    }
}

public ArrayList<String> getAllTasks() {

    ArrayList<String> contactList = new ArrayList<String>();

    String selectQuery = "SELECT * FROM ToDo";

    SQLiteDatabase db = this.getWritableDatabase();
    Cursor cursor = db.rawQuery(selectQuery, null);

    if (cursor.moveToFirst()) {
        do {
            contactList.add(cursor.getString(0));
        } while (cursor.moveToNext());
    }

    return contactList;
}

public ArrayList<String> getTaskByItem(int item) {

    ArrayList<String> contactList = new ArrayList<String>();

    String selectQuery = "SELECT * FROM ToDo where task=" + item;

    SQLiteDatabase db = this.getWritableDatabase();
    Cursor cursor = db.rawQuery(selectQuery, null);

    if (cursor.moveToFirst()) {
        contactList.add(cursor.getString(1));
    }

    return contactList;
}
}

```

MainActivity.java

```
package ty.practical5;

import android.content.Intent;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.EditText;
import android.widget.ListView;

import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {

    private ArrayList<String> items;
    private ArrayAdapter<String> itemsAdapter;
    private ListView lvData;
    private ToDoDatabaseHelper dbAccess;

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

        dbAccess = new ToDoDatabaseHelper(this);
        lvData = (ListView) findViewById(R.id.lvData);
        items = new ArrayList<String>();
        readItems();

        itemsAdapter = new ArrayAdapter<String>(this,
        android.R.layout.simple_list_item_1, items);
        lvData.setAdapter(itemsAdapter);

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

                Intent intent = new Intent(MainActivity.this, TaskDetails.class);
                intent.putExtra("data",
                lvData.getItemAtPosition(position).toString());
                startActivity(intent);

                // Refresh the adapter
                refreshListView();
            }
        });

        public void AddItem(View v) {
            EditText txtItem = (EditText) findViewById(R.id.txtItem);
            String itemText = txtItem.getText().toString();
            itemsAdapter.add(itemText);
            txtItem.setText("");
            dbAccess.addTask(itemText);
        }

        public void readItems() {
            try {
                items = new ArrayList<String>(dbAccess.getAllTasks());
            } catch (Exception e) {
                items = new ArrayList<String>();
            }
        }
    }
}
```

```

        public void refreshListView() {

            itemsAdapter.notifyDataSetChanged();
        }
    }
}

```

TaskDetails.java

```

package ty.practical5;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.MenuItem;
import android.view.View;
import android.widget.EditText;

public class TaskDetails extends AppCompatActivity {

    private toDoDatabaseHelper dbAccess;
    String oldValue="";
    EditText txtData;

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

        //code to enable the back button
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        //code to fetch the selected list item data in the previous activity
        dbAccess = new toDoDatabaseHelper(this);

        Intent intent = getIntent();
        oldValue = intent.getStringExtra("data");

        txtData = (EditText) findViewById(R.id.txtData);
        txtData.setText(intent.getStringExtra("data"));
    }

    //code for delete button to delete the task
    protected void Delete(View v) {
        dbAccess.deleteTask(txtData.getText().toString());
        Intent intent = new Intent(TaskDetails.this, MainActivity.class) ;
        intent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
        startActivity(intent);
        finish();
    }

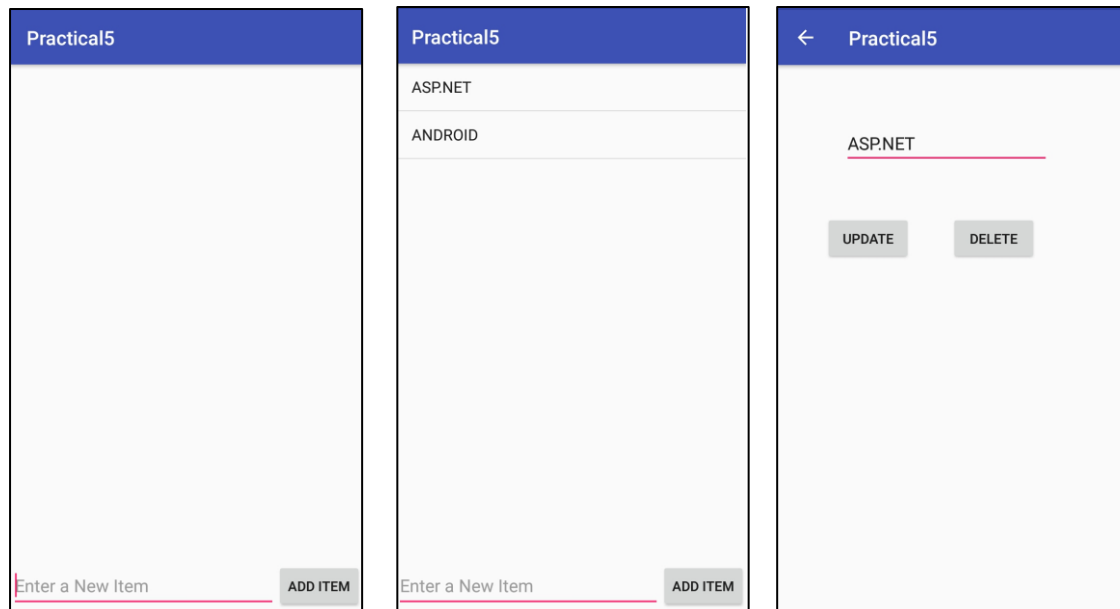
    //code for delete button to update the task
    protected void Update(View v) {
        dbAccess.updateTask(oldValue, txtData.getText().toString());
        Intent intent = new Intent(TaskDetails.this, MainActivity.class) ;
        intent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
        startActivity(intent);
        finish();
    }

    //code to close the current activity and move to the previous
    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        finish();
        return super.onOptionsItemSelected(item);
    }
}

```

```
}
}
```

Output:



6. Develop an application for connecting to the internet and sending email.

XML Code:

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="ty.practical6.MainActivity">

    <Button
        android:id="@+id/btnSendEmail"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="92dp"
        android:onClick="sendEmail"
        android:text="Compose Email"
        android:layout_alignTop="@+id/txtMessage"
        android:layout_centerHorizontal="true" />

    <EditText
        android:id="@+id/txtMessage"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Message"
        android:inputType="textMultiLine"
        android:singleLine="true"
        android:layout_marginTop="48dp"
        android:layout_below="@+id/txtSubject"

```

```

        android:layout_centerHorizontal="true" />

<EditText
    android:id="@+id/txtEmailTo"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="To"
    android:inputType="textEmailAddress"
    android:layout_marginTop="22dp"
    android:layout_alignParentTop="true"
    android:layout_alignStart="@+id/txtSubject" />

<EditText
    android:id="@+id/txtSubject"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignStart="@+id/txtMessage"
    android:layout_below="@+id/txtEmailTo"
    android:layout_marginTop="43dp"
    android:ems="10"
    android:hint="Subject"
    android:inputType="text" />

</RelativeLayout>

```

Source Code:

MainActivity.java

```

package ty.practical6;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

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

    public void sendEmail(View v) {

        EditText txtEmailTo = (EditText) findViewById(R.id.txtEmailTo);
        EditText txtSubject = (EditText) findViewById(R.id.txtSubject);
        EditText txtMessage = (EditText) findViewById(R.id.txtMessage);

        String[] TO = {txtEmailTo.getText().toString()};
        String[] CC = {""};
        String subject = txtSubject.getText().toString();
        String msg = txtMessage.getText().toString();

        Intent emailIntent = new Intent(Intent.ACTION_SEND);

        emailIntent.setData(Uri.parse("mailto:"));
        emailIntent.setType("text/plain");
        emailIntent.putExtra(Intent.EXTRA_EMAIL, TO);
    }
}

```

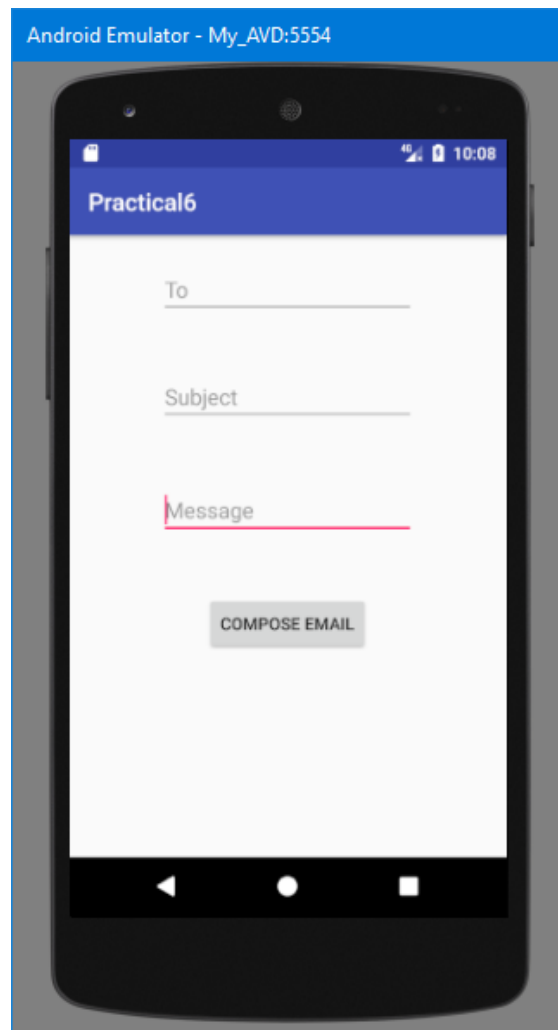
```

emailIntent.putExtra(Intent.EXTRA_CC, CC);
emailIntent.putExtra(Intent.EXTRA_SUBJECT, subject);
emailIntent.putExtra(Intent.EXTRA_TEXT, msg);

try {
    startActivity(Intent.createChooser(emailIntent, "Send mail..."));
    finish();
} catch (android.content.ActivityNotFoundException ex) {
    Toast.makeText(MainActivity.this, "No email client app installed.",
        Toast.LENGTH_SHORT).show();
}
}
}

```

Output:



On clicking COMPOSE EMAIL button a list of apps will be displayed select a relevant email client app e.g. Gmail and the contents given as input here will be passed to Gmail app's email compose screen.

[Well, email sent here is done using an intent but this can be even done using an external email API service this method of sending email is not necessary to be included as a part of practical but can be explained in theory.]

7. Develop an application for working with graphics and animation.

XML Code:

[Here, along with the layout xml file 4 other xml files are to be created which define the Shapes and Animation Transition Effect Data]

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="ty.practical7.MainActivity"
    android:id="@+id/mainLayout">

    <ImageView
        android:id="@+id/imgShape1"
        android:layout_width="150dp"
        android:layout_height="150dp"
        app:srcCompat="@drawable/shape1"
        android:layout_alignParentTop="true"
        android:layout_alignParentEnd="true" />

    <ImageView
        android:id="@+id/imgShape2"
        android:layout_width="200dp"
        android:layout_height="200dp"
        android:layout_alignEnd="@+id/imgShape1"
        android:layout_alignParentBottom="true" />

    <Button
        android:id="@+id/btnAnimation"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="75dp"
        android:layout_marginEnd="32dp"
        android:onClick="Animation"
        android:text="Animation"
        android:layout_alignParentBottom="true"
        android:layout_toStartOf="@+id/imgShape2" />

</RelativeLayout>
```

shape1.xml

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="oval" >
    <solid android:color="#3F51B5" />
    <size android:width="300dp" android:height="300dp"></size>
</shape>
```

shape2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="rectangle" >
    <solid android:color="#303F9F" />
    <size android:width="300dp" android:height="300dp"></size>
</shape>
```

shape3.xml

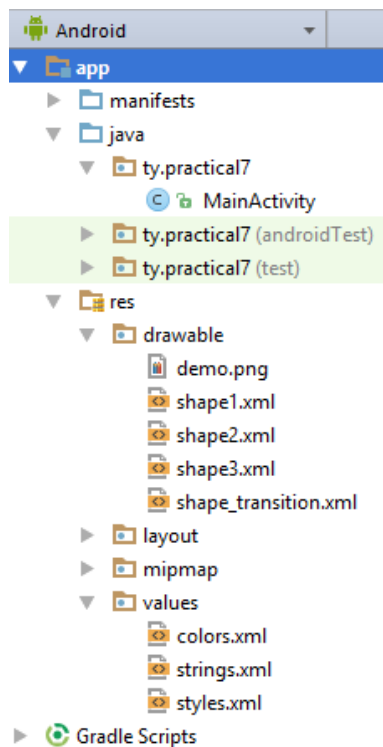
```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="oval" >
    <solid android:color="#FF4081" />
    <size android:width="300dp" android:height="300dp"></size>
</shape>
```

shape_transition.xml

```
<?xml version="1.0" encoding="utf-8"?>
<transition xmlns:android="http://schemas.android.com/apk/res/android" >
    <item android:drawable="@drawable/shape1" android:right="100dp"/>
    <item android:drawable="@drawable/shape3" android:left="100dp"/>
</transition>
```

Image: Add a drawable object an Image in the res/drawable folder, here for example an image file name demo.png is used.

Project Structure:



Source Code:

```
package ty.practical7;

import android.graphics.Color;
import android.graphics.drawable.ShapeDrawable;
import android.graphics.drawable.TransitionDrawable;
import android.graphics.drawable.shapes.RectShape;
import android.os.Bundle;
import android.support.v4.content.res.ResourcesCompat;
import android.support.v7.app.AppCompatActivity;
```



```

import android.view.View;
import android.widget.ImageView;
import android.widget.RelativeLayout;

public class MainActivity extends AppCompatActivity {

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

        DrawableGraphic();
        ShapeDrawableGraphic();
    }

    public void DrawableGraphic(){
        //Get Parent Layout
        RelativeLayout rl = (RelativeLayout)findViewById(R.id.mainLayout);

        //Create a dynamic ImageView object to place it within the Relative Layout
        ImageView demoimg = new ImageView(MainActivity.this);

        //Get the Image Source from @drawable, here demo.png
        demoimg.setImageResource(R.drawable.demo);

        //Specify the placement of ImageView within the RelativeLayout
        RelativeLayout.LayoutParams params = new RelativeLayout.LayoutParams(300,
300);

        //Add rule to align the image to the left of the parent.
        params.addRule(RelativeLayout.ALIGN_PARENT_LEFT);
        demoimg.setLayoutParams(params);

        //Add ImageView within the Relative Layout
        rl.addView(demoimg);
    }

    public void ShapeDrawableGraphic()
    {
        //Assign Shape Properties
        int alpha = 127;
        int width = 300;
        int height = 300;
        int padding = 10;

        //Get Parent Layout
        RelativeLayout rl = (RelativeLayout)findViewById(R.id.mainLayout);

        // Create Shape 2
        ShapeDrawable shape2 = new ShapeDrawable(new RectShape());
        shape2.getPaint().setColor(Color.CYAN);
        shape2.setIntrinsicHeight(height);
        shape2.setIntrinsicWidth(width);
        shape2.setAlpha(alpha);

        // Put Shape 2 into an ImageView
        ImageView shape2View = new ImageView(getApplicationContext());
        shape2View.setImageDrawable(shape2);
        shape2View.setPadding(padding, padding, padding, padding);

        //Specify the placement of ImageView within the RelativeLayout
        RelativeLayout.LayoutParams s2params = new
RelativeLayout.LayoutParams(height, width);

        //Add rule to align the image to the left of the parent.
        s2params.addRule(RelativeLayout.CENTER_IN_PARENT);
        shape2View.setLayoutParams(s2params );
    }
}

```

```

        //Add ImageView within the Relative Layout
        rl.addView(shape2View);
    }

    public void Animation(View v){
        //Get the Shape Transition Drawable Objects
        TransitionDrawable transition = (TransitionDrawable)
        ResourcesCompat.getDrawable(getResources(), R.drawable.shape_transition, null);

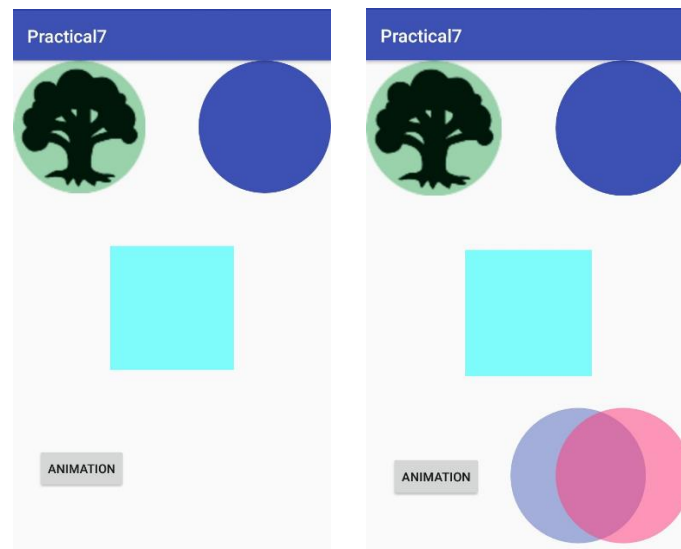
        //Apply Effect
        transition.setCrossFadeEnabled(true);

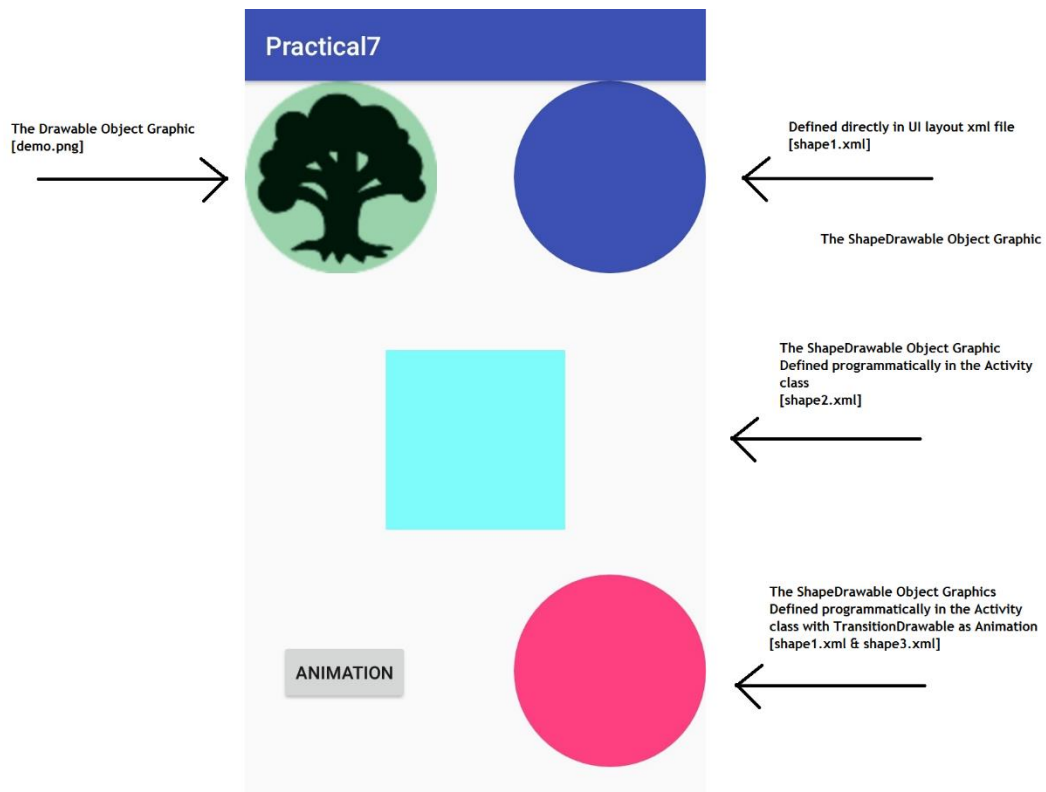
        //Assign the effect to an ImageView object
        ((ImageView) findViewById (R.id.imgShape2)).setImageDrawable(transition);

        //Set the Transition Effect Time
        transition.startTransition(5000);
    }
}

```

Output:





8. Develop an application for working with device camera.

XML Code:

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="ty.practical8.MainActivity">

    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:onClick="takePhotos"
        android:text="Take a Photo"></Button>

    <ImageView
        android:id="@+id/imageView1"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:layout_alignParentTop="true"
        android:layout_alignParentStart="true">
    </ImageView>

</RelativeLayout>
```

Source Code:

MainActivity.java

```
package ty.practical8;

import android.content.Intent;
import android.graphics.Bitmap;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {

    private static final int REQUEST_CODE = 1;
    ImageView imageView;

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

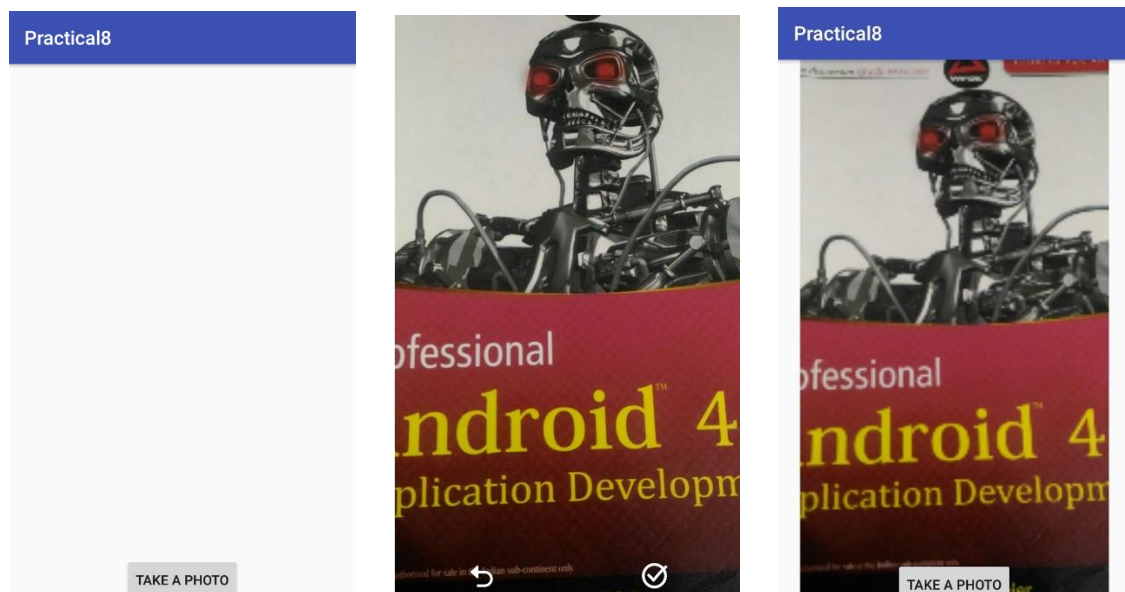
        imageView = (ImageView) this.findViewById(R.id.imageView1);
    }

    public void takePhotos(View v){
        Intent cameraIntent = new
        Intent(android.provider.MediaStore.ACTION_IMAGE_CAPTURE);
        startActivityForResult(cameraIntent, REQUEST_CODE);
    }

    @Override
    protected void onActivityResult(int requestCode, int resultCode, Intent data) {
        super.onActivityResult(requestCode, resultCode, data);

        if (requestCode == REQUEST_CODE && resultCode == RESULT_OK && data != null)
        {
            Bitmap photo = (Bitmap) data.getExtras().get("data");
            imageView.setImageBitmap(photo);
        }
    }
}
```

Output:



9. Develop an application for working with location based services.

XML Code:

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="ty.practical9.MainActivity">

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Get Current Location"
        android:id="@+id/btnLocation"/>

    <TextView
        android:id="@+id/txtLocation"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/btnLocation" />

</RelativeLayout>
```

Source Code:

MainActivity.java

```
package ty.practical9;

import android.content.Context;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
```

```

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity implements LocationListener {

    Button btnLocation;
    TextView txtLocation;

    LocationManager locationManager;

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

        btnLocation = (Button)findViewById(R.id.btnLocation);
        txtLocation = (TextView)findViewById(R.id.txtLocation);

        btnLocation.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                getLocation();
            }
        });

        void getLocation() {
            try {
                locationManager = (LocationManager)
getSystemService(Context.LOCATION_SERVICE);

locationManager.requestLocationUpdates(LocationManager.NETWORK_PROVIDER, 5000, 0,
this);
            }
            catch (SecurityException e) {
                e.printStackTrace();
            }
        }

        @Override
        public void onLocationChanged(Location location) {
            txtLocation.setText("Current Location: " + location.getLatitude() + ", " +
location.getLongitude());
            Log.d("data", + location.getLatitude() + ", " + location.getLongitude());
        }

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

        }

        @Override
        public void onProviderEnabled(String provider) {

        }

        @Override
        public void onProviderDisabled(String provider) {
            Toast.makeText(MainActivity.this, "Please Enable GPS and Internet",
Toast.LENGTH_SHORT).show();
        }
    }
}

```

Permission:

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

Output:



10. Using Worker thread write Android code for a click listener that downloads an image from a separate thread and displays it in an ImageView.

XML Code:

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"
    android:orientation="vertical"
    tools:context="ty.practical10.MainActivity"
    android:weightSum="1">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_alignParentTop="true"/>

    <Button
        android:id="@+id/btnDownload"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
```

```

        android:onClick="DownloadImage"
        android:text="Download Image" />

</RelativeLayout>

```

Source Code:

MainActivity.java

```

package ty.practical10;

import android.graphics.drawable.Drawable;
import android.os.AsyncTask;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.util.Log;
import android.view.View;
import android.widget.ImageView;

import java.io.IOException;
import java.net.URL;

import static ty.practical10.R.id.imageView;

public class MainActivity extends AppCompatActivity {

    ImageView imageView;
    Drawable d;
    //Image URL to be downloaded from the internet
    String IMAGE_URL = "http://via.placeholder.com/500x500";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        imageView = (ImageView) findViewById(imageView);
    }

    public void DownloadImage(View v)
    {
        new DownloadImageTask().execute(IMAGE_URL) ;
    }

    private class DownloadImageTask extends AsyncTask<String, Void, Drawable> {
        /** The system calls this to perform work in a worker thread and
         * delivers it the parameters given to AsyncTask.execute() */
        protected Drawable doInBackground(String... urls) {
            return loadImageFromNetwork(urls[0]);
        }

        /** The system calls this to perform work in the UI thread and delivers
         * the result from doInBackground() */
        protected void onPostExecute(Drawable result) {
            imageView.setImageDrawable(result);
        }
    }

    //Download Image From Network
    private Drawable loadImageFromNetwork(String imageUrl)
    {
        Drawable drawable = null;
        try {
            drawable = Drawable.createFromStream(new
            URL(imageUrl).openStream(), "image");

```



```
    } catch (IOException e) {  
        Log.d("Error", e.getMessage());  
    }  
    return drawable ;  
}  
}
```

Permission:

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

Output:

