

ButterKnife: a View binding Library

Butter Knife 是一个流行的 Android 视图绑定库，简化了将 UI 元素绑定到代码中的字段的过程。然而，请注意，Butter Knife 现在已被弃用，现代 Android 开发建议使用 Android Jetpack 中引入的 View Binding 代替。尽管如此，如果你需要在 Android 项目中使用 Butter Knife 6.1.0（一个在 2015 年左右发布的旧版本）和一个 .jar 文件，这里是你在 2025 年 3 月 3 日的 Android Studio 中可以做到的方法。

前提条件

- 安装了 Android Studio（任何较新的版本都应该可以工作，尽管根据你的版本，过程可能会有所不同）。
- 设置了一个基本的 Android 项目。
- 下载了 butterknife-6.1.0.jar 文件。你通常可以在像 Maven Central 这样的仓库中找到旧版本，或者通过归档源（如果你本地有 .jar 文件）。

在 Android 开发中使用 butterknife-6.1.0.jar 的步骤

步骤 1：将 .jar 文件添加到你的项目中

1. 定位 libs 文件夹：

- 在你的 Android Studio 项目中，导航到 app 模块。
- 在 app 文件夹中，找到或创建一个名为 libs 的文件夹。如果它不存在，右键点击 app 文件夹，选择 New > Directory，并将其命名为 libs。

2. 复制 .jar 文件：

- 将 butterknife-6.1.0.jar 文件复制到 libs 文件夹中。你可以通过将文件拖放到 Android Studio 中的 libs 文件夹中，或者通过文件资源管理器手动将其放置在那里。

3. 同步 .jar 文件与 Gradle：

- 打开 app 模块的 build.gradle 文件（位于 app/build.gradle）。
- 在 dependencies 块下添加以下行，以包含 libs 文件夹中的所有 .jar 文件：

```
dependencies {  
    compile fileTree(dir: 'libs', include: ['*.jar'])  
}
```

- 通过点击 Android Studio 中的“Sync Project with Gradle Files”按钮来同步你的项目。

步骤 2：配置你的项目 由于 Butter Knife 6.1.0 使用注释处理，你不需要为这个特定版本添加注释处理器依赖（与 8.x 及以上的版本不同）。.jar 文件包含运行时库，Butter Knife 6.1.0 依赖运行时反射而不是编译时代码生成来实现大部分功能。

然而，确保你的项目设置为支持 Java 注释：- 在你的 app/build.gradle 中，确保 Java 版本兼容（Butter Knife 6.1.0 与 Java 6+ 兼容）：

```
gradle android {  
    compileOptions {  
        sourceCompatibility  
        JavaVersion.VERSION_1_6  
        targetCompatibility JavaVersion.VERSION_1_6  
    }  
}
```

步骤 3：在代码中使用 Butter Knife

1. 添加 Butter Knife 注释：

- 在你的 Activity 或 Fragment 中，导入 Butter Knife 并使用 `@InjectView`（该版本中使用的注释）对你的视图进行注释。例如：

```
import android.os.Bundle;
import android.widget.Button;
import android.widget.TextView;
import butterknife.InjectView;
import butterknife.ButterKnife;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    @InjectView(R.id.my_button)
    Button myButton;

    @InjectView(R.id.my_text)
    TextView myText;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ButterKnife.inject(this); // 绑定视图

        // 示例用法
        myButton.setOnClickListener(v -> myText.setText(" 按钮被点击! "));
    }
}
```

2. XML 布局：

- 确保你的布局文件（例如 `res/layout/activity_main.xml`）包含具有相应 ID 的视图：

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <TextView
        android:id="@+id/my_text"
```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World" />

<Button
    android:id="@+id/my_button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text=" 点击我" />
</LinearLayout>

```

3. 绑定视图：

- onCreate 中的 `ButterKnife.inject(this)` 调用将注释字段绑定到布局中的视图。请注意，在 6.1.0 版本中，使用 `inject` 而不是 `bind`（后者在 7.x 和 8.x 版本中引入）。

步骤 4：运行你的项目

- 构建并运行你的项目。如果一切设置正确，Butter Knife 将绑定你的视图，你应该会看到 UI 正常工作。

重要说明

- **版本限制：**Butter Knife 6.1.0 非常旧（2015 年发布），缺少在后续版本中引入的功能，例如通过注释处理器的编译时代码生成以及对 `@BindView` 而不是 `@InjectView` 的支持。它在运行时使用反射，这可能会比现代替代方案在性能上略有影响。
- **已弃用状态：**Butter Knife 不再维护（开发大约在 2020 年停止），官方建议使用 View Binding 或 Jetpack Compose 进行新项目。
- **替代设置：**如果你使用的是较新的版本（例如 10.2.3），你需要添加注释处理器依赖（例如 `annotationProcessor 'com.jakewharton:butterknife-compiler:10.2.3'`），但对于 6.1.0 版本和 `.jar` 文件，这不是必需的。
- **故障排除：**
 - 如果你在 Butter Knife 类上遇到“无法解析符号”错误，请确保 `.jar` 文件正确添加到 `libs` 文件夹中，并且在 Gradle 依赖中包含。
 - 如果应用程序在 `NoClassDefFoundError` 中崩溃，请双重检查 `.jar` 文件是否在 `libs` 文件夹中，并且项目是否已同步。

推荐

虽然这个指南向你展示了如何使用 `butterknife-6.1.0.jar`，但考虑迁移到 View Binding 以获得更现代、受支持的方法。以下是一个快速比较：- **Butter Knife 6.1.0：**使用 `@InjectView`，运行时反射，手动 `.jar` 集成。- **View Binding：**内置于 Android，无外部依赖，在编译时生成绑定类，并且是空安全的。

你想要关于迁移到 View Binding 的指导吗？