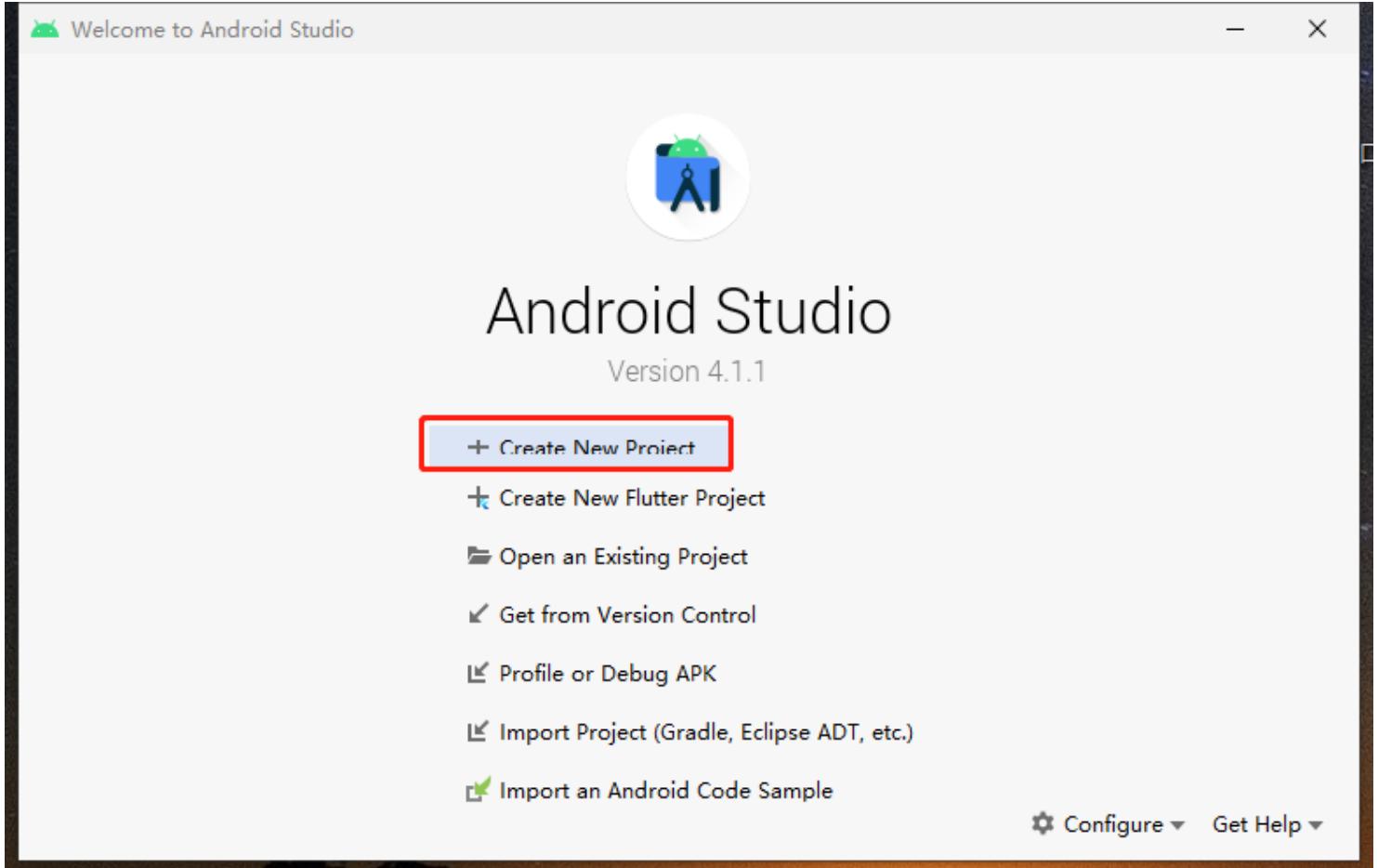


how to make android app from webpage - KinCony

Use Android Studio create new project

Open Android Studio, if you first time use Android Studio, click “create New Project”.

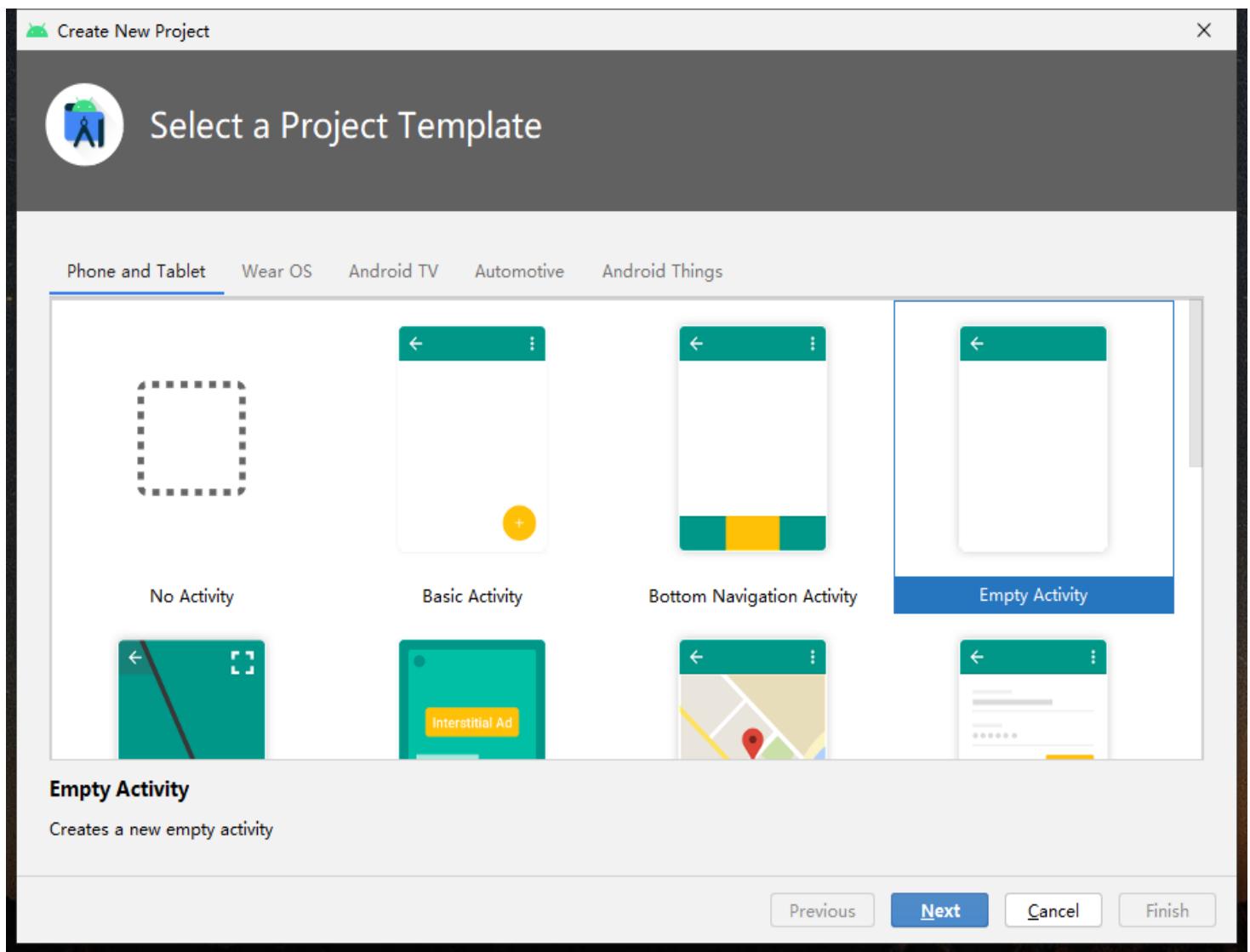


if not first time use Android Studio, Android Studio will auto open last project, now you can click File->New-->New Project....

The screenshot shows the Android Studio interface. The top navigation bar includes 'File', 'Edit', 'View', 'Navigate', 'Code', 'Analyze', 'Refactor', 'Build', 'Run', 'Tools', 'VCS', 'Window', and 'Help'. A red box highlights the 'File' menu, specifically the 'New' section where 'New Project...' is selected. The main workspace displays the code for 'MainActivity.java' which contains logic for loading a URL into a WebView. The bottom status bar shows 'Event Log', 'Layout Inspector', '1:1 CRLF UTF-8 4 spaces', and other developer tools.

```
MainActivity extends AppCompatActivity {  
    String url = "http://iot.kincony.com:1888/ui/#!/0?socketId=yJktDF85HAA85JIAACe";  
    WebView webView;  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        webView = findViewById(R.id.web_view);  
        webView.getSettings().setJavaScriptEnabled(true);  
        webView.setWebViewClient(new WebViewClient());  
        webView.loadUrl(url);  
    }  
}
```

Select “Empty Activity”, click “Next”.



Input text for “Name, Package name, Save location” , others as default, not need to change, click “Finish”.

Name: Project name.

Package name: The unique ID of the project cannot be the same as other applications, otherwise the installation will fail. It's better to start with your own domain name, such as com.kincony

Save location: project file save path.

 Create New Project

X



Configure Your Project



Empty Activity

Creates a new empty activity

Name

MyApplication

Package name

com.lm.myapplication

Save location

C:\Users\wengw\Desktop\MyApplication

Language

Java

Minimum SDK API 16: Android 4.1 (Jelly Bean)

 Your app will run on approximately **99.8%** of devices.

[Help me choose](#)

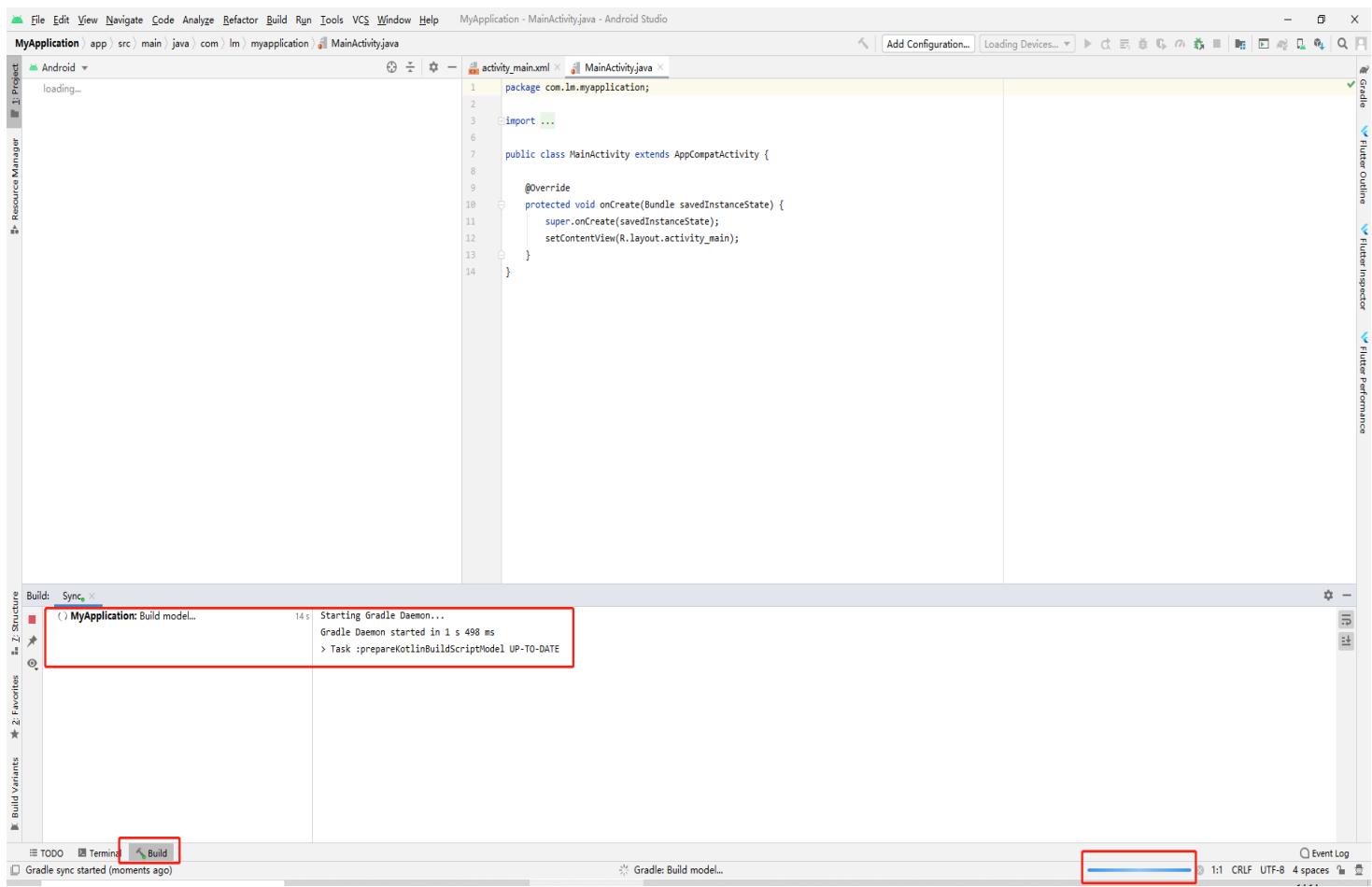
Use legacy android support libraries 

[Previous](#)

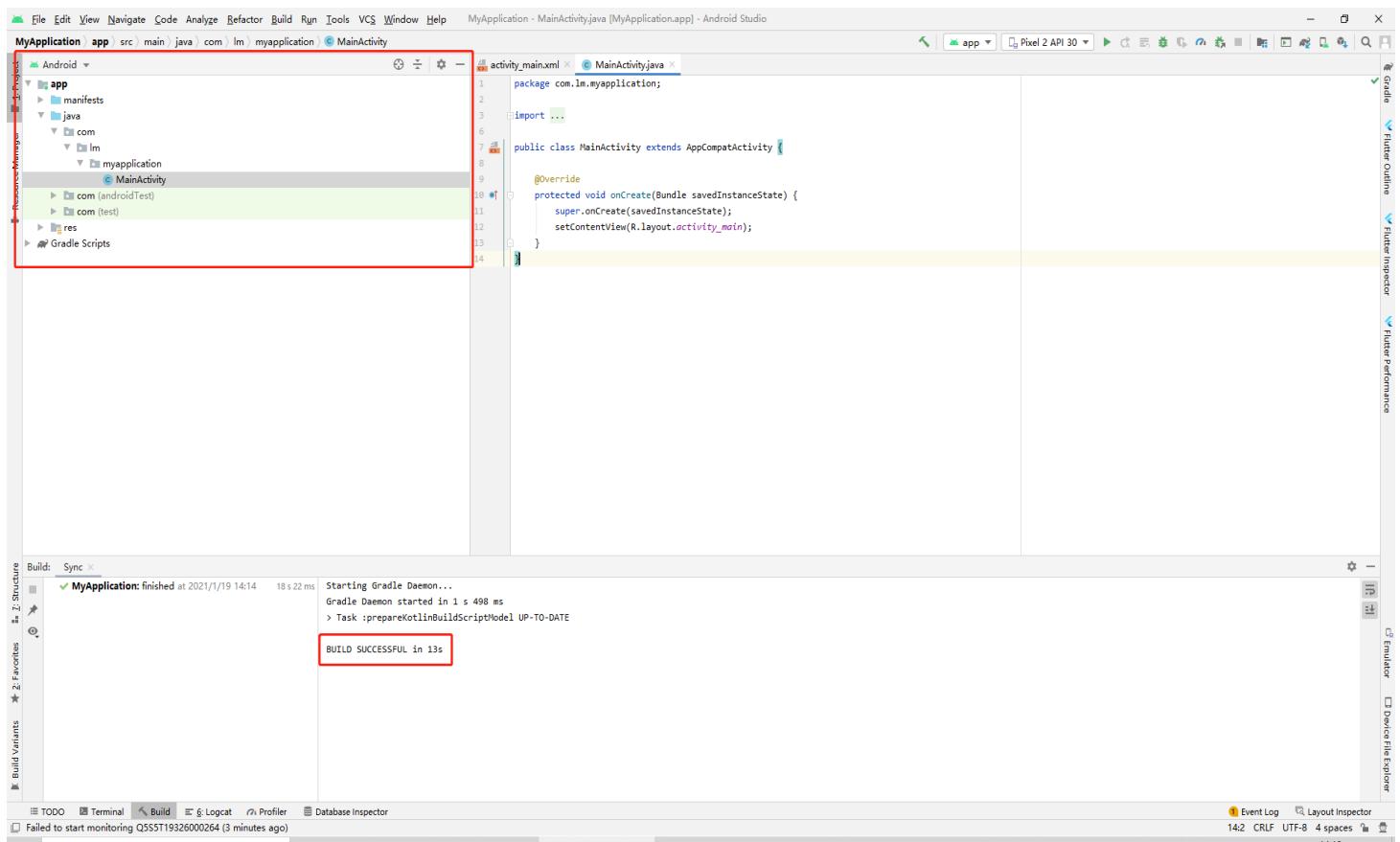
[Next](#)

[Cancel](#)

[Finish](#)



When Andorid Studio startup, you can write code now.



Add network access

Open AndroidManifest.xml file, before <application> insert code:

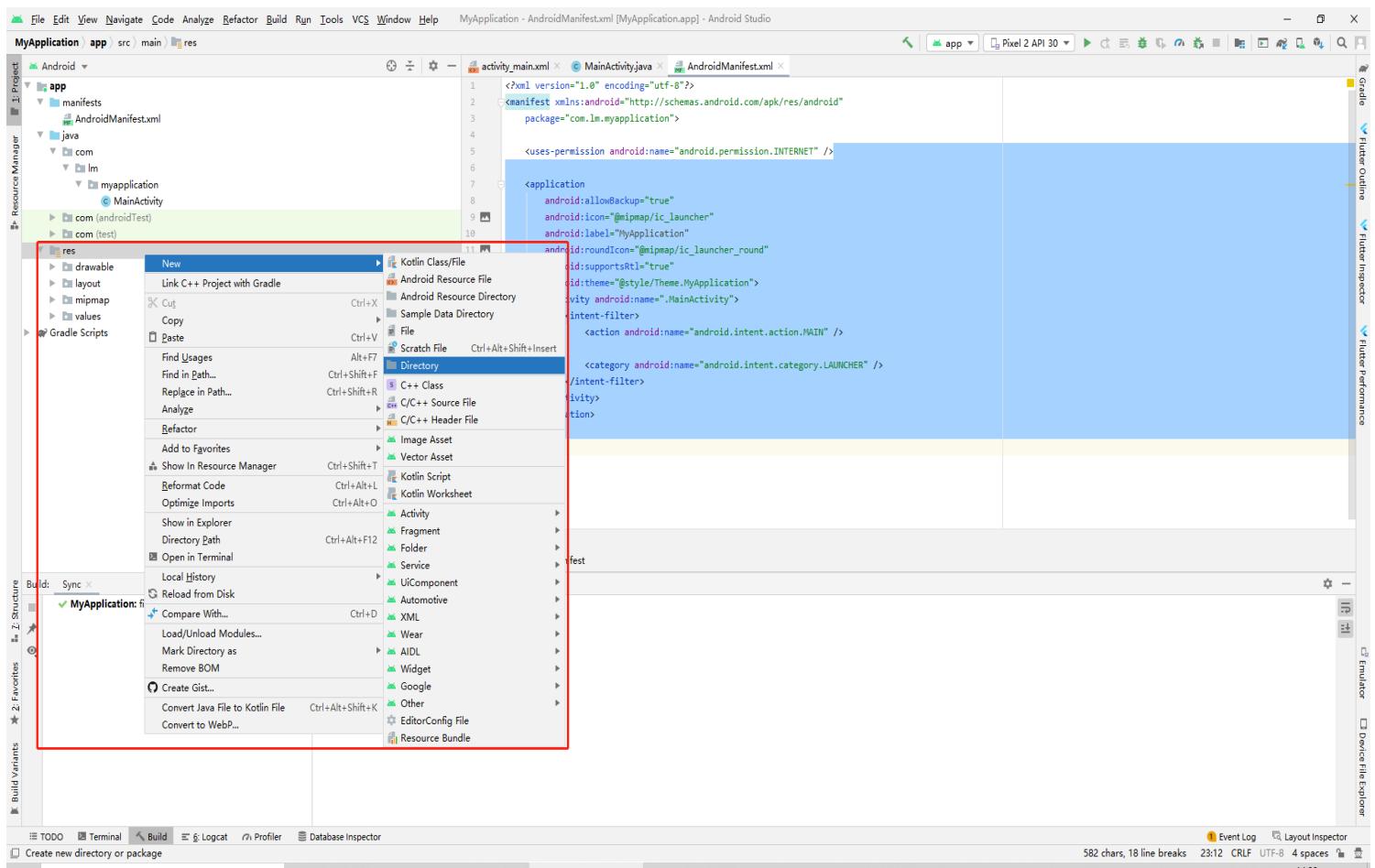
```
<uses-permission android:name="android.permission.INTERNET"/>
```

The screenshot shows the Android Studio interface with the following details:

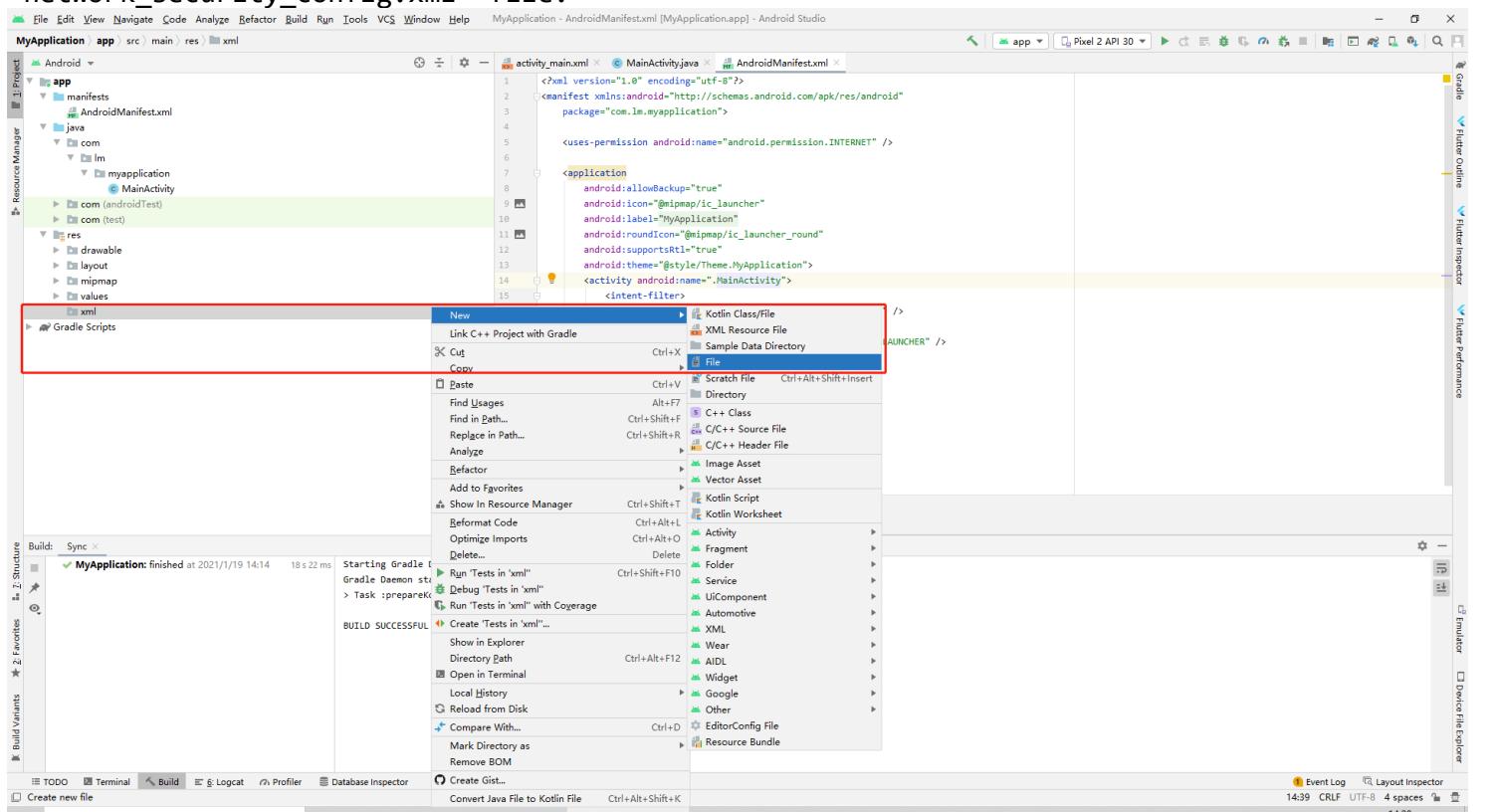
- Project Structure:** The left sidebar shows the project structure under "MyApplication > app > src > main". The "AndroidManifest.xml" file is selected and highlighted with a red box.
- Code Editor:** The main editor window displays the XML code for the manifest. A specific line, "`<uses-permission android:name="android.permission.INTERNET" />`", is highlighted with a red box.
- Build Output:** The bottom-left pane shows the build log:
 - Starting Gradle Daemon...
 - Gradle Daemon started in 1 s 498 ms
 - > Task :prepareKotlinBuildScriptModel UP-TO-DATE
 - BUILD SUCCESSFUL in 13s
- Bottom Bar:** The bottom navigation bar includes tabs for TODO, Terminal, Build, Logcat, Profiler, Database Inspector, Event Log, Layout Inspector, and other developer tools.

Support http protocol

Click “res” folder, mouse right button click “New-->Directory” create “xml”folder

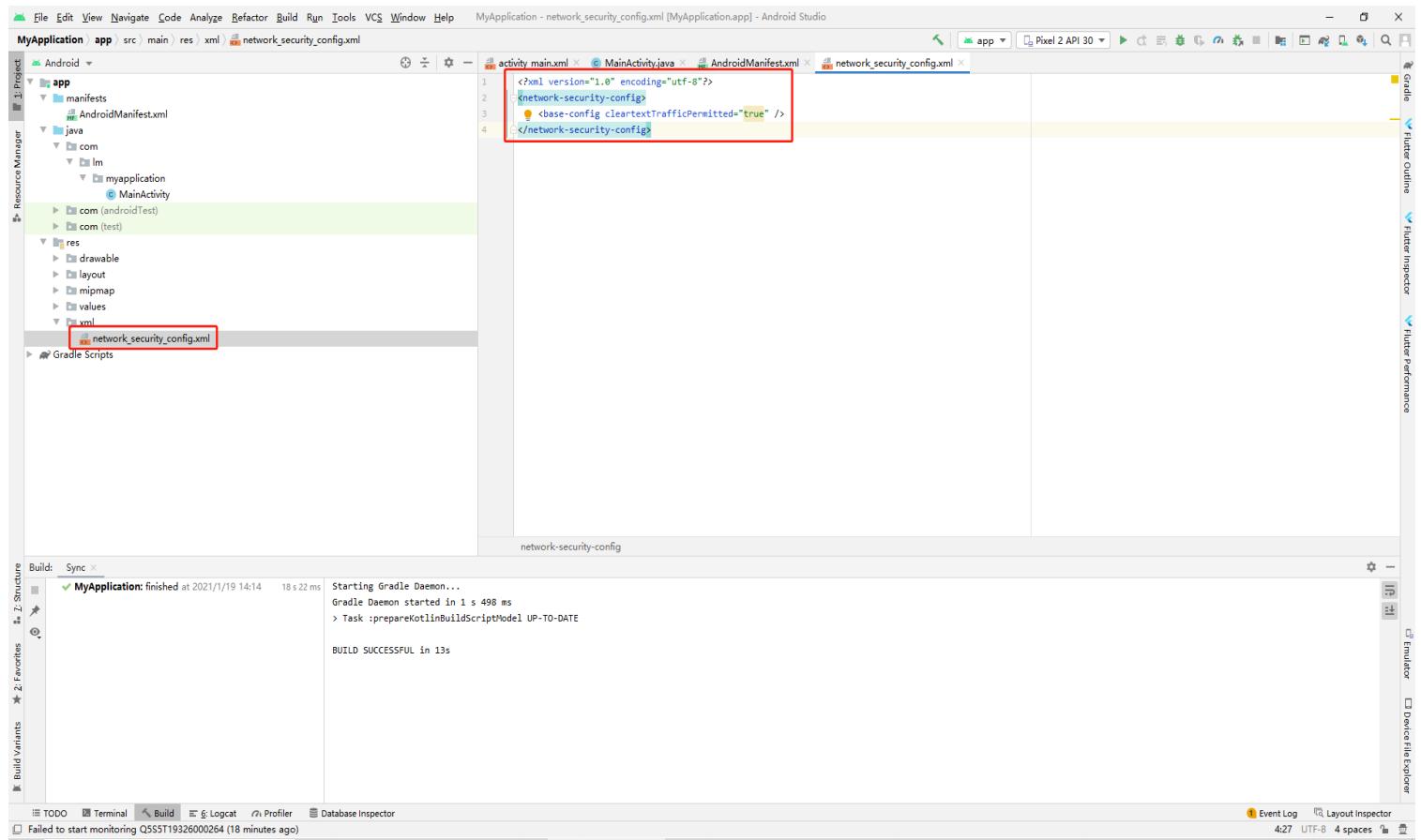


Chose “xml” folder, mouse right button click “New-->File” create “network_security_config.xml” file.



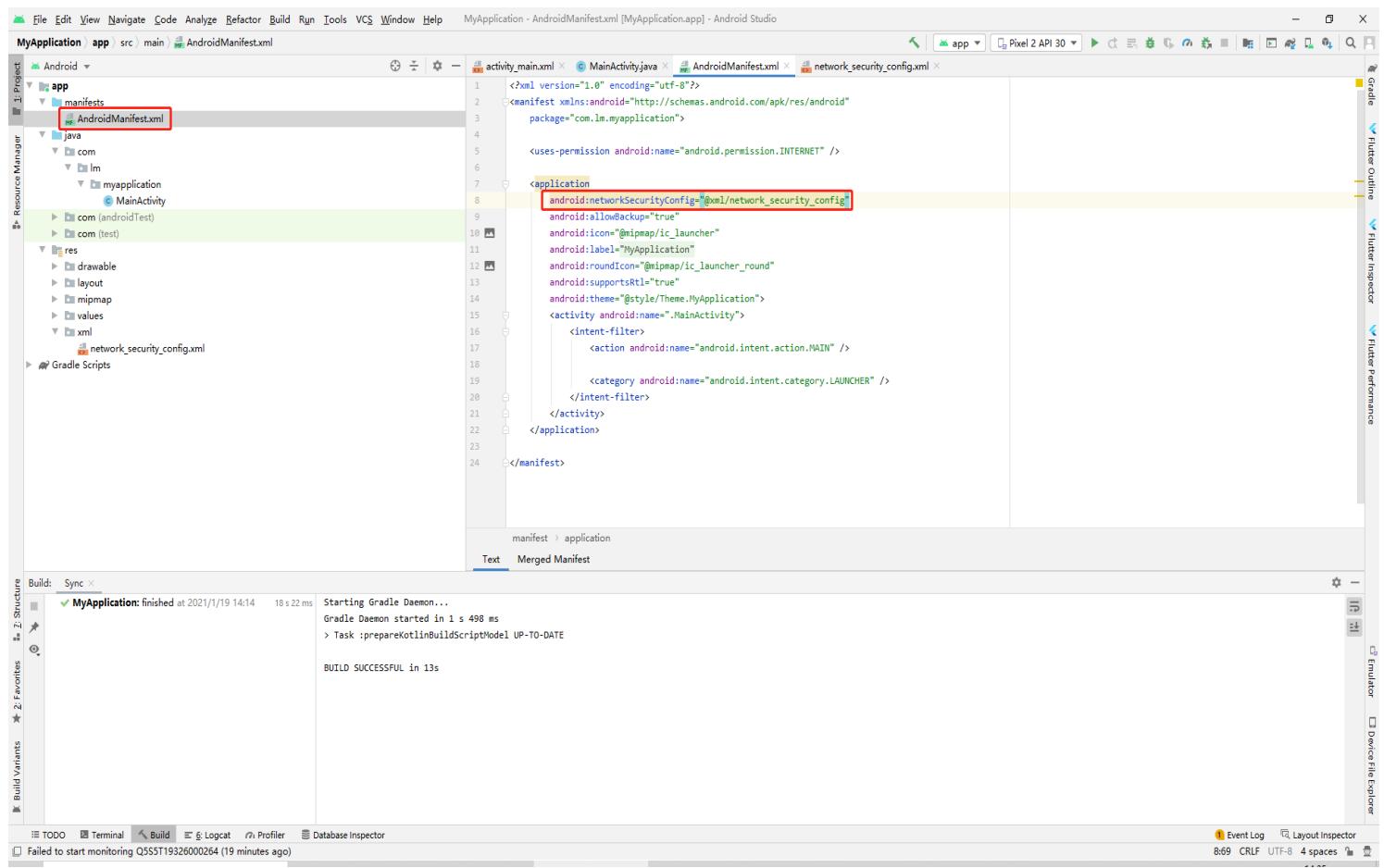
Modify “network_security_config.xml” input these code:

```
<?xml version="1.0" encoding="utf-8"?>
<network-security-config>
    <base-config cleartextTrafficPermitted="true" />
</network-security-config>
```



Open “AndroidManifest.xml” again, between <application> add code:

```
        android:networkSecurityConfig="@xml/network_security_config"
```



Add WebView

Open "activity_main.xml", modify code as follows:

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout 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=".MainActivity">

    <WebView
        android:id="@+id/web_view"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"

    <WebView
        android:id="@+id/web_view"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Build: Sync

MyApplication: finished at 2021/1/19 14:14 18 s 22 ms Starting Gradle Daemon...
Gradle Daemon started in 1 s 498 ms
> Task :prepareKotlinBuildScriptModel UP-TO-DATE

BUILD SUCCESSFUL in 13s

Event Log Layout Inspector
8:1 CRLF UTF-8 4 spaces

Use WebView load webpage

Open “MainActivity”, define website url you want to open, such as:

```
private String url = " http://iot.kincony.com:1880/ui/#!/0?socketid=yJktDF85HVA4B5J1AACe";
```

This is your url.

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

MyApplication - MainActivity.java [MyApplication.app] - Android Studio

1 Project 1 app

Resource Manager

activity_main.xml MainActivity.java AndroidManifest.xml network_security_config.xml

```
1 package com.lm.myapplication;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.os.Bundle;
6
7 public class MainActivity extends AppCompatActivity {
8
9     private String url = "http://iot.kincony.com:1880/ui/#!/0?socketId=yJktDF85HVA4B53jAACe";
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_main);
15     }
16 }
```

Build Variants 2 Favorites 2 Structure

Event Log Layout Inspector

B5 CRLF UTF-8 4 spaces

Config WebView as red region:

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

MyApplication - MainActivity.java [MyApplication.app] - Android Studio

1 Project 1 app

Resource Manager

activity_main.xml MainActivity.java AndroidManifest.xml network_security_config.xml

```
1 package com.lm.myapplication;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.os.Build;
6 import android.os.Bundle;
7 import android.webkit.WebSettings;
8 import android.webkit.WebView;
9
10 public class MainActivity extends AppCompatActivity {
11
12     private String url = "http://iot.kincony.com:1880/ui/#!/0?socketId=yJktDF85HVA4B53jAACe";
13
14     private WebView webView;
15
16     @Override
17     protected void onCreate(Bundle savedInstanceState) {
18         super.onCreate(savedInstanceState);
19         setContentView(R.layout.activity_main);
20
21         webView = findViewById(R.id.web_view);
22         WebSettings webSettings = webView.getSettings();
23         webSettings.setJavaScriptEnabled(true);
24         webSettings.setUseWideViewPort(true);
25         webSettings.setLoadWithOverviewMode(true);
26         webSettings.setAllowFileAccess(true);
27         webSettings.setAllowContentAccess(true);
28         webSettings.setJavaScriptCanOpenWindowsAutomatically(true);
29         webSettings.setBlockNetworkImage(false);
30         webSettings.setTextZoom(100);
31         webSettings.setSupportZoom(true);
32         webSettings.setBuiltInZoomControls(false);
33         webSettingssetDisplayZoomControls(false);
34         webSettings.setDefaultTextEncodingName("UTF-8");
35         if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.JELLY_BEAN) {
36             webSettings.setAllowFileAccessFromFileURLs(true);
37             webSettings.setAllowUniversalAccessFromFileURLs(true);
38         }
39         if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.OLLLIPOP) {
40             webSettings.setMixedContentMode(WebSettings.MIXED_CONTENT_ALWAYS_ALLOW);
41         }
42         webSettings.setDomStorageEnabled(true);
43         webSettings.setAppCacheEnabled(true);
44         webSettings.setCacheMode(WebSettings.LOAD_DEFAULT);
```

Build Variants 2 Favorites 2 Structure

Event Log Layout Inspector

44:1 CRLF UTF-8 4 spaces

Code list:

```
import android.os.Build;
import android.os.Bundle;
import android.webkit.WebSettings;
import android.webkit.WebView;

private WebView webView;

webView = findViewById(R.id.web_view);
WebSettings webSettings = webView.getSettings();
webSettings.setJavaScriptEnabled(true);
webSettings.setUseWideViewPort(true);
webSettings.setLoadWithOverviewMode(true);
webSettings.setAllowFileAccess(true);
webSettings.setAllowContentAccess(true);
webSettings.setJavaScriptCanOpenWindowsAutomatically(true);
webSettings.setBlockNetworkImage(false);
webSettings.setTextZoom(100);
webSettings.setSupportZoom(true);
webSettings.setBuiltInZoomControls(false);
webSettingssetDisplayZoomControls(false);
webSettings.setDefaultTextEncodingName("UTF-8");
if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.JELLY_BEAN) {
    webSettings.setAllowFileAccessFromFileURLs(true);
    webSettings.setAllowUniversalAccessFromFileURLs(true);
}
if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
    webSettings.setMixedContentMode(WebSettings.MIXED_CONTENT_ALWAYS_ALLOW);
}
webSettings.setDomStorageEnabled(true);
```

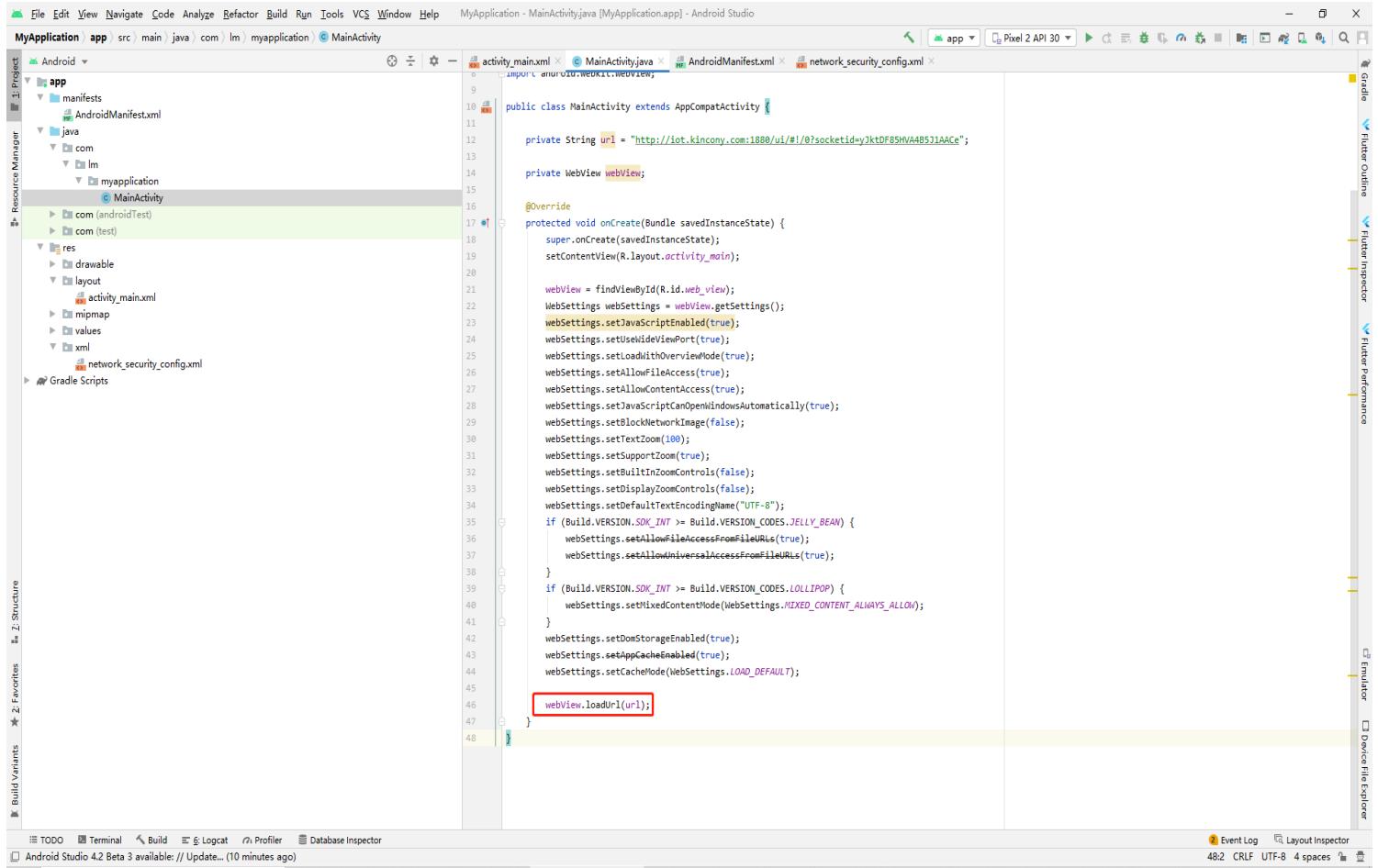
```

webSettings.setAppCacheEnabled(true);

webView.loadUrl(url);

```

Config completed.



The screenshot shows the Android Studio interface with the project 'MyApplication' open. The main editor window displays the `MainActivity.java` file. The line `webView.loadUrl(url);` is highlighted with a red rectangle. The code in the file is as follows:

```

public class MainActivity extends AppCompatActivity {

    private String url = "http://iot.kincony.com:1880/ui/#/?socketId=y3ktDF8SHVA4B5J1A4Ce";

    private WebView webView;

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

        webView = findViewById(R.id.web_view);
        WebSettings webSettings = webView.getSettings();
        webSettings.setJavaScriptEnabled(true);
        webSettings.setUseWideViewPort(true);
        webSettings.setLayoutAlgorithm(LayoutAlgorithm.NATIVE);
        webSettings.setAllowFileAccess(true);
        webSettings.setAllowContentAccess(true);
        webSettings.setJavaScriptCanOpenWindowsAutomatically(true);
        webSettings.setBlockNetworkImage(false);
        webSettings.setTextZoom(100);
        webSettings.setSupportZoom(true);
        webSettings.setBuiltInZoomControls(false);
        webSettingssetDisplayZoomControls(false);
        webSettings.setDefaultTextEncodingName("UTF-8");
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.JELLY_BEAN) {
            webSettings.setAllowFileAccessFromFileURLs(true);
            webSettings.setAllowUniversalAccessFromFileURLs(true);
        }
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
            webSettings.setMixedContentMode(WebSettings.MIXED_CONTENT_ALWAYS_ALLOW);
        }
        webSettings.setDomStorageEnabled(true);
        webSettings.setAppCacheEnabled(true);
        webSettings.setCacheMode(webSettings.LOAD_DEFAULT);

        webView.loadUrl(url);
    }
}

```

Return button and click event

Open “MainActivity” add these code:

```

1 package com.lm.myapplication;
2
3 import android.os.Build;
4 import android.os.Bundle;
5 import android.webkit.WebSettings;
6 import android.webkit.WebView;
7
8 public class MainActivity extends AppCompatActivity {
9
10    private String url = "http://iot.kincony.com:1888/ui/#/socketid=y3ktDF8SHVA4BS31AAce";
11
12    private WebView webView;
13
14    @Override
15        public void onBackPressed() {
16            if (webView.canGoBack()) {
17                webView.goBack();
18            } else {
19                super.onBackPressed();
20            }
21
22        }
23
24    @Override
25        protected void onCreate(Bundle savedInstanceState) {
26            super.onCreate(savedInstanceState);
27            setContentView(R.layout.activity_main);
28
29            webView = findViewById(R.id.web_view);
30            WebSettings webSettings = webView.getSettings();
31            webSettings.setJavaScriptEnabled(true);
32            webSettings.setUseWideViewPort(true);
33            webSettings.setLoadWithOverviewMode(true);
34            webSettings.setAllowFileAccess(true);
35            webSettings.setAllowContentAccess(true);
36            webSettings.setJavaScriptCanOpenWindowsAutomatically(true);
37            webSettings.setBlockNetworkImage(false);
38            webSettings.setTextZoom(100);
39            webSettings.setSupportZoom(true);
40            webSettings.setBuiltInZoomControls(false);
41            webSettingssetDisplayZoomControls(false);
42            webSettings.setDefaultTextEncodingName("UTF-8");
43
44            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.JELLY_BEAN) {
45                webSettings.setAllowFileAccessFromFileURLs(true);
46                webSettings.setAllowUniversalAccessFromFileURLs(true);
47            }
48
49        }
50
51    }
52
53
54
55
56
57
58
59
5

```

Code list:

@Override

```

public void onBackPressed() {
    if(webView.canGoBack()) {
        webView.goBack();
    } else {
        super.onBackPressed();
    }
}

```

Debug

Click Build-->Build Bunle(s)/APK(s)-->Build APK(s) Compile the project into an APK file

The screenshot shows the Android Studio interface with the project 'MyApplication' open. The 'Build' menu is highlighted, and the 'Build APK(s)' option is selected. A red box highlights this menu path. The code editor shows the MainActivity.java file with Java code for a webview application. The bottom status bar indicates 'Android Studio 4.2 Beta 3 available // Update... (20 minutes ago)'. The bottom right corner shows the event log with 'Event Log' and 'Layout Inspector' tabs.

```

    package com.lm.myapplication;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Build;
import android.os.Bundle;
import android.webkit.WebSettings;
import android.webkit.WebView;

public class MainActivity extends AppCompatActivity {

    private String url = "http://iot.kincony.com:1880/ui/#!/socketid=yJktDF85HVAAB5j1AAce";

    @Override
    public void onBackPressed() {
        if (webView.canGoBack()) {
            webView.goBack();
        } else {
            super.onBackPressed();
        }
    }

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

        webView = findViewById(R.id.web_view);
        WebSettings webSettings = webView.getSettings();
        webSettings.setJavaScriptEnabled(true);
        webSettings.setUseWideViewPort(true);
        webSettings.setLoadWithOverviewMode(true);
        webSettings.setAllowFileAccess(true);
        webSettings.setAllowContentAccess(true);
        webSettings.setJavaScriptCanOpenWindowsAutomatically(true);
        webSettings.setBlockNetworkImage(false);
        webSettings.setTextSize(100);
        webSettings.setSupportZoom(true);
        webSettings.setBuiltInZoomControls(false);
        webSettingssetDisplayZoomControls(false);
        webSettings.setDefaultTextEncodingName("UTF-8");
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.JELLY_BEAN) {
            webSettings.setAllowFileAccessFromFileURLs(true);
            webSettings.setAllowUniversalAccessFromFileURLs(true);
        }
    }
}

```

Waiting Android Studio Compile, it will takes about 1 minute.

The screenshot shows the Android Studio interface with the project 'MyApplication' open. The 'Build' tab is selected, and the 'Sync' button is highlighted. A red box highlights the 'Build: Sync' button. Below it, the 'Build Output' section shows the build tasks and their execution times. A red box highlights the first task, 'Build: Run tasks...'. The bottom status bar indicates 'Executing tasks: [app:assembleDebug] in project C:\Users\wengw\Desktop\MyApplication (moments ago)'. The bottom right corner shows the event log with 'Event Log' and 'Layout Inspector' tabs, and a red box highlights the progress bar.

Task	Time
app:generateDebugResources	4 s
app:preCompileDebug	9 s
app:processDebugManifestForPackage	641 ms
app:mergeDebugJavaResource	323 ms
app:checkDebugDuplicateClasses	193 ms
app:mergeDebugNativeLibs	17 ms

After compiling, a prompt will pop up. Click the “locate” button on the prompt to directly open the APK file directory

The screenshot shows the Android Studio interface. The code editor displays `MainActivity.java` with Java code for a webview application. Below the code editor is the build log window, which shows a successful build process. A red box highlights a message in the bottom right corner: "Build APK(s) APK(s) generated successfully for 1 module: Module 'MyApplication.app' locate or...".

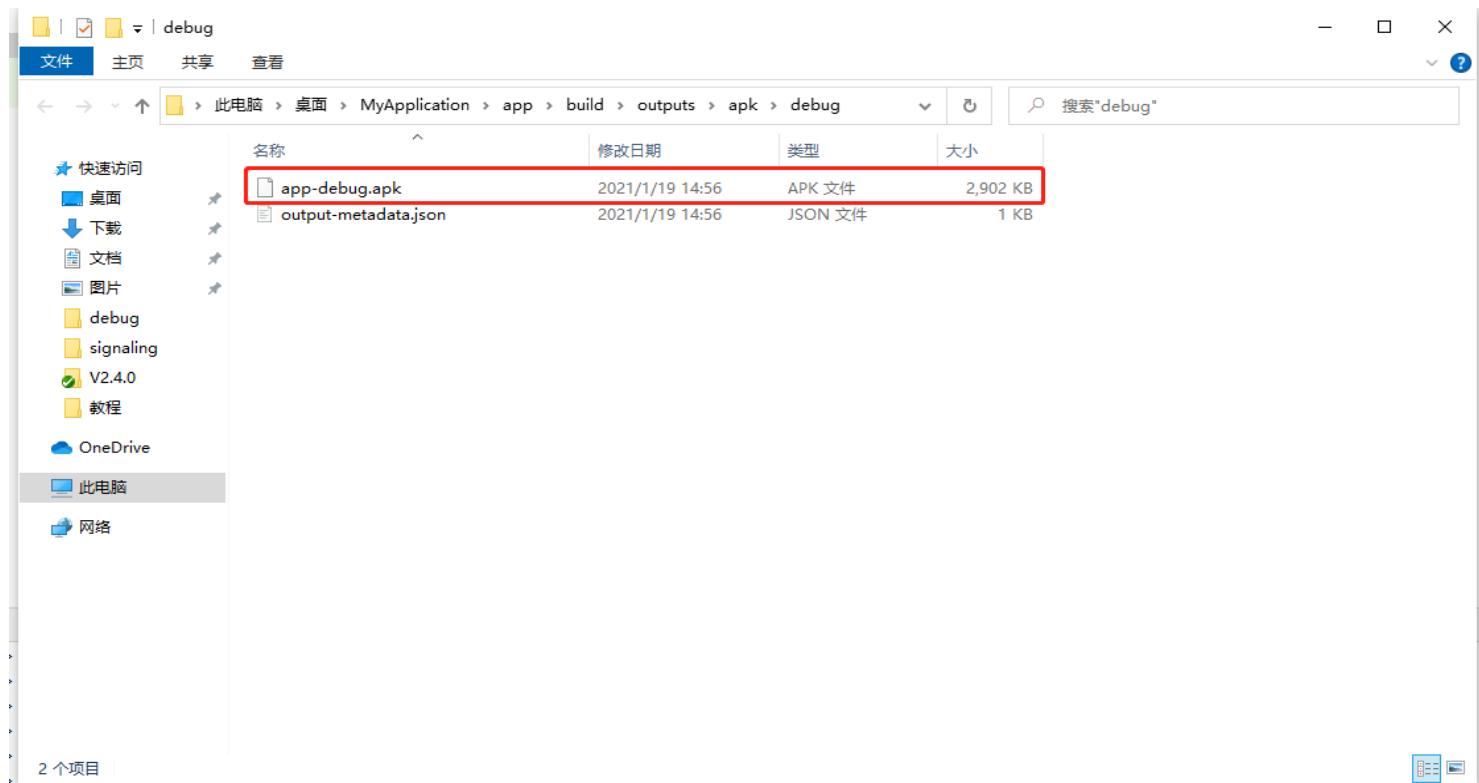
```

1 package com.lm.myapplication;
2
3 import android.os.Build;
4 import android.os.Bundle;
5 import android.webkit.WebSettings;
6 import android.webkit.WebView;
7
8
9 public class MainActivity extends AppCompatActivity {
10
11     private String url = "http://iot.kincony.com:1880/ui#!/@?socketId=yJktDF85H/A4B5Jla6Ce";
12
13     private WebView webView;
14
15     @Override
16     public void onBackPressed() {
17         if (webView.canGoBack()) {
18             webView.goBack();
19         } else {
20             super.onBackPressed();
21         }
22     }
23
24
25     @Override
26     protected void onCreate(Bundle savedInstanceState) {
27         super.onCreate(savedInstanceState);
28         setContentView(R.layout.activity_main);
29
30         webView = findViewById(R.id.web_view);

```

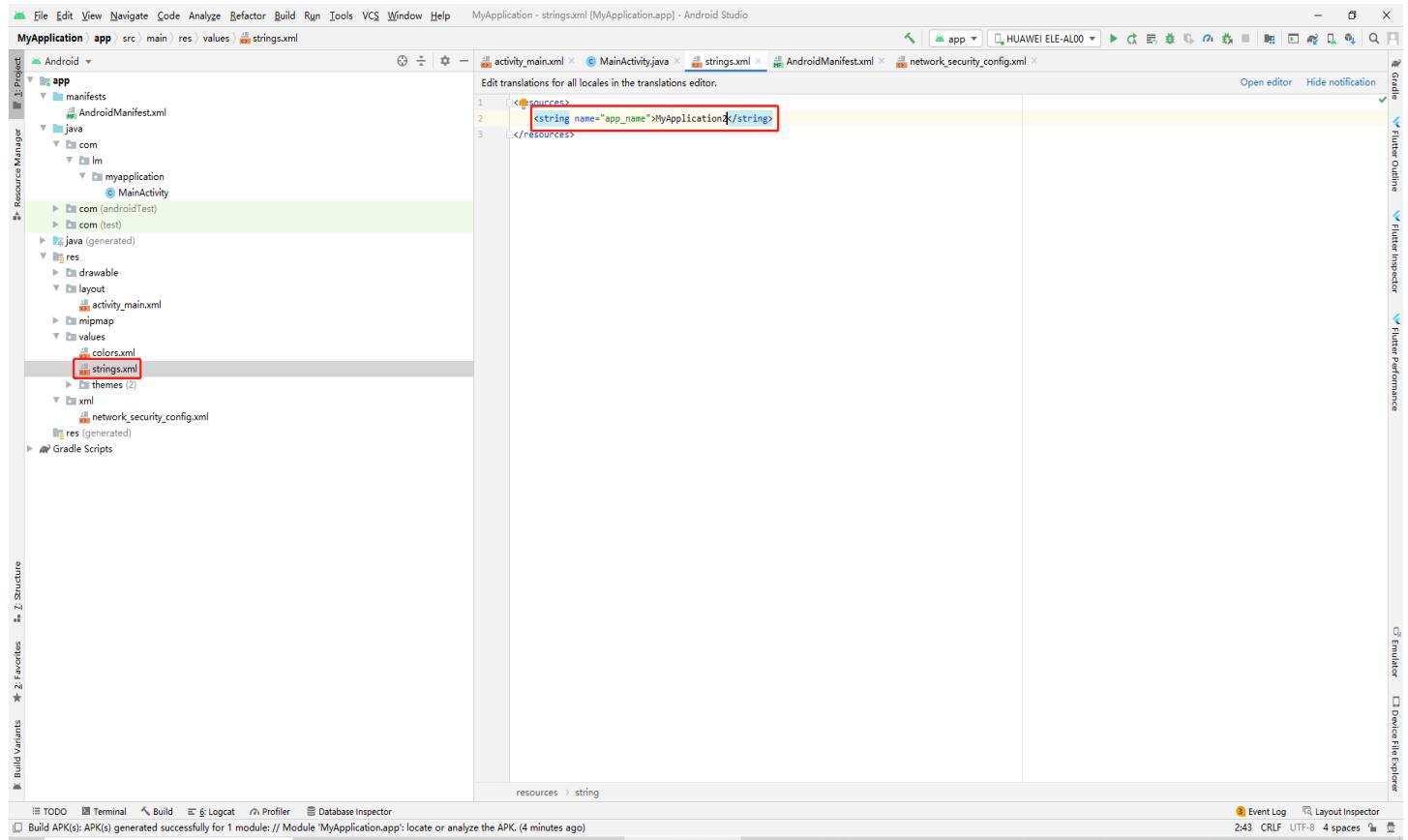
If there is no pop-up prompt, you can also search under the save location path defined before.

"Save location" path: `app/build/outputs/apk/debug`



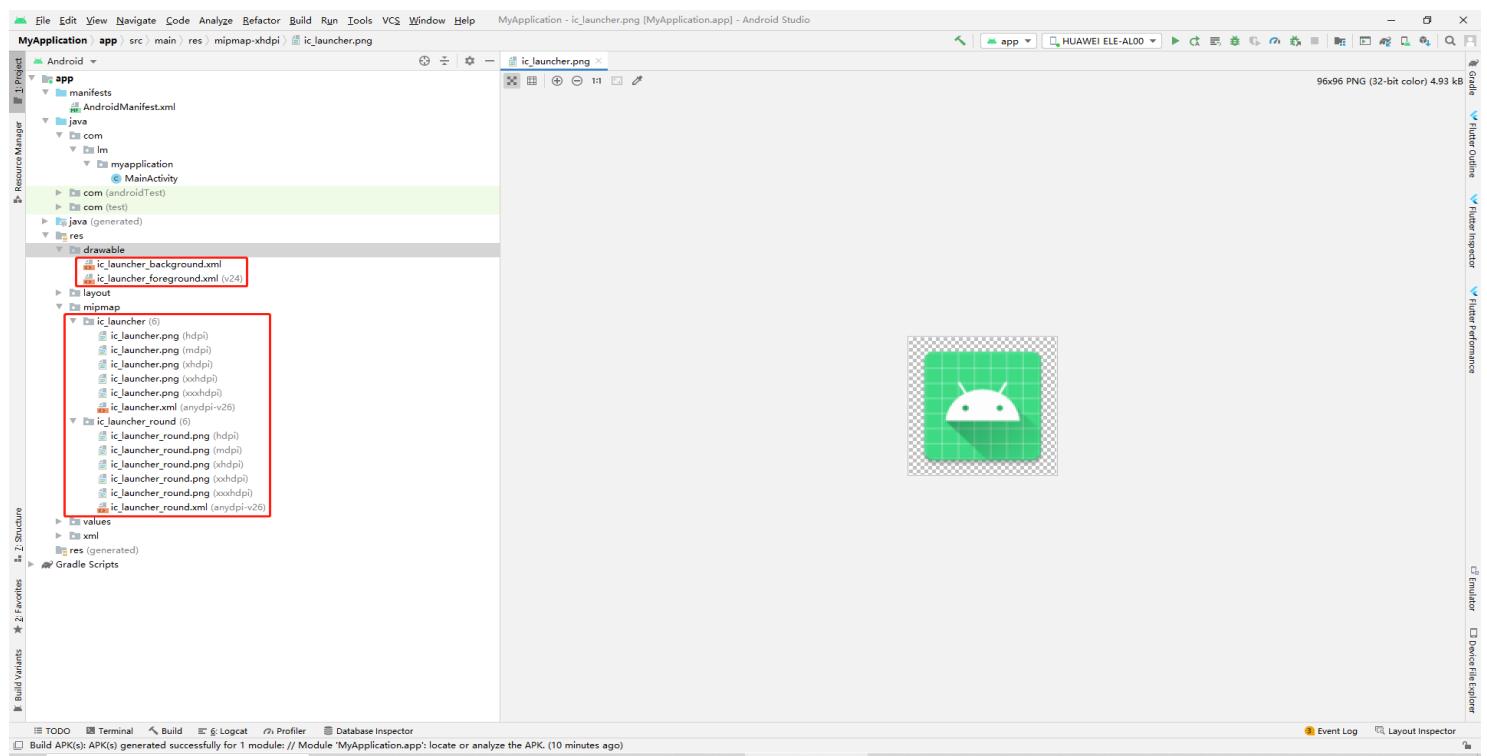
Modify application name

Open res/values/string.xml, modify app_name. Now our application name is "MyApplication2"

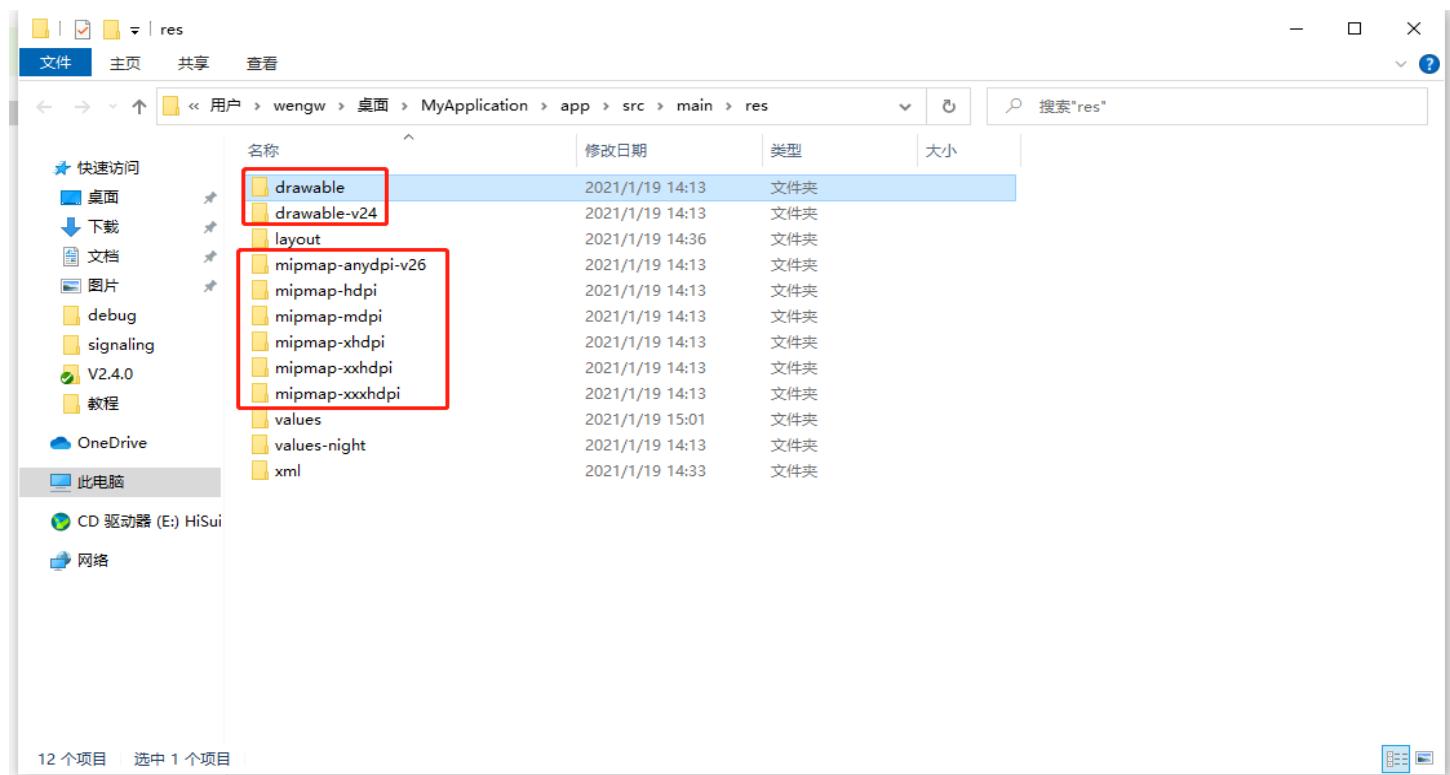


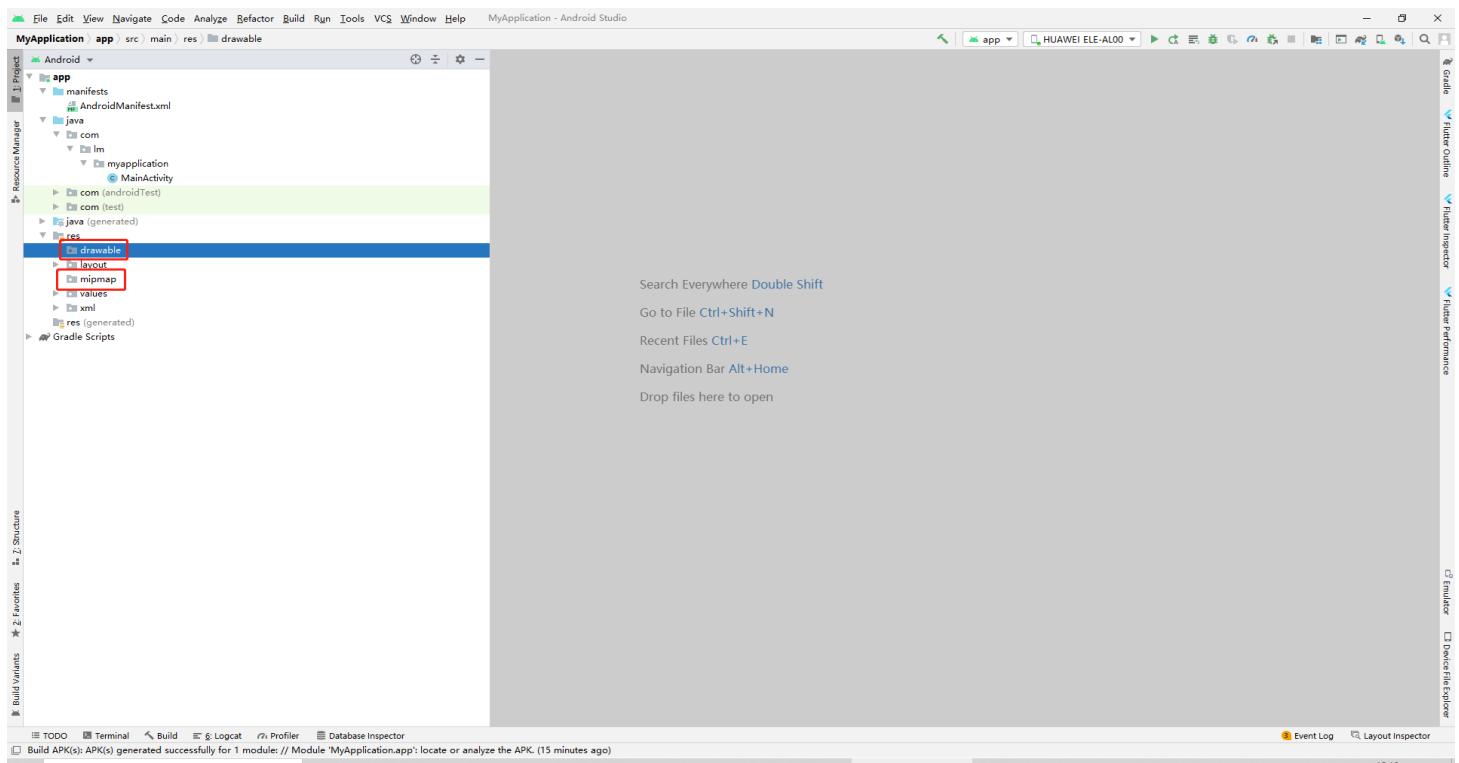
Modify application ico

"res/drawable" and "res/mipmap" folder, Here is the icon generated by Android studio for us by default. We can replace it to make the icon our own. IC_launcher.png Represents a square icon, IC_launcher_round.png Represents a circular icon, which use for different mobile phones.

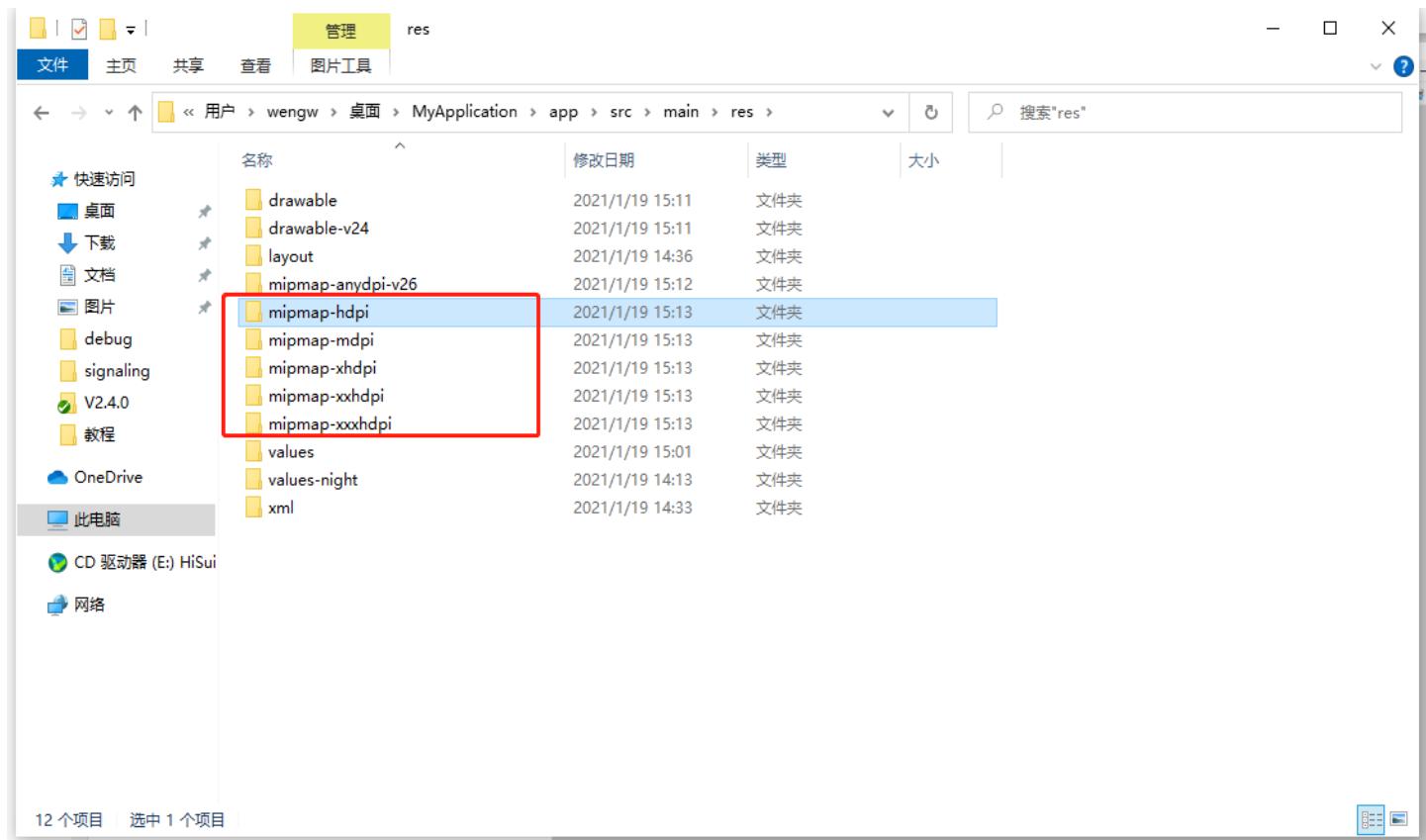


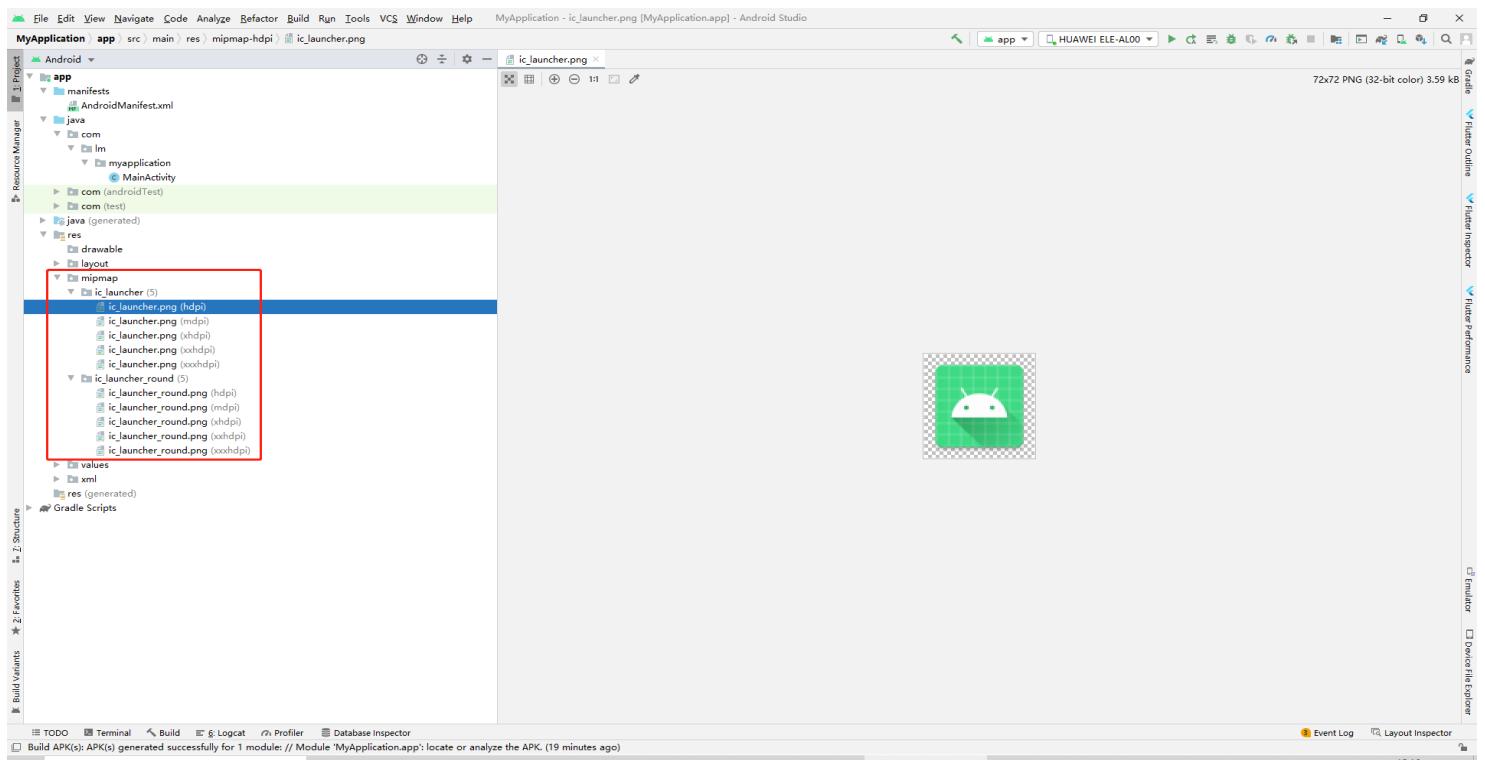
Open Save location: app/src/main/res folder, delete default folder.





Copy our new file and folder to this path

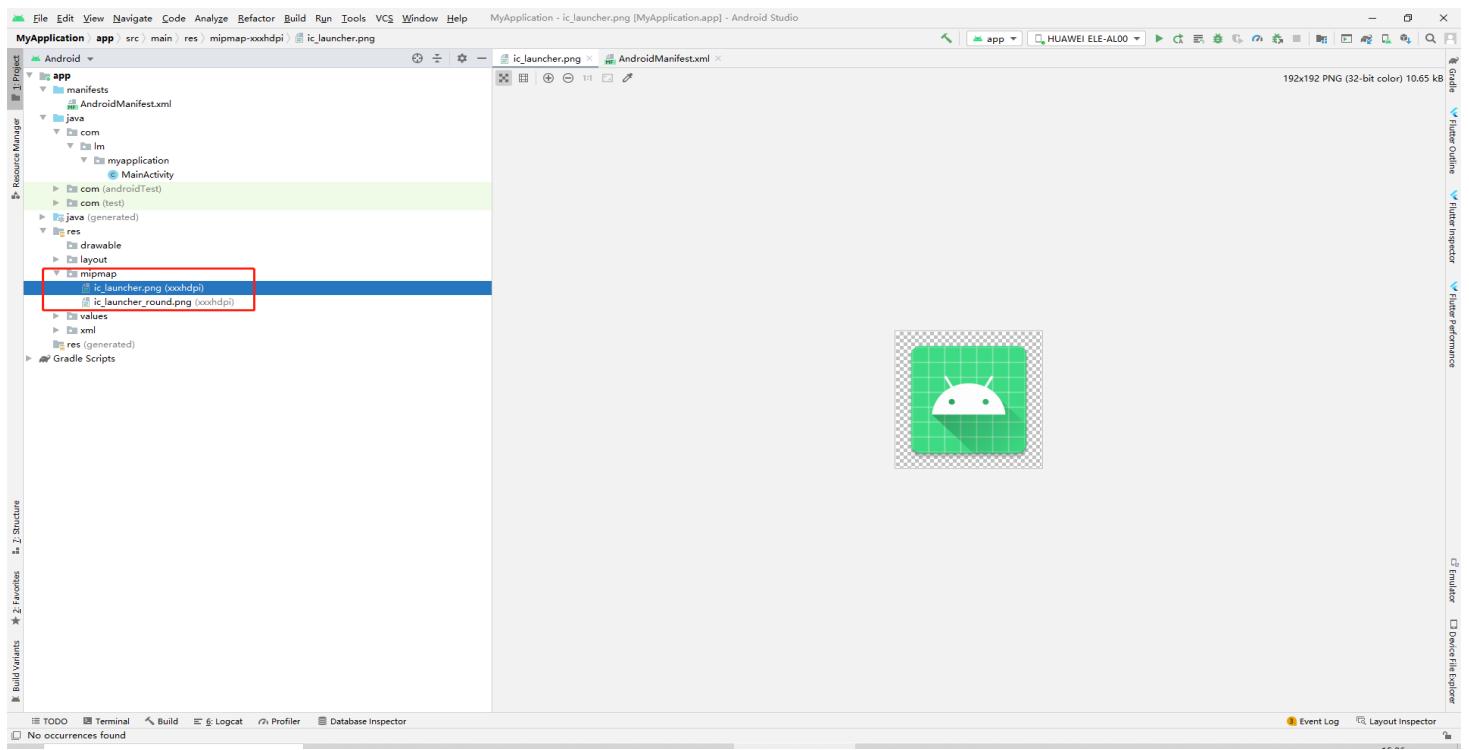




It should be noted that different resolutions are required for different folders.

folder	ico resolutions
<code>mipmap-hdpi</code>	72*72
<code>mipmap-mdpi</code>	48*48
<code>mipmap-xhdpi</code>	96*96
<code>mipmap-xxhdpi</code>	144*144
<code>mipmap-xxxhdpi</code>	192*192

If you not make all types of ico, you only need make ico for "`mipmap-xxhdpi`" or "`mipmap-xxxhdpi`" folder is ok.



After change the ico file, click “Build-->Build Bunle(s)/APK(s)-->Build APK(s)” recompile the project to APK file.