Load image from camera or gallery

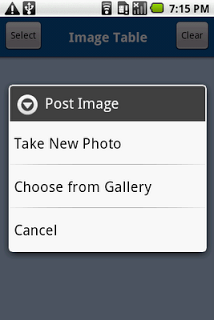
This post is to demonstrate how to load image from camera or from gallery.  
  
It contain select and clear button at the title bar of the activity, and a text view will contain the title of the acticity.  
  
On clicking select button, a context menu will be opened with three options called "Take New Photo", "Choose from Gallery" and "Cancel".  
  
On Selecting Take New Photo option, the android camera intent will be called and camera view will be opened, the user can take photo, on clicking the capture button in the camera, the image will be stored in the folder of the sdcard, and the taken image will be loaded into the activity.  
  
For accessing the camera hardware of the mobile, the following lines must to be add in the manifest file.  
  
   **<uses-permission android:name="android.permission.CAMERA" />  
      
    <uses-feature android:name="android.hardware.camera" />  
    <uses-feature android:name="android.hardware.camera.autofocus" />**  
  
For storing the image in the sdcard, you have to add the following permission in the manifest file.  
  
**<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE"/>**  
  
On Selecting Choose from Gallery option, the android gallery page will be opened, from the gallery you can select image, and the selected image will be added next to the main activity.  
  
Below is the code for loading the image into the table layout with dynamic table row. You can take new picture and load in the table layout or you can select image from gallery and load in the table layout.

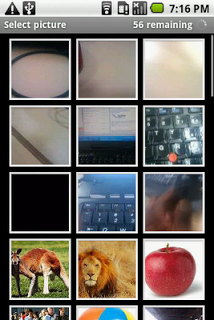
**package cm.camera.table;  
  
import java.io.File;  
import java.util.ArrayList;  
import java.util.Calendar;  
  
import android.app.Activity;  
import android.app.ProgressDialog;  
import android.content.Context;  
import android.content.Intent;  
import android.database.Cursor;  
import android.graphics.Bitmap;  
import android.graphics.BitmapFactory;  
import android.graphics.drawable.BitmapDrawable;  
import android.graphics.drawable.Drawable;  
import android.net.Uri;  
import android.os.AsyncTask;  
import android.os.Bundle;  
import android.os.Environment;  
import android.provider.MediaStore;  
import android.util.Log;  
import android.view.ContextMenu;  
import android.view.Gravity;  
import android.view.MenuInflater;  
import android.view.MenuItem;  
import android.view.View;  
import android.view.Window;  
import android.view.ContextMenu.ContextMenuInfo;  
import android.view.View.OnClickListener;  
import android.view.ViewGroup.LayoutParams;  
import android.widget.Button;  
import android.widget.ImageView;  
import android.widget.TableLayout;  
import android.widget.TableRow;  
import android.widget.TextView;  
  
public class LoadImage extends Activity   
{  
    Activity activity=null;  
    Context context=null;  
      
    Button header\_left\_btn=null;  
    Button header\_right\_btn=null;  
    TextView header\_text=null;  
    TableLayout image\_table=null;  
      
    ArrayList<String> image\_list=new ArrayList<String>();  
    ArrayList<Drawable> image\_drawable=new ArrayList<Drawable>();  
    String path="";  
      
    /\*\* Called when the activity is first created. \*/  
    @Override  
    public void onCreate(Bundle savedInstanceState)   
    {  
        super.onCreate(savedInstanceState);  
        requestWindowFeature(Window.FEATURE\_CUSTOM\_TITLE);  
        setContentView(R.layout.main);  
        getWindow().setFeatureInt(Window.FEATURE\_CUSTOM\_TITLE,R.layout.header);  
          
        activity=LoadImage.this;  
        context=LoadImage.this;  
          
        header\_left\_btn=(Button)findViewById(R.id.header\_left\_btn);  
        header\_right\_btn=(Button)findViewById(R.id.header\_right\_btn);  
        header\_text=(TextView)findViewById(R.id.header\_text);  
        image\_table=(TableLayout)findViewById(R.id.image\_table);  
          
        header\_text.setText("Image Table");  
        header\_left\_btn.setText("Select");  
        header\_right\_btn.setText("Clear");  
        registerForContextMenu(header\_left\_btn);  
          
        header\_left\_btn.setOnClickListener(new OnClickListener(){  
  
            @Override  
            public void onClick(View v)   
            {  
                // TODO Auto-generated method stub  
                openContextMenu(header\_left\_btn);  
            }  
        });  
          
        header\_right\_btn.setOnClickListener(new OnClickListener(){  
  
            @Override  
            public void onClick(View v)   
            {  
                // TODO Auto-generated method stub  
                image\_list.clear();  
                image\_drawable.clear();  
                deletePhotos();  
                updateImageTable();  
            }  
        });  
    }  
      
    public void deletePhotos()  
    {  
        String folder=Environment.getExternalStorageDirectory() +"/LoadImg";  
        File f=new File(folder);  
        if(f.isDirectory())  
        {  
            File[] files=f.listFiles();  
            Log.v("Load Image", "Total Files To Delete=====>>>>>"+files.length);  
            for(int i=0;i<files.length;i++)  
            {  
                String fpath=folder+File.separator+files[i].getName().toString().trim();  
                System.out.println("File Full Path======>>>"+fpath);  
                File nf=new File(fpath);  
                if(nf.exists())  
                {  
                    nf.delete();  
                }  
            }  
        }  
    }  
      
    @Override  
    public void onCreateContextMenu(ContextMenu menu, View v, ContextMenuInfo menuInfo)  
    {  
          
        super.onCreateContextMenu(menu, v, menuInfo);  
        menu.setHeaderTitle("Post Image");  
        MenuInflater inflater = getMenuInflater();  
        inflater.inflate(R.menu.camer\_menu, menu);  
    }  
      
    @Override  
    public boolean onContextItemSelected(MenuItem item)  
    {  
      switch (item.getItemId())  
      {  
          case R.id.take\_photo:  
              //Toast.makeText(context, "Selected Take Photo", Toast.LENGTH\_SHORT).show();  
              takePhoto();  
              break;  
            
          case R.id.choose\_gallery:  
              //Toast.makeText(context, "Selected Gallery", Toast.LENGTH\_SHORT).show();  
              Intent photoPickerIntent = new Intent(Intent.ACTION\_PICK);  
              photoPickerIntent.setType("image/\*");  
              startActivityForResult(photoPickerIntent, 1);  
                
              break;  
          
          case R.id.share\_cancel:  
              closeContextMenu();  
              break;  
          default:  
            return super.onContextItemSelected(item);  
      }  
      return true;  
    }  
      
    public void takePhoto()   
    {  
         Intent intent = new Intent("android.media.action.IMAGE\_CAPTURE");  
         File folder = new File(Environment.getExternalStorageDirectory() + "/LoadImg");  
  
         if(!folder.exists())  
         {  
             folder.mkdir();  
         }           
         final Calendar c = Calendar.getInstance();  
         String new\_Date= c.get(Calendar.DAY\_OF\_MONTH)+"-"+((c.get(Calendar.MONTH))+1)   +"-"+c.get(Calendar.YEAR) +" " + c.get(Calendar.HOUR) + "-" + c.get(Calendar.MINUTE)+ "-"+ c.get(Calendar.SECOND);  
         path=String.format(Environment.getExternalStorageDirectory() +"/LoadImg/%s.png","LoadImg("+new\_Date+")");  
         File photo = new File(path);   
         intent.putExtra(MediaStore.EXTRA\_OUTPUT,Uri.fromFile(photo));  
         startActivityForResult(intent, 2);  
    }  
      
    @Override  
    public void onActivityResult(int requestCode, int resultCode, Intent data)   
    {  
        super.onActivityResult(requestCode, resultCode, data);  
          
        if(requestCode==1)  
        {  
            Uri photoUri = data.getData();  
            if (photoUri != null)  
            {  
                String[] filePathColumn = {MediaStore.Images.Media.DATA};  
                Cursor cursor = getContentResolver().query(photoUri, filePathColumn, null, null, null);  
                cursor.moveToFirst();  
                int columnIndex = cursor.getColumnIndex(filePathColumn[0]);  
                String filePath = cursor.getString(columnIndex);  
                cursor.close();  
                Log.v("Load Image", "Gallery File Path=====>>>"+filePath);  
                image\_list.add(filePath);  
                Log.v("Load Image", "Image List Size=====>>>"+image\_list.size());  
                  
                //updateImageTable();  
                new GetImages().execute();  
            }  
        }  
          
        if(requestCode==2)  
        {  
            Log.v("Load Image", "Camera File Path=====>>>"+path);  
            image\_list.add(path);  
             Log.v("Load Image", "Image List Size=====>>>"+image\_list.size());  
            //updateImageTable();  
             new GetImages().execute();  
        }  
    }  
      
    public void updateImageTable()  
    {  
        image\_table.removeAllViews();  
          
        if(image\_drawable.size() > 0)  
        {  
            for(int i=0; i<image\_drawable.size(); i++)  
            {  
                TableRow tableRow=new TableRow(this);  
                tableRow.setLayoutParams(new LayoutParams(LayoutParams.FILL\_PARENT, LayoutParams.WRAP\_CONTENT));  
                tableRow.setGravity(Gravity.CENTER\_HORIZONTAL);  
                tableRow.setPadding(5, 5, 5, 5);  
                for(int j=0; j<1; j++)  
                {  
                    ImageView image=new ImageView(this);  
                    image.setLayoutParams(new LayoutParams(LayoutParams.FILL\_PARENT, LayoutParams.WRAP\_CONTENT));  
                      
                    /\*Bitmap bitmap = BitmapFactory.decodeFile(image\_list.get(i).toString().trim());  
                    bitmap = Bitmap.createScaledBitmap(bitmap,500, 500, true);  
                    Drawable d=loadImagefromurl(bitmap);\*/  
                    image.setBackgroundDrawable(image\_drawable.get(i));  
                      
                    tableRow.addView(image, 200, 200);  
                }  
                image\_table.addView(tableRow);  
            }  
        }  
    }  
      
    public Drawable loadImagefromurl(Bitmap icon)  
    {  
        Drawable d=new BitmapDrawable(icon);          
        return d;  
    }  
      
    public class GetImages extends AsyncTask<Void, Void, Void>   
    {  
        public ProgressDialog progDialog=null;  
          
        protected void onPreExecute()   
        {  
            progDialog=ProgressDialog.show(context, "", "Loading...",true);  
        }  
        @Override  
        protected Void doInBackground(Void... params)   
        {  
            image\_drawable.clear();  
            for(int i=0; i<image\_list.size(); i++)  
            {  
                Bitmap bitmap = BitmapFactory.decodeFile(image\_list.get(i).toString().trim());  
                bitmap = Bitmap.createScaledBitmap(bitmap,500, 500, true);  
                Drawable d=loadImagefromurl(bitmap);  
                  
                image\_drawable.add(d);  
            }  
            return null;  
        }      
              
        protected void onPostExecute(Void result)   
        {  
            if(progDialog.isShowing())  
            {  
                progDialog.dismiss();  
            }  
            updateImageTable();  
        }  
    }  
}**

Below is the layout, that are used for loading the image. The name of the layout called main.xml.

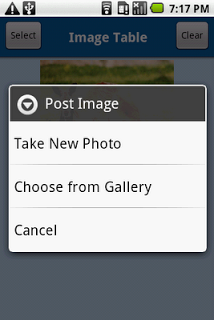
**<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    android:orientation="vertical"  
    android:layout\_width="fill\_parent"  
    android:layout\_height="fill\_parent"  
    android:background="#BFD6E8">  
      
    <ScrollView android:id="@+id/image\_scroll"  
        android:layout\_width="fill\_parent"  
        android:layout\_height="wrap\_content">  
          
        <TableLayout android:id="@+id/image\_table"  
            android:layout\_width="fill\_parent"  
            android:layout\_height="wrap\_content">  
        </TableLayout>  
          
    </ScrollView>  
  
</LinearLayout>**  
  
I added the Custom title bar in the activity. For creating custom title bar, please Refer the link below.  
  
<http://tjkannan.blogspot.com/2012/01/custom-title-bar.html>  
  
Below is the sample output screens.

[](http://3.bp.blogspot.com/-RimX5ITMNYI/TwxNWCVeMaI/AAAAAAAAAEY/-14NiKj-Bz4/s1600/image_table_01.png)

[](http://2.bp.blogspot.com/-b5382Xv6PCA/TwxNdBugQvI/AAAAAAAAAEg/1UP4LK_a1EQ/s1600/image_table_02.png)

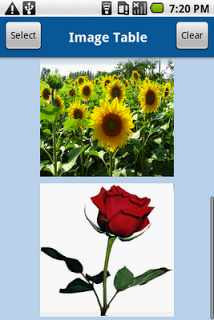
[](http://3.bp.blogspot.com/-UXRp09wVjnM/TwxNkPwxc_I/AAAAAAAAAEo/uYYsxBYqWTc/s1600/image_table_03.png)

[](http://4.bp.blogspot.com/-QHhfdqjGTKw/TwxNvqw-lLI/AAAAAAAAAEw/R_mK8myFozc/s1600/image_table_04.png)

[](http://4.bp.blogspot.com/-l9ki4Yuxe2g/TwxN6VSlYiI/AAAAAAAAAE4/t6ApDzDV4UU/s1600/image_table_05.png)

[](http://1.bp.blogspot.com/-IuXJvgxohnE/TwxOCpwJ_1I/AAAAAAAAAFA/0kMO-FIFAgM/s1600/image_table_06.png)

[](http://2.bp.blogspot.com/-Fbh5kit_GLo/TwxOQ5Kd5nI/AAAAAAAAAFI/4ILXT-aKwy0/s1600/image_table_07.png)

[](http://1.bp.blogspot.com/-OBP6IX8saN8/TwxOd-2ZXAI/AAAAAAAAAFQ/k7NERVqrW2c/s1600/image_table_08.png)

[](http://1.bp.blogspot.com/-DjhWEWjpqb0/TwxOm3oN8OI/AAAAAAAAAFY/Da9t5klq4MY/s1600/image_table_09.png)

Posted by [Kannan - Android Blogs](https://www.blogger.com/profile/11900544453567020381" \o "author profile)at [06:44](http://tjkannan.blogspot.ru/2012/01/load-image-from-camera-or-gallery.html)