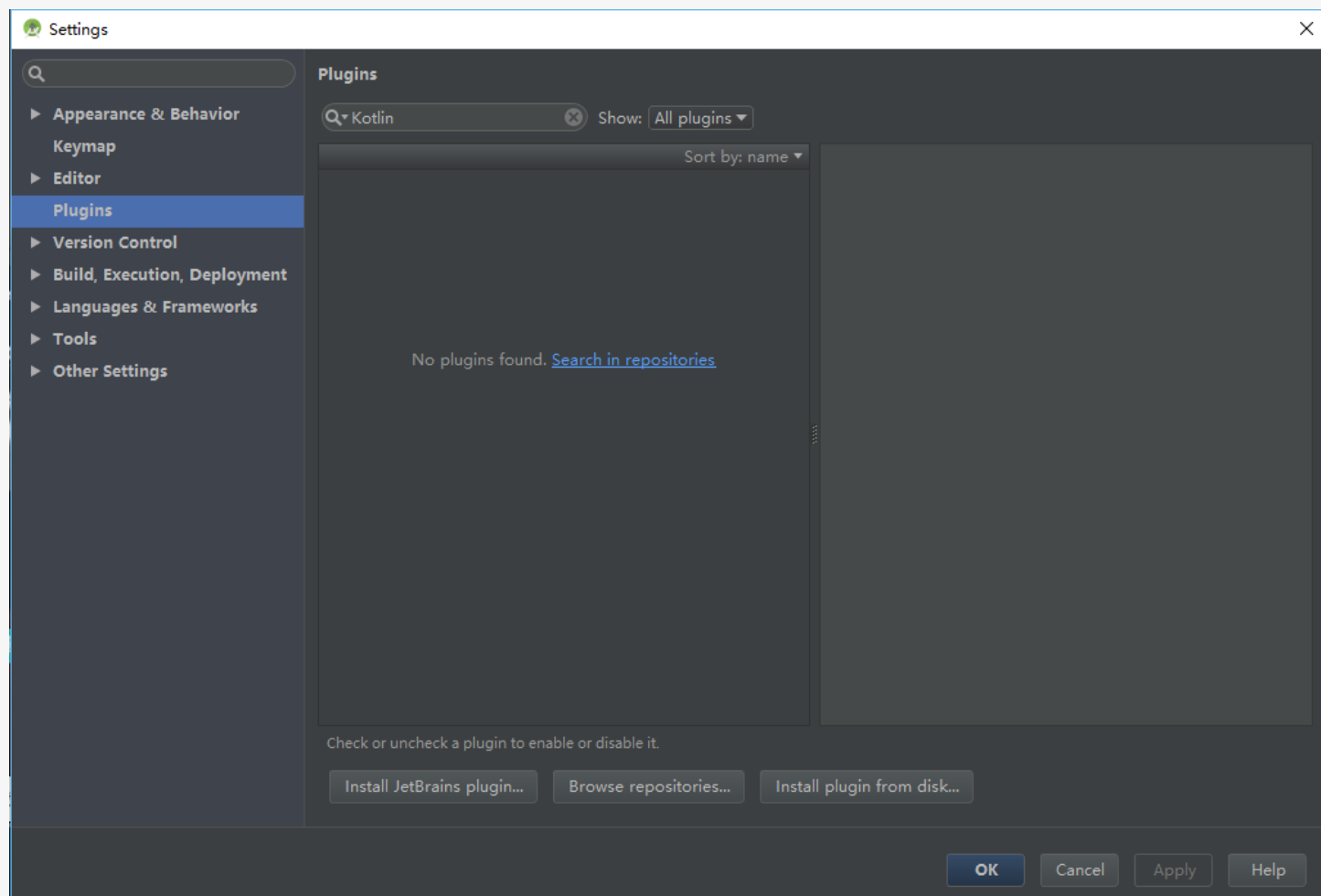


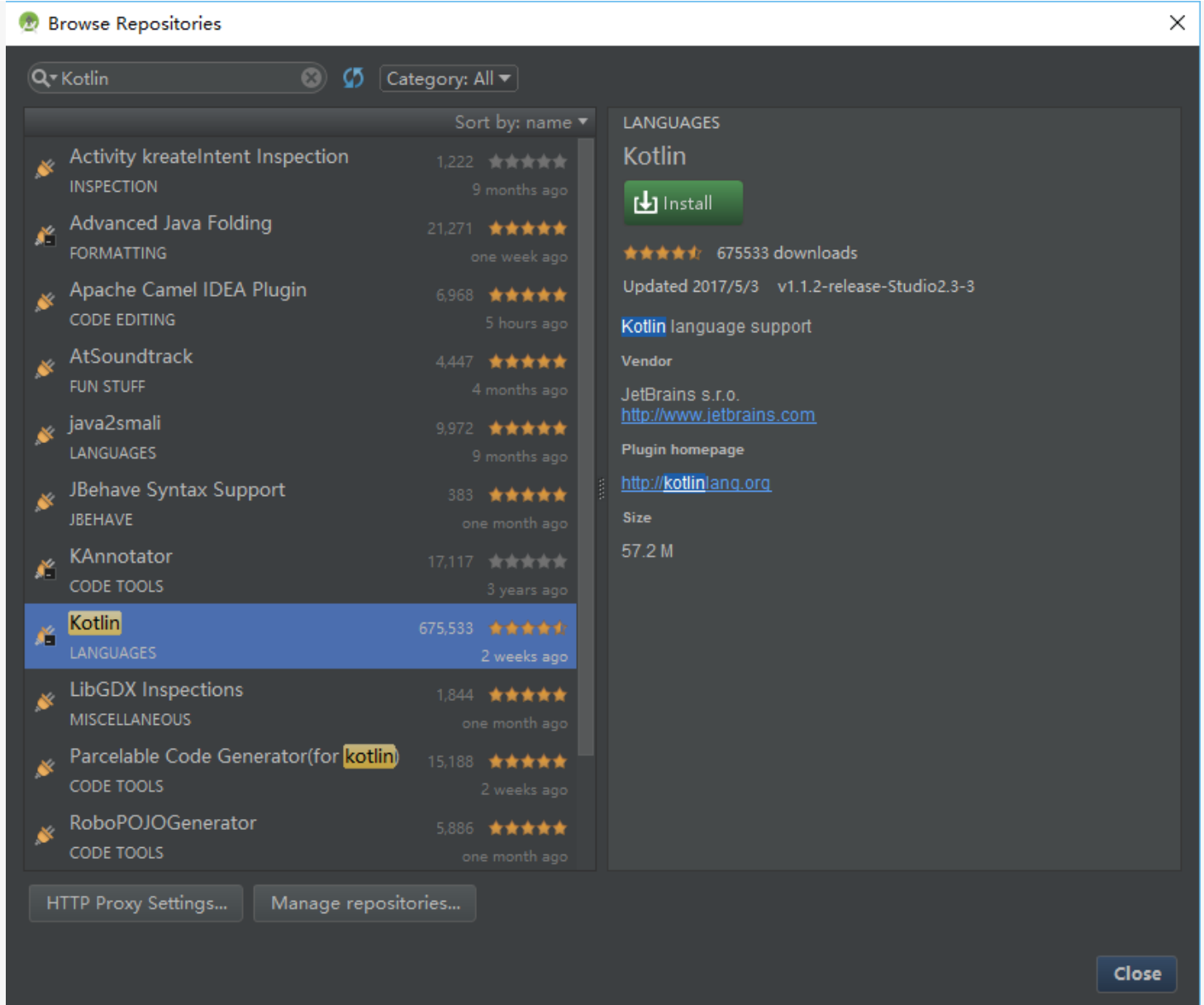
Kotlin Android 环境搭建

安装 Kotlin 插件

Android Studio 从 3.0 (preview) 版本开始将内置安装 Kotlin 插件。

打开 Settings (Mac 为 Preferences) 面板，在右侧找到 Plugins 选项 (快捷键 Ctrl+, Mac 下为 command+)，搜索框输入 "Kotlin" 查找，点击 Search in repositories(仓库中搜索)，然后安装即可，安装完成之后需要重启 Android Studio。





创建新工程

选择 Start a new Android Studio project 或者 File | New project , 大多数选项均有默认值 , 只需要按几次"回车"键即可。



Configure your new project

Application name:

Company domain:

Package name: [Edit](#)

☐ Include C++ support

Project location: ...

[Cancel](#) [Previous](#) [Next](#) [Finish](#)

Android Studio 3.0 在当前对话框中提供启用 Kotlin 支持的选项，勾选后可以跳过 "配置 Kotlin 工程 (Configuring Kotlin in the project)" 的步骤。

选择 Android 版本:



Select the form factors your app will run on


Different platforms may require separate SDKs

- ☒ Phone and Tablet
- Minimum SDK [Help me choose](#)
- Lower API levels target more devices, but have fewer features available.
By targeting API 19 and later, your app will run on approximately 73.9% of the devices that are active on the Google Play Store.
- ☐ Wear
- Minimum SDK
- ☐ TV
- Minimum SDK
- ☐ Android Auto

[Cancel](#) [Previous](#) [Next](#) [Finish](#)

选择需要创建的 Activity 样式:

Create New Project

 **New Project**
Android Studio

Configure your new project

Application name:

Company domain:

Package name: [Edit](#)

☐ Include C++ support

Project location: ...

Cancel


Previous

Next

Finish

命名该 Activity:

Create New Project

 **Target Android Devices**

Select the form factors your app will run on

Different platforms may require separate SDKs

☒ Phone and Tablet

Minimum SDK ▾

Lower API levels target more devices, but have fewer features available.
By targeting API 19 and later, your app will run on approximately **73.9%** of the devices that are active on the Google Play Store.
[Help me choose](#)

☐ Wear

Minimum SDK ▾

☐ TV

Minimum SDK ▾

☐ Android Auto

Cancel

Previous

Next

Finish

在 Android Studio 3.0 中，可以选择使用 Kotlin 创建 activity，因此也不需要"将Java 代码转换为 Kotlin (Converting Java code to Kotlin)"这一步骤。

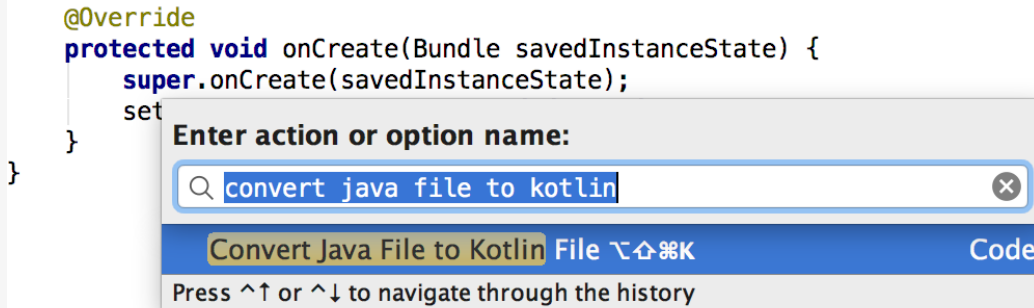
早期版本中则会先使用 Java 创建 activity，然后再使用自动转换工具 进行转换。

将 Java 代码转换为 Kotlin

重新打开Android Studio，新建一个Android项目吧，添加一个默认的MainActivity

打开 MainActivity.java 文件，通过菜单栏依次调出 Code | Convert Java File to Kotlin File：

```
public class MainActivity extends AppCompatActivity {
```



转换完成后即可看到使用 Kotlin 编写的 activity。

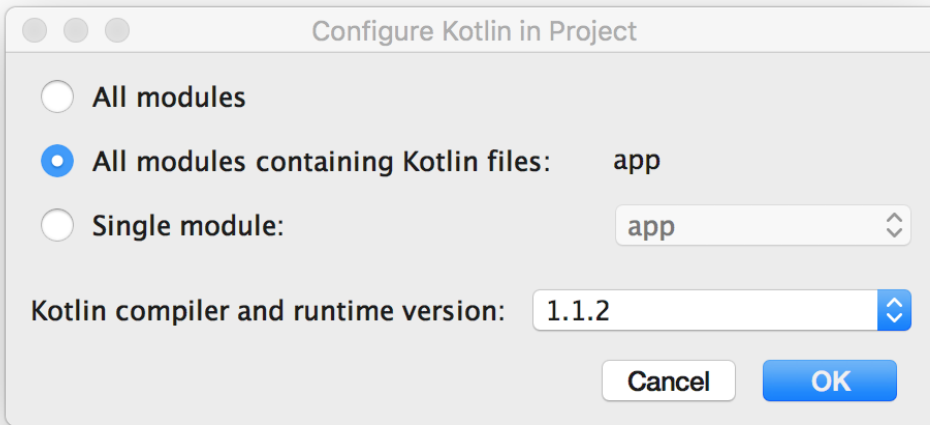
```
class MainActivity : AppCompatActivity() {  
  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.activity_main)  
    }  
}
```

工程中配置 Kotlin

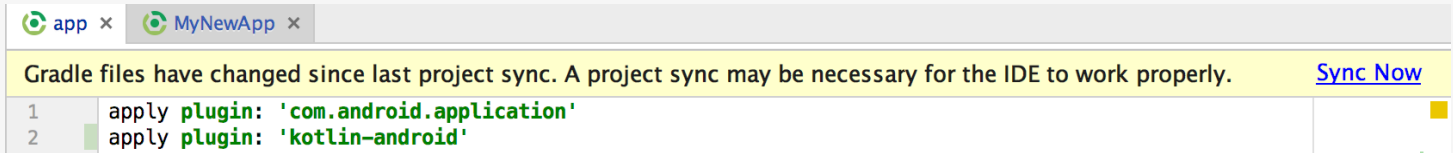
在开始编辑此文件时，Android Studio 会提示当前工程还未配置 Kotlin，根据提示完成操作即可，或者可以在菜单栏中选择 Tools



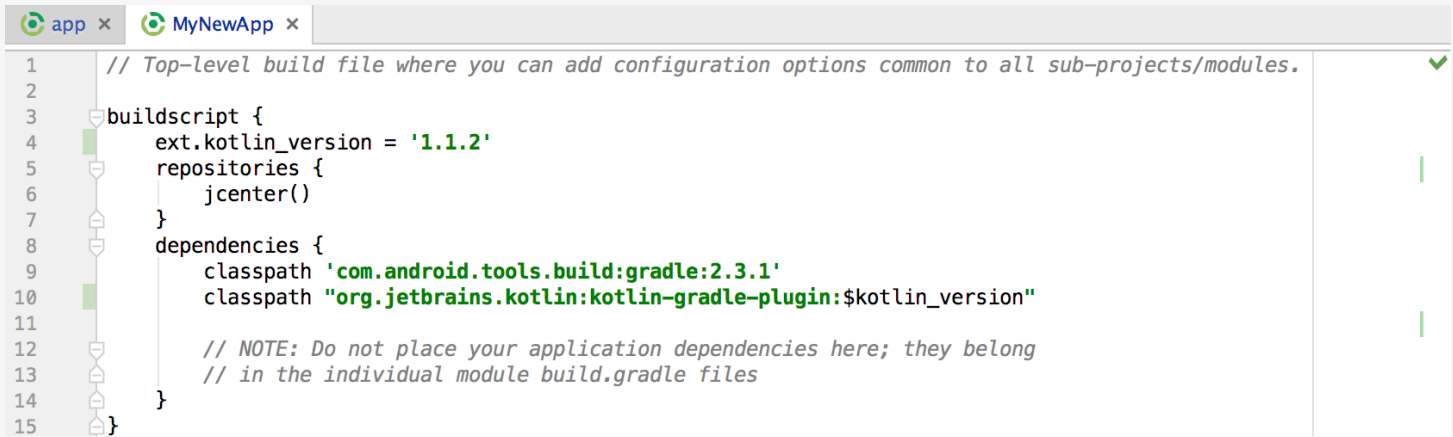
选择配置时有如下对话框，选择已安装的最新版本即可。



Kotlin 配置完成后，应用程序的 build.gradle 文件会更新。你能看到新增了 apply plugin: 'kotlin-android' 及其依赖。



同步工程，在提示框中点击"立即同步 (Sync Now)"或者使用 Sync Project with Gradle Files命令。



← Kotlin 使用命令行编译

Kotlin 基础语法 →

点我分享笔记