Creating A Camera App In Xamarin Android App Using Visual Studio 2015

In this article, you will learn, how to create the Camera app in Xamarin Android app, using Visual Studio 2015.

Delpin Susai Raj Oct 27 2016

6 5 33.1k

Download Free .NET & JAVA Files API

Introduction

Xamarin is a platform to develop cross-platform and multi-platform apps (for example, Windows phone, Android, iOS). In the Xamarin platform, the code sharing concept is used. In Xamarin Studio, Visual Studio is also available. Camera app is used to open the camera and take snaps, using the app.

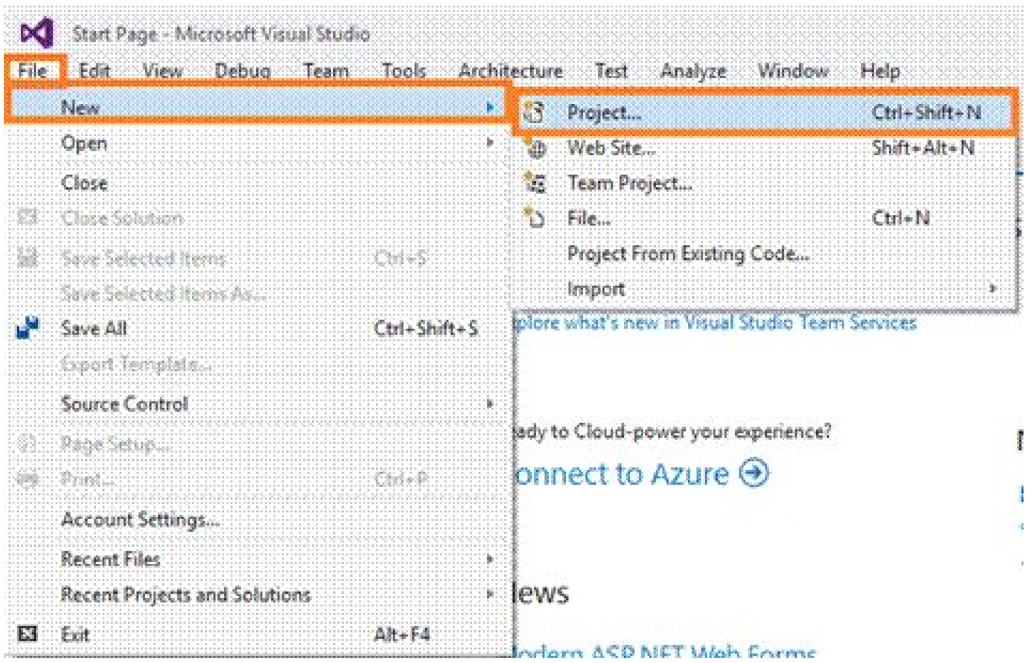
Prerequisites

- Visual Studio 2015 Update 3.

The steps, mentioned below are required to be followed in order to create a Camera App in Xamarin Android app, using Visual Studio 2015.

Step 1

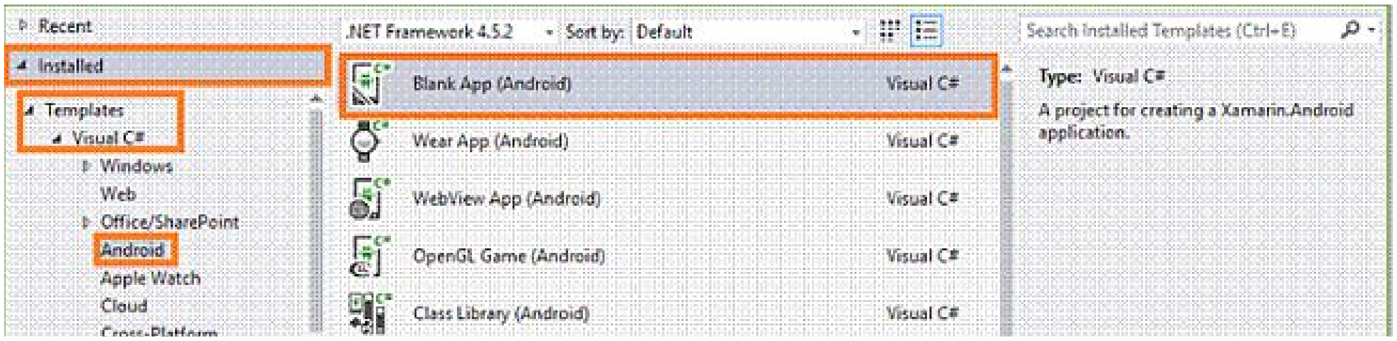
Click File--> select New--> select Project. The project needs to be clicked after opening all the types of projects in Visual Studio or click ing (Ctrl+Shift+N).



Step 2

After opening the New Project, select Installed-->Templates-->Visual C#-->Android-->choose the Blank app (Android).

Now, give your Android app; a name (Ex:sample) and give the path of your project. Afterwards, click OK.


Recent
Installed
Templates
Visual C#
Windows
Web
Office/SharePoint
Android
Apple Watch
Cloud
Cross-Platform
.NET Framework 4.5.2
Sort by: Default
Search Installed Templates (Ctrl+E)
Blank App (Android) Visual C#
Wear App (Android) Visual C#
WebView App (Android) Visual C#
OpenGL Game (Android) Visual C#
Class Library (Android) Visual C#

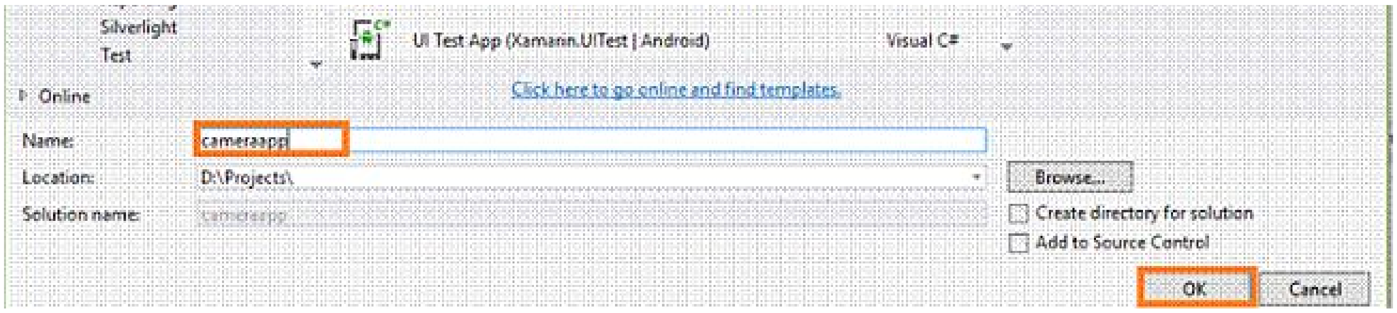
Type: Visual C#

A project for creating a Xamarin.Android application.

cus

C# Corner

orner Announces Python MVP Program


Silverlight
Test
UI Test App (Xamarin.UITest | Android) Visual C#
Online
Click here to go online and find templates.
Browse...
Create directory for solution
Add to Source Control
OK
Cancel

Name: cameraapp

Location: D:\Projects\

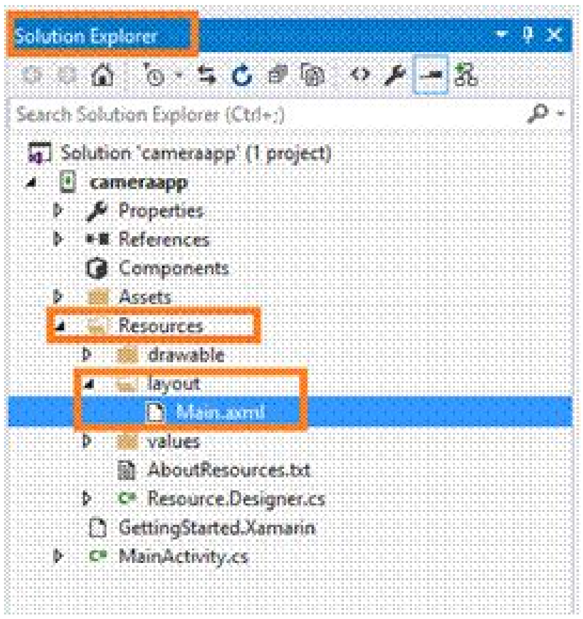
Solution name: cameraapp

Step 3

Now, go to the Solution Explorer. In Solution Explorer, get all the files and sources in your project.

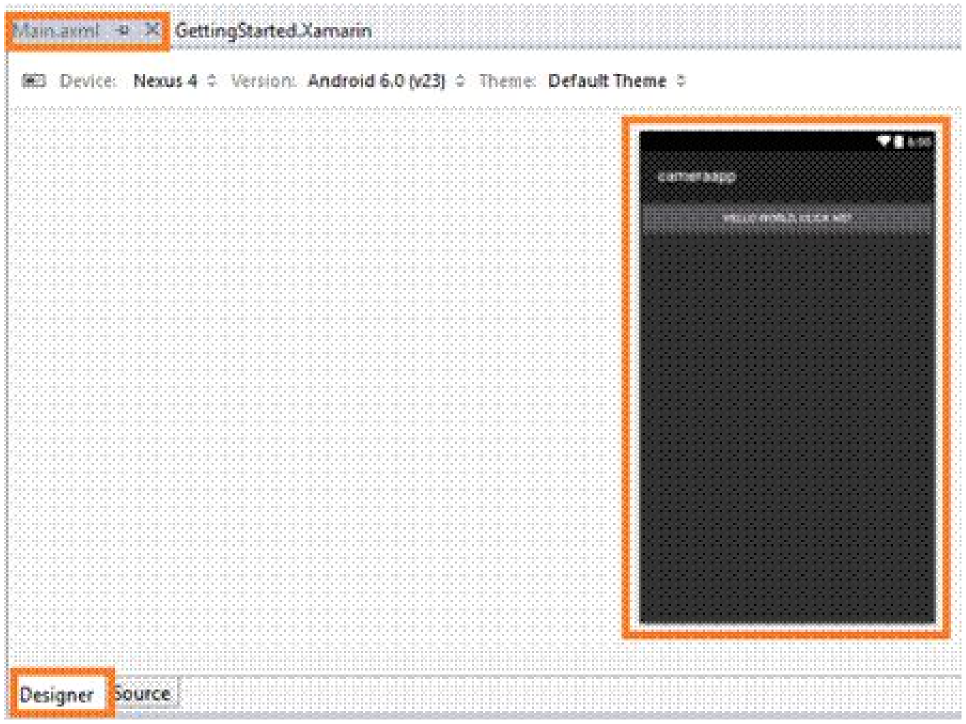
Now, select Resource-->Layout-->double click to open main.axml page. To write XAML code, you need to select a source.

Choose the Designer Window, if you want to design. Hence, you can design your app.



Step 4

After opening main.axml, file will open the main page designer. You can design this ꞁ as per your wish.



Now, delete the Default hello world button. Go to the source panel and you can see the button coding. You need to delete it.

After deleting XAML code, delete C# button action code.

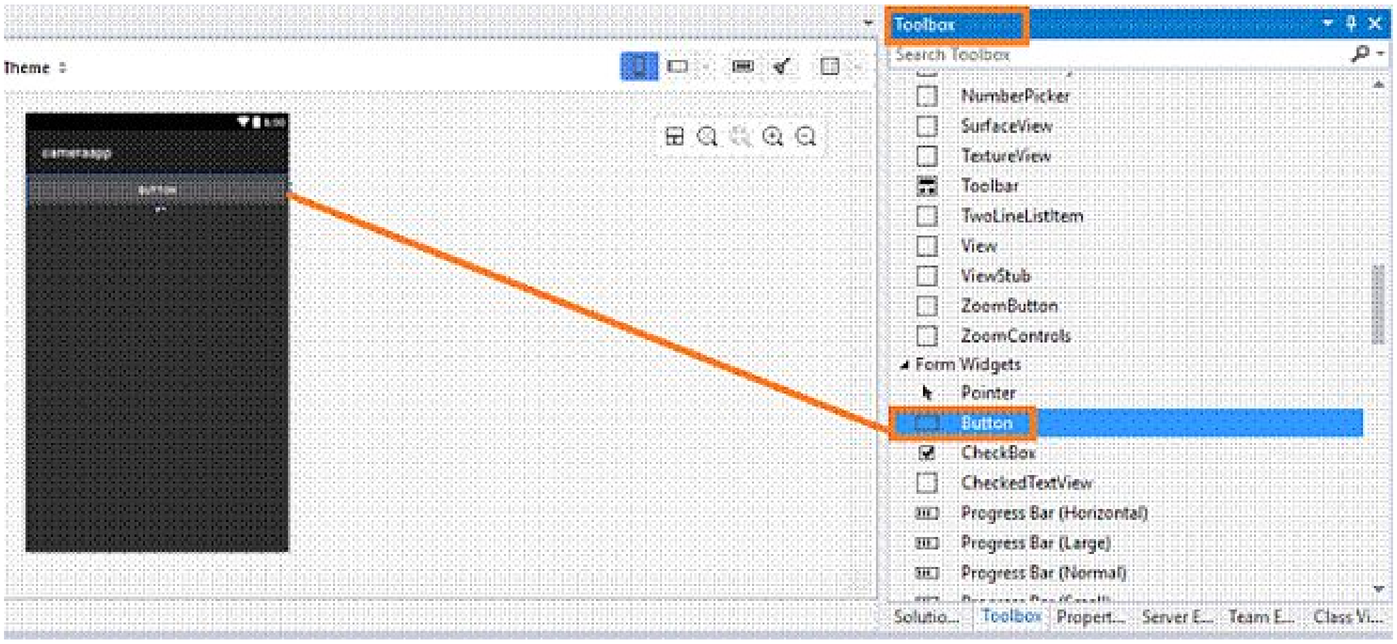
Go to the MainActivity.cs page. You need to delete the button code.

Step 5

Now, go to the toolbox Window. In the toolbox Window, get all the types of the tools and controls.

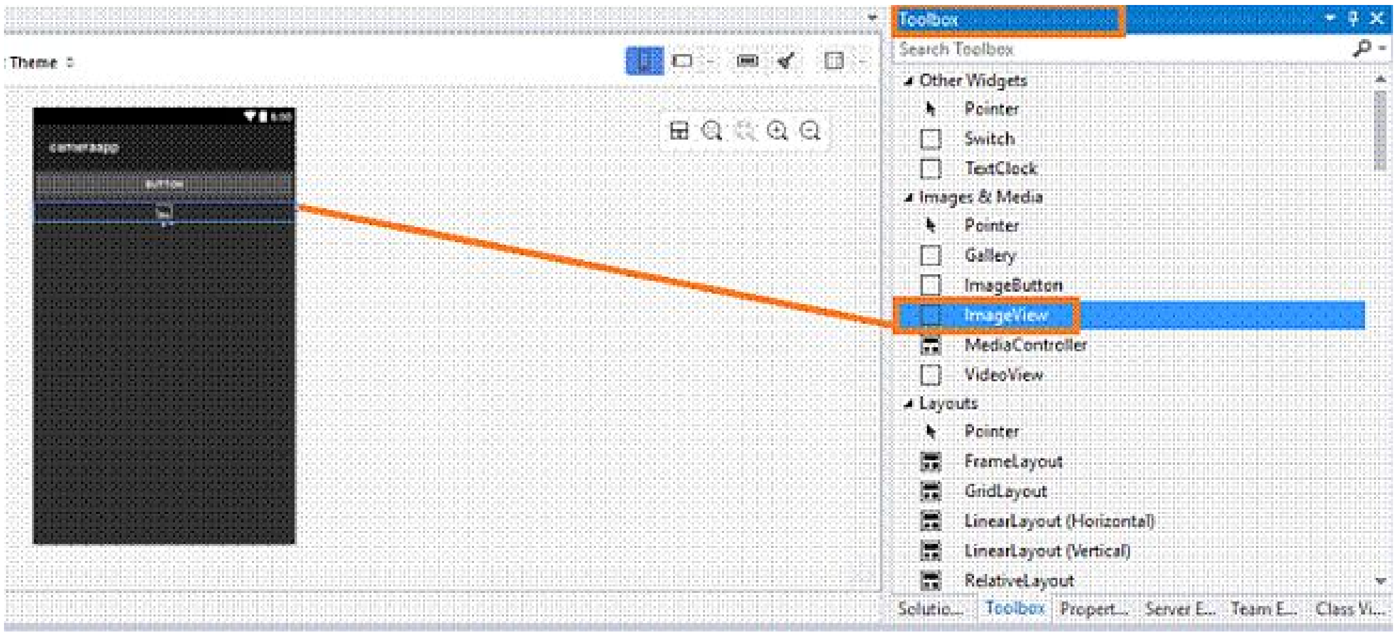
You need to go to the toolbox Window and scroll down. You will see all the tools and controls.

You need to drag and drop the button.



Step 6

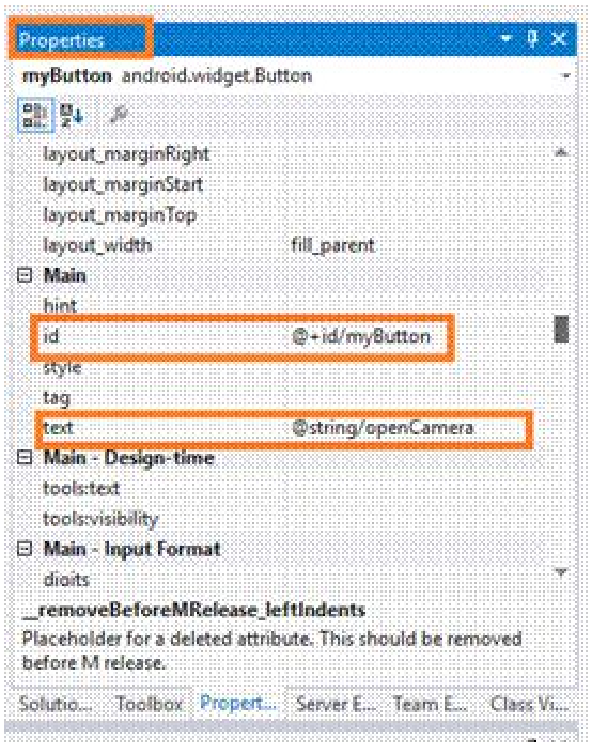
You need to drag and drop the ImageView.



Step 7

Now, go to the properties Window. You need to edit the Button Id value and Text value

(EX: android:id="@+id/myButton" android:text="@string/openCamera").


Properties
Solutio... Toolbox Propert... Server E... Team E... Class Vi...

myButton android.widget.Button

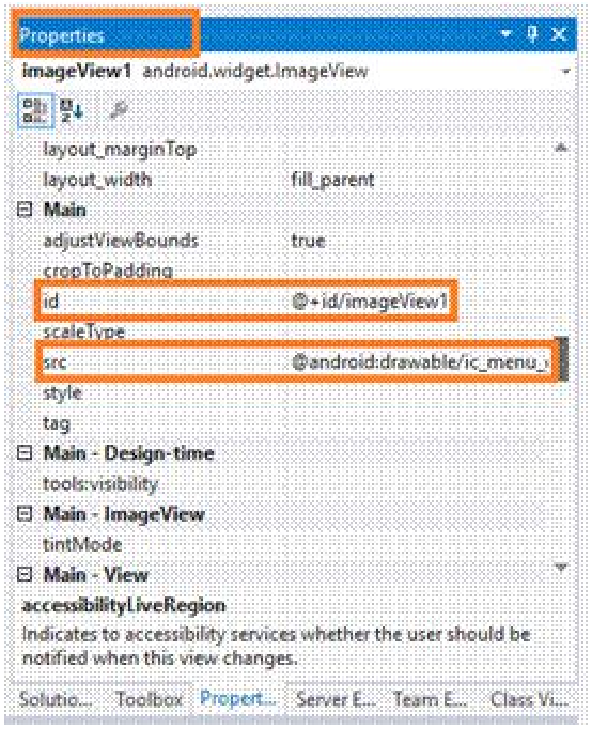
layout_marginRight	
layout_marginStart	
layout_marginTop	
layout_width	fill_parent
Main	
hint	
id	@+id/myButton
style	
tag	
text	@string/openCamera
Main - Design-time	
tools:text	
tools:visibility	
Main - Input Format	
digits	

__removeBeforeMRelease_leftIndents
Placeholder for a deleted attribute. This should be removed before M release.

Step 8

Now, you will edit ImageView Id value and src value.

(Ex: android:id="@+id/imageView1" android:src="@android:drawable/ic_menu_gallery").


Properties
Solutio... Toolbox Propert... Server E... Team E... Class Vi...

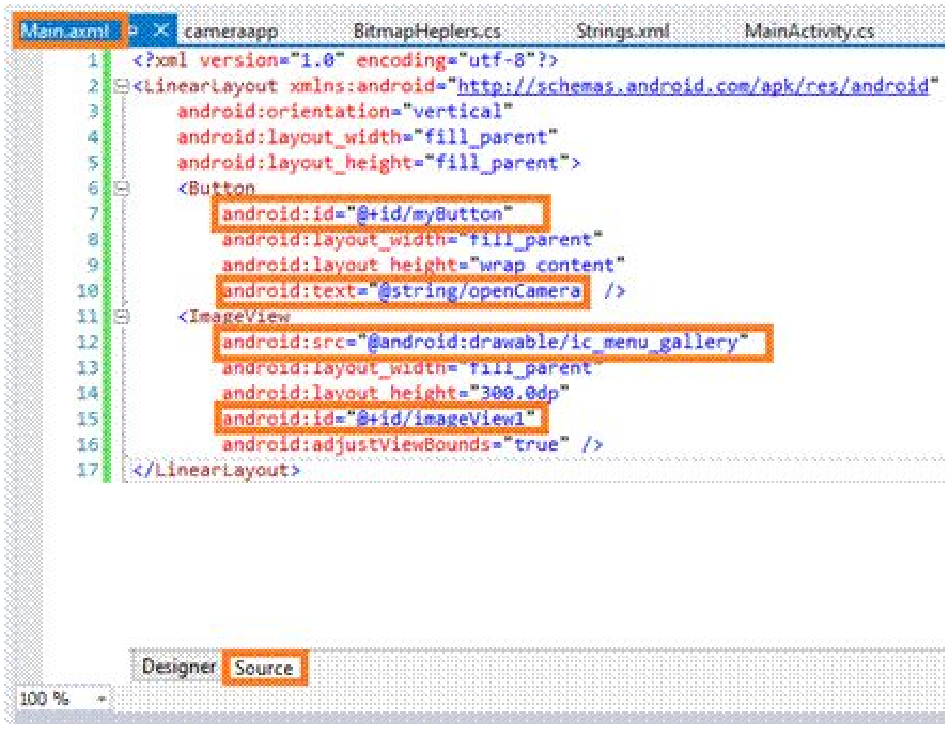
imageView1 android.widget.ImageView

layout_marginTop	
layout_width	fill_parent
Main	
adjustViewBounds	true
cropToPadding	
id	@+id/imageView1
scaleType	
src	@android:drawable/ic_menu_
style	
tag	
Main - Design-time	
tools:visibility	
Main - ImageView	
tintMode	
Main - View	

accessibilityLiveRegion
Indicates to accessibility services whether the user should be notified when this view changes.

Step 9

In this step, go to the Main.axml page Source Panel. Note, the ImageView and Button Id value and source value.



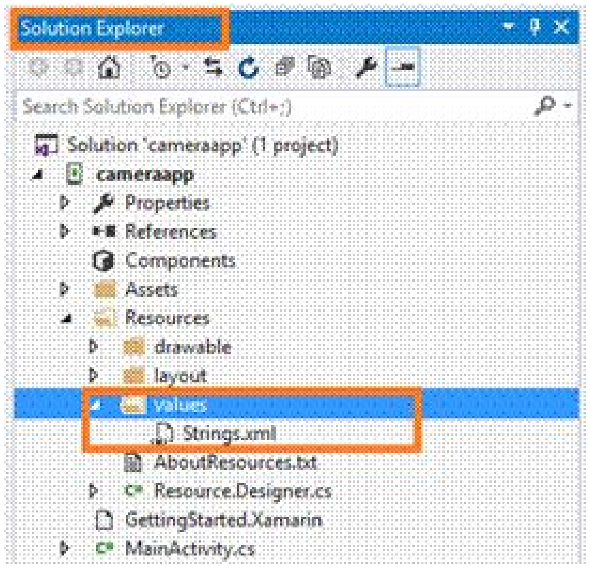
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
    <Button
        android:id="@+id/myButton"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/openCamera" />
    <ImageView
        android:src="@android:drawable/ic_menu_gallery"
        android:layout_width="fill_parent"
        android:layout_height="300.0dp"
        android:id="@+id/imageView1"
        android:adjustViewBounds="true" />
</LinearLayout>
```

Main.axml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/re
    <Button android:id="@+id/myButton" android:layout_width="f
    <ImageView android:src="@android:drawable/ic_menu_gallery"
</LinearLayout>
```

Step 10

In this step, open the String.xml page. Go to Solution Explorer-->Resource-->values-->String.xml.

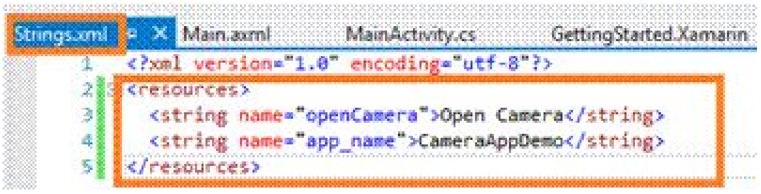


Step 11

Afterwards, open String.xml file. Write XML code, mentioned below.

String.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="openCamera">Open Camera</string>
    <string name="app_name">CameraAppDemo</string>
</resources>
```

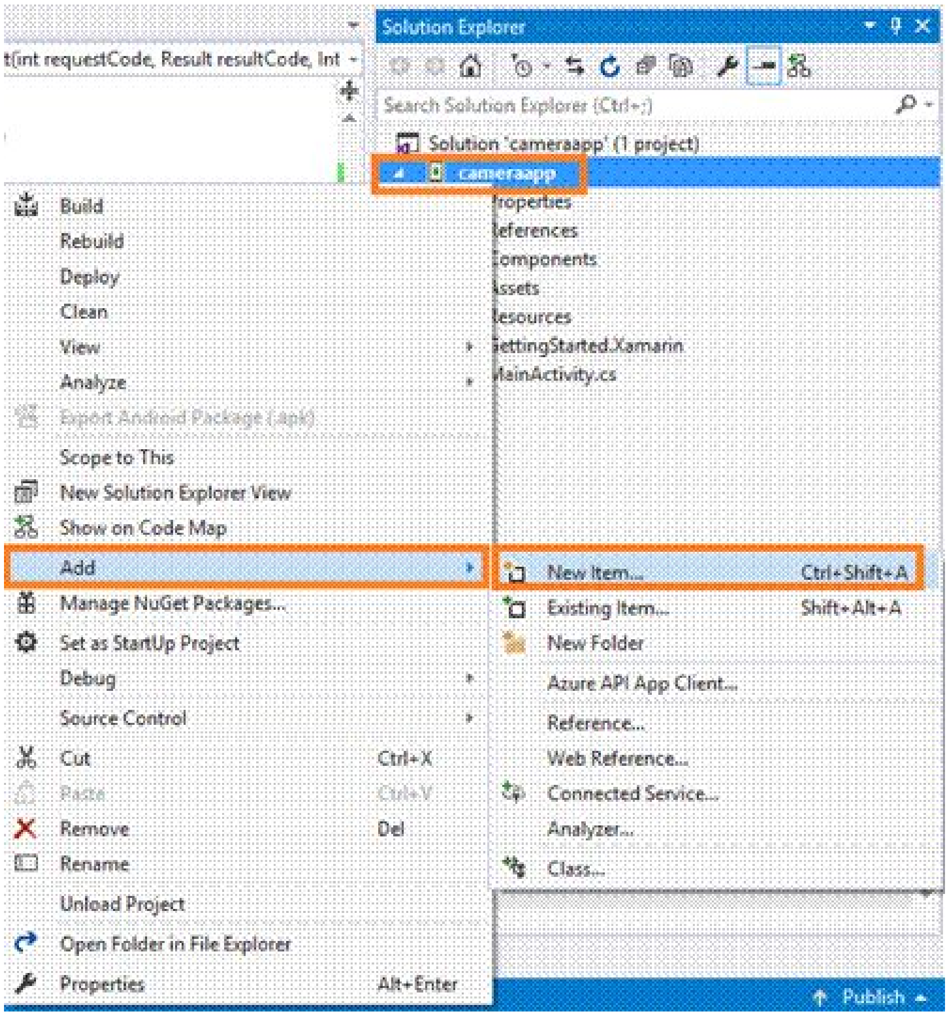


```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <string name="openCamera">Open Camera</string>
  <string name="app_name">CameraAppDemo</string>
</resources>
```

Step 12

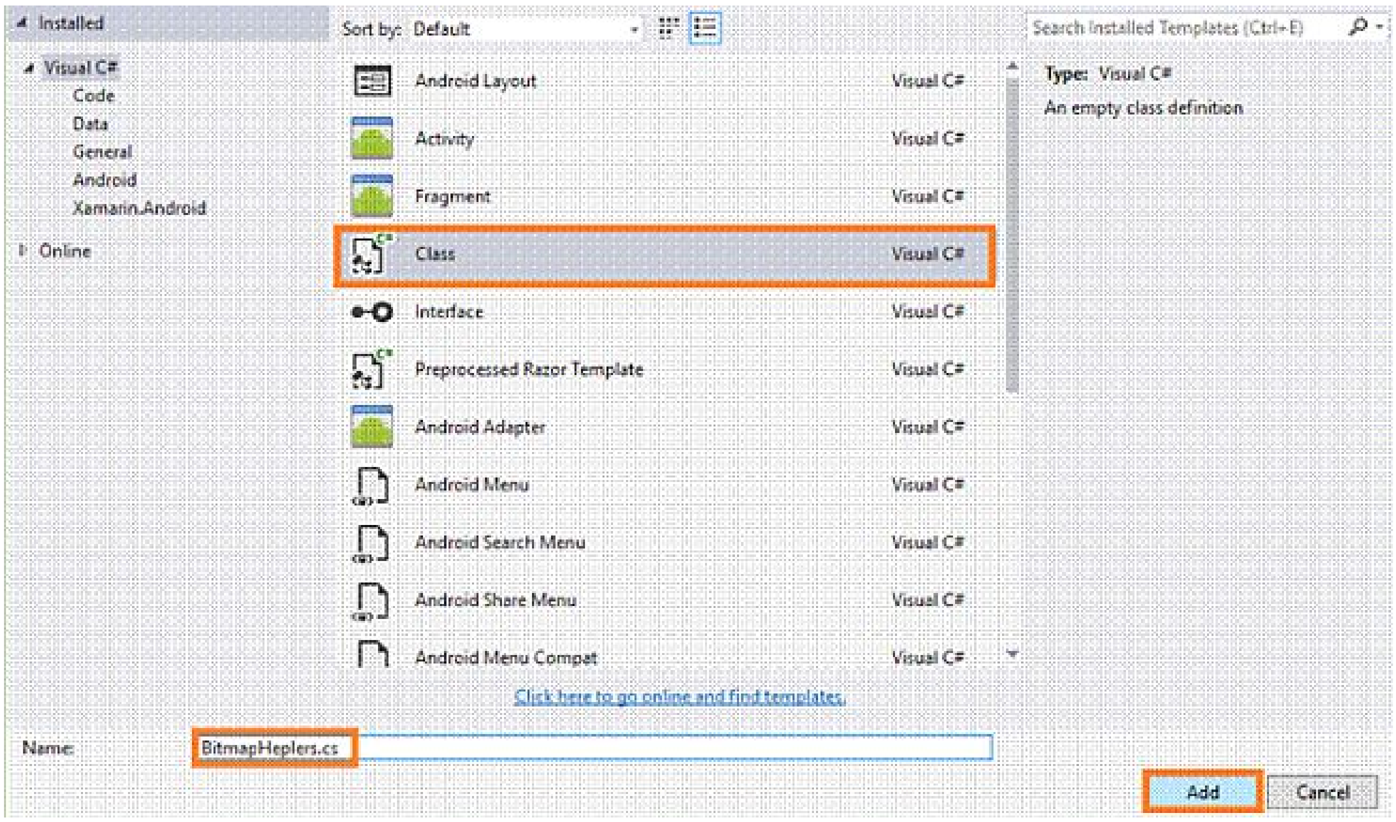
In this step, add one class file.

Go to Solution Explorer-->Resource-->Right click -->Add-->New Item.



Step 13

Now, choose the Class file and give name BitmapHelpers.cs.

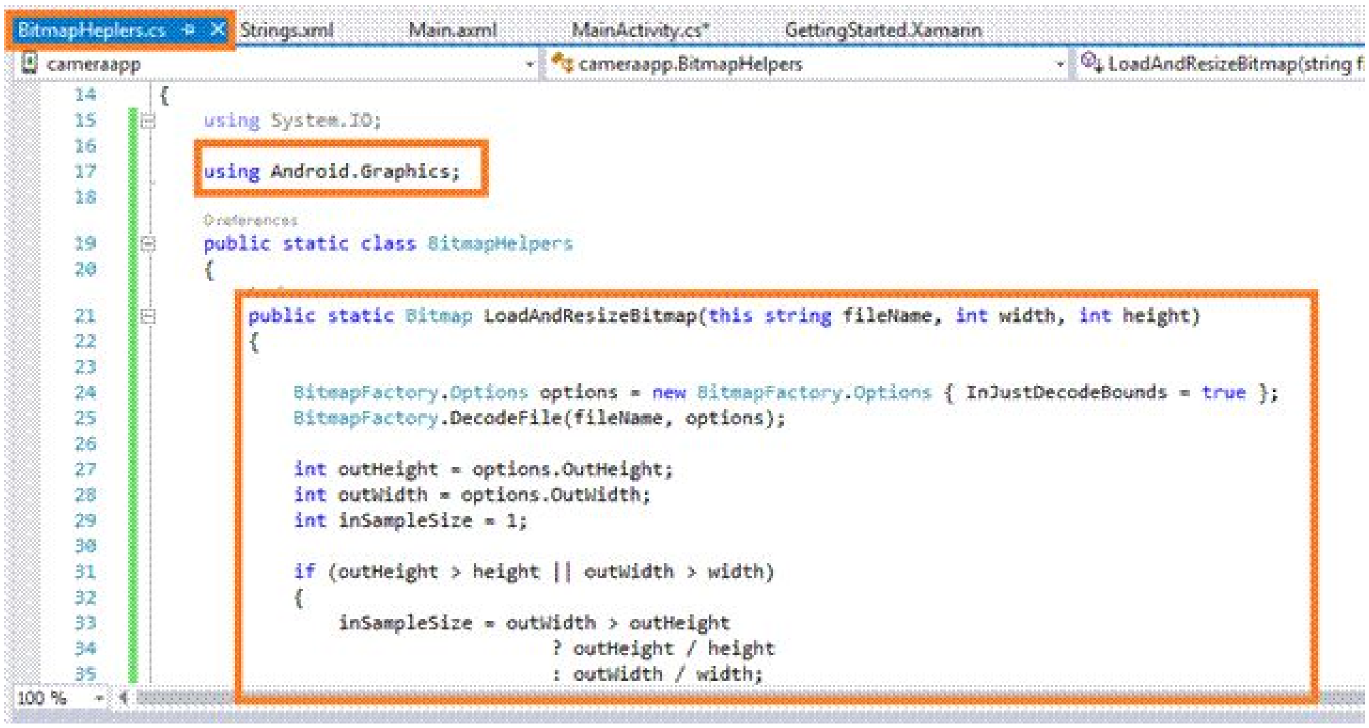


Step 14

Afterwards, create the class file. Write the code, mentioned below.

BitmapHelpers.cs

```
namespace cameraapp {
    using System.IO;
    using Android.Graphics;
    public static class BitmapHelpers {
        public static Bitmap LoadAndResizeBitmap(this string f
            BitmapFactory.Options options = new BitmapFactory.
                InJustDecodeBounds = true
            };
            BitmapFactory.DecodeFile(fileName, options);
            int outHeight = options.OutHeight;
            int outWidth = options.OutWidth;
            int inSampleSize = 1;
            if (outHeight > height || outWidth > width) {
                inSampleSize = outWidth > outHeight ?
                    outHeight / height :
                    outWidth / width;
            }
            options.InSampleSize = inSampleSize;
            options.InJustDecodeBounds = false;
            Bitmap resizedBitmap = BitmapFactory.DecodeFile(fi
            return resizedBitmap;
            }
        }
}
```



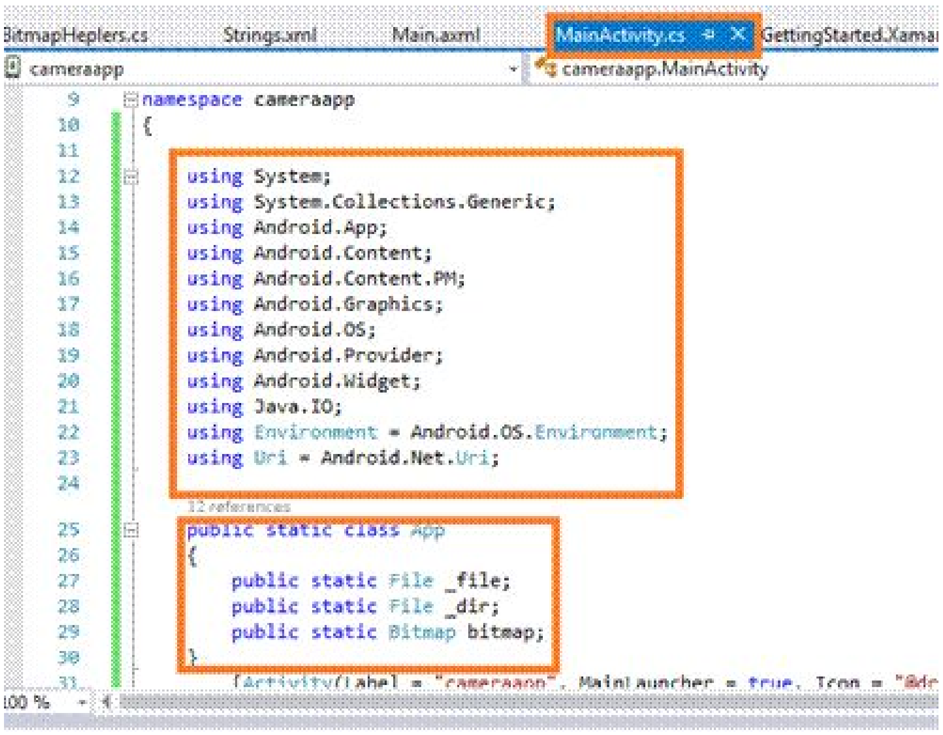
```
{
    using System.IO;

    using Android.Graphics;

    public static class BitmapHelpers
    {
        public static Bitmap LoadAndResizeBitmap(this string fileName, int width, int height)
        {

            BitmapFactory.Options options = new BitmapFactory.Options { InJustDecodeBounds = true };
            BitmapFactory.DecodeFile(fileName, options);

            int outHeight = options.OutHeight;
            int outWidth = options.OutWidth;
            int inSampleSize = 1;

            if (outHeight > height || outWidth > width)
            {
                inSampleSize = outWidth > outHeight
                                    ? outHeight / height
                                    : outWidth / width;
```

Step 15

Now, go to the MainActivity.cs page. Write the namespaces and variables, mentioned below.

MainActivity.cs

```
using Java.IO;
using Environment = Android.OS.Environment;
using Uri = Android.Net.Uri;
public static class App {
    public static File _file;
    public static File _dir;
    public static Bitmap bitmap;
}
```



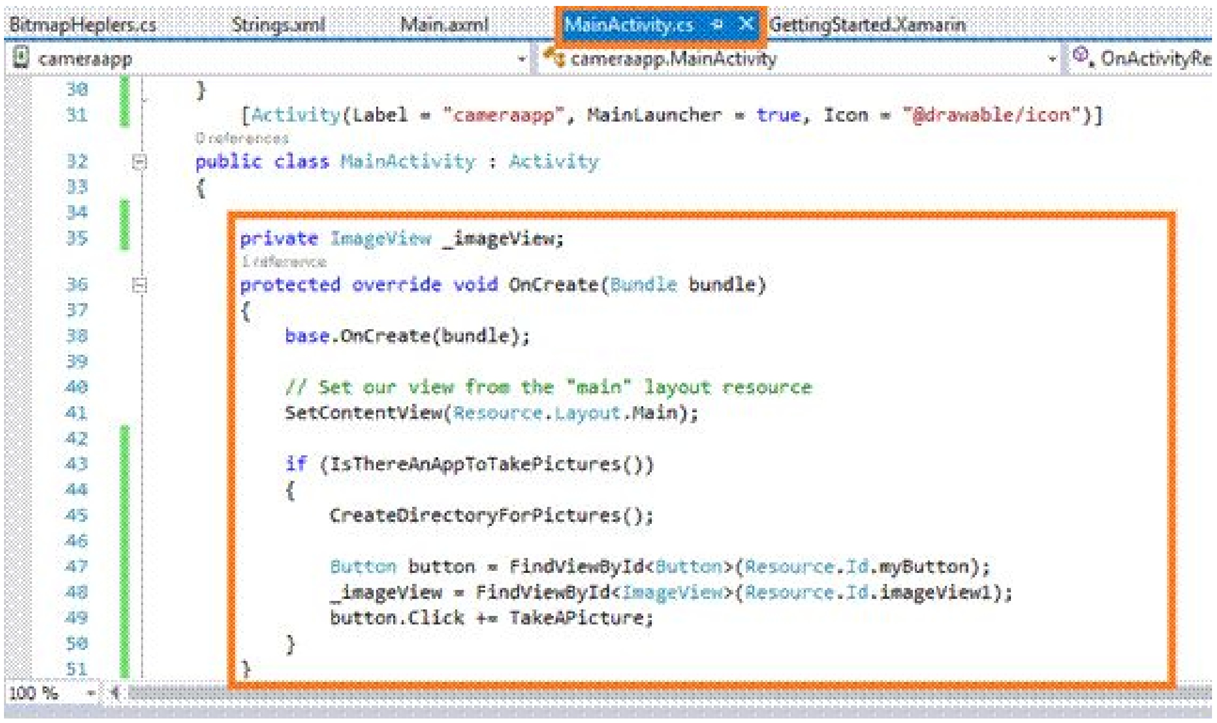
```
namespace cameraapp
{

    using System;
    using System.Collections.Generic;
    using Android.App;
    using Android.Content;
    using Android.Content.PM;
    using Android.Graphics;
    using Android.OS;
    using Android.Provider;
    using Android.Widget;
    using Java.IO;
    using Environment = Android.OS.Environment;
    using Uri = Android.Net.Uri;

    public static class App
    {
        public static File _file;
        public static File _dir;
        public static Bitmap bitmap;
    }
        [Activity(Label = "cameraapp", MainLauncher = true, Icon = "@dr
```

Step 16

MainActivity.cs

Now, go to the MainActivity.cs page. Write the code, mentioned below.

```
public class MainActivity: Activity {
    private ImageView _imageView;
    protected override void OnCreate(Bundle bundle) {
        base.OnCreate(bundle);
        // Set our view from the "main" layout resource
        SetContentView(Resource.Layout.Main);
        if (IsThereAnAppToTakePictures()) {
            CreateDirectoryForPictures();
            Button button = FindViewById < Button > (Resource.
            _imageView = FindViewById < ImageView > (Resource.
            button.Click += TakeAPicture;
        }
    }
    protected override void OnActivityResult(int requestCode,
        base.OnActivityResult(requestCode, resultCode, data);
        Intent mediaScanIntent = new Intent(Intent.ActionMedia
        Uri contentUri = Uri.FromFile(App._file);
        mediaScanIntent.SetData(contentUri);
        SendBroadcast(mediaScanIntent);
        int height = Resources.DisplayMetrics.HeightPixels;
        int width = _imageView.Height;
        App.bitmap = App._file.Path.LoadAndResizeBitmap(width,
        if (App.bitmap != null) {
                _imageView.SetImageBitmap(App.bitmap);
                App.bitmap = null;
            }
            GC.Collect();
        }
        private void CreateDirectoryForPictures() {
            App._dir = new File(
                Environment.GetExternalStoragePublicDirectory(
                    Environment.DirectoryPictures), "CameraAppDemc
            if (!App._dir.Exists()) {
                App._dir.Mkdirs();
            }
        }
        private bool IsThereAnAppToTakePictures() {
            Intent intent = new Intent(MediaStore.ActionImageCaptu
            IList < ResolveInfo > availableActivities =
                PackageManager.QueryIntentActivities(intent, Packa
            return availableActivities != null && availableActivit
        }
        private void TakeAPicture(object sender, EventArgs eventAr
            Intent intent = new Intent(MediaStore.ActionImageCaptu
            App._file = new File(App._dir, String.Format("myPhoto_
            intent.PutExtra(MediaStore.ExtraOutput, Uri.FromFile(A
            StartActivityForResult(intent, 0);
        }
}
```

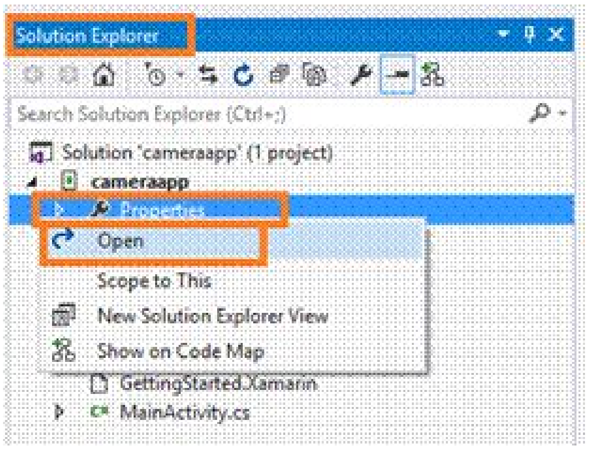


```
}
    [Activity(Label = "cameraapp", MainLauncher = true, Icon = "@drawable/icon")]
    0 references
public class MainActivity : Activity
{
    private ImageView _imageView;
    1 reference
    protected override void OnCreate(Bundle bundle)
    {
        base.OnCreate(bundle);

        // Set our view from the "main" layout resource
        SetContentView(Resource.Layout.Main);

        if (IsThereAnAppToTakePictures())
        {
            CreateDirectoryForPictures();

            Button button = FindViewById<Button>(Resource.Id.myButton);
            _imageView = FindViewById<ImageView>(Resource.Id.imageView1);
            button.Click += TakeAPicture;
        }
    }
```

Step 17

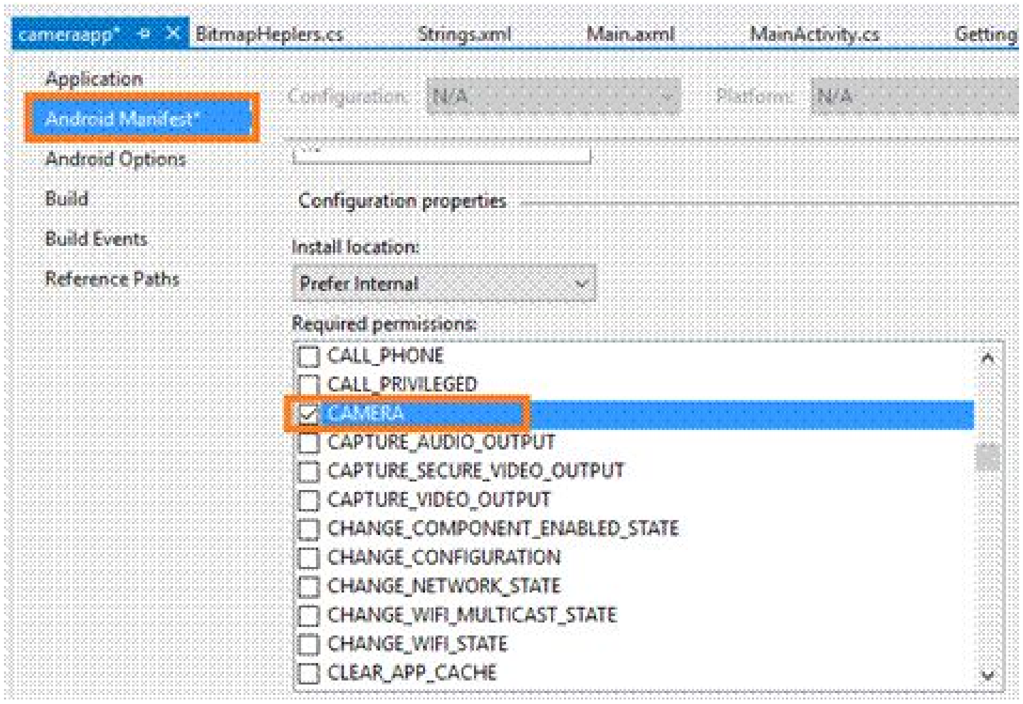
In this step, give the required permissions in your app.

Go to Solution Explorer--> properties-->Right click-->Open.



Step 18

Afterwards, open the properties options. Select Android Manifest-->Required Permissions-->Check CAMERA.



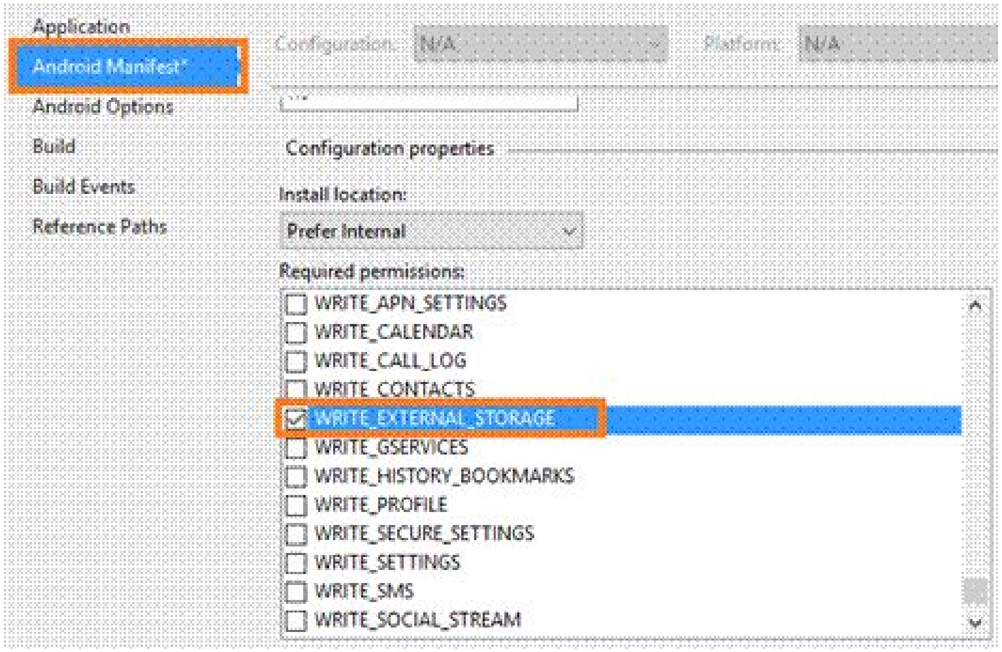
Install location: Prefer Internal

Required permissions:

- [ ] CALL_PHONE
- [ ] CALL_PRIVILEGED
- [x] CAMERA
- [ ] CAPTURE_AUDIO_OUTPUT
- [ ] CAPTURE_SECURE_VIDEO_OUTPUT
- [ ] CAPTURE_VIDEO_OUTPUT
- [ ] CHANGE_COMPONENT_ENABLED_STATE
- [ ] CHANGE_CONFIGURATION
- [ ] CHANGE_NETWORK_STATE
- [ ] CHANGE_WIFI_MULTICAST_STATE
- [ ] CHANGE_WIFI_STATE
- [ ] CLEAR_APP_CACHE

Step 19

Select Android Manifest-->Required Permissions-->Check WRITE_EXTERNAL_STORAGE.



Configuration properties

Install location: Prefer Internal

Required permissions:

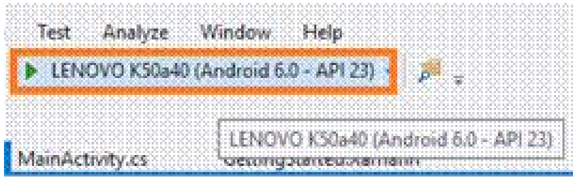
- [ ] WRITE_APN_SETTINGS
- [ ] WRITE_CALENDAR
- [ ] WRITE_CALL_LOG
- [ ] WRITE_CONTACTS
- [x] WRITE_EXTERNAL_STORAGE
- [ ] WRITE_GSERVICES
- [ ] WRITE_HISTORY_BOOKMARKS
- [ ] WRITE_PROFILE
- [ ] WRITE_SECURE_SETTINGS
- [ ] WRITE_SETTINGS
- [ ] WRITE_SMS
- [ ] WRITE_SOCIAL_STREAM

Step 20

If you have Android Virtual device, run the app on it, else connect your Android phone and run the app on it.

Simply, connect your phone and go to Visual Studio. The connected phone will show up in Run menu.

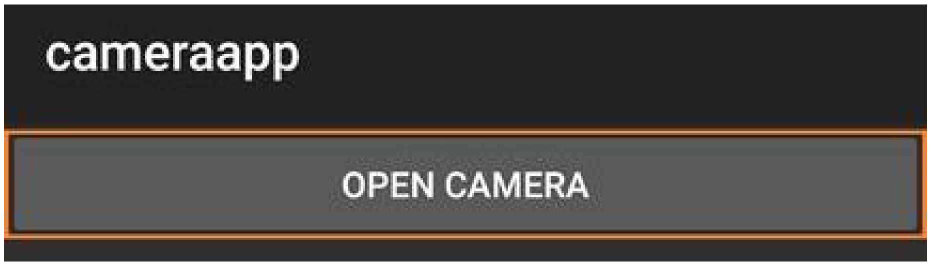
(Ex:LENOVO A6020a40(Android 5.1-API 22)). Click Run option.


Test
Analyze
Window
Help
LENOVO K50a40 (Android 6.0 - API 23)
LENOVO K50a40 (Android 6.0 - API 23)
MainActivity.cs

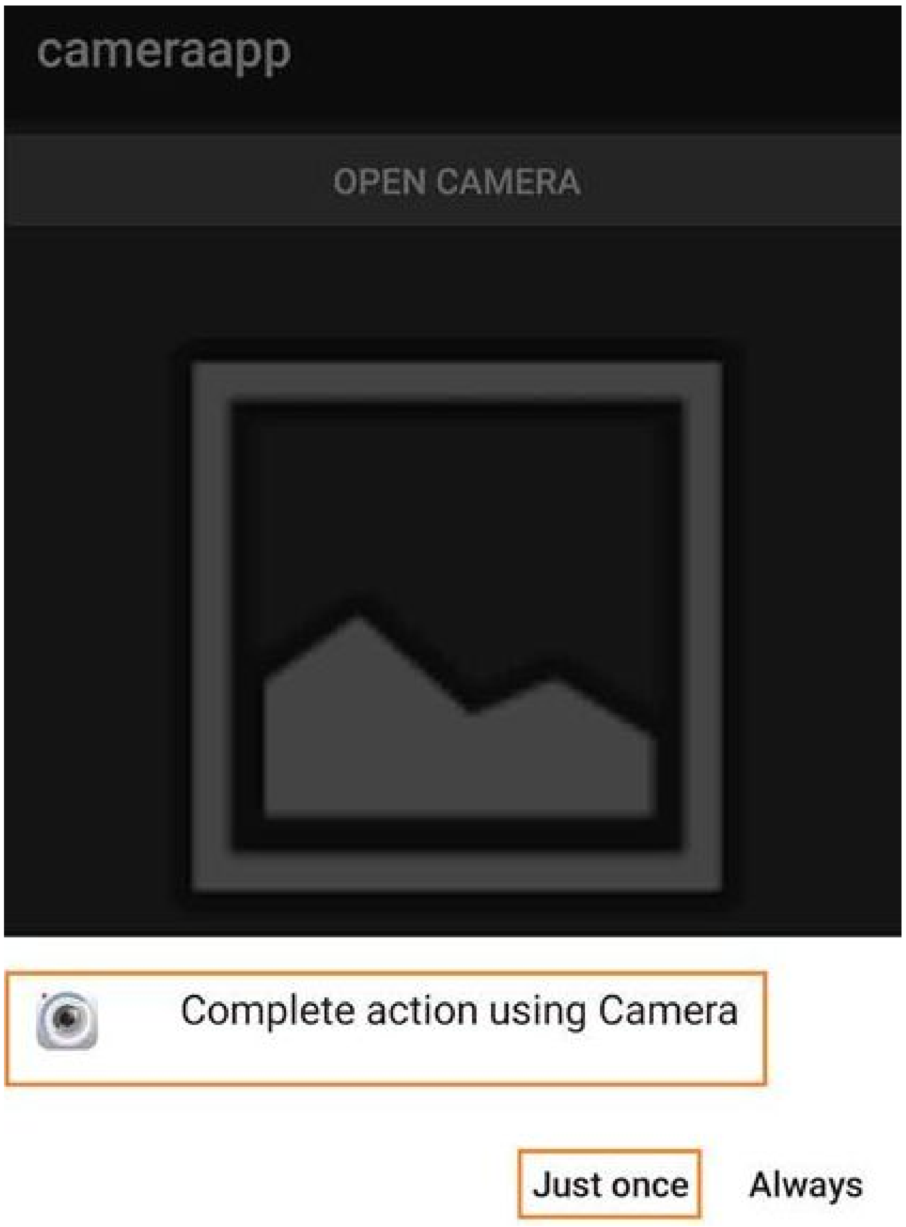
Output

After few seconds, the app will start running on your phone.

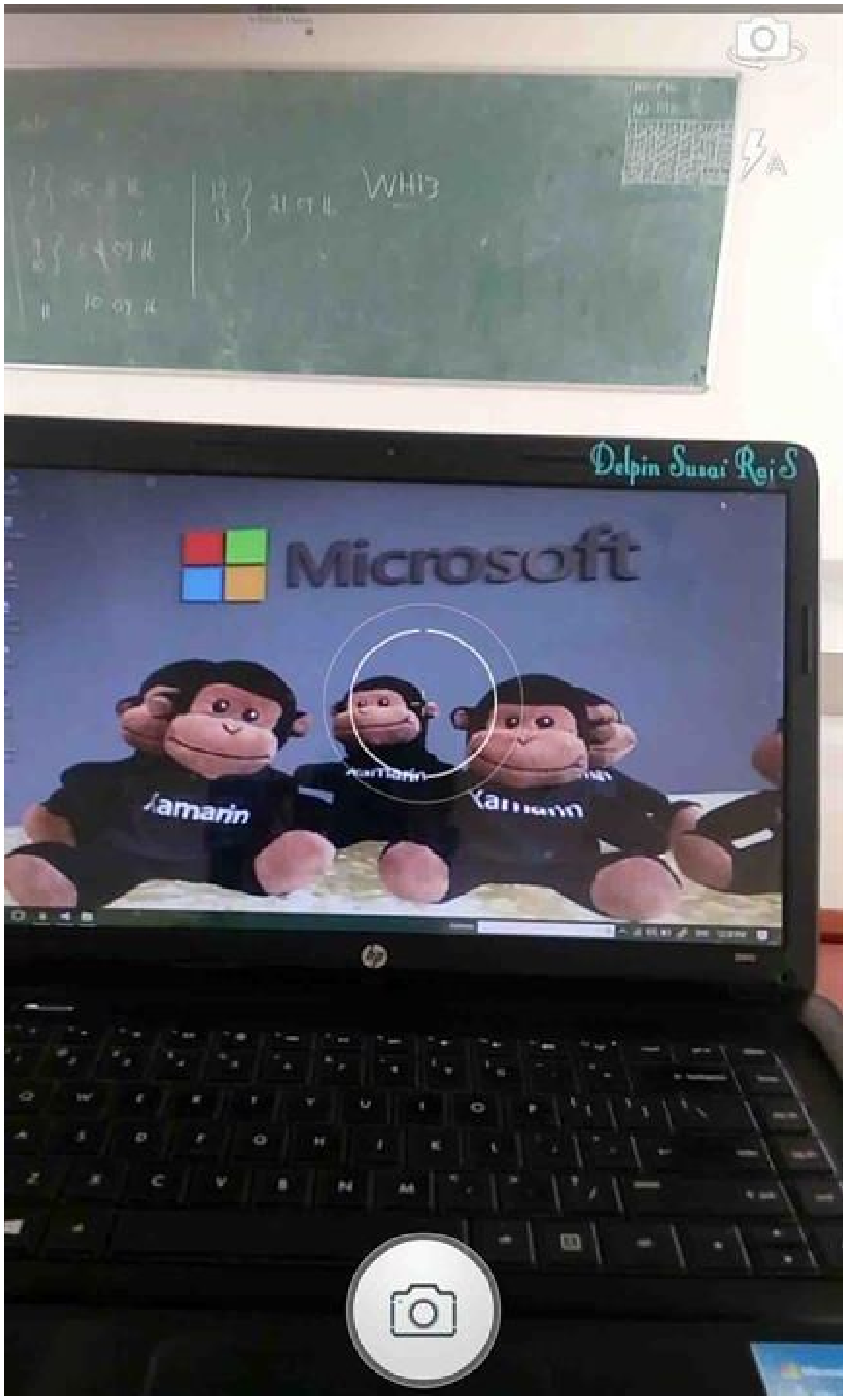
You can click open camera button.



Now, choose camera and click just once.



You will see the camera app is working successfully.



Summary

Hence, this was the process of creating a Camera App in Xamarin Android app, using Visual Studio 2015.

Android App Camera App Visual Studio 2015 Xamarin

Delpin Susai Raj TOP 50

Xamarin Developer | Author | Blogger | Speaker | C#Corner MVP(Most Valuable Professional) | DZone MVB(Most Valuable Blogger)

http://xamarinmonkeys.blogspot.in/

42 2.6m 3

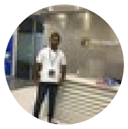
6 5

Type your comment here and press Enter Key (Minimum 10 characters)

How do i save the photo to my device after capturing it?

Mfundo Xolo Nov 05, 2018

1799 2 0 0 1 Reply



Hi Mfundo Xolo, please check this article TakeAPicture() function, and also enable WRITE_EXTERNAL_STORAGE permission.

Delpin Susai Raj Nov 08, 2018

42 29.1k 2.6m 0

Nice article...I need to capture image after 5 seconds of opening camera. How can i integrate? without user interaction i need to capture

Sagar Patil Mar 14, 2018

1704 97 427 2 0 Reply

How add crop functionality in this camera please guide me bro

Umair Ilyas Dec 08, 2017

1724 77 0 1 0 Reply

Thank you..

Joe Wilson Oct 28, 2016

228 7.9k 276.5k 1 0 Reply

TRENDING UP

01 What Is Replication In MongoDB? How To Configure Replication In MongoDB?

02 C# Corner Author Posts Analytics With Angular 8

03 .NET Interview Questions

04 Consume HTTPS Service With Self-Signed Certificate In Xamarin.Forms

05 Repository Design Pattern In .NET CORE WEB API

06 Interfaces In C# 8.0

07 Verify OTP Automatically In Android Without SMS Read Permission Using Xamarin.Forms

08 Top 10 Project Management Software

09 CRUD Operation Using Repository Design Pattern In ASP.NET MVC

10 Add Dynamic Image In PDF And Download PDF In Angular 7

View All