7. Develop an app to capture a photo and store it into SDCard, and extend this app to display all the images captured in the grid view.
a) Utilize the Camera functionality.
b) Implement writing data to the SD card.

7a)
activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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:gravity="center_horizontal"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <ImageView
        android:id="@+id/imgCamera"
        android:layout_width="400dp"
        android:layout_height="240dp"
        android:scaleType="fitXY" />
    <Button
        android:id="@+id/btnCamera"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="21dp"
        android:text="Open Camera"/>
</LinearLayout>
```

MainActivity.java

```
package com.example.program7;

import androidx.activity.result.ActivityResult;
import androidx.activity.result.ActivityResultCallback;
import androidx.activity.result.ActivityResultLauncher;
import androidx.activity.result.contract.ActivityResultContracts;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.graphics.Bitmap;
import android.os.Bundle;
import android.provider.MediaStore;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {
    private final int CAMERA_REQ_CODE = 100;
    ImageView imgCamera;
    ActivityResultLauncher<Intent> activityResultLauncher;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);

        imgCamera = findViewById(R.id.imgCamera);
        Button btnCamera = findViewById(R.id.btnCamera);

        activityResultLauncher = registerForActivityResult(new
ActivityResultContracts.StartActivityForResult(), new
ActivityResultCallback<ActivityResult>() {
                    @Override
                    public void onActivityResult(ActivityResult result) {
                        if (result.getResultCode() == RESULT_OK) {
                            if (result.getResultCode() == CAMERA_REQ_CODE)
{
                                //for camera
                                Bitmap img = (Bitmap)
(result.getData().getExtras().get("data"));
                                imgCamera.setImageBitmap(img);

                            }
                        }
                    }


        });
        btnCamera.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent iCamera = new
Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                activityResultLauncher.launch(iCamera);
              //  startActivityForResult(iCamera, CAMERA_REQ_CODE);
            }
        });
    }

}
```

7b)
activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    tools:ignore="ExtraText">

    <Button
        android:id="@+id/buttonSelectedImage"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="select_image"/>
    <ImageView
        android:id="@+id/selectedImage"
        android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        android:adjustViewBounds="true"
        android:contentDescription="@string/app_name" />
    android:adjustViewBounds="true"
    android:contentDescription="@string/app_name" />

</LinearLayout>
```

MainActivity.java

```
package com.example.program7a;

import androidx.activity.result.ActivityResultLauncher;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;
import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.net.Uri;
import android.os.Bundle;
import android.provider.MediaStore;
import android.view.View;
import android.widget.ImageView;
import android.widget.Toast;
import java.io.InputStream;

public class MainActivity extends AppCompatActivity
{
    private static final int REQUEST_CODE_STORAGE_PERMISSION = 1;
    private static final int REQUEST_CODE_SELECT_IMAGE = 2;

    private ImageView imageSelected;
    ActivityResultLauncher<Intent> activityResultLauncher;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        imageSelected = findViewById(R.id.selectedImage);

        findViewById(R.id.buttonSelectedImage).setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if (ContextCompat.checkSelfPermission(
                        getApplicationContext(),
Manifest.permission.READ_EXTERNAL_STORAGE
                ) != PackageManager.PERMISSION_GRANTED) {
                    ActivityCompat.requestPermissions(
                            MainActivity.this,
                            new
String[]{Manifest.permission.READ_EXTERNAL_STORAGE},
                            REQUEST_CODE_STORAGE_PERMISSION);
                } else {
                    selectImage();
```

```
                }
            }
        });

    }


    private void selectImage()
    {
        Intent intent = new

Intent(Intent.ACTION_PICK,MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
        if(intent.resolveActivity(getPackageManager()) != null){
            startActivityForResult(intent,REQUEST_CODE_SELECT_IMAGE);
        }
    }
    @Override
    public void onRequestPermissionsResult(int requestCode,@Nullable
String[]
            permissions,@Nullable int[] grantResults)
    {

super.onRequestPermissionsResult(requestCode,permissions,grantResults);

        if(requestCode == REQUEST_CODE_STORAGE_PERMISSION &&
grantResults.length > 0)
        {
            if(grantResults[0] == PackageManager.PERMISSION_GRANTED)
            {
                selectImage();
            }
            else
            {
                Toast.makeText(this, "Permission Denied",
Toast.LENGTH_SHORT).show();
            }
        }
    }

    @Override
    protected void onActivityResult(int requestCode, int resultCode,
@Nullable Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
        if(requestCode == REQUEST_CODE_SELECT_IMAGE && resultCode ==
RESULT_OK){
            if(data != null){
                Uri selectedImageUri = data.getData();
                if(selectedImageUri != null){
                    try{

                        InputStream inputStream =
getContentResolver().openInputStream(selectedImageUri);
                        Bitmap bitmap =
BitmapFactory.decodeStream(inputStream);
                        imageSelected.setImageBitmap(bitmap);
                    }catch (Exception exception){
                        Toast.makeText(this,exception.getMessage(),
Toast.LENGTH_SHORT).show();
                    }
                }
```

```
                }
            }
        }
}
```

AndroidManifest.xml

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