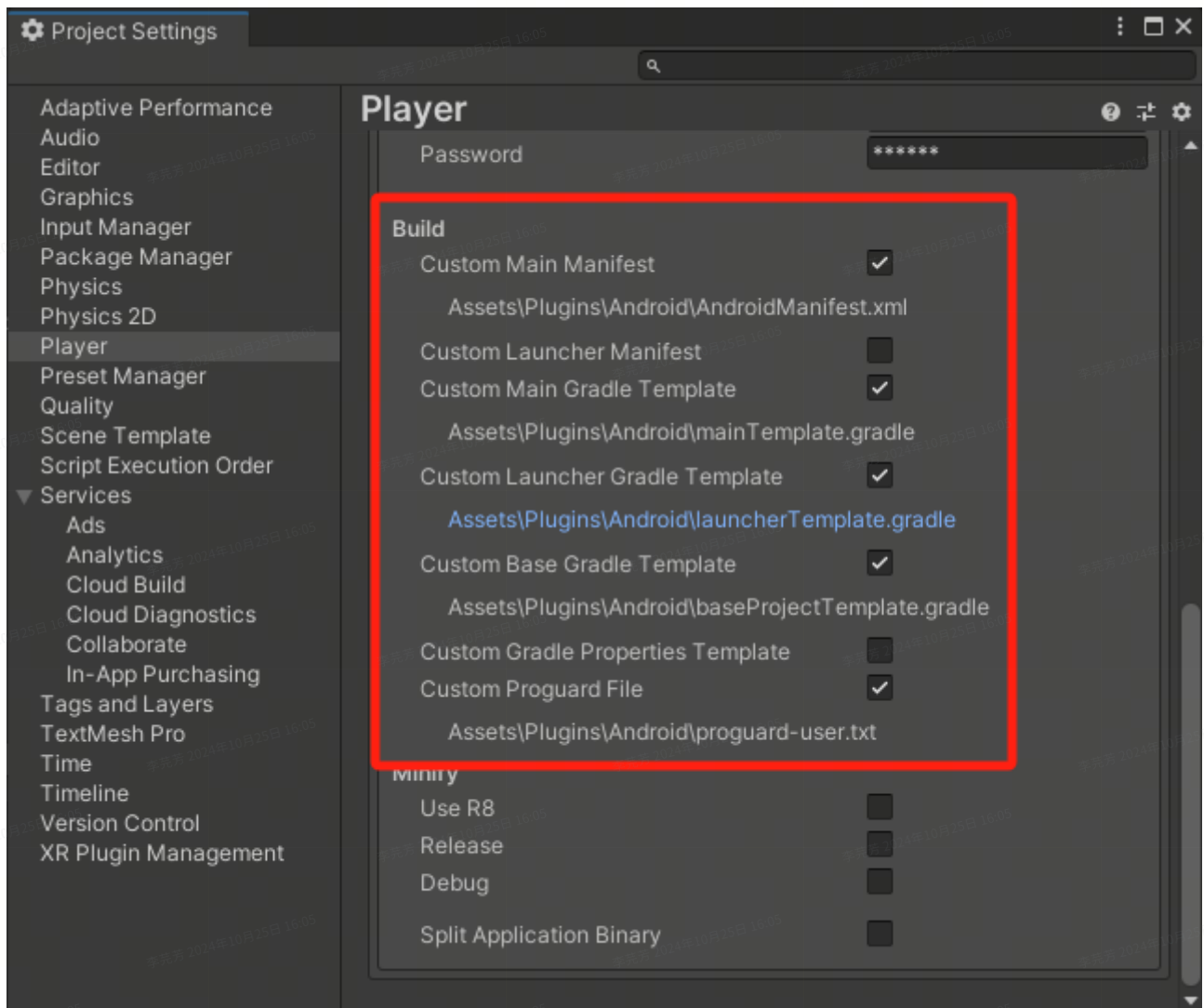


# BrainX Max Mediation-Unity Integration Guide

## Steps for Integrating with Unity Project:

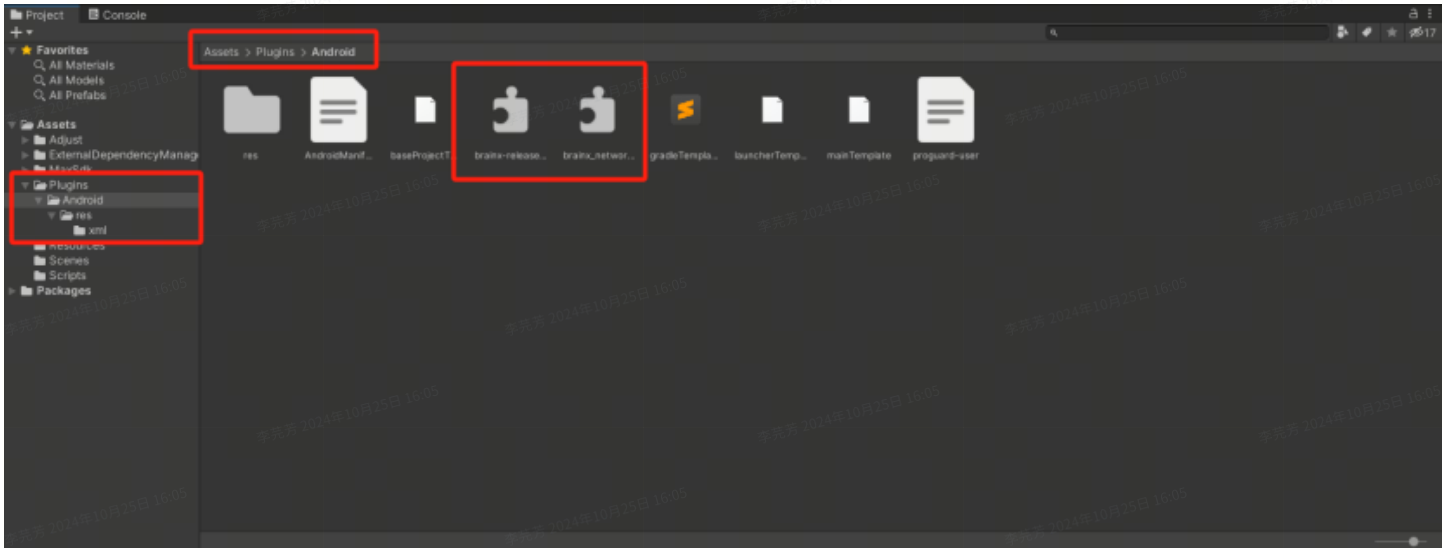
1. **Follow the Max Unity Integration Guide:** Access the official Max Unity integration guide using the following link: [Max Unity Integration Guide](#)
2. **Configure Unity:**
  - Open Unity, go to **File** > **Build Settings** > **Player Settings**. Under **Player**, select the **Android** tab.
  - Check the required configurations.



## 3. Add Required Files:

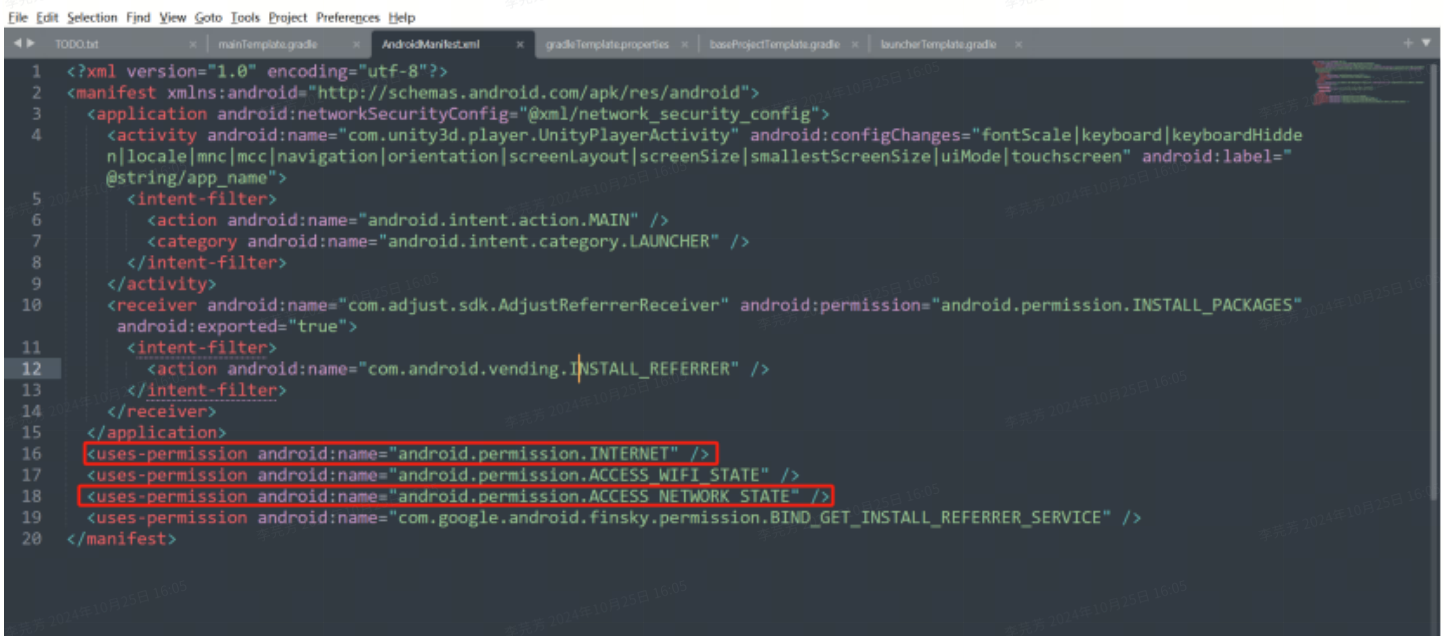
- Place the following files into the **Assets/Plugins/Android** directory:

- brainx\_network\_max-release.aar
- brainx-release\_2.0.0.3.aar



#### 4. Update Android Manifest:

- Open `Assets/Plugins/Android/AndroidManifest.xml` and add the following permissions:
- `<uses-permission android:name="android.permission.INTERNET" />`
- `<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />`



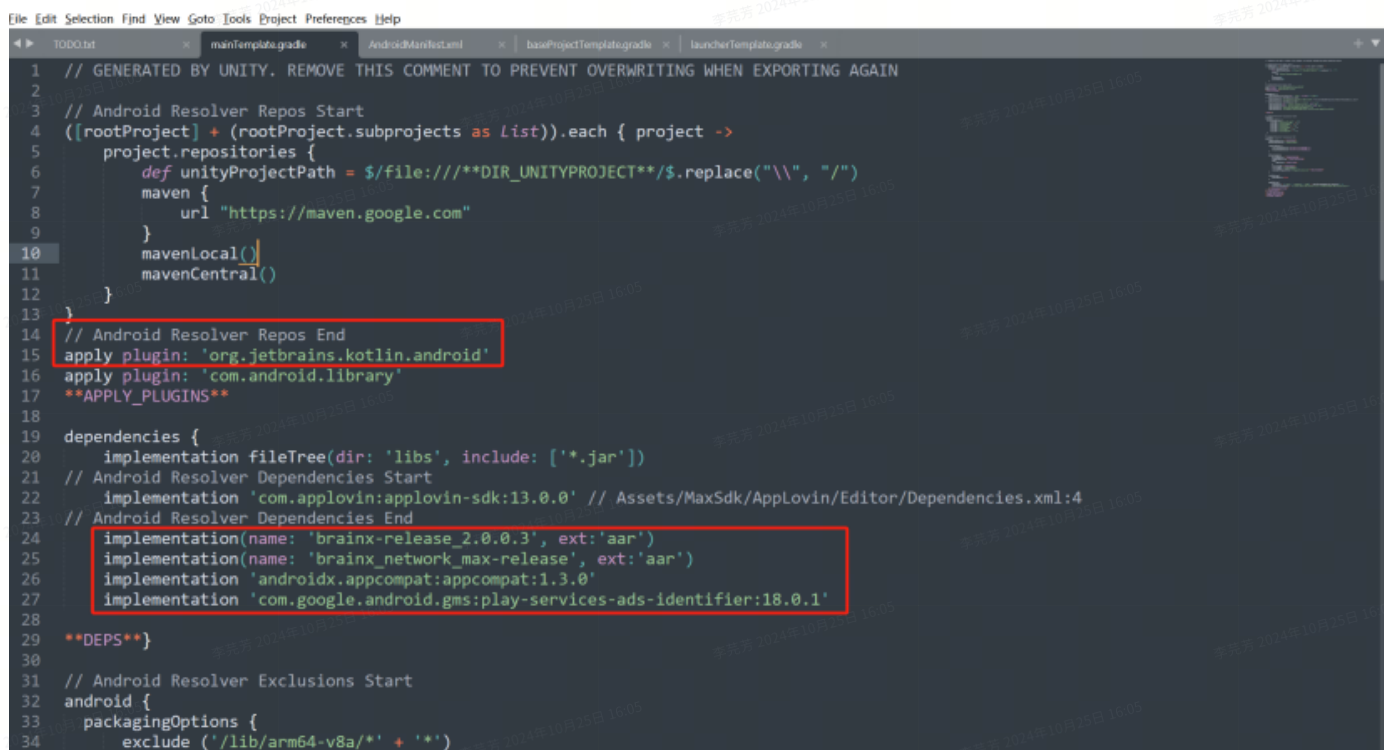
#### 5. Open `Assets\Plugins\Android\mainTemplate.gradle`.

- a. Add the following lines:

- Above `apply plugin: 'com.android.library'`, insert:
- `apply plugin: 'org.jetbrains.kotlin.android'`

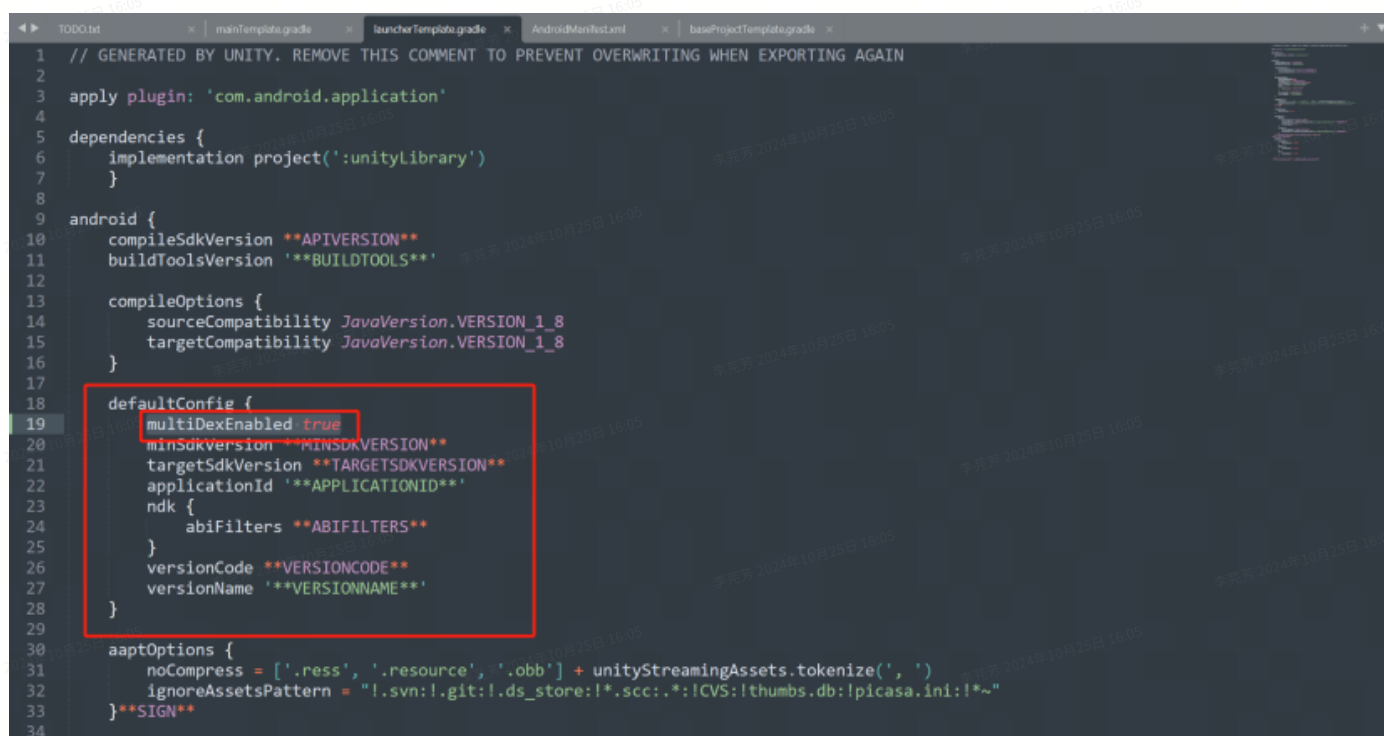
b. In the dependencies section, add:

- `implementation(name: 'brainx-release_2.0.0.3', ext:'aar')`
- `implementation(name: 'brainx_network_max-release', ext:'aar')`
- `implementation 'androidx.appcompat:appcompat:1.3.0'`
- `implementation 'com.google.android.gms:play-services-ads-identifier:18.0.1'`



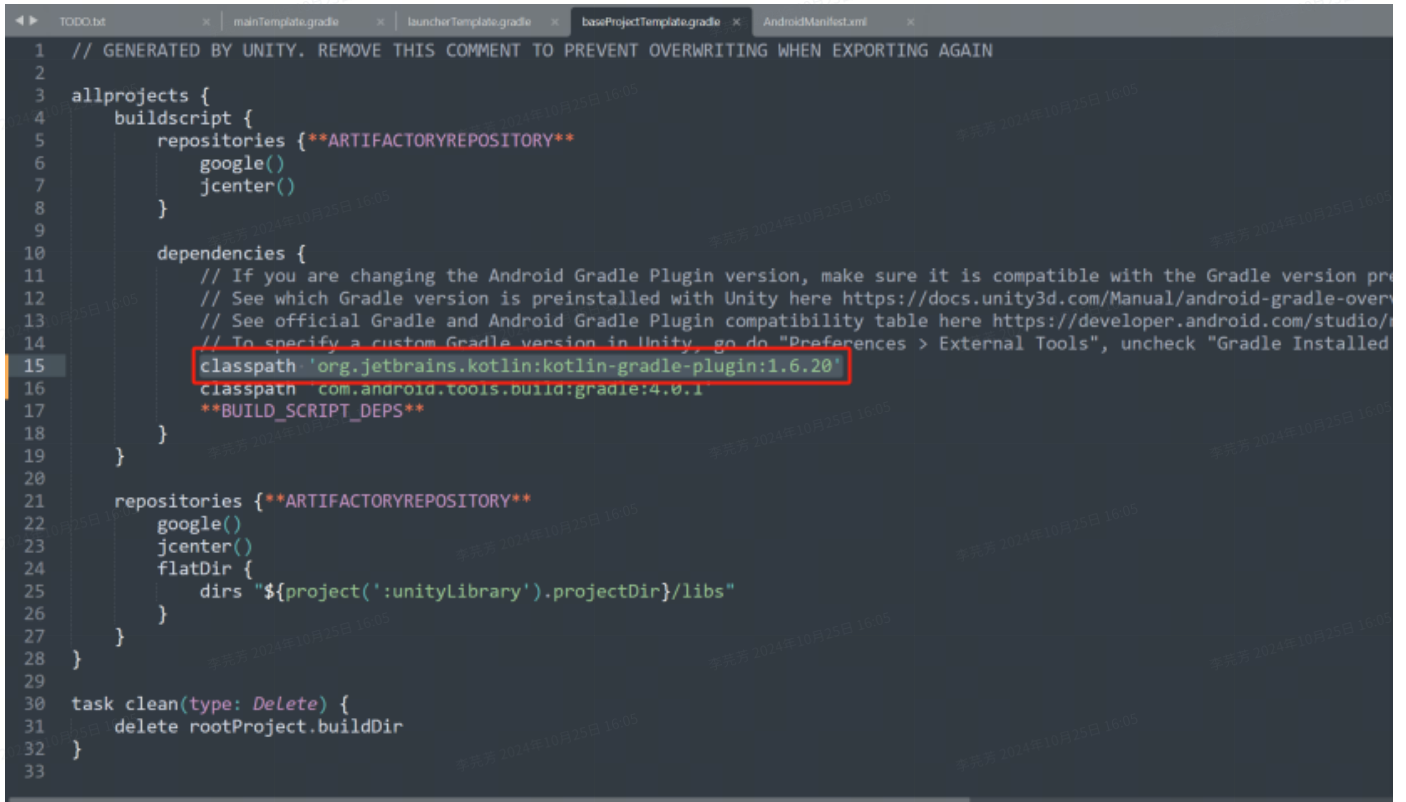
```
1 // GENERATED BY UNITY. REMOVE THIS COMMENT TO PREVENT OVERWRITING WHEN EXPORTING AGAIN
2
3 // Android Resolver Repos Start
4 ([rootProject] + (rootProject.subprojects as List)).each { project ->
5     project.repositories {
6         def unityProjectPath = $/file:///**DIR_UNITYPROJECT**/$.replace("\\", "/")
7         maven {
8             url "https://maven.google.com"
9         }
10        mavenLocal()
11        mavenCentral()
12    }
13 }
14 // Android Resolver Repos End
15 apply plugin: 'org.jetbrains.kotlin.android'
16 apply plugin: 'com.android.library'
17 **APPLY_PLUGINS**
18
19 dependencies {
20     implementation fileTree(dir: 'libs', include: ['*.jar'])
21     // Android Resolver Dependencies Start
22     implementation 'com.applovin:applovin-sdk:13.0.0' // Assets/MaxSdk/AppLovin/Editor/Dependencies.xml:4
23     // Android Resolver Dependencies End
24     implementation(name: 'brainx-release_2.0.0.3', ext:'aar')
25     implementation(name: 'brainx_network_max-release', ext:'aar')
26     implementation 'androidx.appcompat:appcompat:1.3.0'
27     implementation 'com.google.android.gms:play-services-ads-identifier:18.0.1'
28 }
29 **DEPS**
30
31 // Android Resolver Exclusions Start
32 android {
33     packagingOptions {
34         exclude ('/lib/arm64-v8a/*' + '*')
```

6. Open `Assets\Plugins\Android\launcherTemplate.gradle` and add `{multiDexEnabled true}` in `defaultConfig`



```
1 // GENERATED BY UNITY. REMOVE THIS COMMENT TO PREVENT OVERWRITING WHEN EXPORTING AGAIN
2
3 apply plugin: 'com.android.application'
4
5 dependencies {
6     implementation project(':unityLibrary')
7 }
8
9 android {
10     compileSdkVersion **APIVERSION**
11     buildToolsVersion '**BUILDTOOLS**'
12
13     compileOptions {
14         sourceCompatibility JavaVersion.VERSION_1_8
15         targetCompatibility JavaVersion.VERSION_1_8
16     }
17
18     defaultConfig {
19         multiDexEnabled true
20         minSdkVersion **MINSDKVERSION**
21         targetSdkVersion **TARGETSDKVERSION**
22         applicationId '**APPLICATIONID**'
23         ndk {
24             abiFilters **ABIFILTERS**
25         }
26         versionCode **VERSIONCODE**
27         versionName '**VERSIONNAME**'
28     }
29
30     aaptOptions {
31         noCompress = ['.ress', '.resource', '.obb'] + unityStreamingAssets.tokenize(', ')
32         ignoreAssetsPattern = "!svn:!git:!ds_store:!*.scc!.*:!CVS:!thumbs.db!picasa.ini!*"
33     }
34     **SIGN**
35 }
```

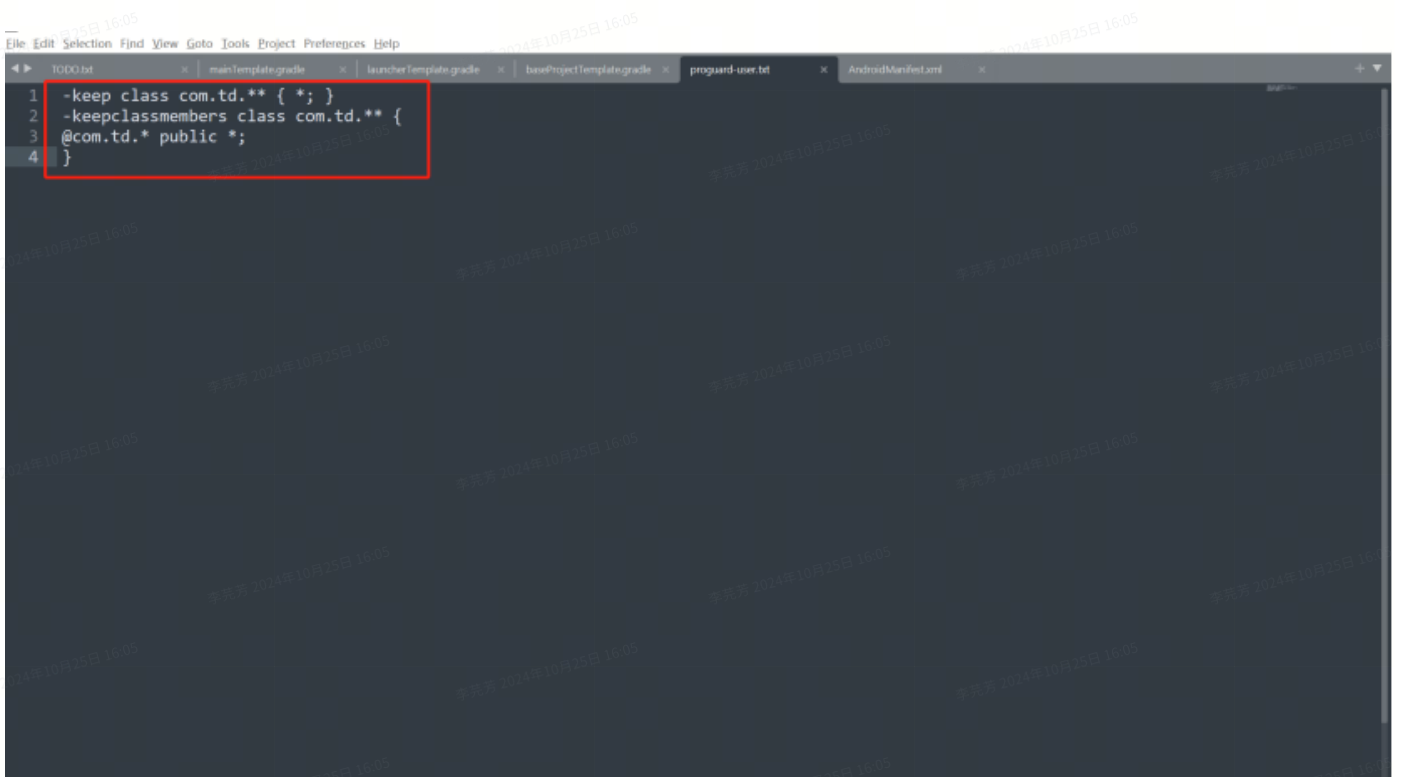
7. Open `Assets\Plugins\Android\baseProjectTemplate.gradle`, add `classpath 'org.jetbrains.kotlin:kotlin-gradle-plugin:1.6.20'` in dependencies



```
1 // GENERATED BY UNITY. REMOVE THIS COMMENT TO PREVENT OVERWRITING WHEN EXPORTING AGAIN
2
3 allprojects {
4     buildscript {
5         repositories {**ARTIFACTORYREPOSITORY**
6             google()
7             jcenter()
8         }
9     }
10
11     dependencies {
12         // If you are changing the Android Gradle Plugin version, make sure it is compatible with the Gradle version provided by Unity.
13         // See which Gradle version is preinstalled with Unity here https://docs.unity3d.com/Manual/android-gradle-overview.html
14         // See official Gradle and Android Gradle Plugin compatibility table here https://developer.android.com/studio/build#compatible-gradle-versions
15         classpath 'org.jetbrains.kotlin:kotlin-gradle-plugin:1.6.20'
16         classpath 'com.android.tools.build:gradle:4.0.1'
17         **BUILD_SCRIPT_DEPS**
18     }
19 }
20
21 repositories {**ARTIFACTORYREPOSITORY**
22     google()
23     jcenter()
24     flatDir {
25         dirs "${project(':unityLibrary').projectDir}/libs"
26     }
27 }
28
29
30 task clean(type: Delete) {
31     delete rootProject.buildDir
32 }
33
```

8. Open `Assets\Plugins\Android\proguard-user.txt` and add the following rules:

- `-keep class com.td.** { *; }`  
`-keepclassmembers class com.td.** {`  
 `@com.td.* public *;`  
`}`



```
1 -keep class com.td.** { *; }
2 -keepclassmembers class com.td.** {
3     @com.td.* public *;
4 }
```

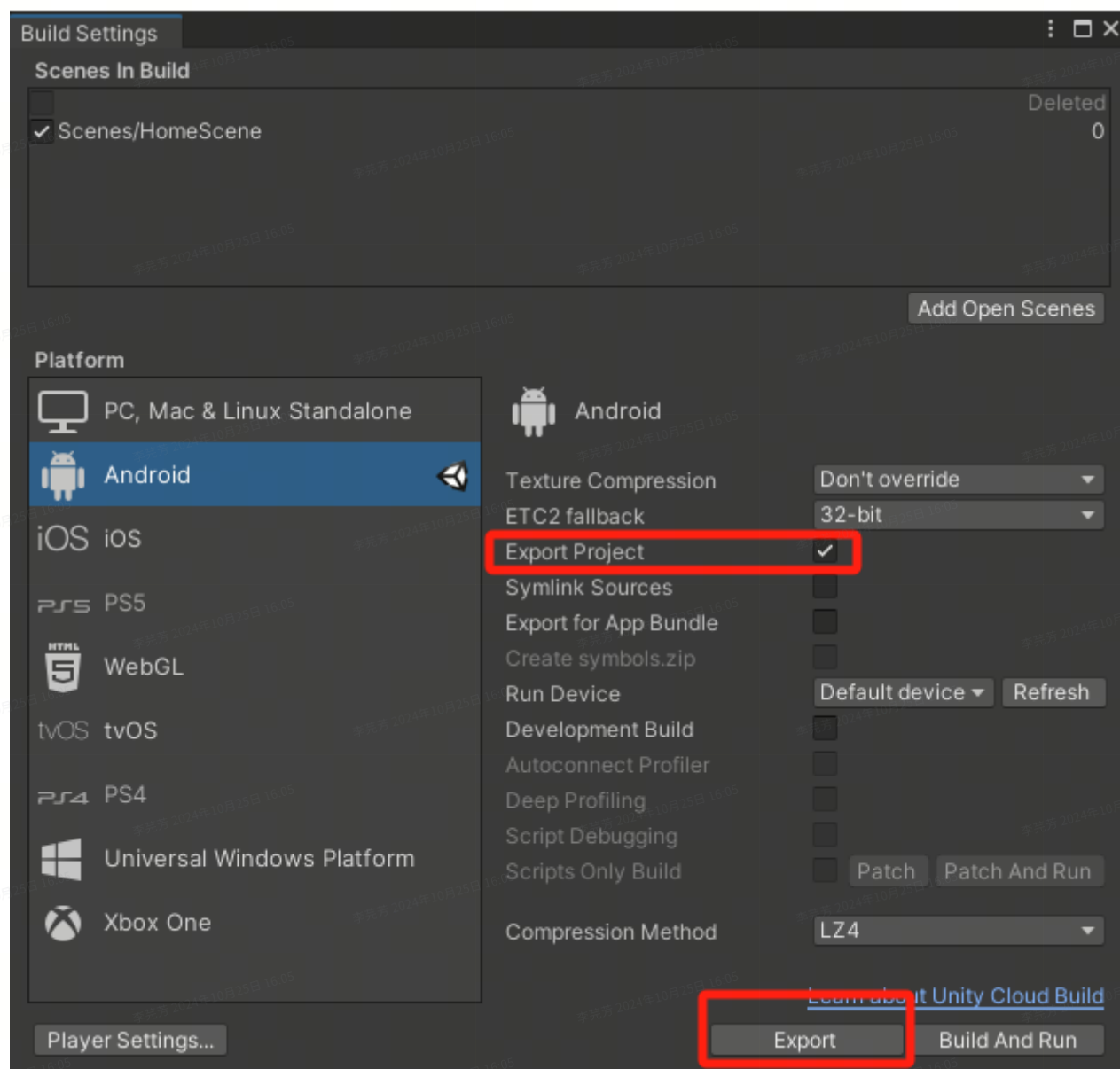
9. **Complete the Configuration:** Once these steps are completed, you can proceed with building the project as usual.

## Alternative Method: Export Unity Project to Android Studio

### 1. Export Unity Project:

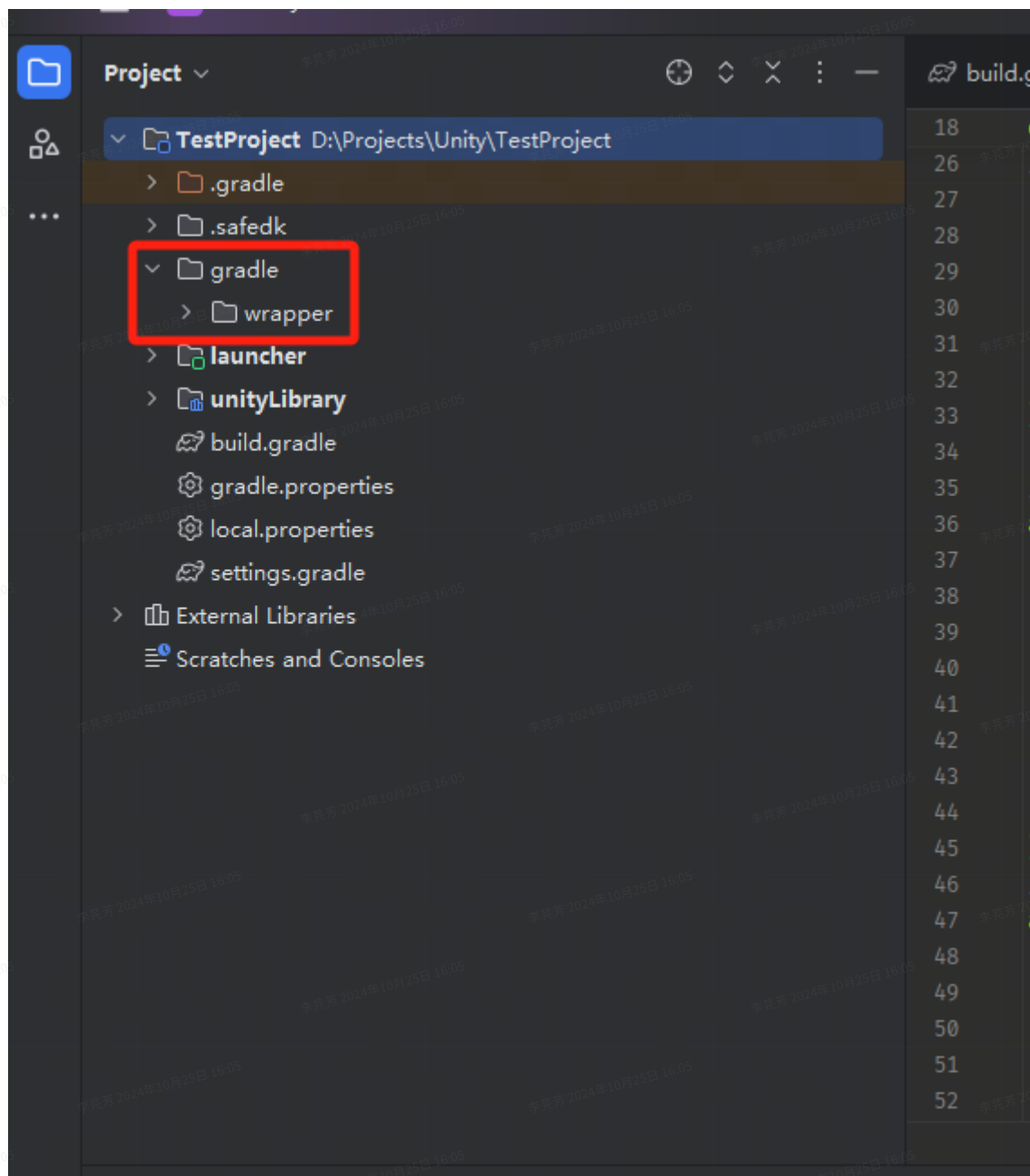
- Follow the Max Unity integration steps as mentioned above.

2. In Unity, go to **File** > **Build Setting**, select **Export Project**, and click **Export**.

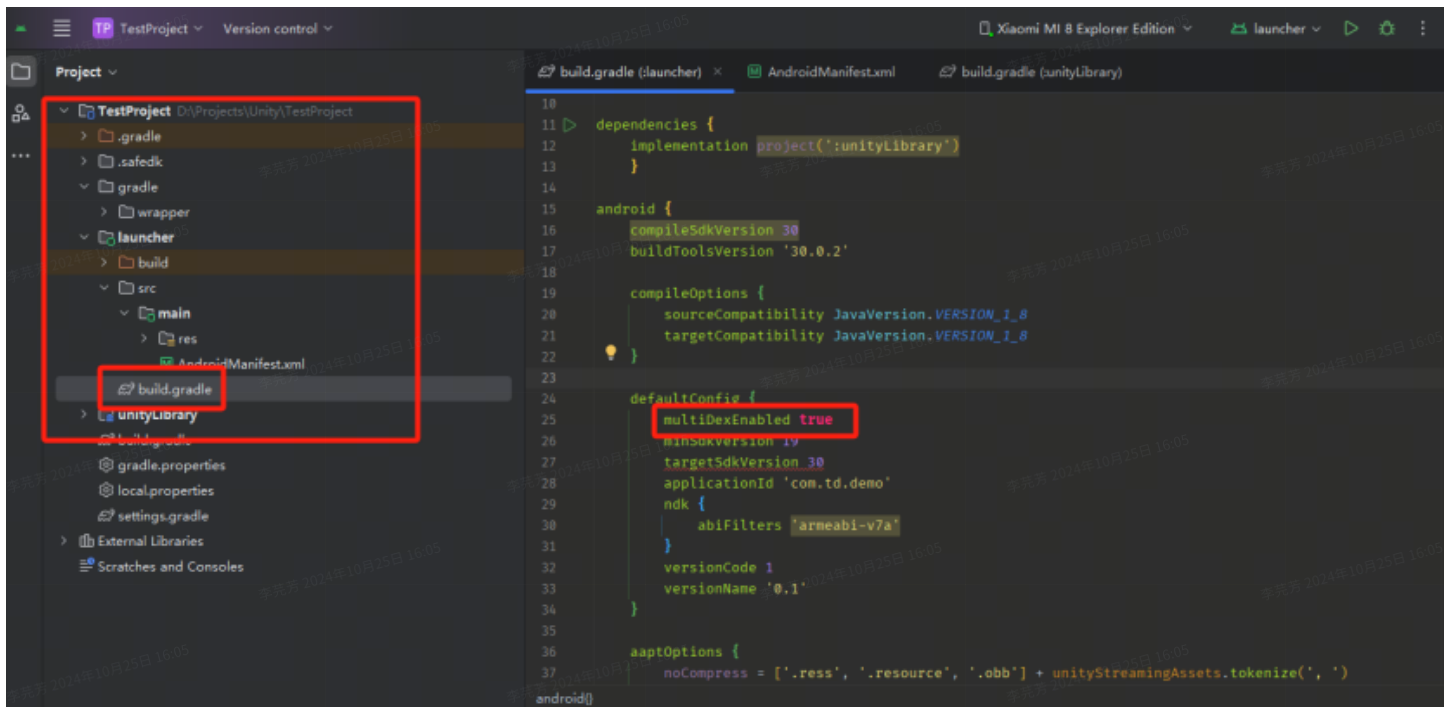


### 3. Open with Android Studio:

- Open the exported project with Android Studio. If the **gradle** folder is missing, extract the provided **gradle.zip** to the project's root directory.



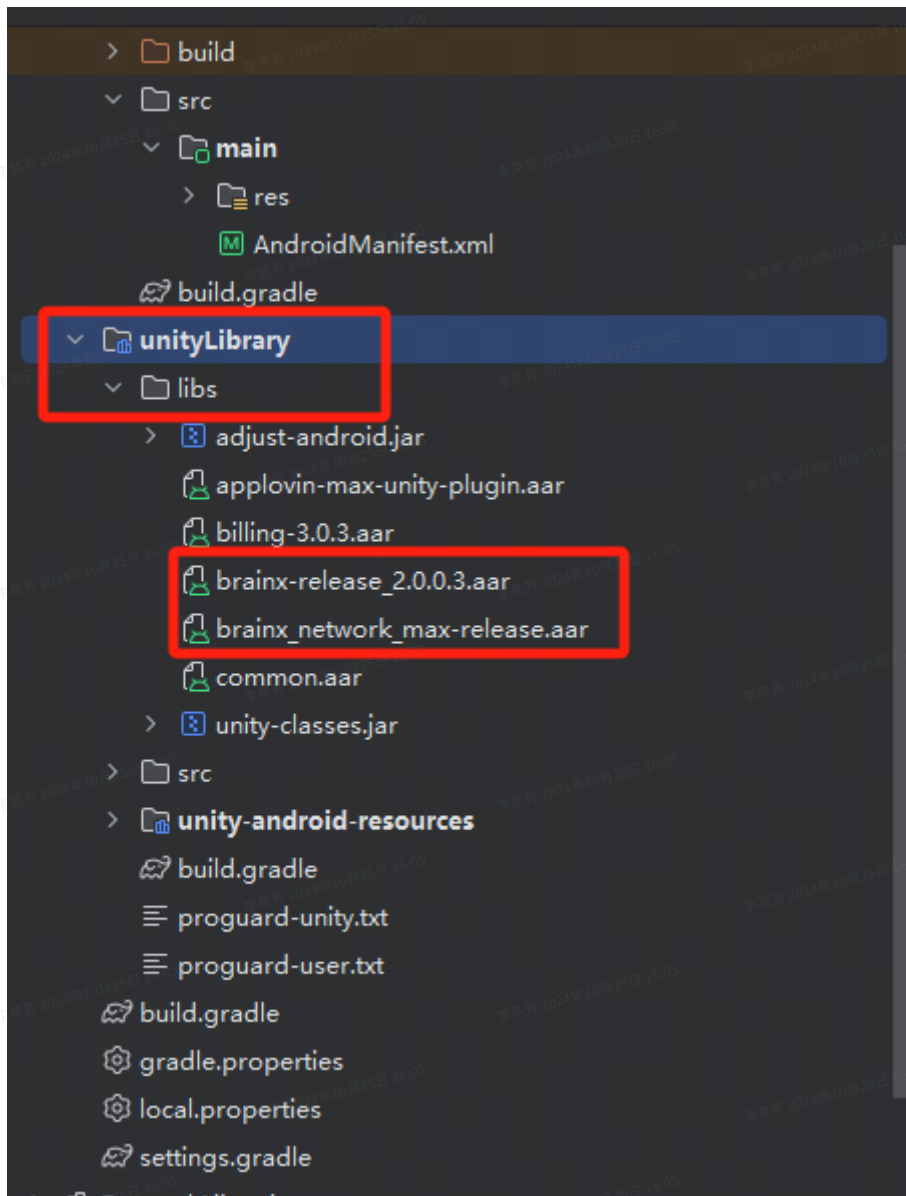
4. Open `launcher\build.gradle` and add: `{multiDexEnabled true}` in `defaultConfig`



## 5. Add Required Files:

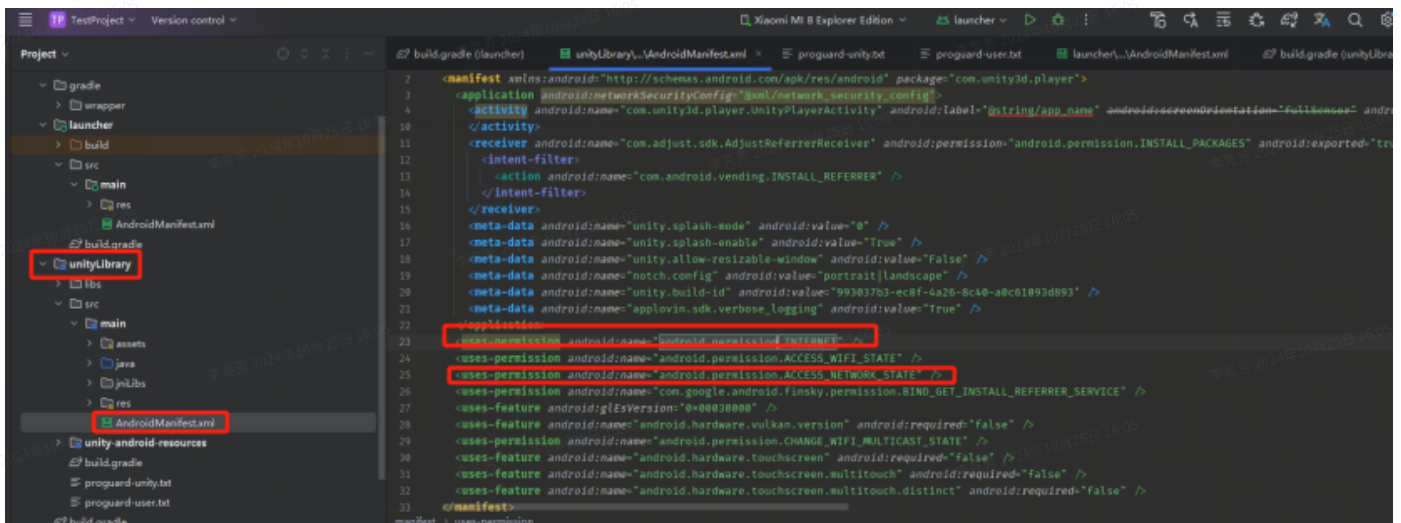
- Place the `brainx_network_max-release.aar` and `brainx-release_2.0.0.3.aar` files into the `unityLibrary\libs` directory.





6. Open `unityLibrary\src\main\AndroidManifest.xml` and add:

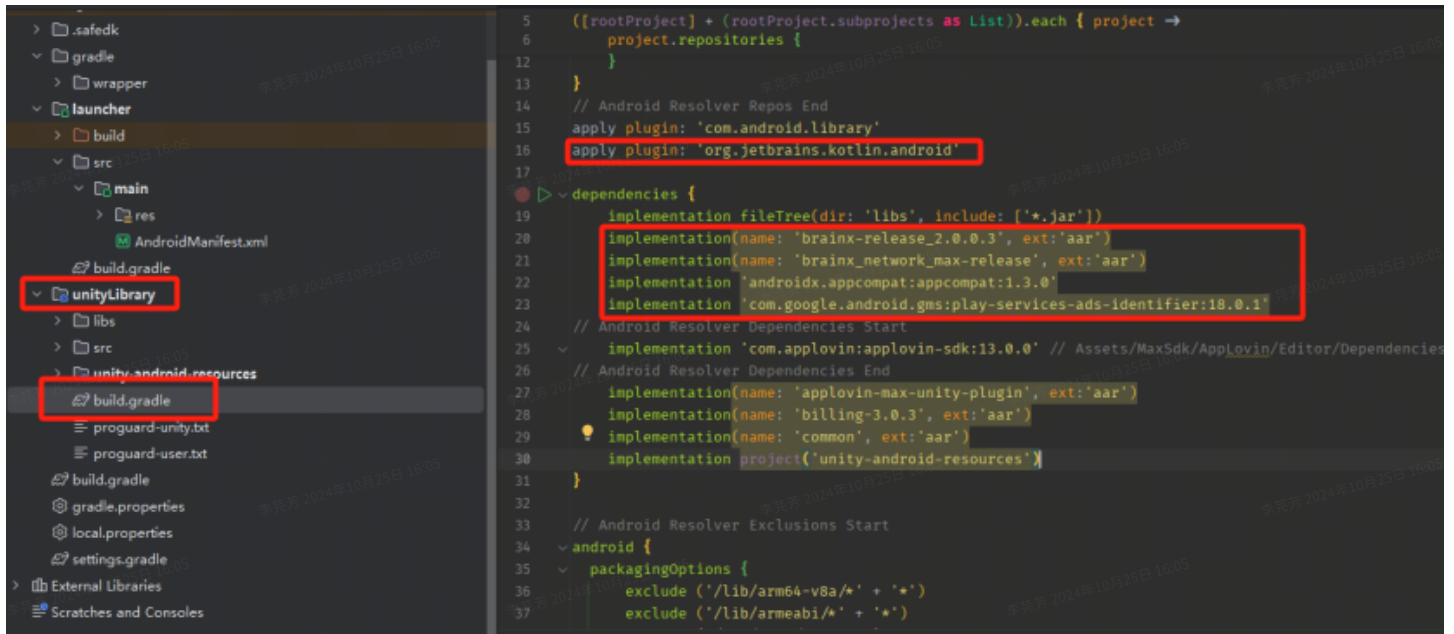
- `<uses-permission android:name="android.permission.INTERNET" />`  
`<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />`



7. Open `unityLibrary\build.gradle` and add:

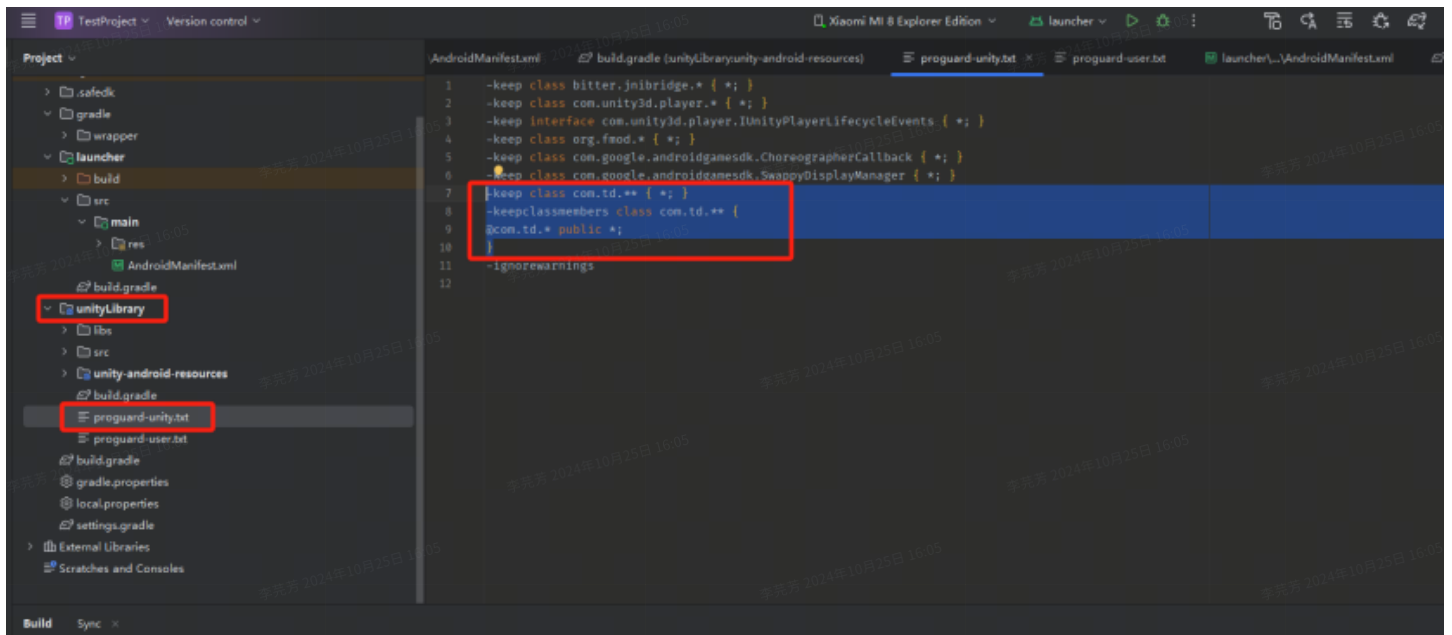


- Below `apply plugin: 'com.android.library'`, insert:
- `apply plugin: 'org.jetbrains.kotlin.android'`
- In the dependencies section, add:
- `implementation(name: 'brainx-release_2.0.0.3', ext:'aar')`
- `implementation(name: 'brainx_network_max-release', ext:'aar')`
- `implementation 'androidx.appcompat:appcompat:1.3.0'`
- `implementation 'com.google.android.gms:play-services-ads-identifier:18.0.1'`



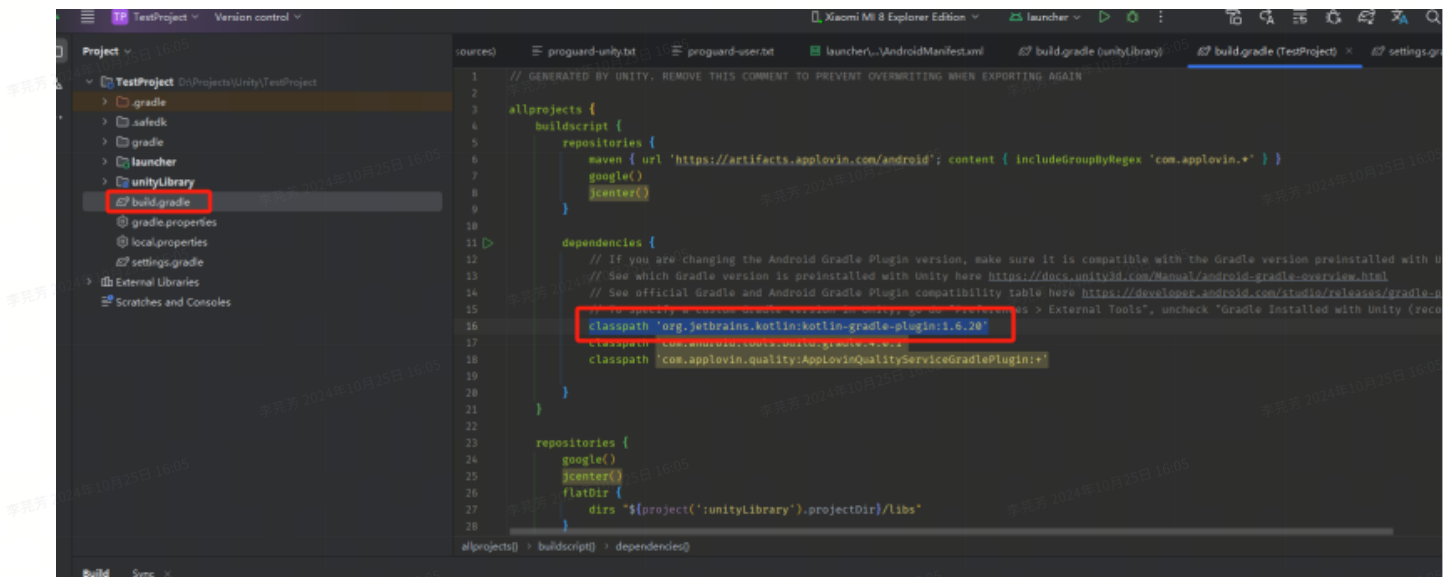
8. Open `unityLibrary\proguard-unity.txt` and add:

- `-keep class com.td.** { *; }`
- `-keepclassmembers class com.td.** {`
- `@com.td.* public *;`
- `}`



9. Open `build.gradle` in the project root and add:

- `classpath 'org.jetbrains.kotlin:kotlin-gradle-plugin:1.6.20'`



10. **Complete the Configuration:** After these steps, you can proceed with building the project.