



福建师范大学
FUJIAN NORMAL UNIVERSITY

构建第一个Kotlin应用

实验内容

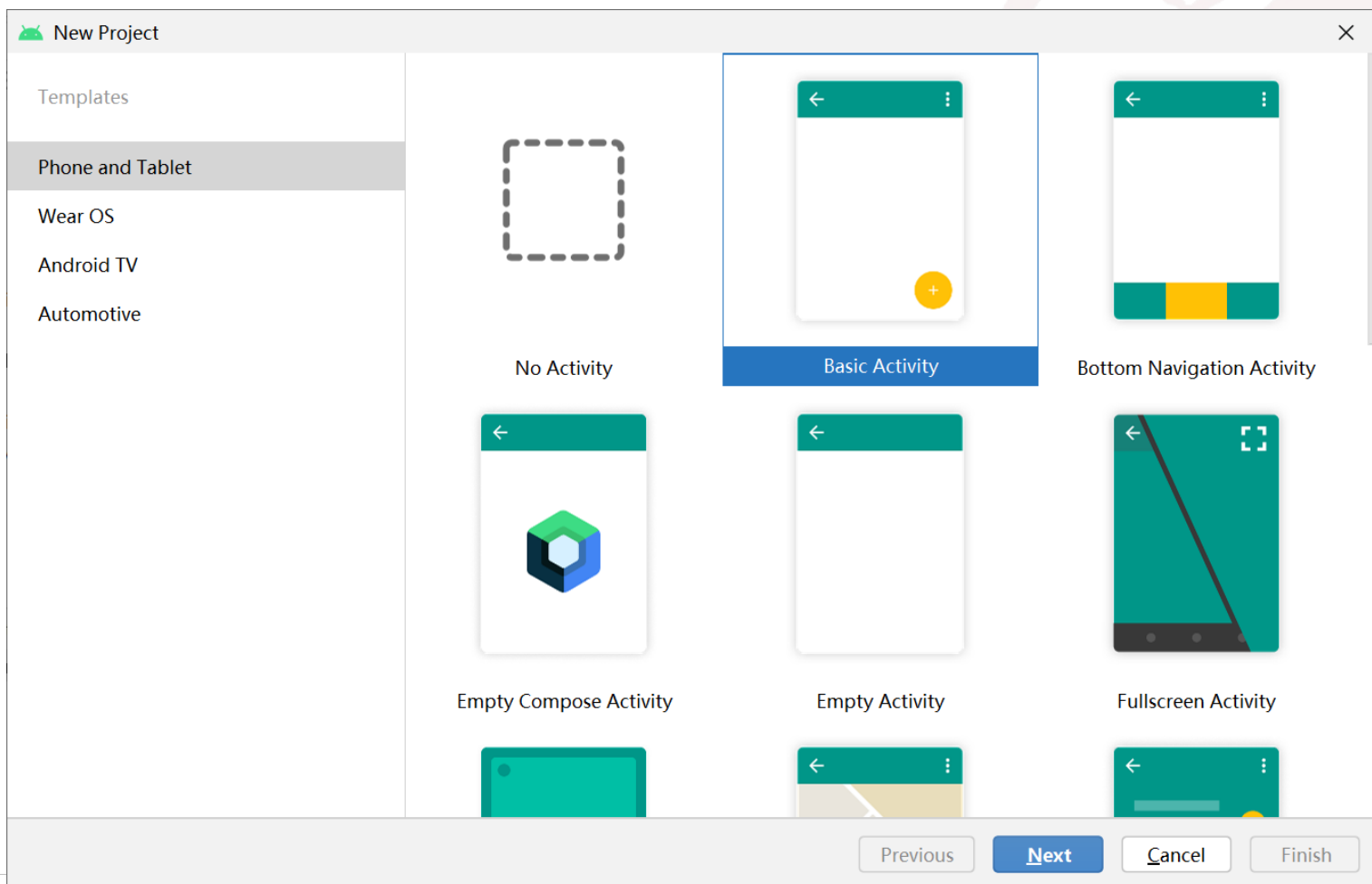
- 安装Android Studio 4.1+ 以上版本，后续更好的使用TensorFlow Lite
- 按照[教程](#)构建首个Kotlin交互应用
- 上传代码至Github，并撰写详细的Readme文档

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

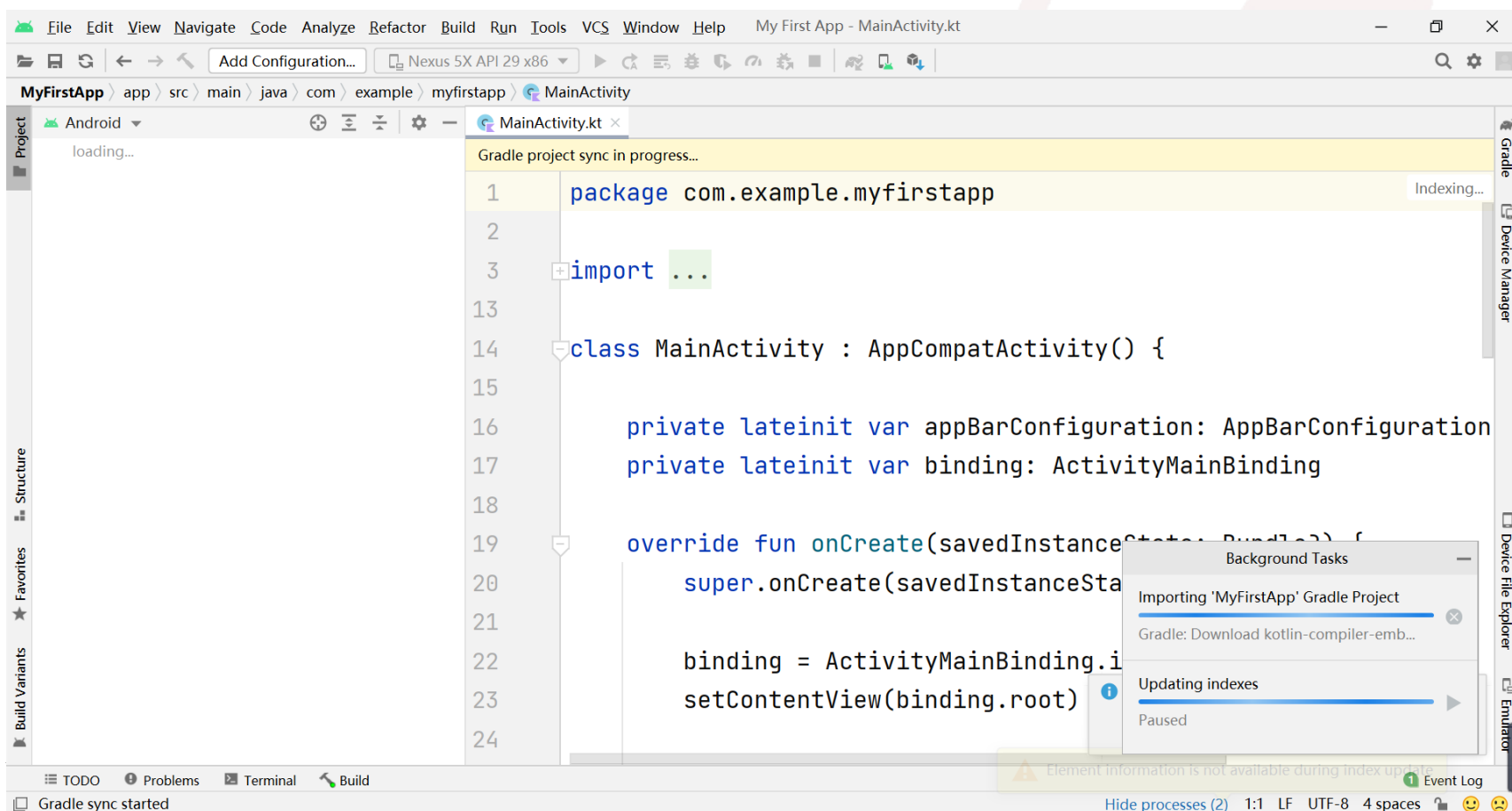
创建第一个Kotlin应用

- 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 an IDE interface for an Android project named "MyFirstApp". The top menu bar includes File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, and Help. The main editor shows the file structure on the left and the source code of MainActivity.kt in the center.

Project Manager (Left):

- Android** (selected)
- app**
 - manifests
 - AndroidManifest.xml (labeled "清单文件" - Manifest file)
 - java
 - com.example.myfirstapp
 - FirstFragment
 - MainActivity (labeled "源代码文件" - Source code file)
 - SecondFragment
 - com.example.myfirstapp (androidTest) (labeled "测试代码" - Test code)
 - com.example.myfirstapp (test)
 - res
 - Gradle Scripts
 - build.gradle (Project: My_First_App)
 - build.gradle (Module: My_First_App.app)
 - gradle.properties (Global Properties)
 - gradle-wrapper.properties (Gradle Version)
 - proguard-rules.pro (ProGuard Rules for My_First_App)
 - gradle.properties (Project Properties)

Main Editor (Center):

```
1 package com.example.myfirstapp
2
3 import ...
4
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 Log (Bottom):

```
Build: Sync x
MyFi 3 min, 6 sec, 472 ms
Starting Gradle Daemon...
Gradle Daemon started in 15 s 369 ms

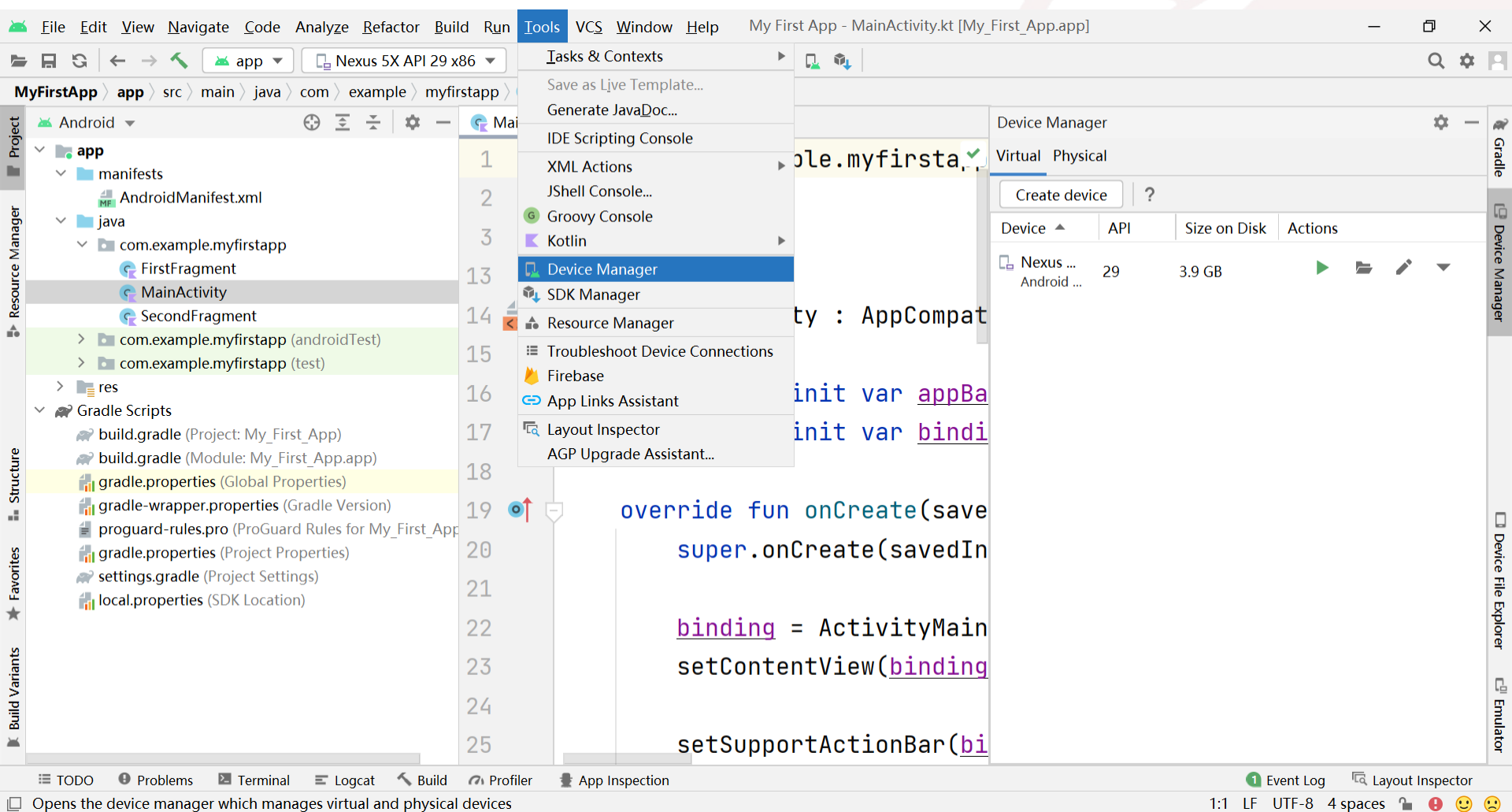
BUILD SUCCESSFUL in 2m 42s
```

Bottom Bar:

- TODO, Problems, Terminal, Logcat, Build, Profiler, App Inspection
- Event Log, Layout Inspector
- 1:1 LF UTF-8 4 spaces

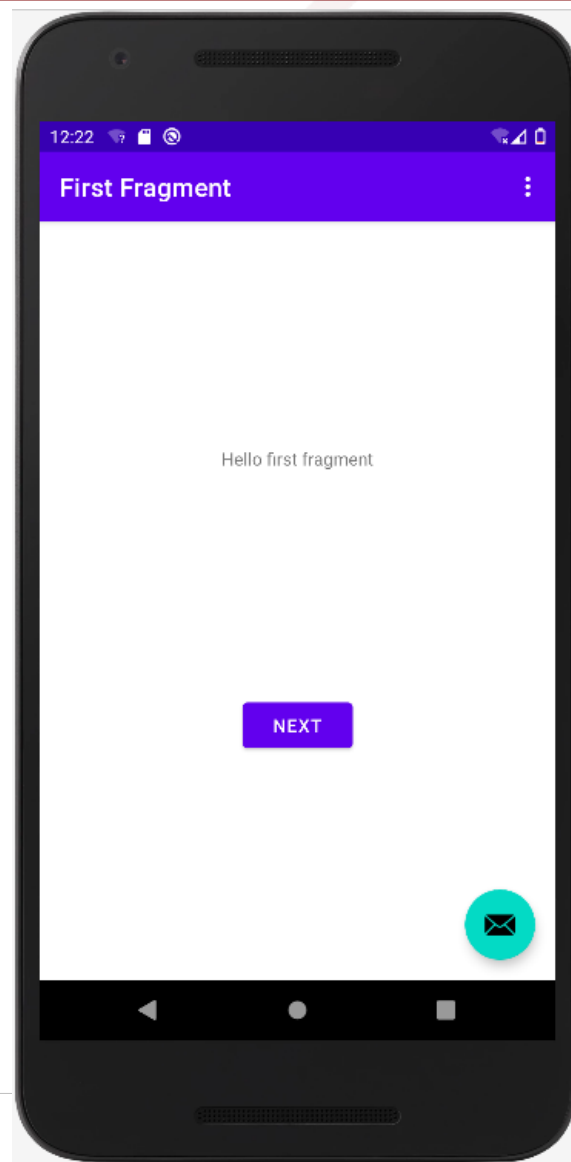
创建虚拟设备（模拟器）

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



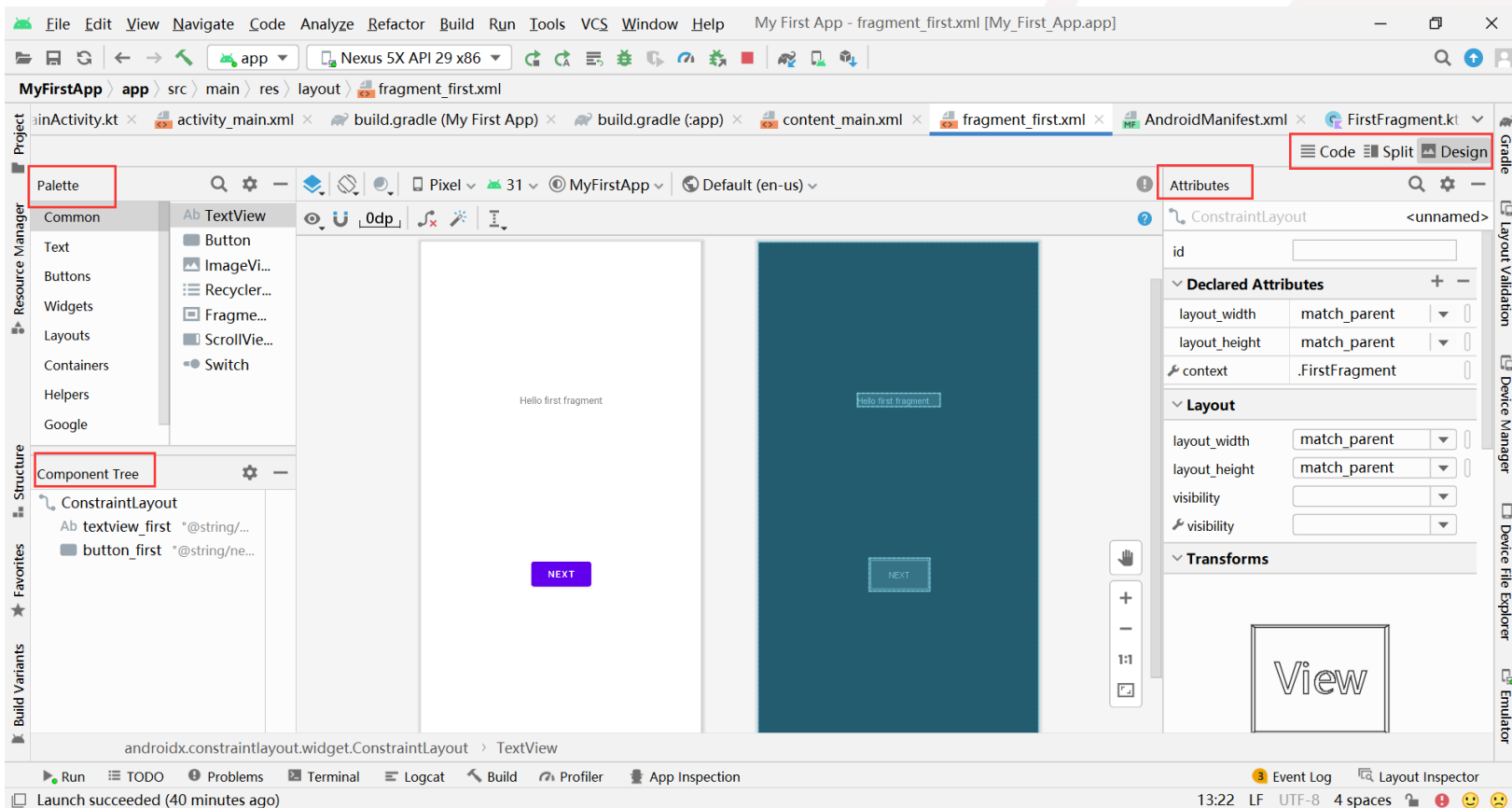
应用程序运行效果图

- 应用程序最终的运行效果：



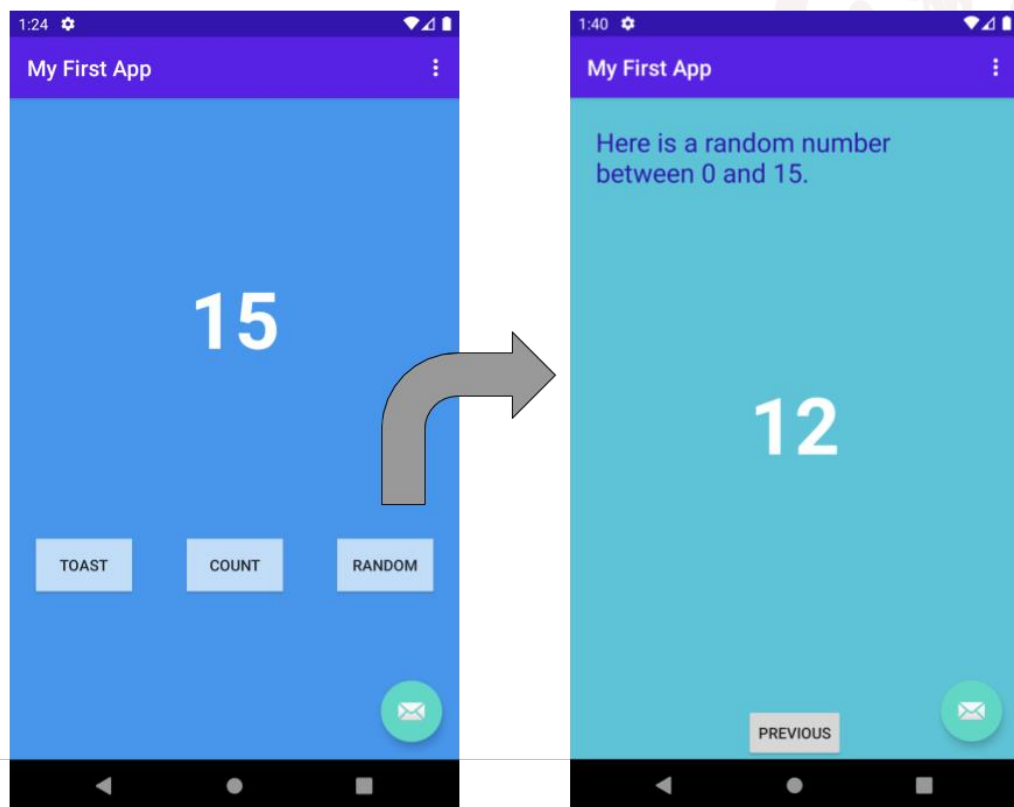
探索界面布局编辑器

- 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](#)