



福建師範大學  
FUJIAN NORMAL UNIVERSITY

# 构建首个Kotlin应用

# 实验内容

---

- 安装Android Studio 4.1+ 以上版本，后续更好的使用TensorFlow Lite
- 按照[教程](#)构建首个Kotlin交互应用
- 上传代码至Github，并撰写详细的Readme文档

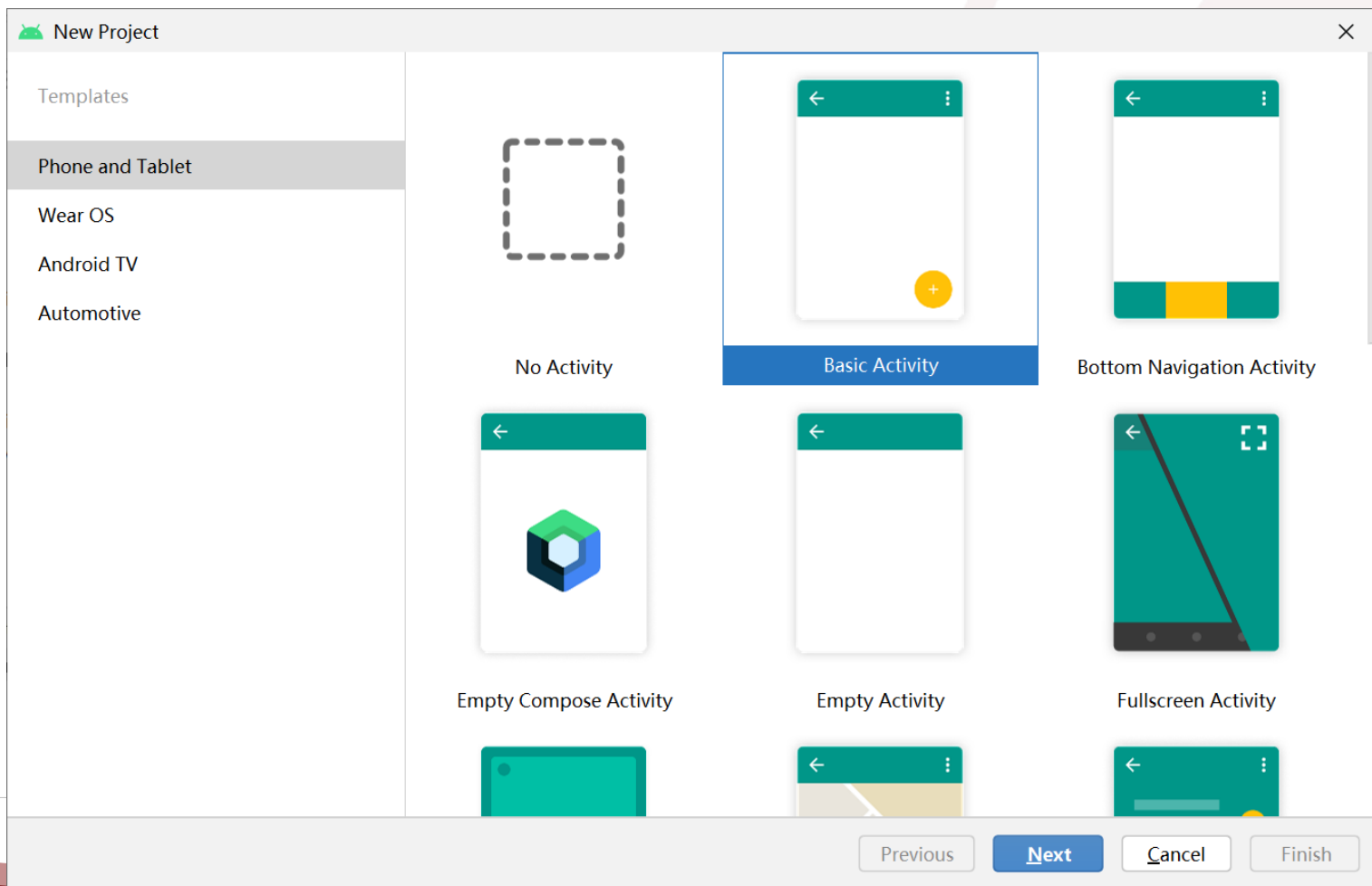
# 实验目的

---

- 掌握Android Studio开发应用的基本流程
- 掌握Android Studio开发组件的基本用法
- 初始Kotlin语言的基本要素
- 掌握Android Navigation的基本用法

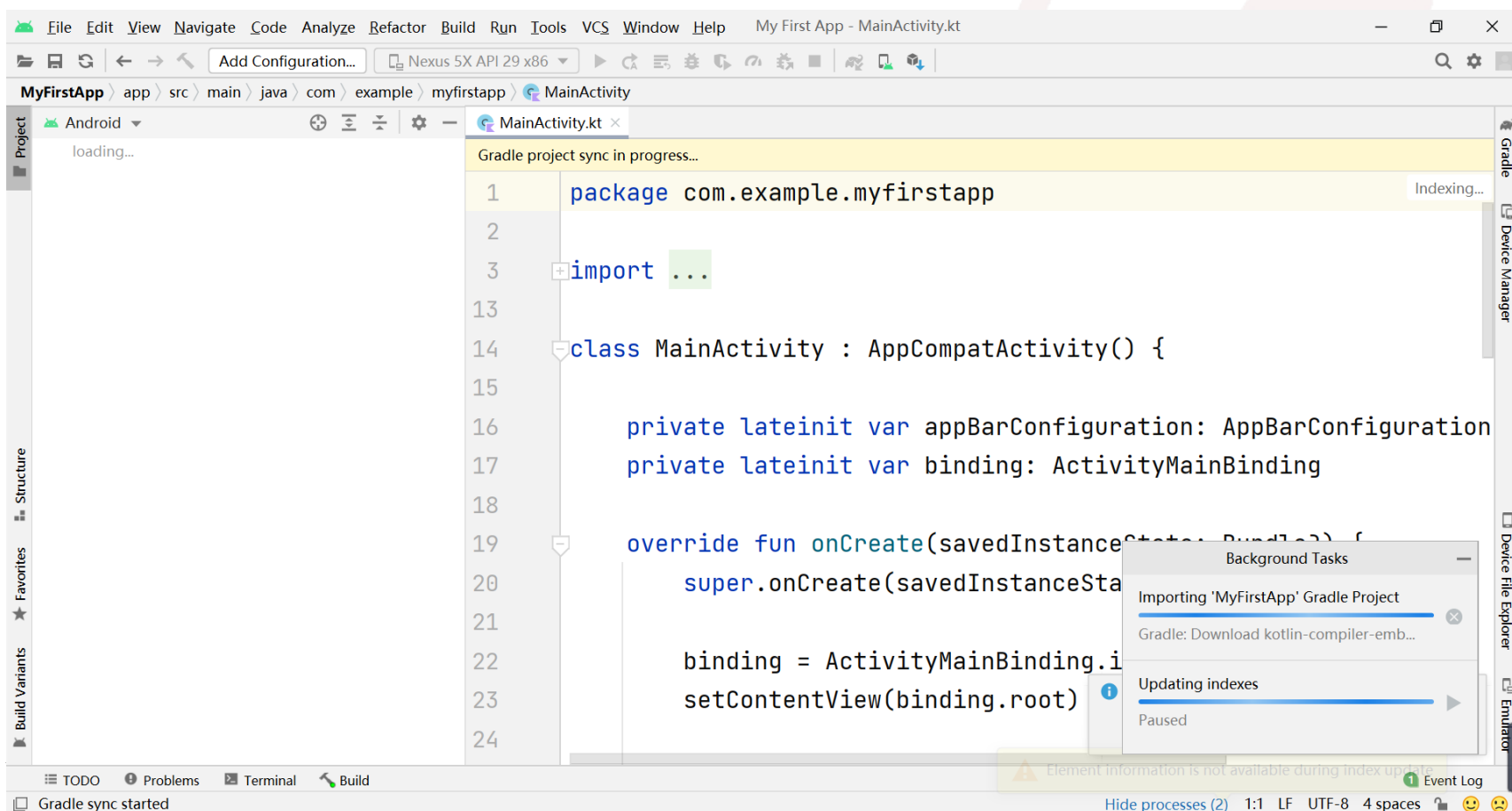
# 创建第一个Kotlin应用

- 选择创建一个Basic Activity，选择Kotlin语言，并命名应用程序



# 编译生成应用

- 第一次 Build Android 应用，将会下载必要的 gradle wrappers和依赖库（需要时间依网络而定）



# 应用程序生成成功

- 生成成功之后呈现如下效果，请查看每个部分的相关代码

The screenshot displays the Android Studio IDE with the following components:

- Project View (Left):** Shows the project structure for 'MyFirstApp'. Annotations include:
  - 清单文件 (Manifest file):** Points to 'AndroidManifest.xml'.
  - 源代码文件 (Source code file):** Points to the 'com.example.myfirstapp' package containing 'FirstFragment', 'MainActivity', and 'SecondFragment'.
  - 测试代码 (Test code):** Points to the test directories 'com.example.myfirstapp (androidTest)' and 'com.example.myfirstapp (test)'.
  - gradle相关文件 (Gradle related files):** Points to the 'Gradle Scripts' directory containing 'build.gradle', 'gradle.properties', 'gradle-wrapper.properties', 'proguard-rules.pro', and 'gradle.properties (Project Properties)'.
- Main Editor:** Displays the 'MainActivity.kt' file with the following code:

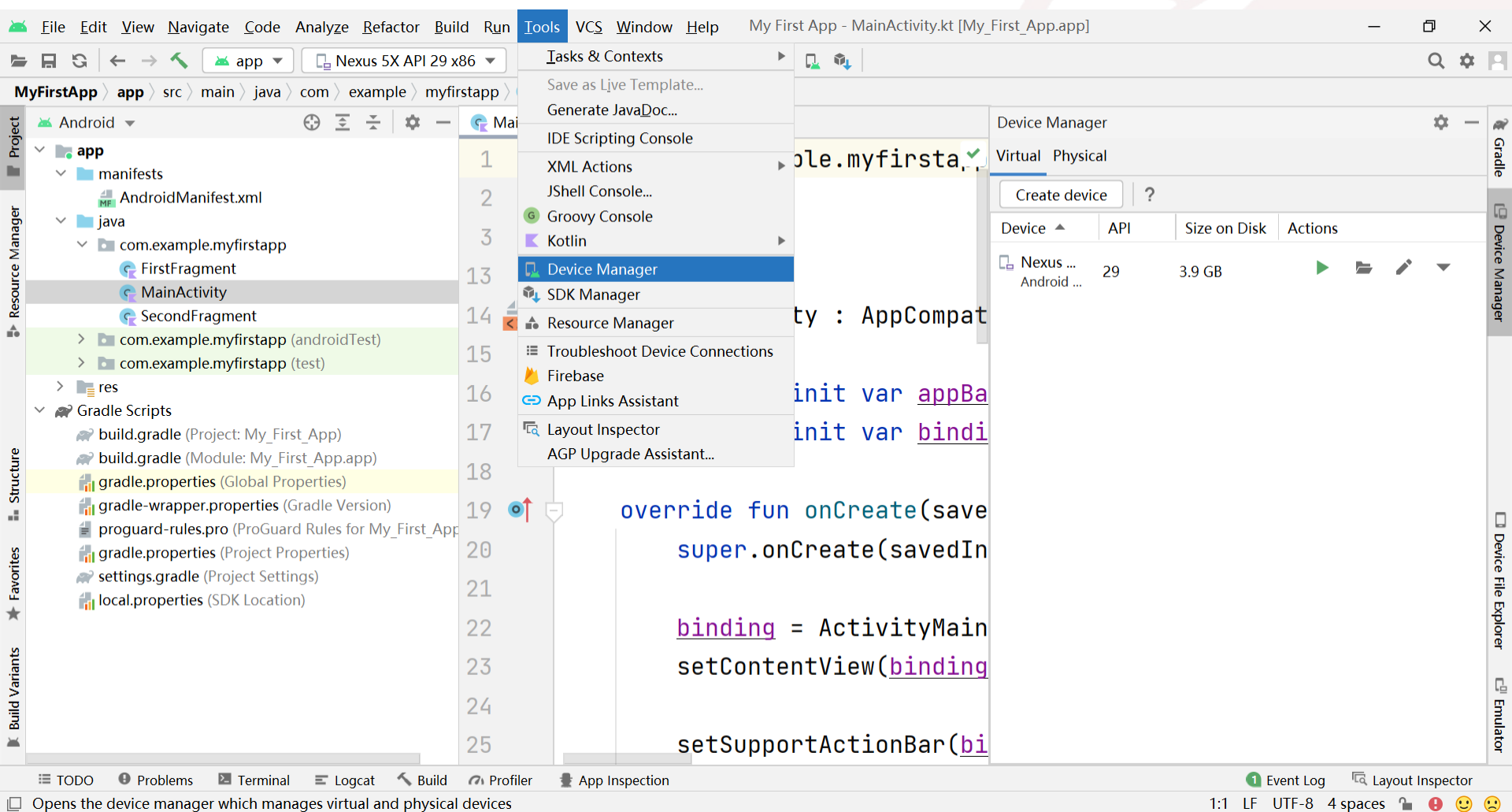
```
1 package com.example.myfirstapp
2
3 import ...
13
14 class MainActivity : AppCompatActivity() {
15
16     private lateinit var appBarConfiguration: AppBarConfiguration
17     private lateinit var binding: ActivityMainBinding
18
19     override fun onCreate(savedInstanceState: Bundle?) {
20         super.onCreate(savedInstanceState)
```
- Build Output (Bottom):** Shows the build process with the following messages:

```
Starting Gradle Daemon...
Gradle Daemon started in 15 s 369 ms

BUILD SUCCESSFUL in 2m 42s
```

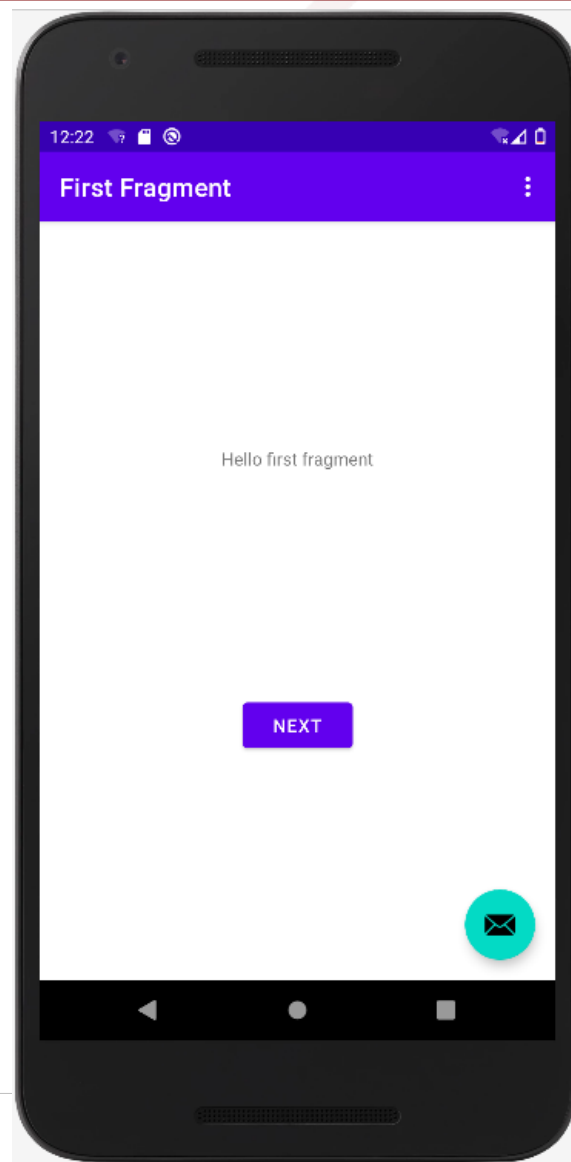
# 创建虚拟设备（模拟器）

- Android Studio支持虚拟设备（模拟器）的创建，以运行应用程序



# 应用程序运行效果图

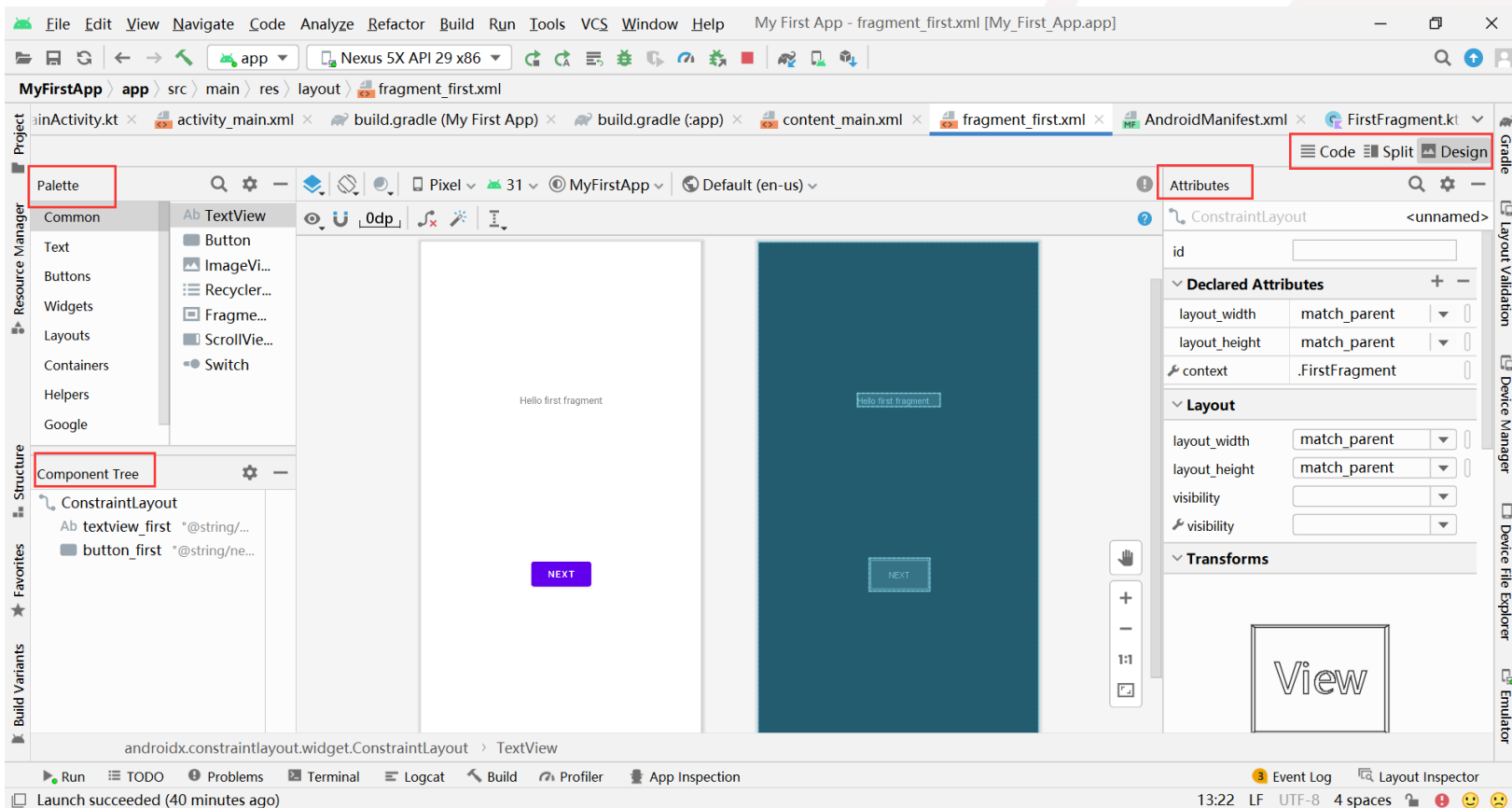
- 应用程序最终的运行效果：





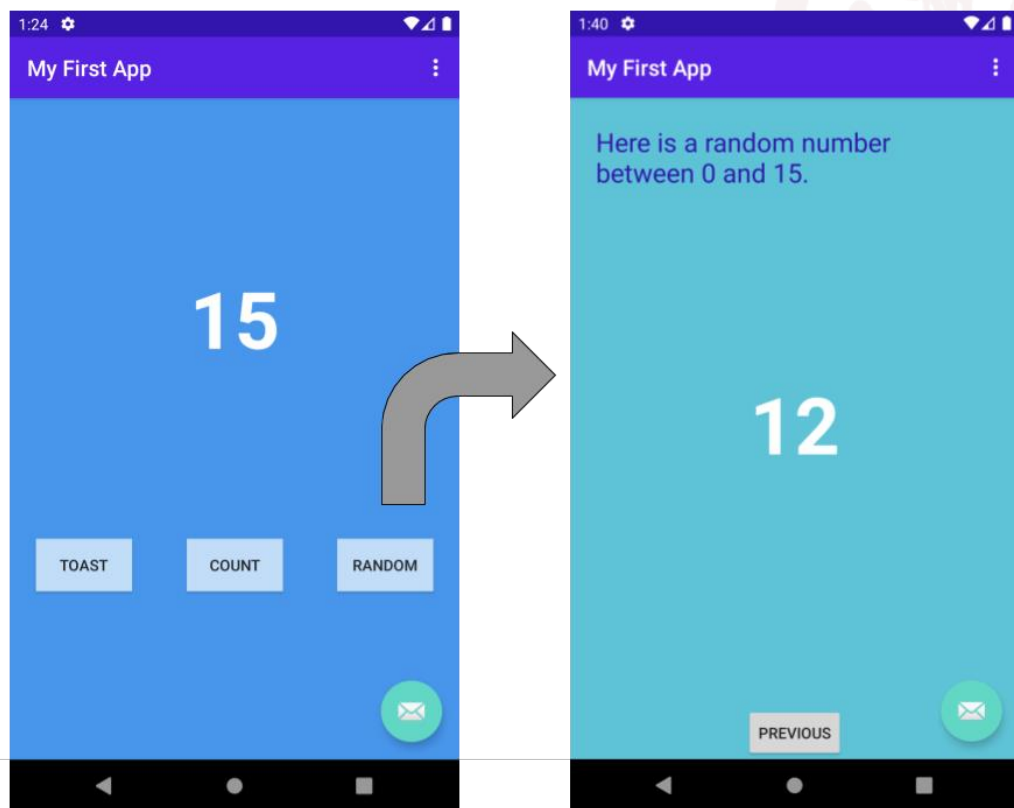
# 探索界面布局编辑器

- 每个界面由一个Fragment组成，初始界面显示的FirstFragment，双击fragment\_first.xml可以查看具体的布局设计界面



# 向页面添加组件并完成交互代码

- 应用程序使用Navigation机制来导航两个页面FirstFragment和SecondFragment，向两个界面添加若干组件，并添加事件代码完成指定功能。



# 参考资源

---

- [Kotlin官方网站](#)
- [Kotlin基本语法](#)
- [Kotlin教程](#)
- [Navigation组件使用入门](#)