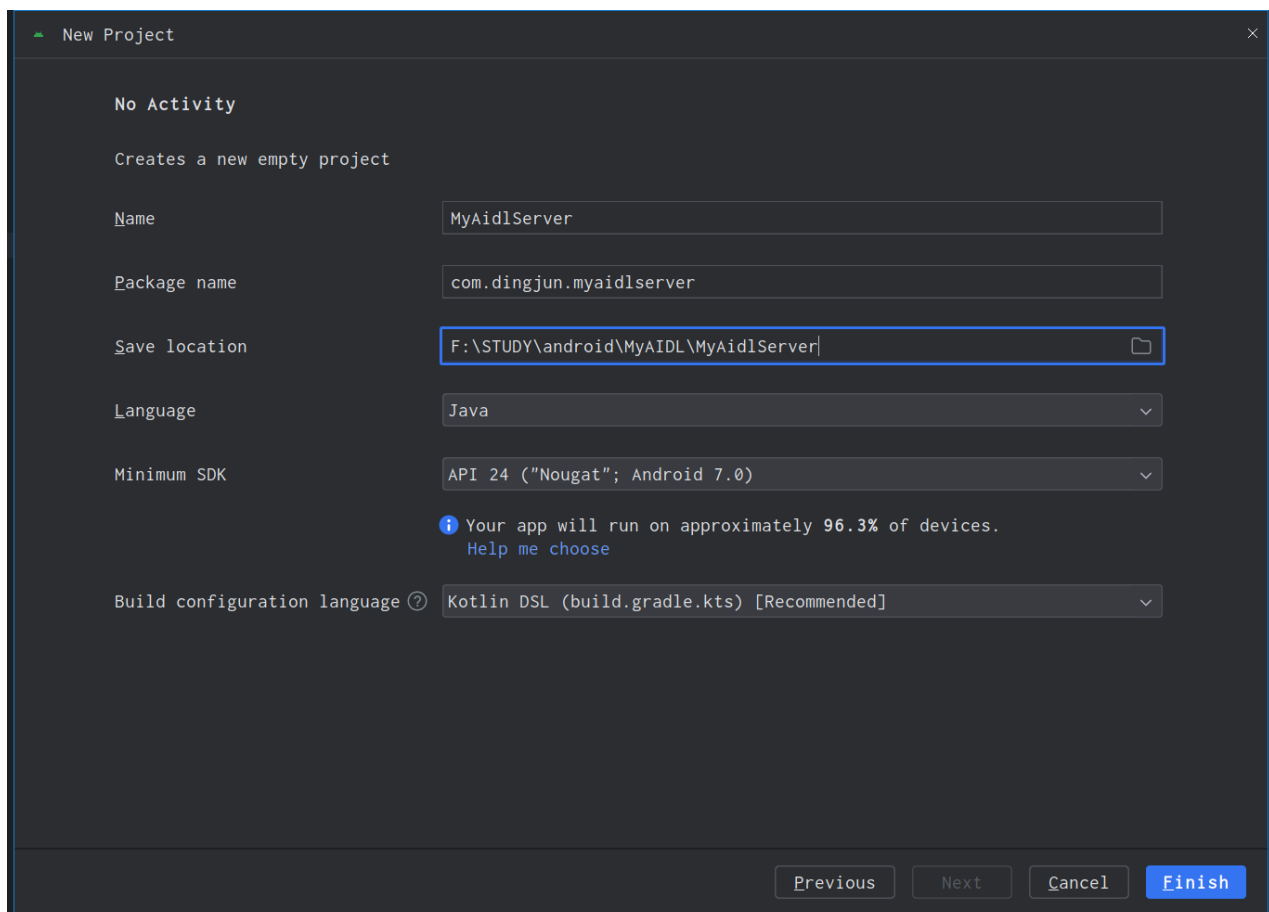
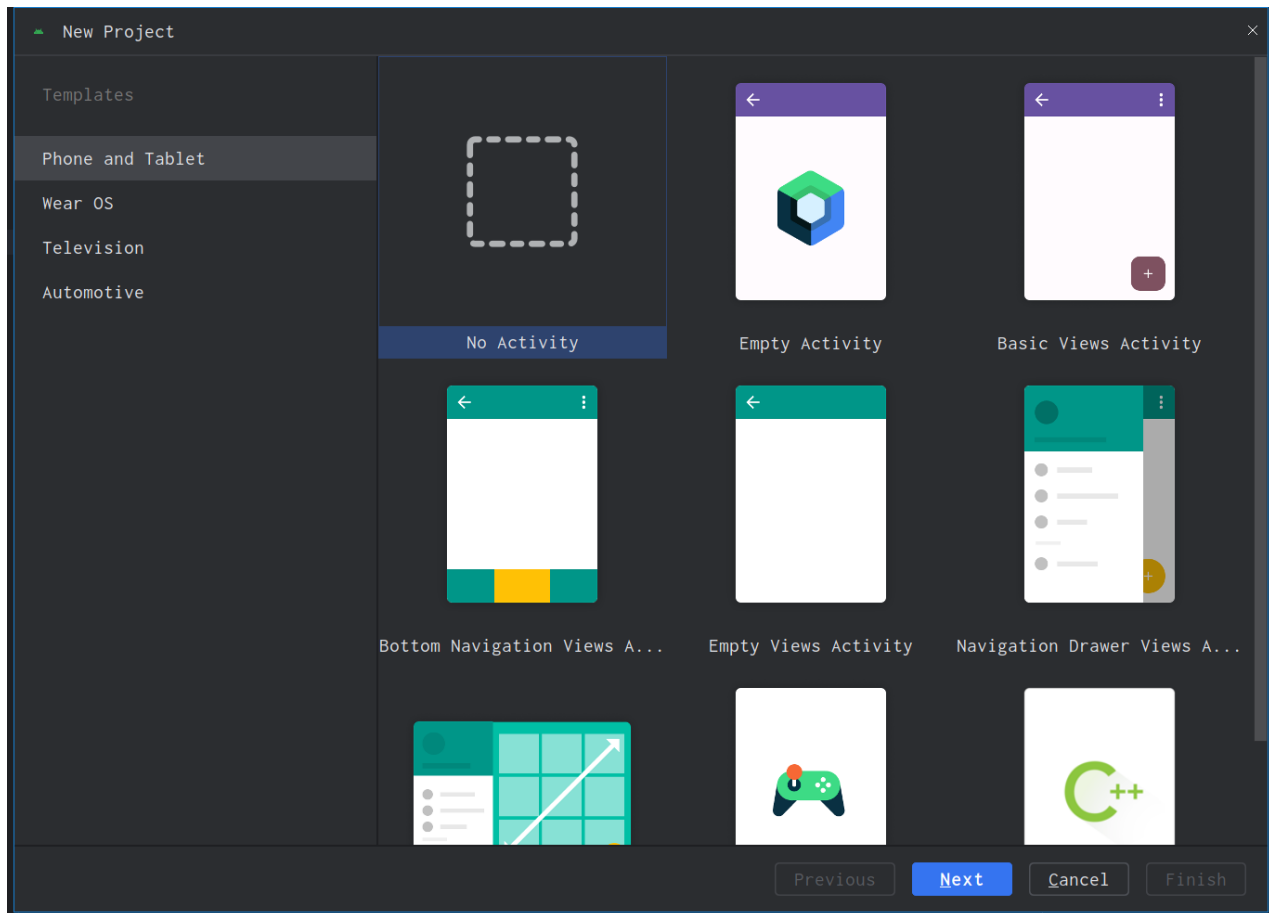
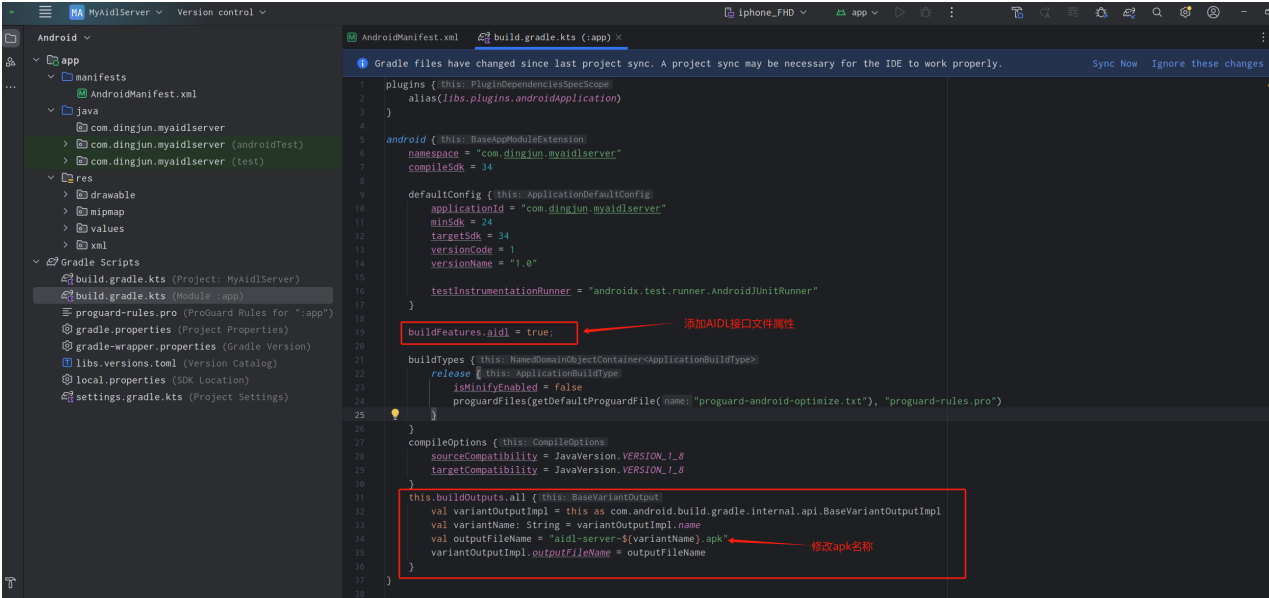


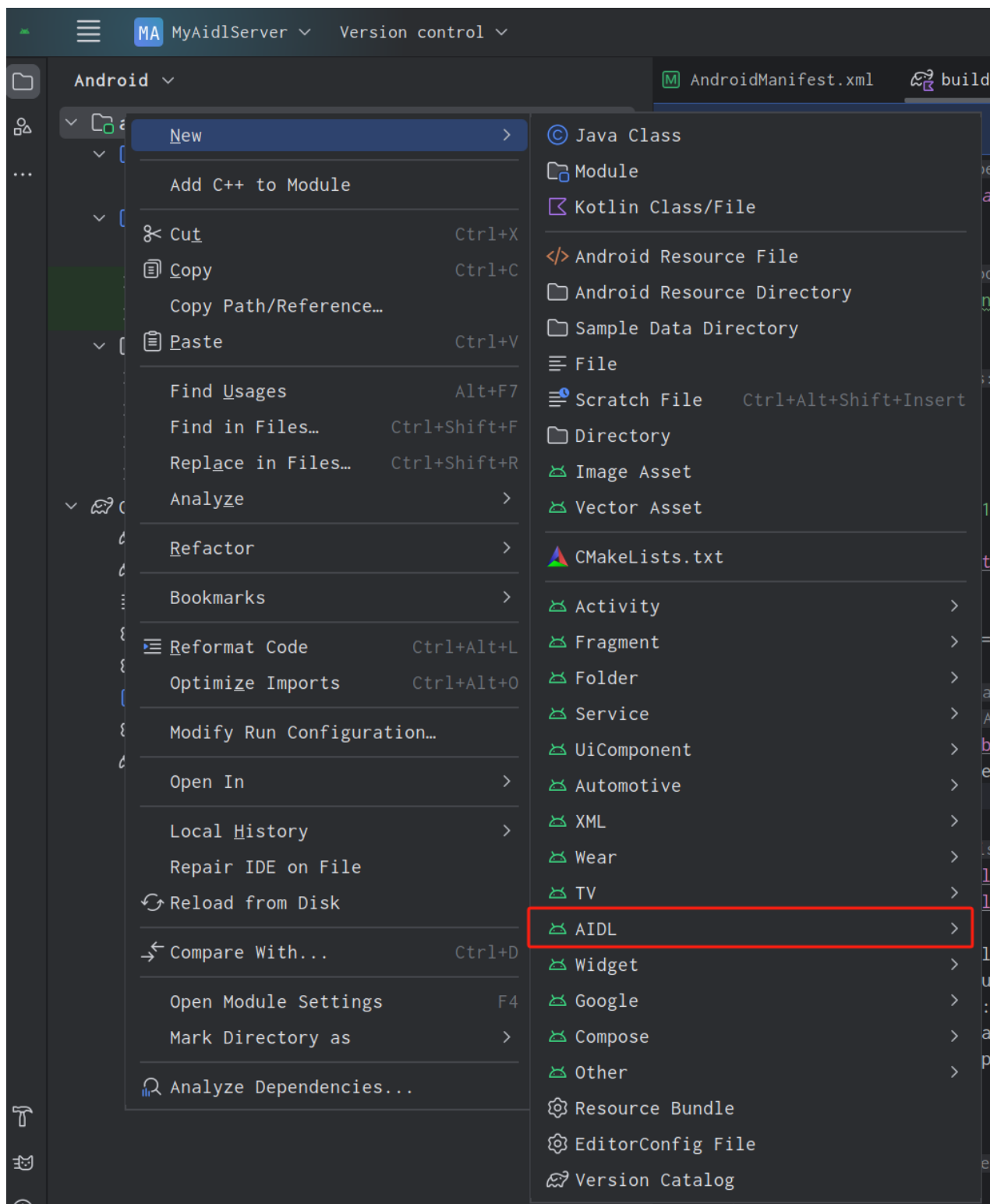
一、服务端

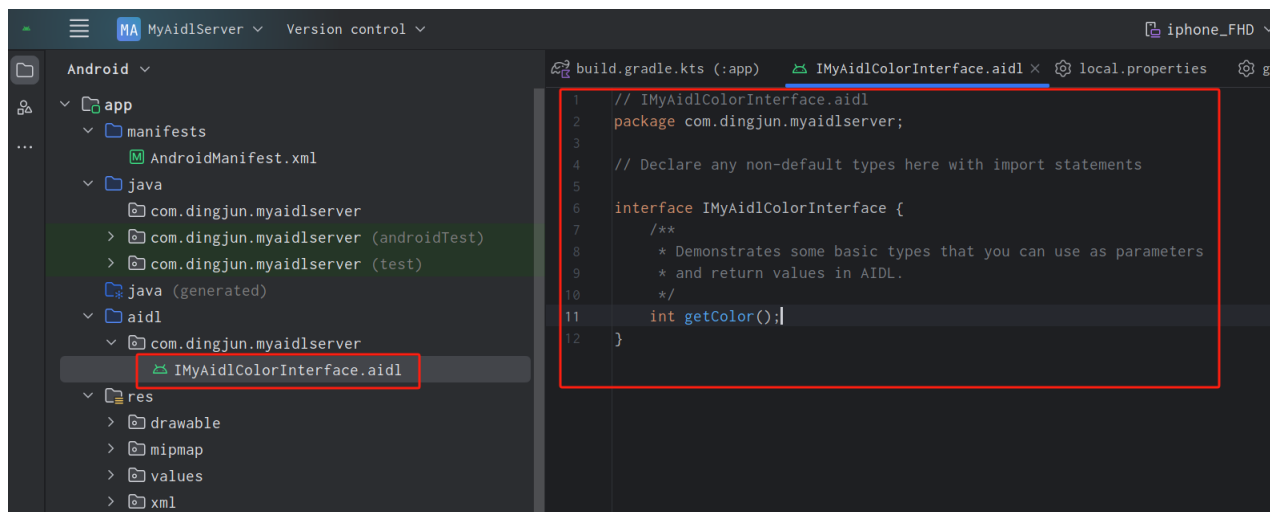
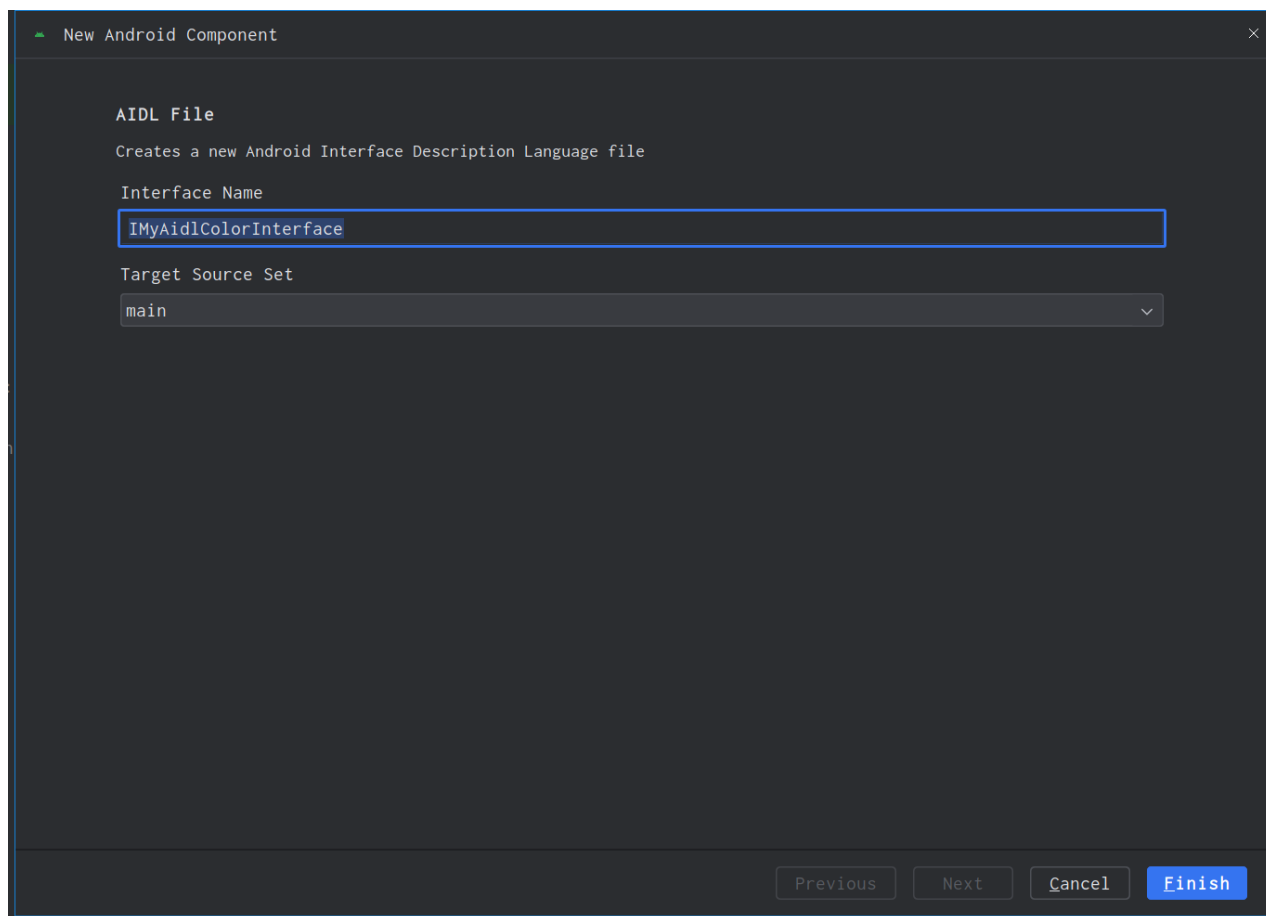
1. 新建server端的AS工程，此工程命名MyAIDLServer (Android Studio的版本为Android Studio Iguana | 2023.2.1)



2. 新建AIDL接口文件，注在建立该文件时候需要将build.gradle.kts文件修改增加aidl属性





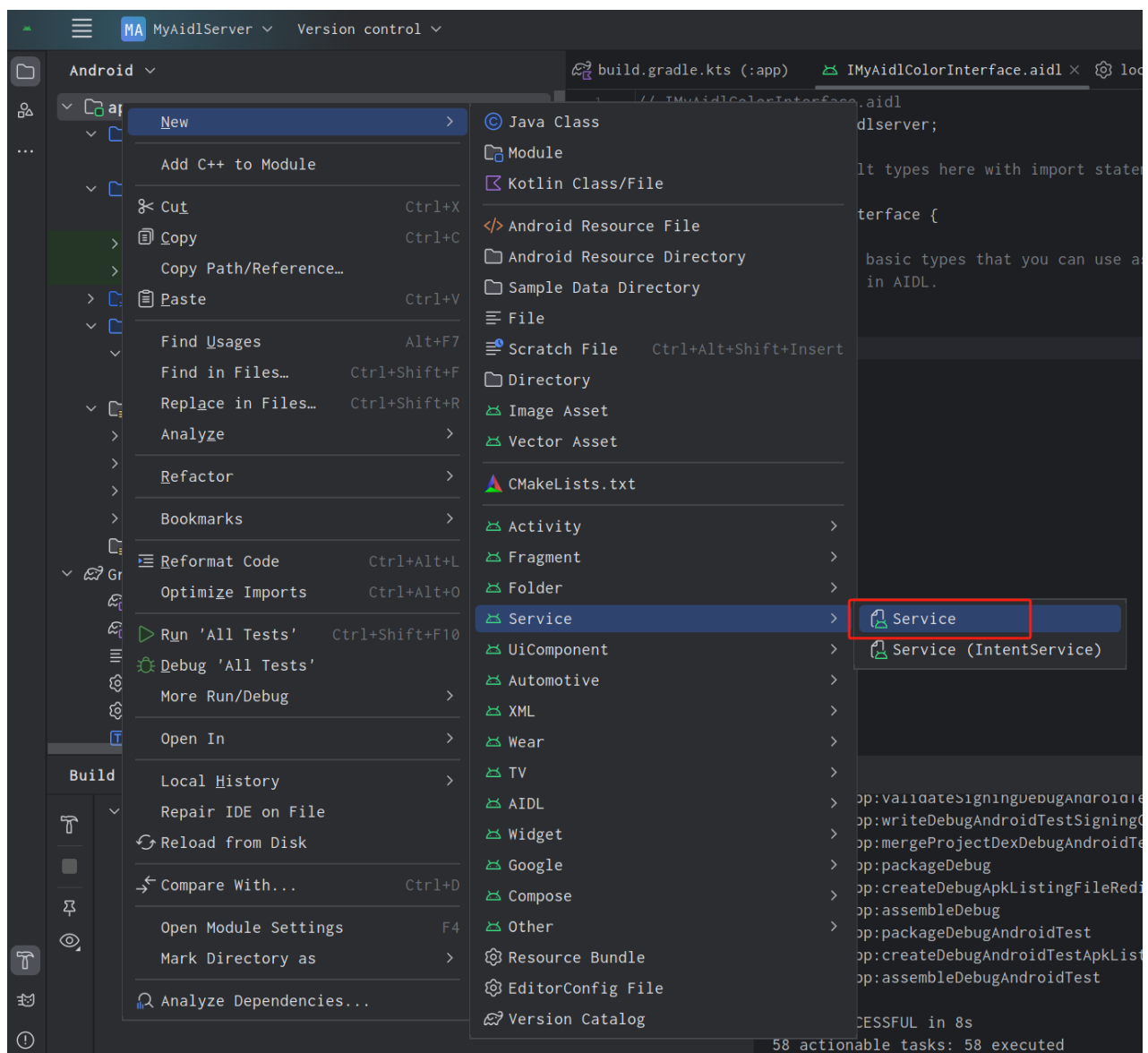


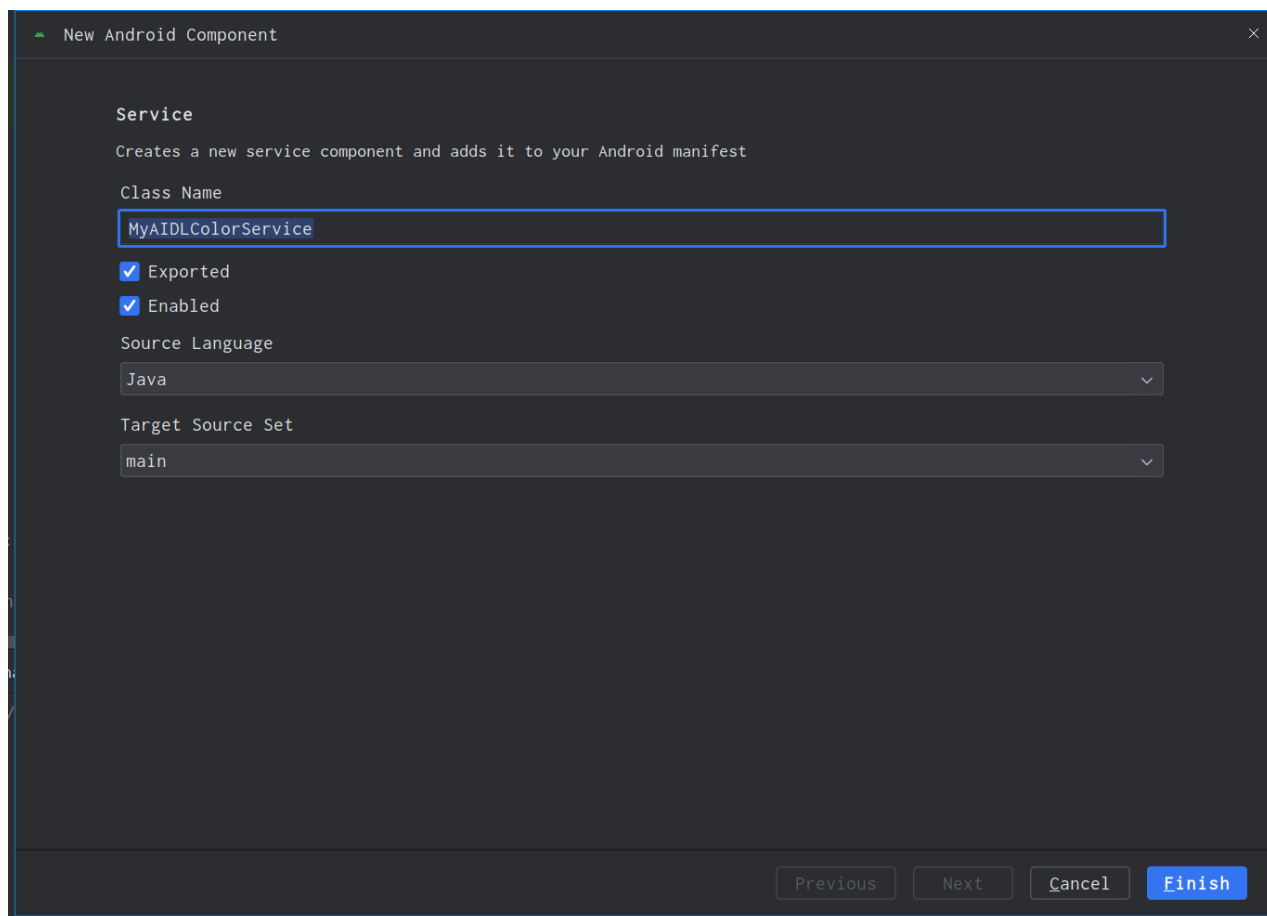
```

1 // IMyAidlColorInterface.aidl
2 package com.dingjun.myaidlserver;
3
4 // Declare any non-default types here with import statements
5
6 interface IMyAidlColorInterface {
7     /**
8      * Demonstrates some basic types that you can use as parameters
9      * and return values in AIDL.
10     */
11     int getColor();
12 }

```

3. 添加 MyAIDLColorService.java 服务文件





MyAIDLColorService.java文件如下：

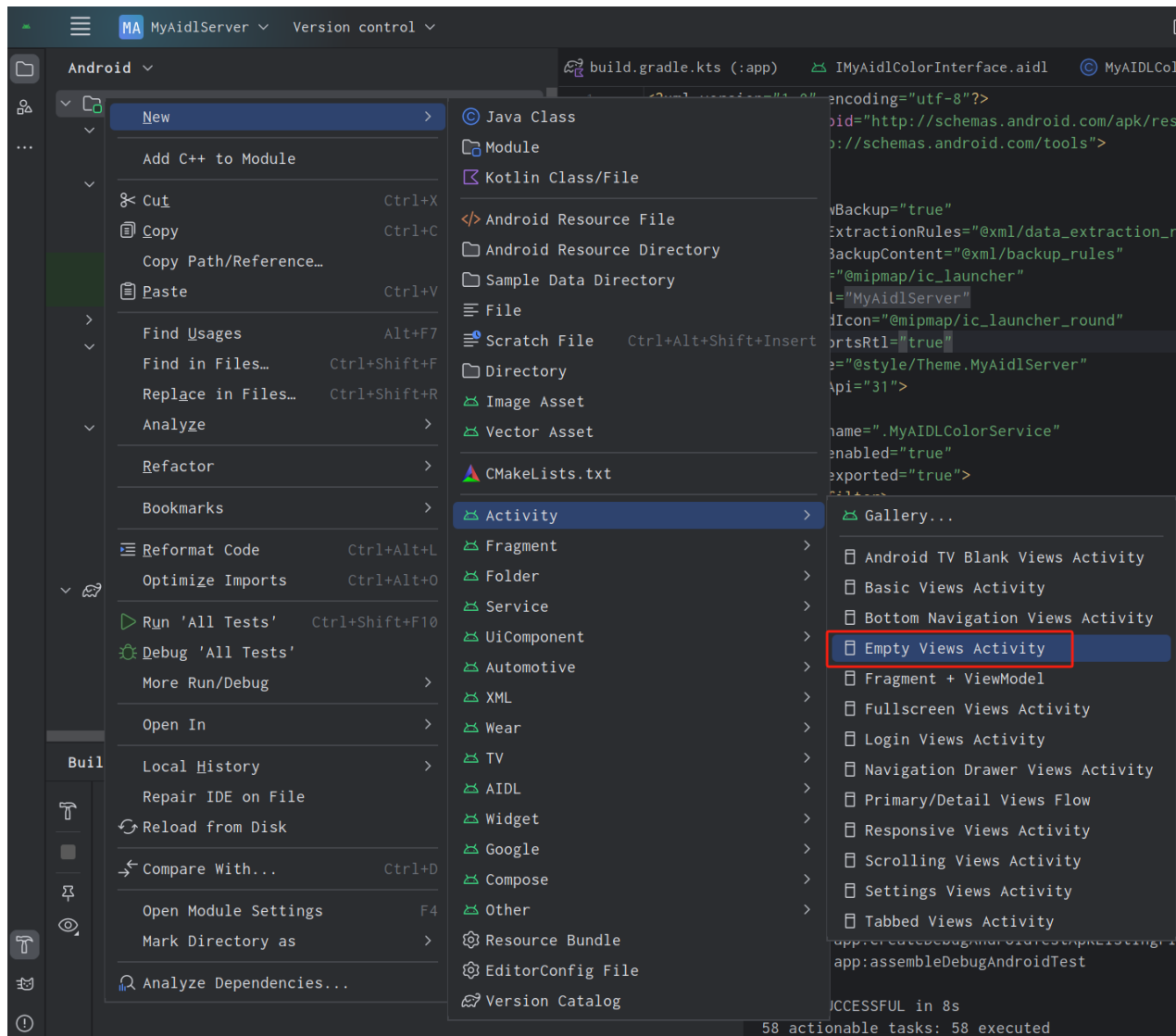
```
1 package com.dingjun.myaidlserver;
2
3 import android.annotation.SuppressLint;
4 import android.app.Notification;
5 import android.app.NotificationChannel;
6 import android.app.NotificationManager;
7 import android.app.Service;
8 import android.content.Context;
9 import android.content.Intent;
10 import android.graphics.Color;
11 import android.os.Build;
12 import android.os.IBinder;
13 import android.os.RemoteException;
14 import android.util.Log;
15
16 import androidx.annotation.RequiresApi;
17
18 import java.util.Random;
19
20 public class MyAIDLColorService extends Service {
21
22     private static final String TAG = "MyAIDLColorService";
23     public MyAIDLColorService() {
24     }
25
26     @RequiresApi(api = Build.VERSION_CODES.O)
27     @Override
28     public void onCreate() {
29         super.onCreate();
30         Log.d(TAG, "onCreate: startServiceForeground.");
31         startServiceForeground();
32     }
33
34     @Override
35     public IBinder onBind(Intent intent) {
36         // TODO: Return the communication channel to the service.
37         return binder;
38     }
39 }
```

```

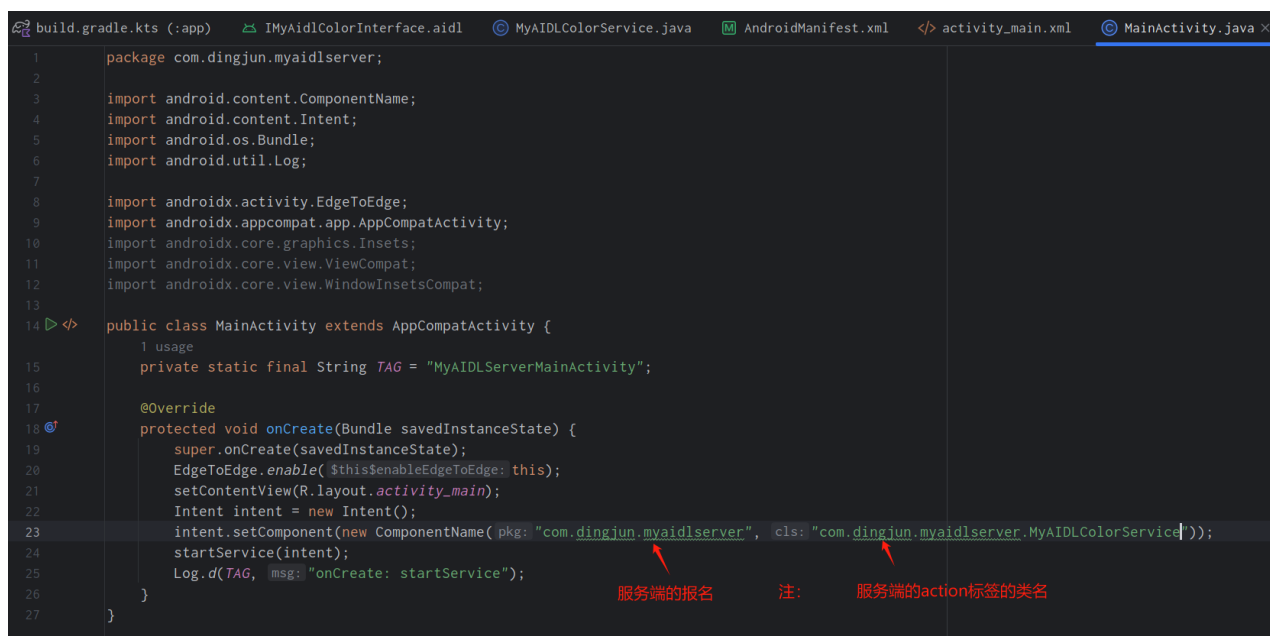
39     private final IMyAidlColorInterface.Stub binder = new IMyAidlColorInterface.Stub() {
40         @Override
41         public int getColor() throws RemoteException {
42             Random random = new Random();
43             int color = Color.argb(255, random.nextInt(256), random.nextInt(256),
random.nextInt(256));
44             String colorHex = Integer.toHexString(color);
45             Log.d(TAG, "getColor: " + colorHex.toUpperCase());
46             return color;
47         }
48     };
49
50     private static final String CHANNEL_ID_STRING =
"com.dingjun.myaidlserver.MyAIDLColorService";
51     private static final int CHANNEL_ID = 0x11;
52
53     @SuppressWarnings("ForegroundServiceType")
54     @RequiresApi(api = Build.VERSION_CODES.O)
55     private void startServiceForeground() {
56         NotificationManager notificationManager = (NotificationManager)
57             getSystemService(Context.NOTIFICATION_SERVICE);
58         NotificationChannel channel;
59         channel = new NotificationChannel(CHANNEL_ID_STRING,
getString(R.string.app_name),
60             NotificationManager.IMPORTANCE_LOW);
61         notificationManager.createNotificationChannel(channel);
62         Notification notification = new Notification.Builder(getApplicationContext(),
63             CHANNEL_ID_STRING).build();
64         startForeground(CHANNEL_ID, notification);
65     }
66 }

```

4. 创建服务端的Activity



Server端的MainActivity.java



```

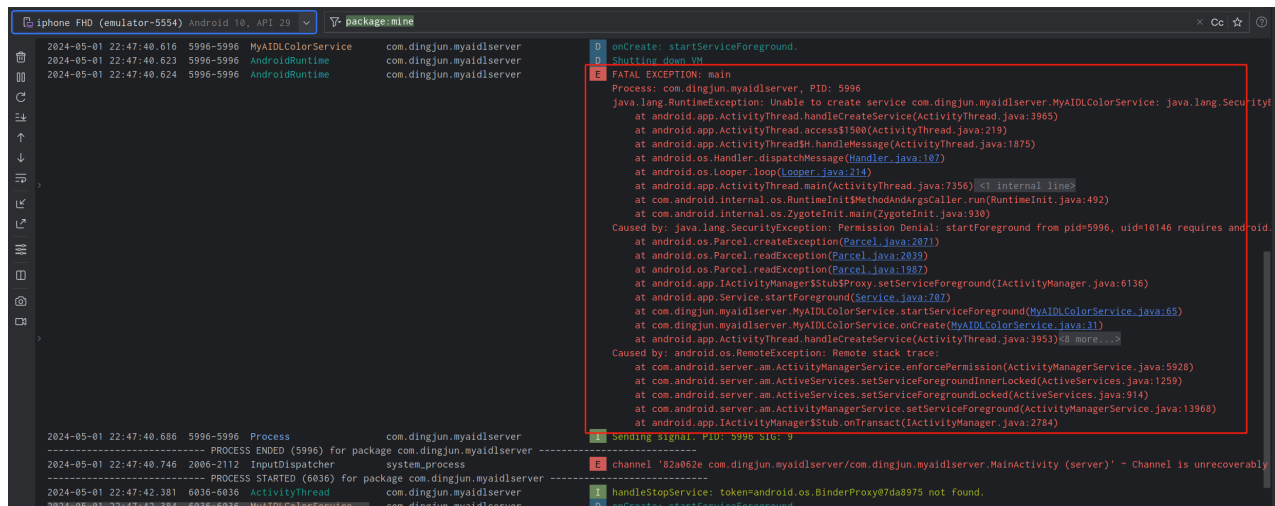
1 package com.dingjun.myaidlserver;
2
3 import android.content.ComponentName;
4 import android.content.Intent;
5 import android.os.Bundle;
6 import android.util.Log;
7
8 import androidx.activity.EdgeToEdge;
9 import androidx.appcompat.app.AppCompatActivity;
10 import androidx.core.graphics.Insets;
11 import androidx.core.view.ViewCompat;
12 import androidx.core.view.WindowInsetsCompat;
13
14 public class MainActivity extends AppCompatActivity {
15     private static final String TAG = "MyAIDLServerMainActivity";
16
17     @Override
18     protected void onCreate(Bundle savedInstanceState) {
19         super.onCreate(savedInstanceState);
20         EdgeToEdge.enable(this);
21         setContentView(R.layout.activity_main);
22         Intent intent = new Intent();
23         intent.setComponent(new ComponentName("com.dingjun.myaidlserver",
24 "com.dingjun.myaidlserver.MyAIDLColorService"));
25         startService(intent);
26         Log.d(TAG, "onCreate: startService");
27     }
28 }

```

5. 在服务端的AndroidManifest.xml文件服务标签里添加intent-filter和服务端的Activity，使用activity来拉起demo服务，内容如下：

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools">
4
5     <application
6         android:allowBackup="true"
7         android:dataExtractionRules="@xml/data_extraction_rules"
8         android:fullBackupContent="@xml/backup_rules"
9         android:icon="@mipmap/ic_launcher"
10        android:label="@string/app_name"
11        android:roundIcon="@mipmap/ic_launcher_round"
12        android:supportsRtl="true"
13        android:theme="@style/Theme.MyAidlServer"
14        tools:targetApi="31">
15        <activity
16            android:name=".MainActivity"
17            android:exported="true">
18            <intent-filter>
19                <action android:name="android.intent.action.MAIN" />
20                <category android:name="android.intent.category.LAUNCHER" />
21            </intent-filter>
22        </activity>
23
24        <service
25            android:name=".MyAIDLColorService"
26            android:enabled="true"
27            android:exported="true">
28            <intent-filter>
29                <action android:name="com.dingjun.myaidlserver.MyAIDLColorService" />
30            </intent-filter>
31        </service>
32    </application>
33
34 </manifest>
```

至此，服务端在模拟器运行时候会报错，问题原因是运行服务没有foreground服务权限



```
2024-05-01 22:47:40.616 5996-5996 MyAIDLColorService com.dingjun.myaidservice
2024-05-01 22:47:40.623 5996-5996 AndroidRuntime com.dingjun.myaidservice
2024-05-01 22:47:40.624 5996-5996 AndroidRuntime com.dingjun.myaidservice

2024-05-01 22:47:40.686 5996-5996 Process com.dingjun.myaidservice
----- PROCESS ENDED (5996) for package com.dingjun.myaidservice -----
2024-05-01 22:47:40.746 2006-2112 InputDispatcher system_process
----- PROCESS STARTED (6036) for package com.dingjun.myaidservice -----
2024-05-01 22:47:42.381 6036-6036 ActivityThread com.dingjun.myaidservice
2024-05-01 22:47:42.384 6036-6036 MyAIDLColorService com.dingjun.myaidservice

D onCreate: startServiceForeground.
D Shutting down VM.
E FATAL EXCEPTION: main
Process: com.dingjun.myaidservice, PID: 5996
java.lang.RuntimeException: Unable to create service com.dingjun.myaidservice.MyAIDLColorService: java.lang.SecurityException: Permission Denial: startForeground from pid=5996, uid=10146 requires android.permission.FOREGROUND_SERVICE
    at android.app.ActivityThread.handleCreateService(ActivityThread.java:3965)
    at android.app.ActivityThread.access$5100(ActivityThread.java:219)
    at android.app.ActivityThread$H.handleMessage(ActivityThread.java:1875)
    at android.os.Handler.dispatchMessage(Handler.java:187)
    at android.os.Looper.loop(Looper.java:214)
    at android.app.ActivityThread.main(ActivityThread.java:7356)
    at com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run(RuntimeInit.java:492)
    at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:930)
Caused by: java.lang.SecurityException: Permission Denial: startForeground from pid=5996, uid=10146 requires android.permission.FOREGROUND_SERVICE
    at android.os.Parcel.createException(Parcel.java:2071)
    at android.os.Parcel.readException(Parcel.java:2039)
    at android.os.Parcel.readException(Parcel.java:1983)
    at android.app.ActivityManager$Stub$Proxy.setServiceForeground(ActivityManager.java:6136)
    at android.app.Service.startForeground(Service.java:787)
    at com.dingjun.myaidservice.MyAIDLColorService.startServiceForeground(MyAIDLColorService.java:65)
    at com.dingjun.myaidservice.MyAIDLColorService.onCreate(MyAIDLColorService.java:31)
    at android.app.ActivityThread.handleCreateService(ActivityThread.java:3953)
    at android.app.ActivityThread.handleCreateService(ActivityThread.java:3953)
Caused by: android.os.RemoteException: Remote stack trace:
    at com.android.server.am.ActivityManagerService.enforcePermission(ActivityManagerService.java:5928)
    at com.android.server.am.ActiveServices.setServiceForegroundInnerLocked(ActiveServices.java:1259)
    at com.android.server.am.ActiveServices.setServiceForegroundLocked(ActiveServices.java:914)
    at com.android.server.am.ActivityManagerService.setServiceForeground(ActivityManagerService.java:13968)
    at android.app.ActivityManager$Stub.onTransact(ActivityManager.java:2784)
    at android.os.Parcel.readException(Parcel.java:2071)
    at android.os.Parcel.readException(Parcel.java:2039)
    at android.os.Parcel.readException(Parcel.java:1983)
    at android.app.ActivityManager$Stub$Proxy.setServiceForeground(ActivityManager.java:6136)
    at android.app.Service.startForeground(Service.java:787)
    at com.dingjun.myaidservice.MyAIDLColorService.startServiceForeground(MyAIDLColorService.java:65)
    at com.dingjun.myaidservice.MyAIDLColorService.onCreate(MyAIDLColorService.java:31)
    at android.app.ActivityThread.handleCreateService(ActivityThread.java:3953)
    at android.app.ActivityThread.handleCreateService(ActivityThread.java:3953)
D Sending signal. PID: 5996 SIG: 9
E channel '82a062e com.dingjun.myaidservice/com.dingjun.myaidservice.MainActivity (server)' ~ Channel is unrecoverably
D handleStopService: token=android.os.BinderProxy@7da8975 not found.
D onCreate: startServiceForeground.
```

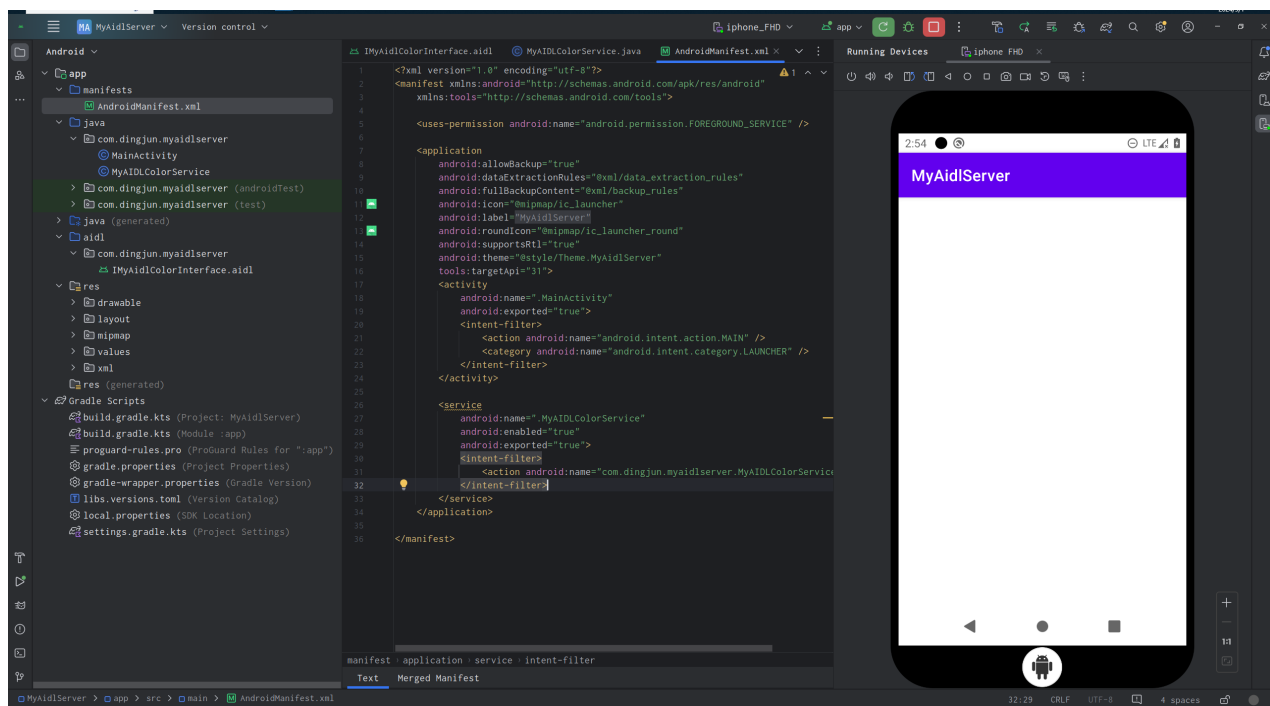
解决办法在服务端的AndroidManifest.xml

添加服务端前台权限 `<uses-permission`

`android:name="android.permission.FOREGROUND_SERVICE" />`

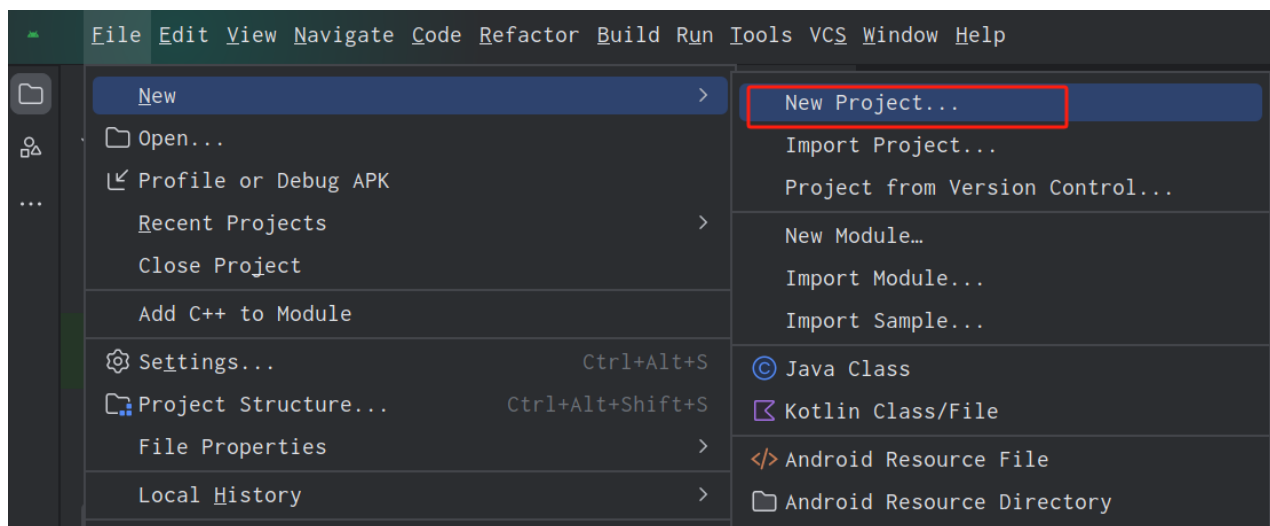
```
1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools">
4      <uses-permission android:name="android.permission.FOREGROUND_SERVICE" />
5      <application
6          android:allowBackup="true"
7          android:dataExtractionRules="@xml/data_extraction_rules"
8          android:fullBackupContent="@xml/backup_rules"
9          android:icon="@mipmap/ic_launcher"
10         android:label="@string/app_name"
11         android:roundIcon="@mipmap/ic_launcher_round"
12         android:supportsRtl="true"
13         android:theme="@style/Theme.MyAidlServer"
14         tools:targetApi="31">
15         <activity
16             android:name=".MainActivity"
17             android:exported="true">
18             <intent-filter>
19                 <action android:name="android.intent.action.MAIN" />
20                 <category android:name="android.intent.category.LAUNCHER" />
21             </intent-filter>
22         </activity>
23         <service
24             android:name=".MyAIDLColorService"
25             android:enabled="true"
26             android:exported="true">
27             <intent-filter>
28                 <action android:name="com.dingjun.myaidlserver.MyAIDLColorService" />
29             </intent-filter>
30         </service>
31     </application>
32
33 </manifest>
```

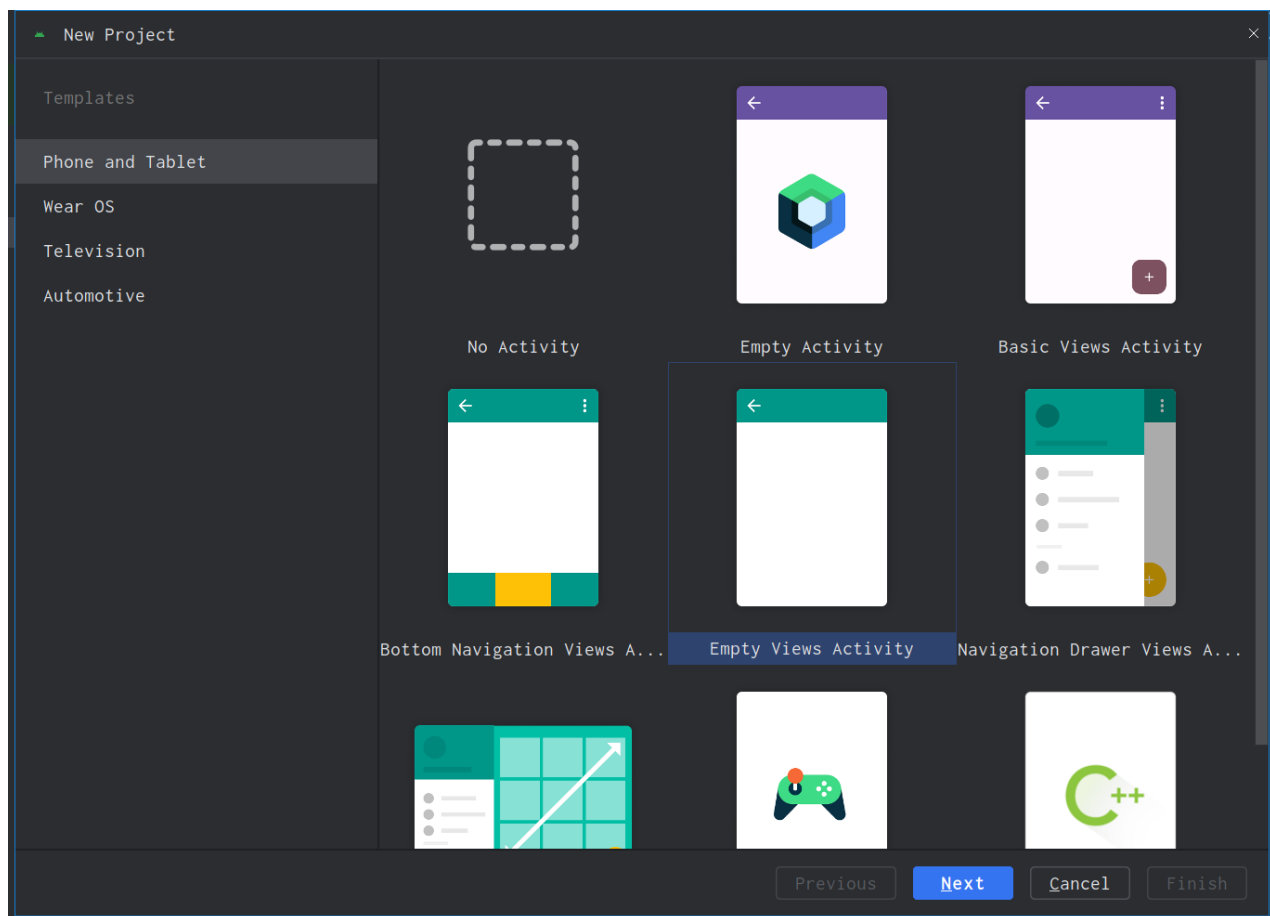
模拟器成功运行：



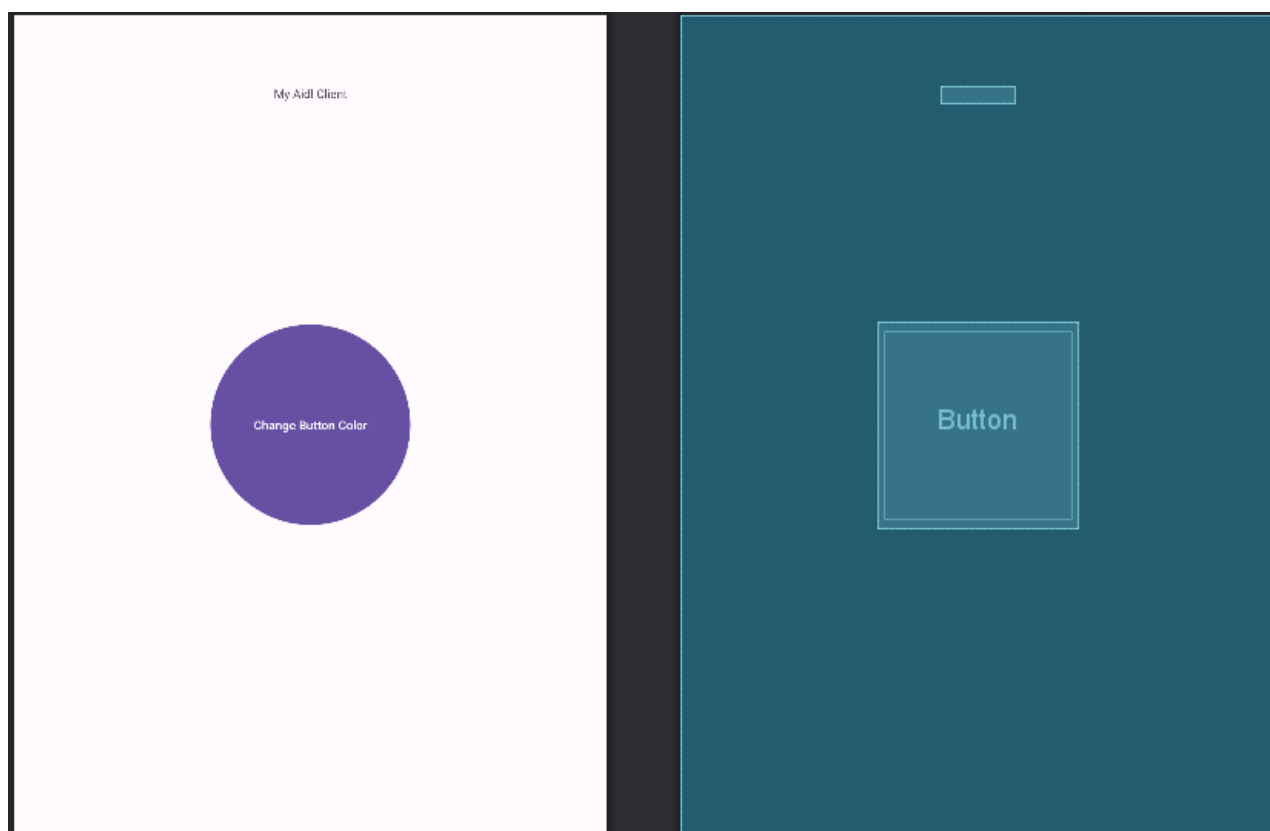
二、客户端

1. 新建客户端的AS工程，此工程命名MyAIDLClient（Android Studio的版本为Android Studio Iguana | 2023.2.1）





2. 调整layout布局

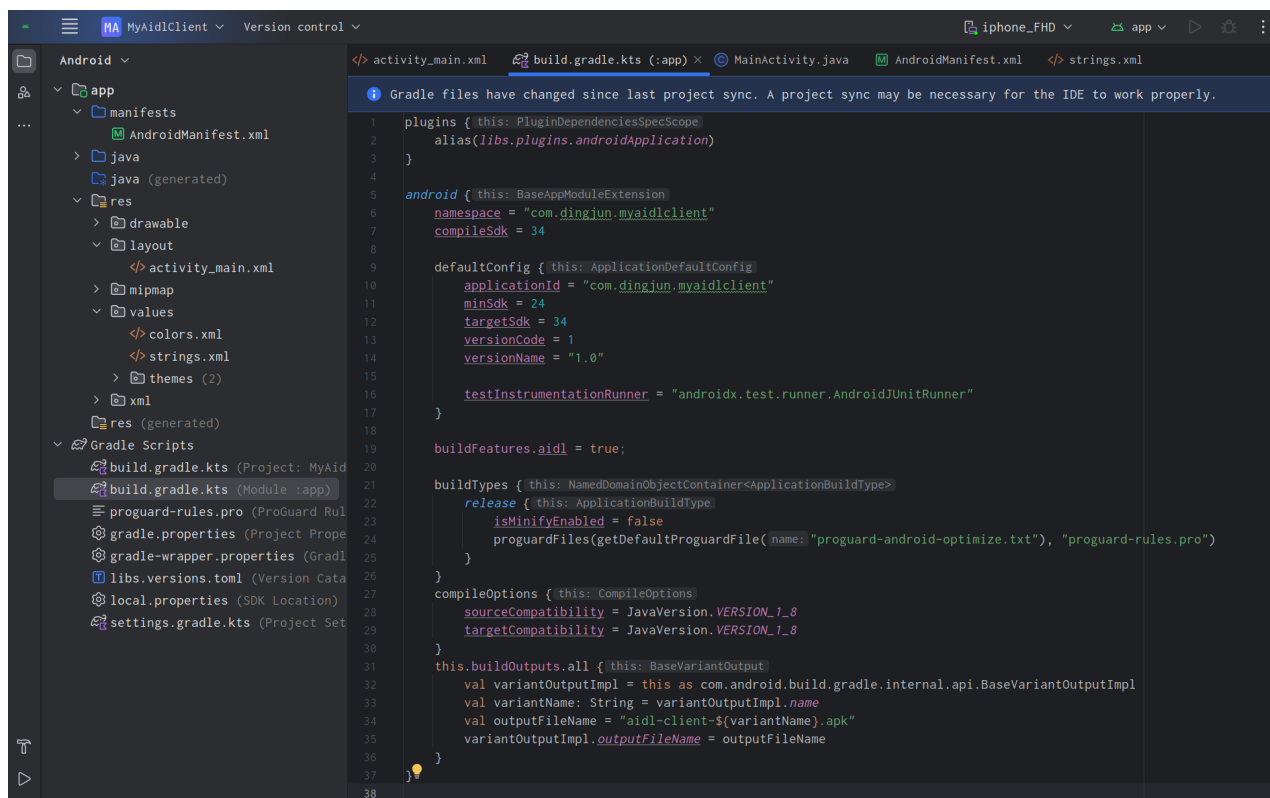


```

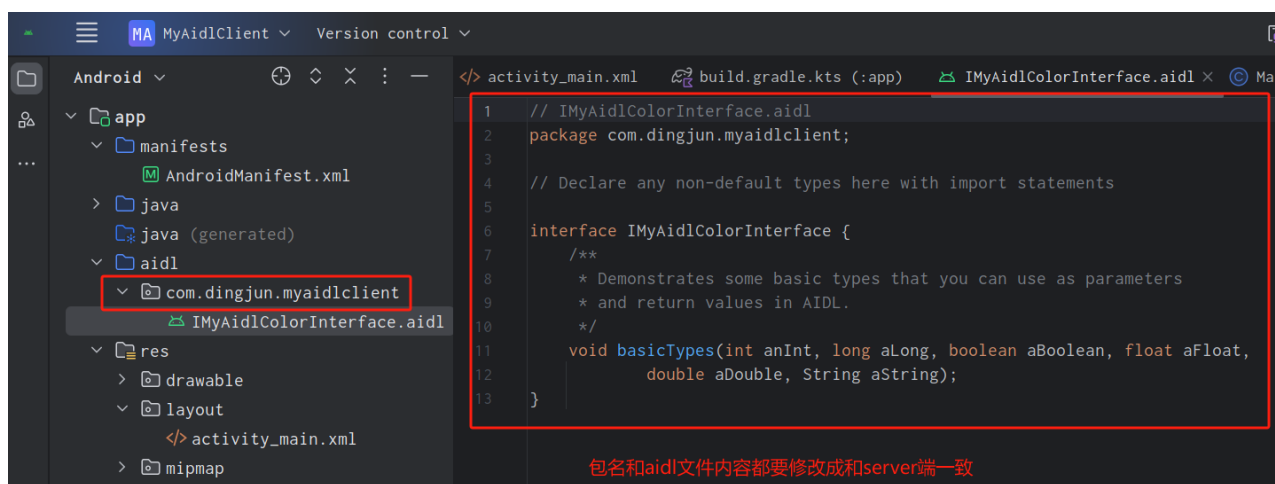
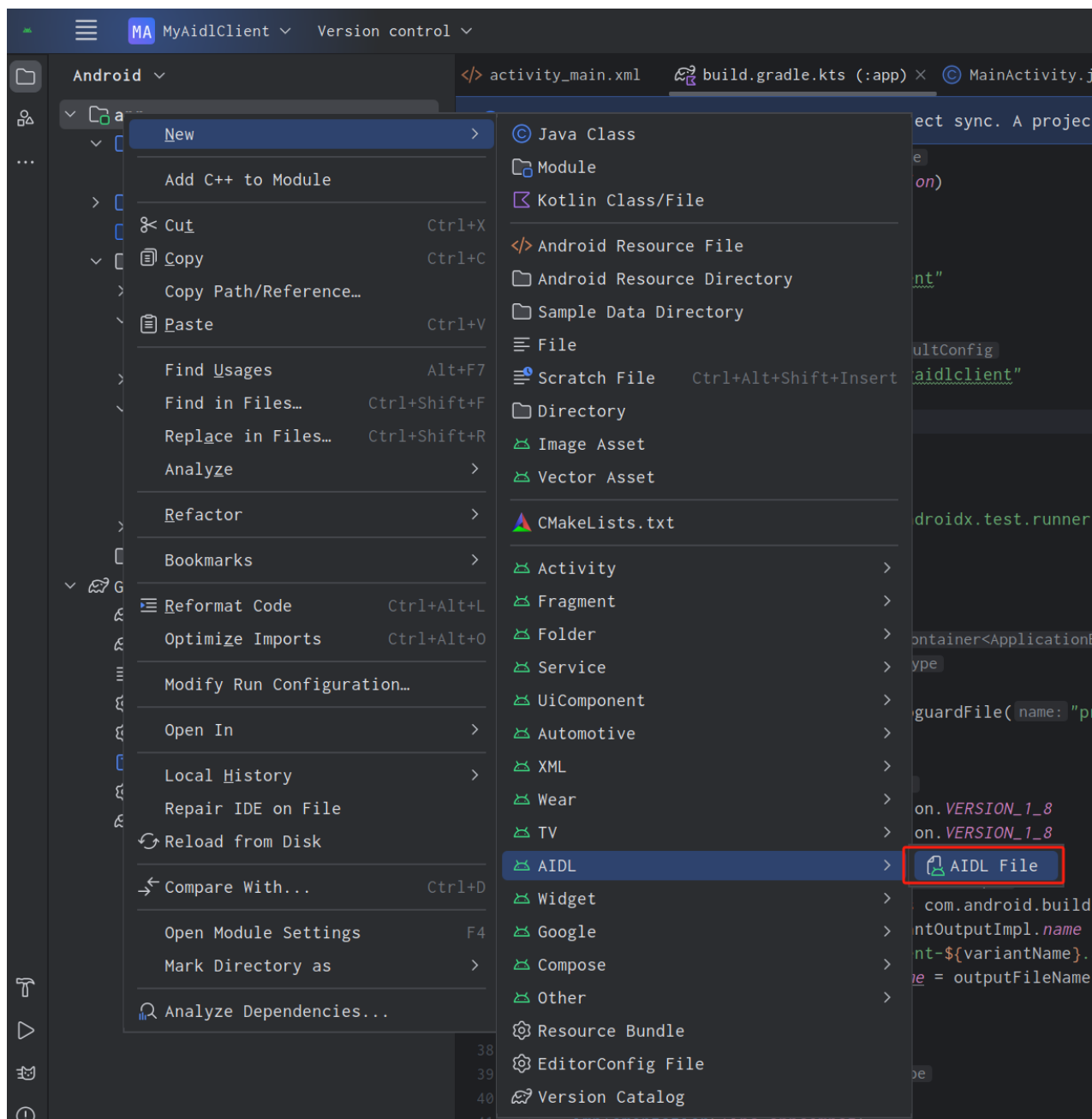
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:id="@+id/main"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context=".MainActivity">
9
10     <TextView
11         android:id="@+id/textView"
12         android:layout_width="wrap_content"
13         android:layout_height="wrap_content"
14         android:text="My Aidl Client"
15         app:layout_constraintBottom_toBottomOf="parent"
16         app:layout_constraintEnd_toEndOf="parent"
17         app:layout_constraintStart_toStartOf="parent"
18         app:layout_constraintTop_toTopOf="parent"
19         app:layout_constraintVertical_bias="0.087" />
20
21     <Button
22         android:id="@+id/btn_color"
23         android:layout_width="233dp"
24         android:layout_height="241dp"
25         android:text="Change Button Color"
26         app:layout_constraintBottom_toBottomOf="parent"
27         app:layout_constraintEnd_toEndOf="parent"
28         app:layout_constraintStart_toStartOf="parent"
29         app:layout_constraintTop_toTopOf="parent" />
30
31 </androidx.constraintlayout.widget.ConstraintLayout>

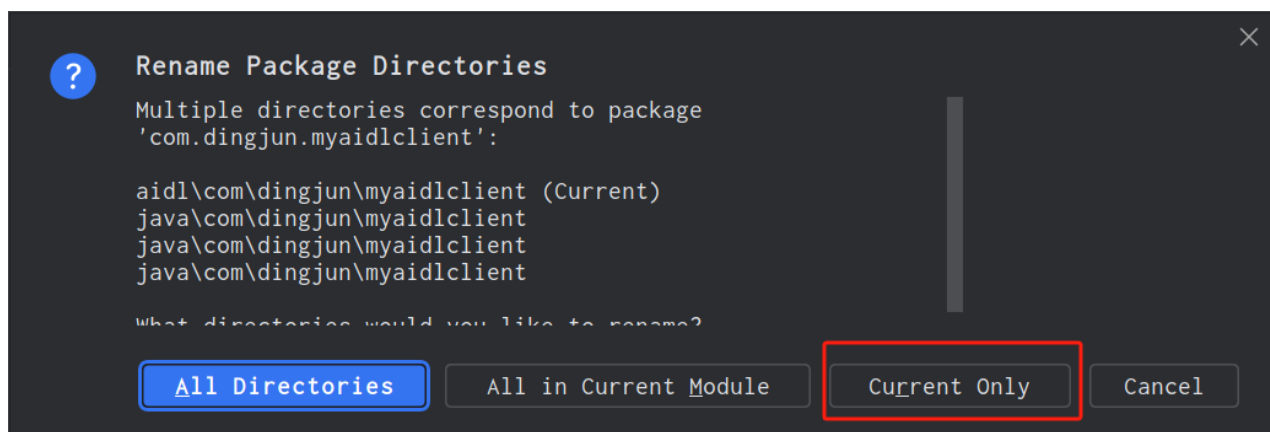
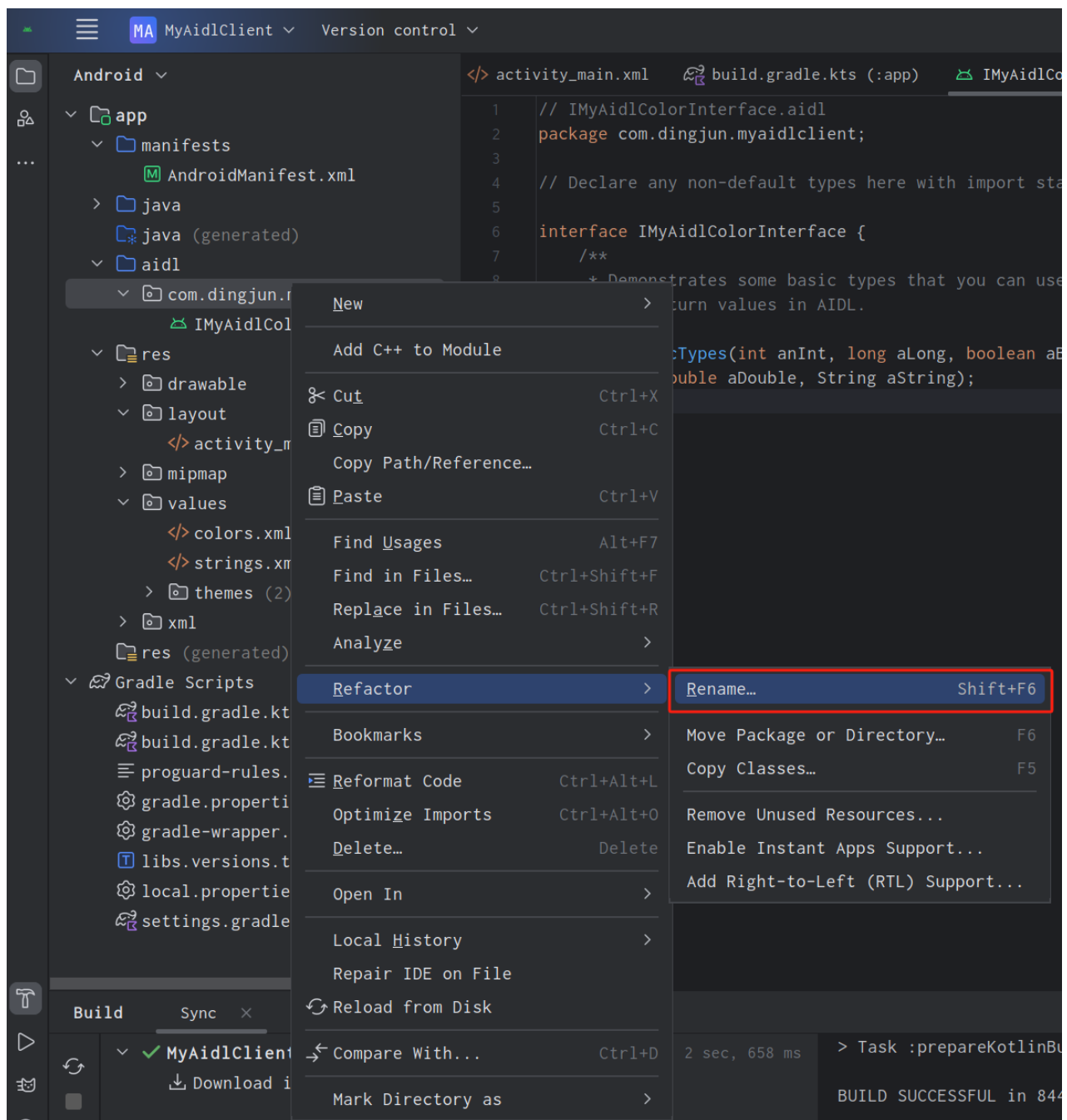
```

3. 新建AIDL接口文件，注在建立该文件时候需要将build.gradle.kts文件修改增加aidl属性

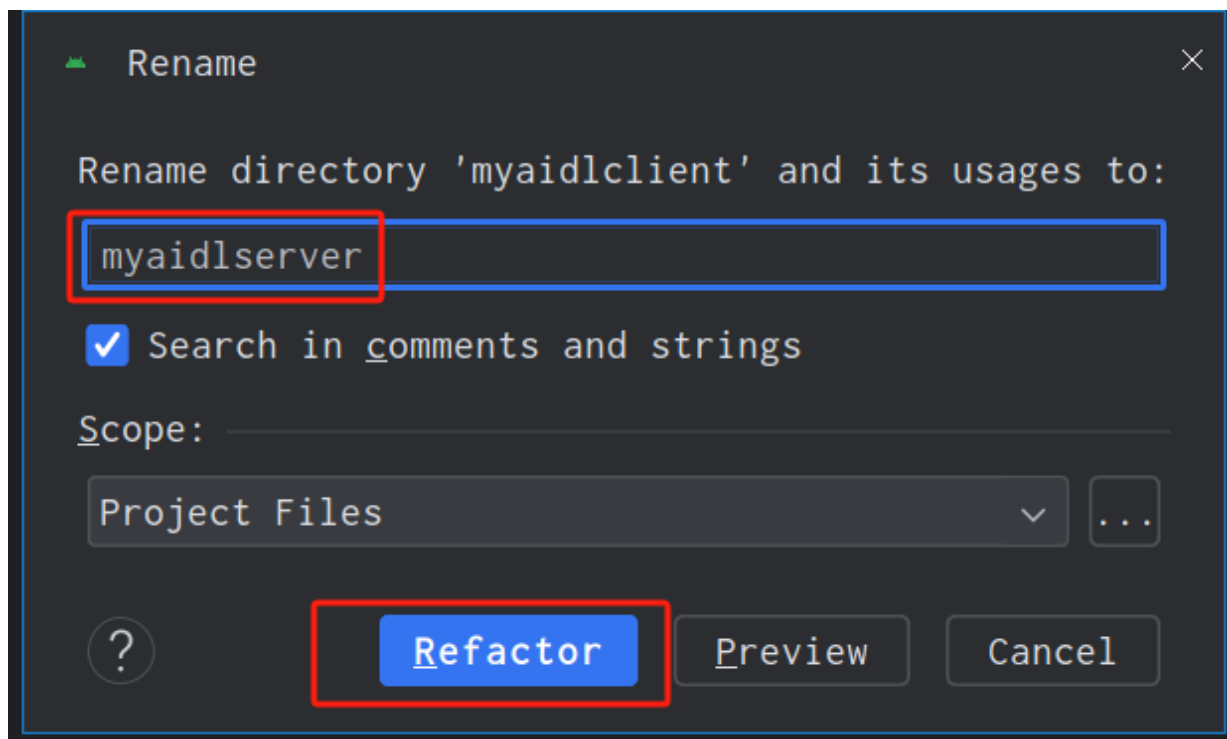


创建IMyAidlColorInterface.aidl接口文件，注：该文件要和server端的文件保持一模一样，包含包名路径也要一样！！！！

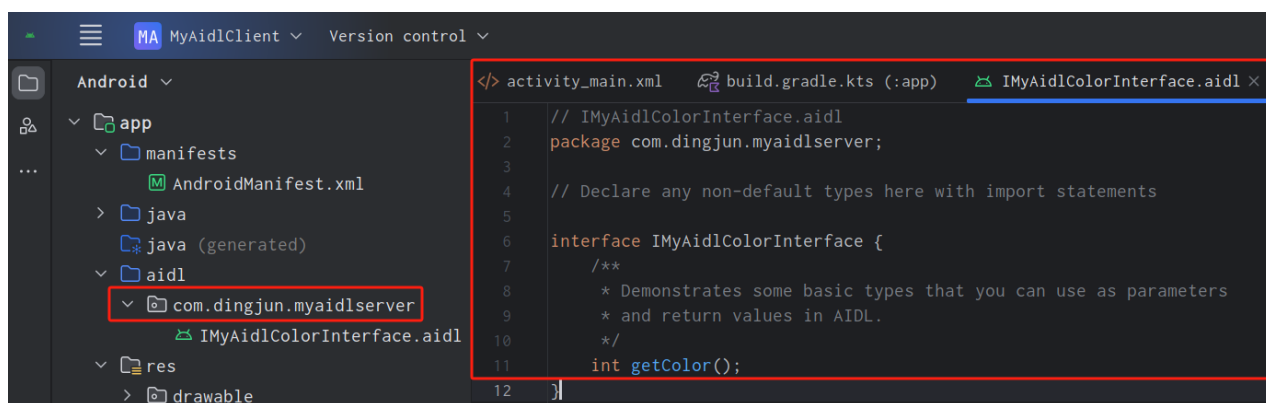
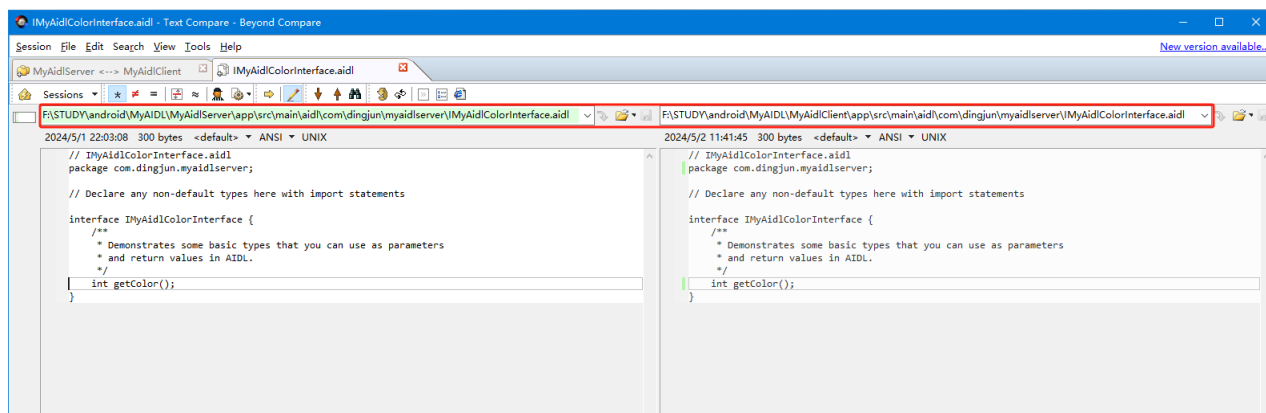




改成如下：



重启一下AS，路径名就会显示正确，保持和server端的AIDL路径和文件保持一致！



4. 在客户端的Activity中创建connection

```
1  IMyAidlColorInterface iAidlColorService;
2
3  private ServiceConnection mConnection = new ServiceConnection() {
4      @Override
5      public void onServiceConnected(ComponentName name, IBinder service) {
6          iAidlColorService = IMyAidlColorInterface.Stub.asInterface(service);
7          Log.d(TAG, "ServiceConnection: Remote Color Service Connected!!!");
8
9      }
10
11     @Override
12     public void onServiceDisconnected(ComponentName name) {
13         Log.d(TAG, "onServiceDisconnected: ");
14         iAidlColorService = null;
15     }
16 };
```

工程重新编译一下;

客户端的MainActivity.java文件:

```
1 package com.dingjun.myaidlclient;
2
3 import android.content.ComponentName;
4 import android.content.Intent;
5 import android.content.ServiceConnection;
6 import android.os.Bundle;
7 import android.os.IBinder;
8 import android.os.RemoteException;
9 import android.util.Log;
10 import android.view.View;
11 import android.widget.Button;
12
13 import androidx.activity.EdgeToEdge;
14 import androidx.appcompat.app.AppCompatActivity;
15 import androidx.core.graphics.Insets;
16 import androidx.core.view.ViewCompat;
17 import androidx.core.view.WindowInsetsCompat;
18
19 import com.dingjun.myaidlserver.IMyAidlColorInterface;
20
21 public class MainActivity extends AppCompatActivity {
22     private static final String TAG = "MyAIDLClientMainActivity";
23
24     IMyAidlColorInterface iAidlColorService;
25
26     private ServiceConnection mConnection = new ServiceConnection() {
27         @Override
28         public void onServiceConnected(ComponentName name, IBinder service) {
29             iAidlColorService = IMyAidlColorInterface.Stub.asInterface(service);
30             Log.d(TAG, "ServiceConnection: Remote Color Service Connected!!!");
31
32         }
33
34         @Override
35         public void onServiceDisconnected(ComponentName name) {
36             Log.d(TAG, "onServiceDisconnected: ");
37             iAidlColorService = null;
38         }
39     }
```

```

39     };
40
41     @Override
42     protected void onCreate(Bundle savedInstanceState) {
43         super.onCreate(savedInstanceState);
44         EdgeToEdge.enable(this);
45         setContentView(R.layout.activity_main);
46         ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -
47     > {
48         Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
49         v.setPadding(systemBars.left, systemBars.top, systemBars.right,
50     systemBars.bottom);
51         return insets;
52     });
53
54     Intent intent = new Intent("com.dingjun.myaidlserver.MyAIDLColorService");
55     intent.setPackage("com.dingjun.myaidlserver");
56
57     bindService(intent, mConnection, BIND_AUTO_CREATE);
58     Log.d(TAG, "onCreate: bindservice called...");
59
60     Button btnColor = findViewById(R.id.btn_color);
61     btnColor.setOnClickListener(new View.OnClickListener() {
62         @Override
63         public void onClick(View v) {
64             try {
65                 int color = iAidlColorService.getColor();
66                 v.setBackgroundColor(color);
67             } catch (RemoteException re) {
68                 re.printStackTrace();
69             }
70         }
71     });
72 }

```

5. 在模拟器里将服务端运行起来，然后再运行客户端；



4:18



LTE



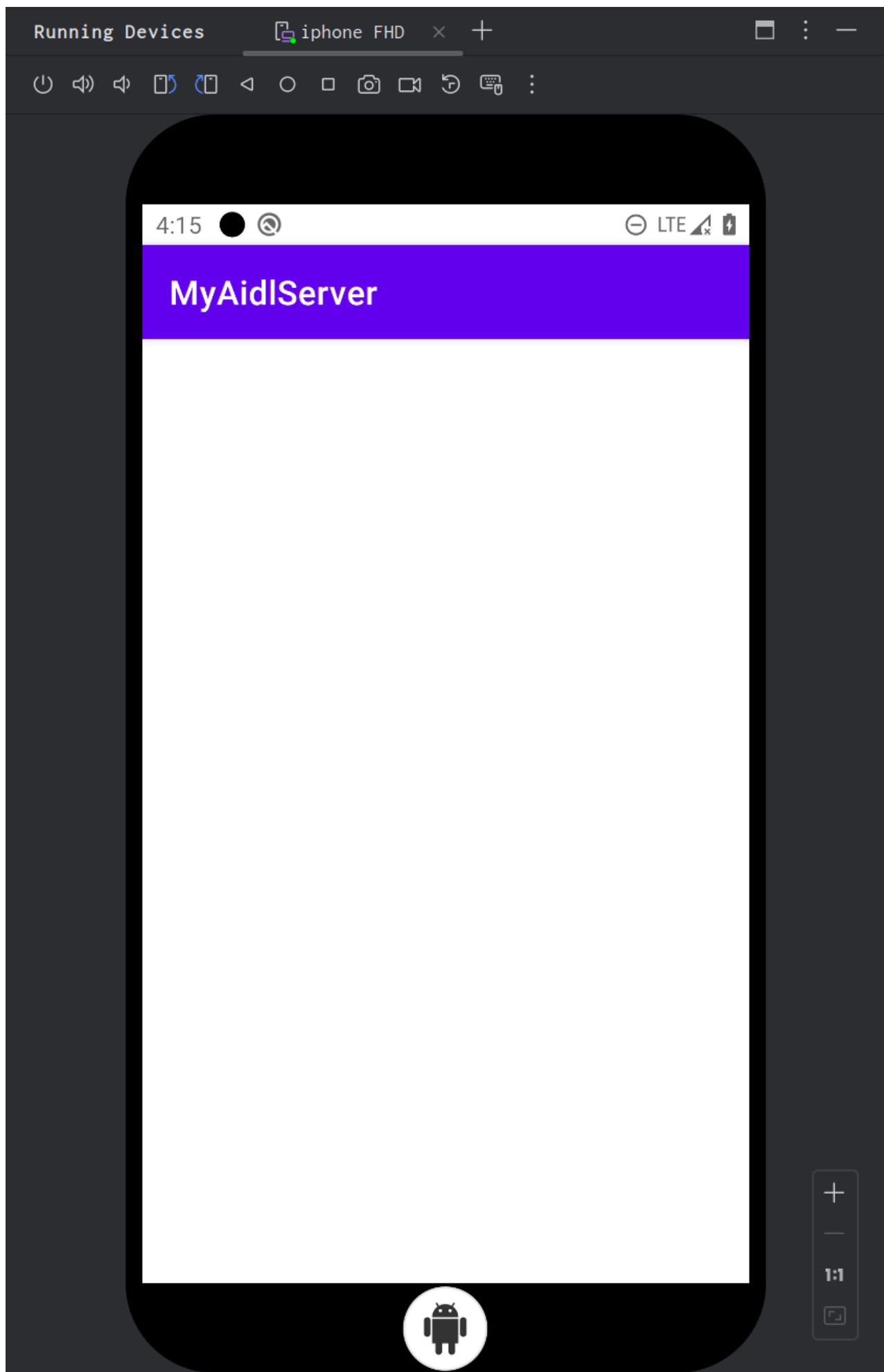
My Aidl Client

Change Button Color

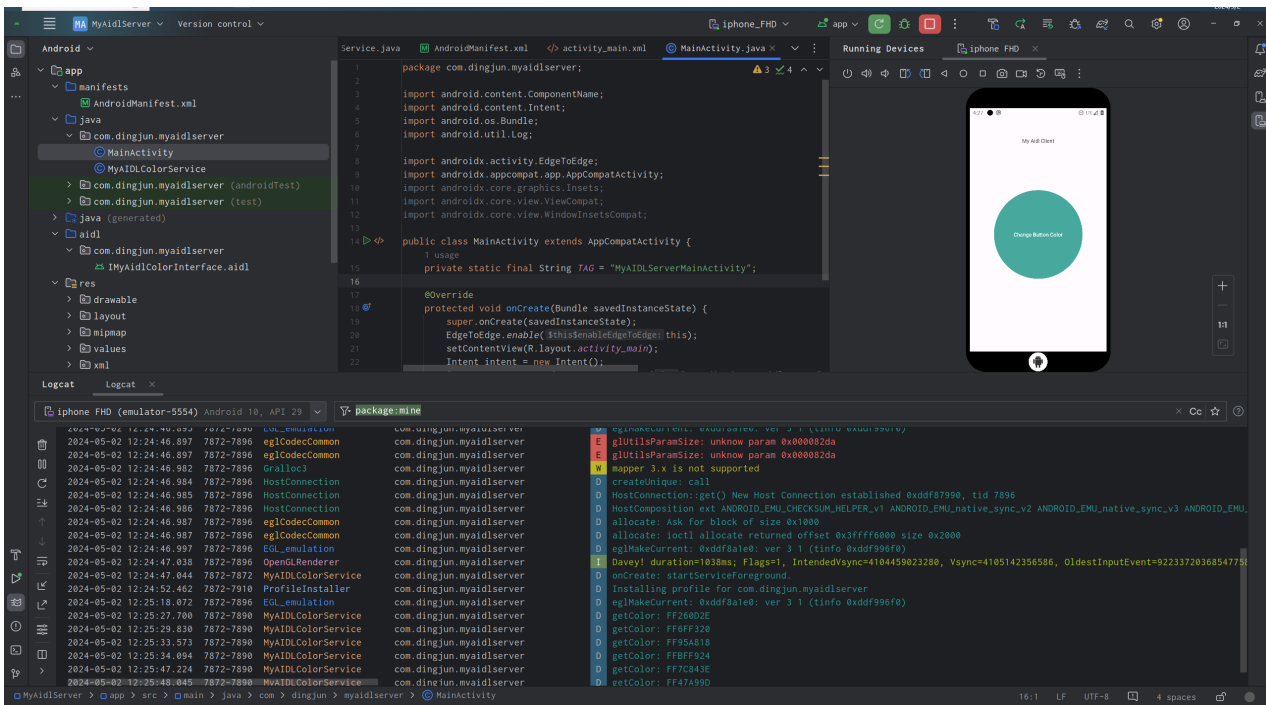


1:1





模拟器中client和server端通信成功！



6. 至此，该版本在android10以下版本，包含android10的真机能够运行成功，但是android11~android13的真机里面就会闪退！

报错log如下：

----- beginning of crash

05-02 12:35:18.003 10378 10378 E AndroidRuntime: FATAL EXCEPTION: main

05-02 12:35:18.003 10378 10378 E AndroidRuntime: Process: com.dingjun.myaidservice, PID: 10378

05-02 12:35:18.003 10378 10378 E AndroidRuntime: java.lang.NullPointerException: Attempt to invoke interface method 'int com.dingjun.myaidservice.IMyAidIColorInterface.getColor()' on a null object reference

05-02 12:35:18.003 10378 10378 E AndroidRuntime: at com.dingjun.myaidservice.MainActivity\$2.onClick(MainActivity.java:63)

05-02 12:35:18.003 10378 10378 E AndroidRuntime: at android.view.View.performClick(View.java:7527)

05-02 12:35:18.003 10378 10378 E AndroidRuntime: at com.google.android.material.button.MaterialButton.performClick(MaterialButton.java:1213)

05-02 12:35:18.003 10378 10378 E AndroidRuntime: at android.view.View.performClickInternal(View.java:7500)

05-02 12:35:18.003 10378 10378 E AndroidRuntime: at android.view.View.-\$\$Nest\$mperformClickInternal(Unknown Source:0)

05-02 12:35:18.003 10378 10378 E AndroidRuntime: at android.view.View\$PerformClick.run(View.java:29538)

05-02 12:35:18.003 10378 10378 E AndroidRuntime: at android.os.Handler.handleCallback(Handler.java:942)

05-02 12:35:18.003 10378 10378 E AndroidRuntime: at android.os.Handler.dispatchMessage(Handler.java:99)

```
05-02 12:35:18.003 10378 10378 E AndroidRuntime: at android.os.Looper.loopOnce(Looper.java:201)
05-02 12:35:18.003 10378 10378 E AndroidRuntime: at android.os.Looper.loop(Looper.java:288)
05-02 12:35:18.003 10378 10378 E AndroidRuntime: at
android.app.ActivityThread.main(ActivityThread.java:7941)
05-02 12:35:18.003 10378 10378 E AndroidRuntime: at java.lang.reflect.Method.invoke(Native
Method)
05-02 12:35:18.003 10378 10378 E AndroidRuntime: at
com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run(RuntimeInit.java:569)
05-02 12:35:18.003 10378 10378 E AndroidRuntime: at
com.android.internal.os.ZygoteInit.main(ZygoteInit.java:1019)
05-02 12:35:18.003 620 694 I libPowerHal: [PD] MTKPOWER_HINT_APP_TOUCH update
cmd:3408600 param:0
```



logcat-AIDL.txt

449.97KB

关于这个报错原因是参考连接：<https://blog.csdn.net/braintt/article/details/120949748>

Android 11 Unable to start service Intent 问题(包可见性, resolveIntent返回null)

今天在Android11上发现了一个的问题，如果目标Service的进程没有启动，那么无论是bindService还是startService都没有办法拉起指定的Service。

网上查了很多资料如下：

- 1.目标Service 设置 android:exported="true"
- 2.目标Service需要声明自定义权限。客户端需要声明权限。
- 3.目标Service需要添加<intent-filter></intent-filter>

上面的方法都试过了**然并卵**，还是报Unable to start service Intent.

实在没办法去翻了下源码~在startService过程中当调用调用pms去解析intent是返回null会打印这个日志。代码片段如下~

ActiveServices # retrieveServiceLocked

```

1      ResolveInfo rInfo = mAm.getPackageManagerInternalLocked().resolveService(service,
2          resolvedType, flags, userId, callingUid);
3      ServiceInfo sInfo = rInfo != null ? rInfo.serviceInfo : null;
4      if (sInfo == null) {
5          Slog.w(TAG_SERVICE, "Unable to start service " + service + " U=" + userId +
6              ": not found");
7          return null;
8      }
9      //....代码省略
10
11

```

PackageManagerService # resolveServiceInternal

```

1      private ResolveInfo resolveServiceInternal(Intent intent, String resolvedType, int
2          flags,
3          int userId, int callingUid) {
4          if (!mUserManager.exists(userId)) return null;
5          flags = updateFlagsForResolve(flags, userId, callingUid, false
6              /*includeInstantApps*/,
7              false /* isImplicitImageCaptureIntentAndNotSetByDpc */);
8          List<ResolveInfo> query = queryIntentServicesInternal(
9              intent, resolvedType, flags, userId, callingUid, false
10              /*includeInstantApps*/);
11          if (query != null) {
12              if (query.size() >= 1) {
13                  // If there is more than one service with the same priority,
14                  // just arbitrarily pick the first one.
15                  return query.get(0);
16              }
17          }
18          return null;
19      }
20

```

看到上面源码吓的我赶紧搜了搜Android11 resolveIntent返回null的问题~

果然不出所料，Android 11引入了**包可见性**

为啥引入包可见性呢？

Goole给出的原因：

- 1.鼓励最小权限原则，需要与那些应用交互，就申请那些包名。
 - 2.帮助 Google Play 等应用商店评估应用的隐私性和安全性。
- 好啦，知道了前因后果那么我们来解决问题吧~

Solve方案

方案一：

```
1 <queries>
2     //你要交互的service的包名
3     <package android:name="com.XXX.XXX" />
4     //...等等包名
5 </queries>
```

方案二：

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

配置完了，赶紧拿起手机试了试，重启手机~ bind目标service，终于bind成功了~

上面的包可见性问题不仅Service有这个问题，Activity也有这么问题哈，记录一下，避免有朋友也遇到同样的问题。

7. 基于以上分析，在客户端的AndroidManifest.xml添加上述解决方案，任意其一即可：

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools">
4      <!-- 添加 start-->
5      <uses-permission android:name="android.permission.QUERY_ALL_PACKAGES"
6          tools:ignore="QueryAllPackagesPermission" />
7      <!-- 添加 end-->
8      <application
9          android:allowBackup="true"
10         android:dataExtractionRules="@xml/data_extraction_rules"
11         android:fullBackupContent="@xml/backup_rules"
12         android:icon="@mipmap/ic_launcher"
13         android:label="@string/app_name"
14         android:roundIcon="@mipmap/ic_launcher_round"
15         android:supportsRtl="true"
16         android:theme="@style/Theme.MyAidlClient"
17         tools:targetApi="31">
18         <activity
19             android:name=".MainActivity"
20             android:exported="true">
21             <intent-filter>
22                 <action android:name="android.intent.action.MAIN" />
23                 <category android:name="android.intent.category.LAUNCHER" />
24             </intent-filter>
25         </activity>
26     </application>
27 </manifest>

```

或者

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools">
4      <!-- 添加 start-->
5      <queries>
6          <package android:name="com.dingjun.myaidlserver"/>
7      </queries>
8      <!-- 添加 end-->
9      <application
10         android:allowBackup="true"
11         android:dataExtractionRules="@xml/data_extraction_rules"
12         android:fullBackupContent="@xml/backup_rules"
13         android:icon="@mipmap/ic_launcher"
14         android:label="@string/app_name"
15         android:roundIcon="@mipmap/ic_launcher_round"
16         android:supportsRtl="true"
17         android:theme="@style/Theme.MyAidlClient"
18         tools:targetApi="31">
19         <activity
20             android:name=".MainActivity"
21             android:exported="true">
22             <intent-filter>
23                 <action android:name="android.intent.action.MAIN" />
24                 <category android:name="android.intent.category.LAUNCHER" />
25             </intent-filter>
26         </activity>
27     </application>
28 </manifest>

```

至此，问题完美解决！

完整工程代码放到我的github上 <https://github.com/hgddingjun/MyAIDL>