

移动 (()) 开发技术

副教授 孙钰

北京林业大学信息学院





第一章：Hello world

01/ 前置知识与参考资料

02/ Android五层架构

03/ 开发环境

04/ 第一个App

05/ Android模拟器

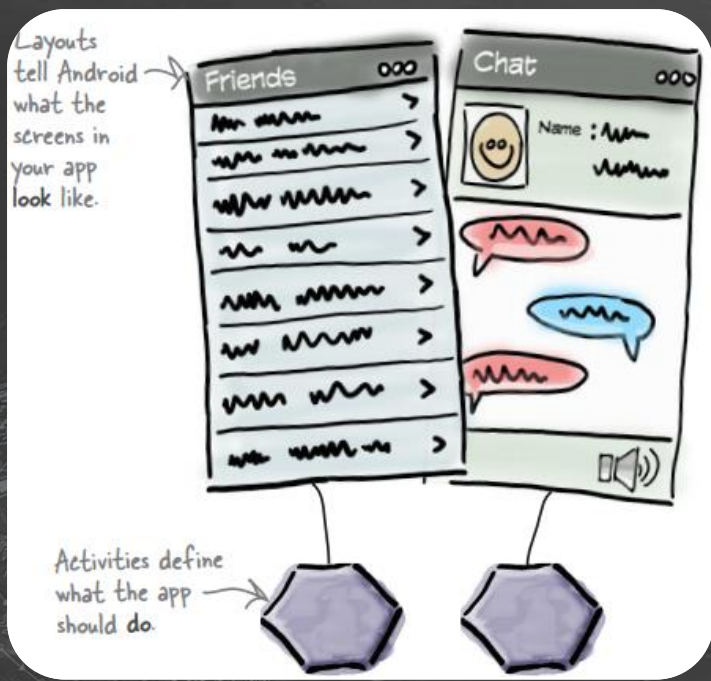
06/ 总结



4、第一个App

► Android开发采用Java和XML混合编程

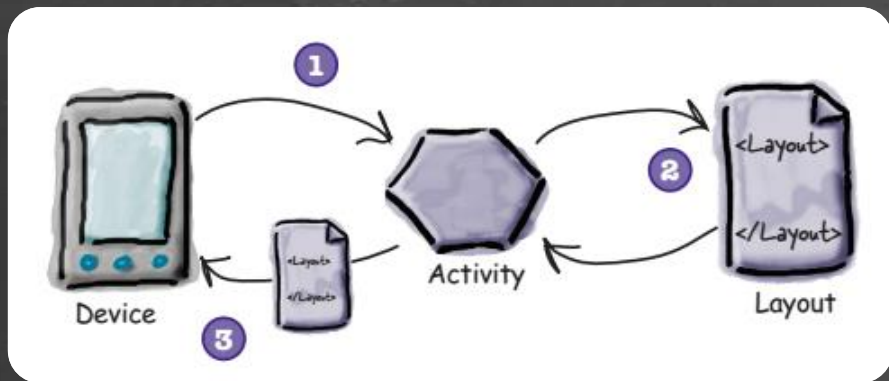
- Layout 使用 XML 定义 App 的外观 (look)
- Activity 使用 Java 定义 App 的行为 (do)



App还需要大量资源文件：
图片、声音等

4、第一个App

App的工作流程：



01 打开App, 系统
新建Activity对象

02 该Activity对象读入
对应的layout

03 系统按Activity要求
在屏幕显示layout

4、第一个App

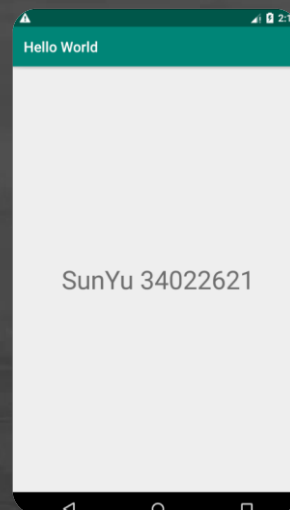
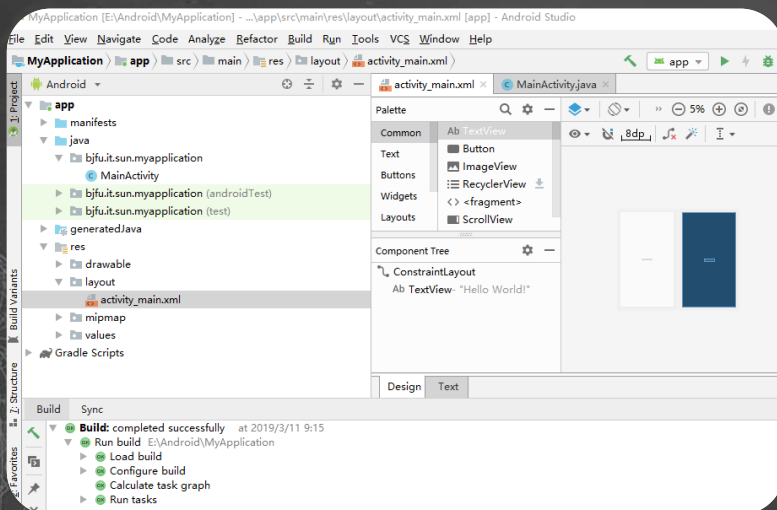
App的开发步骤:

01 建立开发环境
IDE+SDK

02 新建工程
Android 版本 Hello World

03 开发App
修改文本内容

04 运行App
使用模拟器 (AVD) 运行App

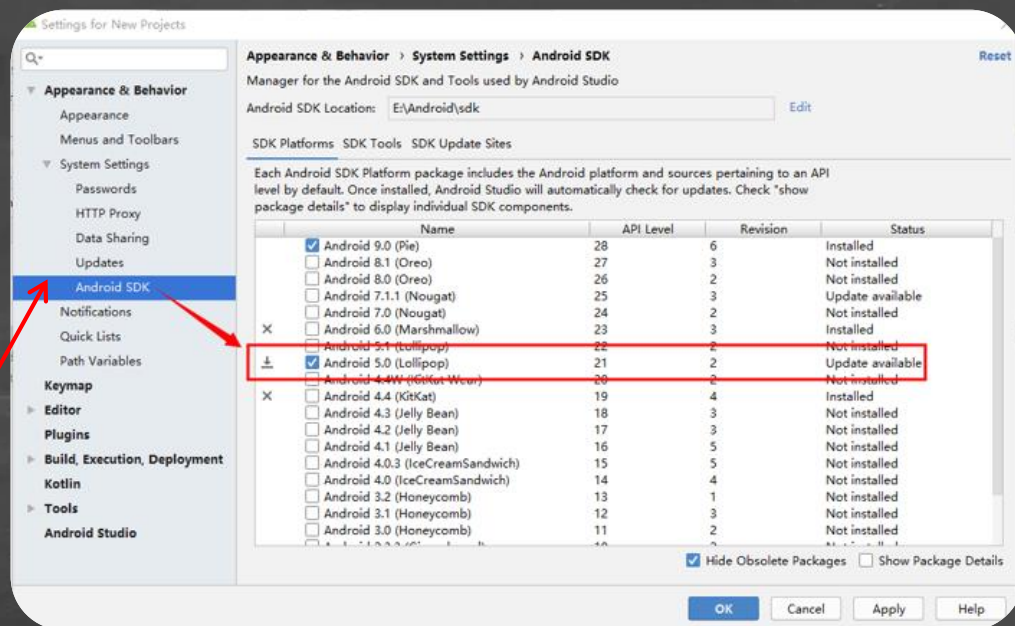
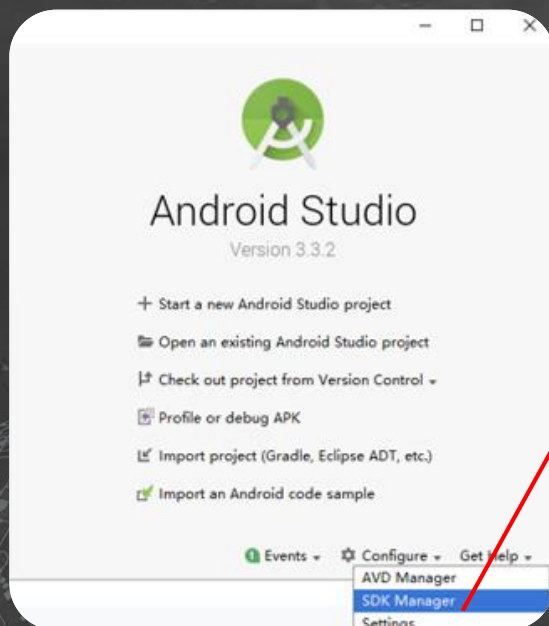




4 第一个App

▶ 1. 建立开发环境: IDE+SDK

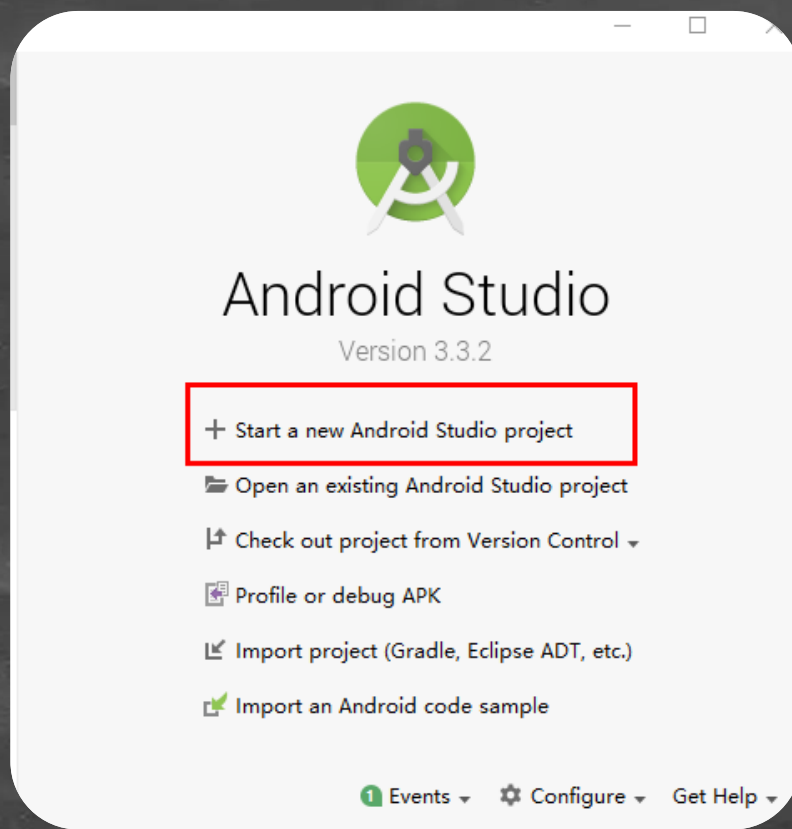
检查SDK, 打开SDK Manager: Configure-->SDK Manager, 确认选中21





4 第一个App

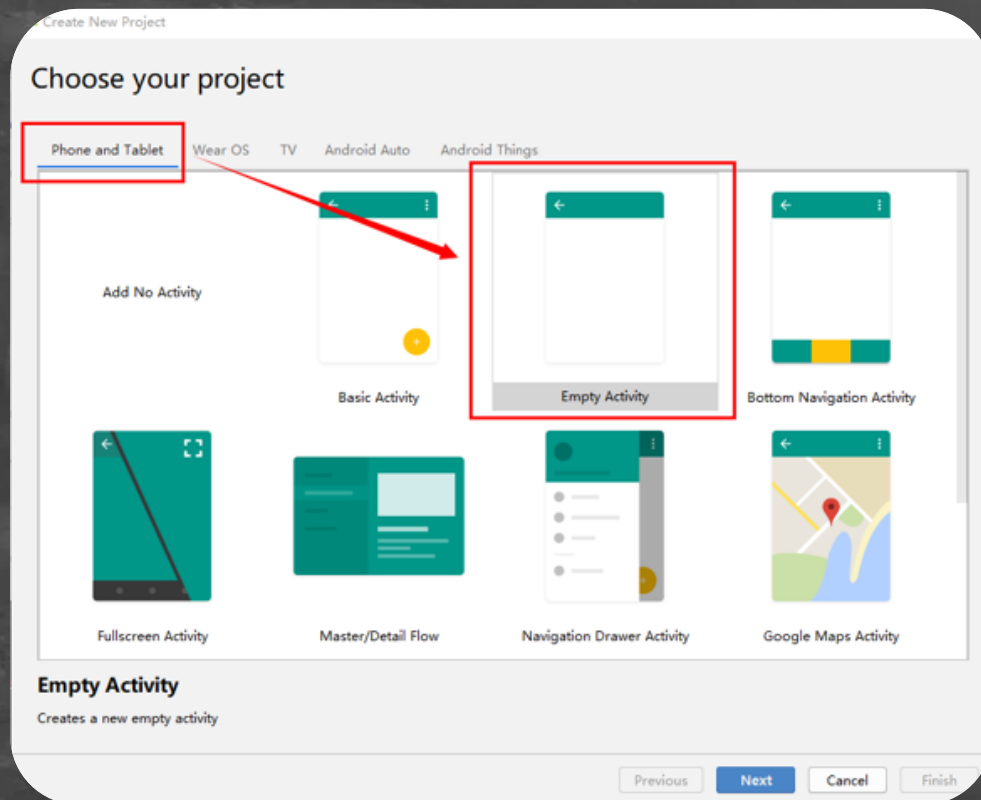
▶ 2. 新建工程：欢迎页新建Android Studio工程





4 第一个App

- ▶ 2. 新建工程：选择Phone and Tablet的Empty Activity模板，点击Next





4 第一个App

2. 新建工程：使用AS向导建立Hello World工程

输入**包名**：学校.院系.名字.App名 bjfu.it.sun.helloworld
升级或重装App时，Android通过包名区别App
SDK选择API 21，App能运行在Android 5.0及更新的手机

Create New Project

Configure your project

Name
Hello World

Package name
bjfu.it.sun.helloworld

Save location
E:\Android\HelloWorld

Language
Java

Minimum API level
API 21: Android 5.0 (Lollipop)

Your app will run on approximately 85.0% of devices.

Help me choose

☐ This project will support instant apps

☐ Use AndroidX artifacts

Creates a new empty activity

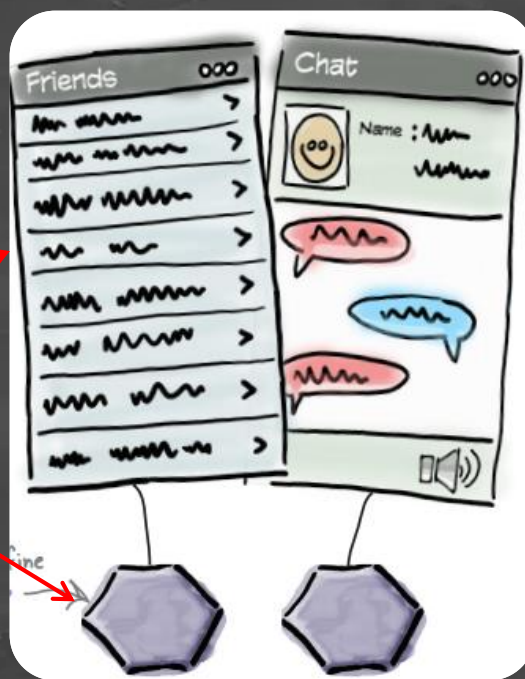
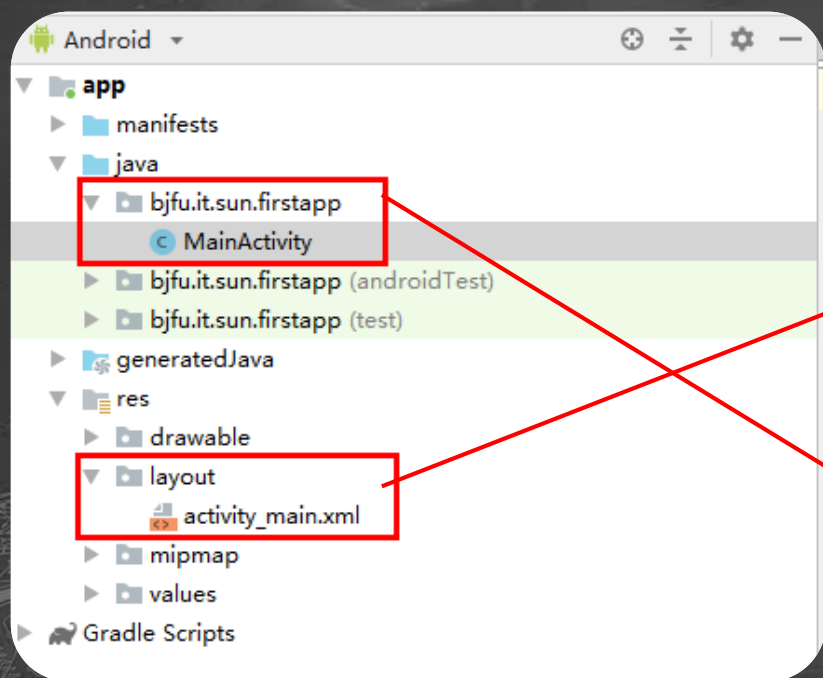
Previous Next Cancel Finish

路径无中文，机房存到D盘



4 第一个App

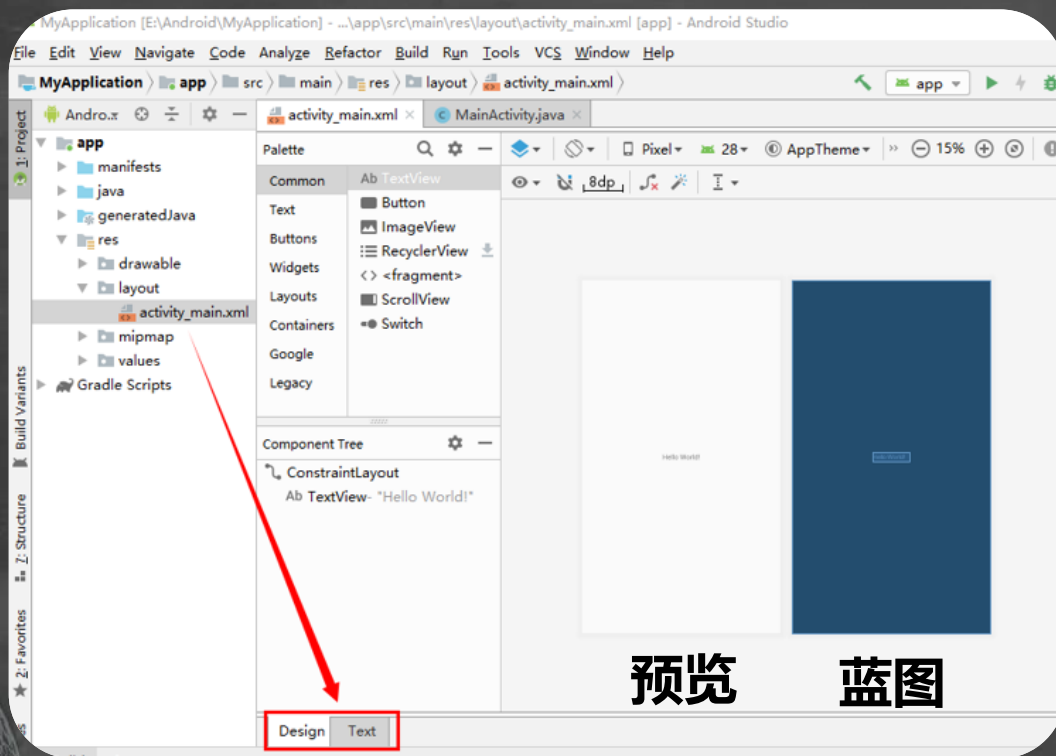
- ▶ 2. 新建工程：找到模板生成的Activity和layout





4 第一个App

▶ 3. 开发App: 打开layout xml文件



预览

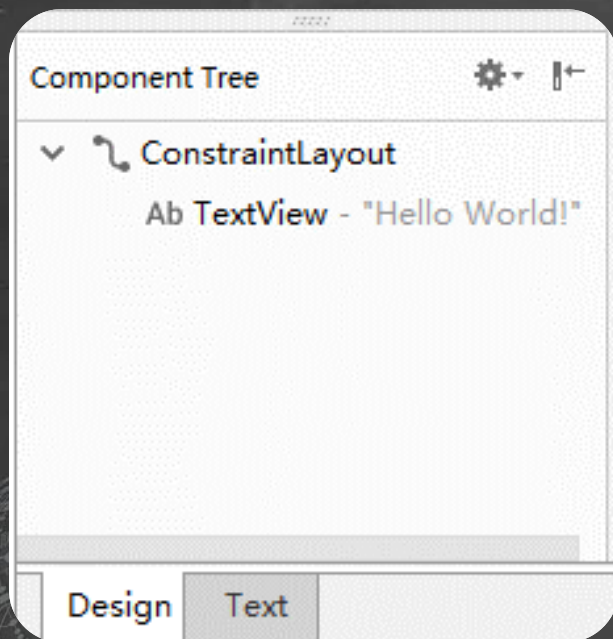
蓝图

在Design和Text标签页间切换



4 第一个App

- ▶ 3. 开发App: IDE自动生成的layout包含
约束布局ConstraintLayout和文本框TextView



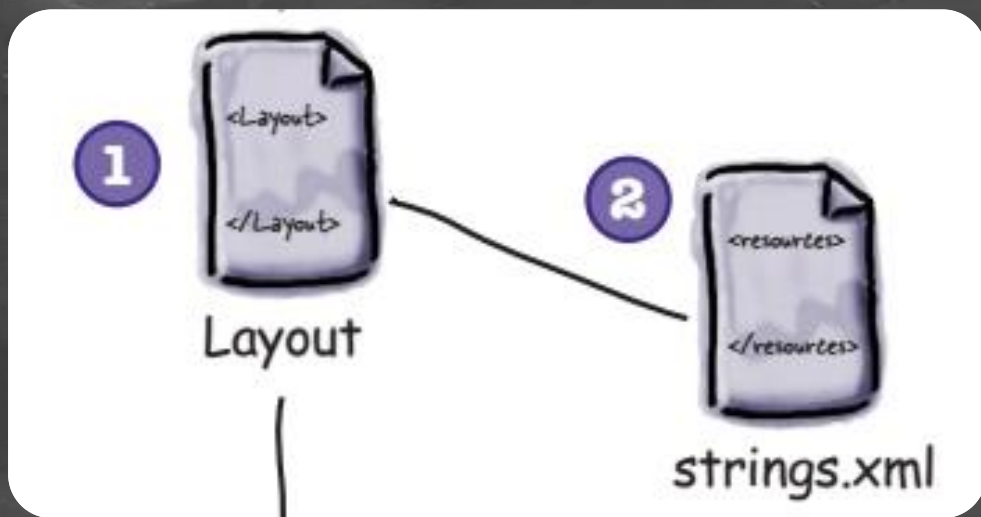
文本框嵌套在约束布局中



4 第一个App

▶ 3. 开发App：更改文本框内容

新建strings.xml中的字符资源



Strings.xml包含：

一个resources元素多个string元素，对应字符资源



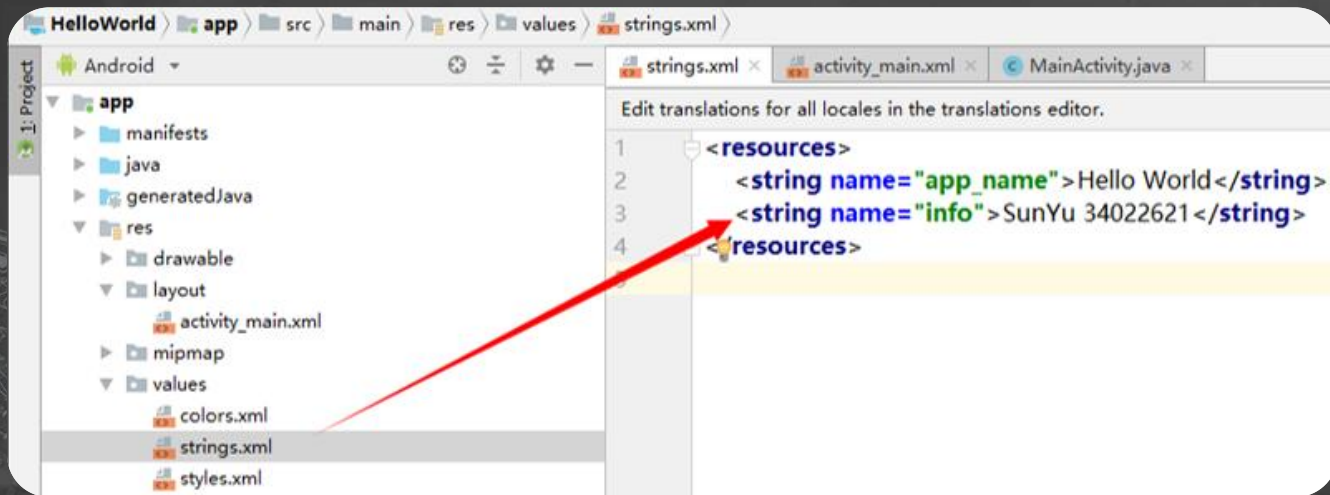
4 第一个App

▶ 3. 开发App: 更改文本框内容

新建strings.xml中的字符资源

在strings.xml中输入姓名学号

```
<string name="info">SunYu 34022621</string>
```

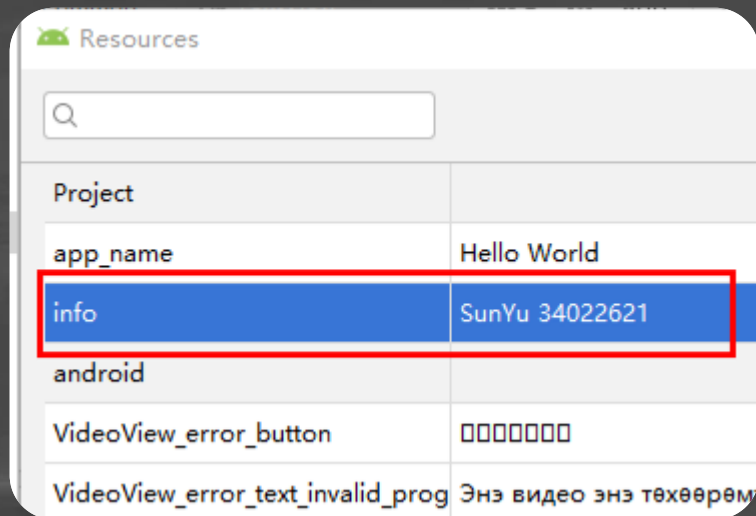
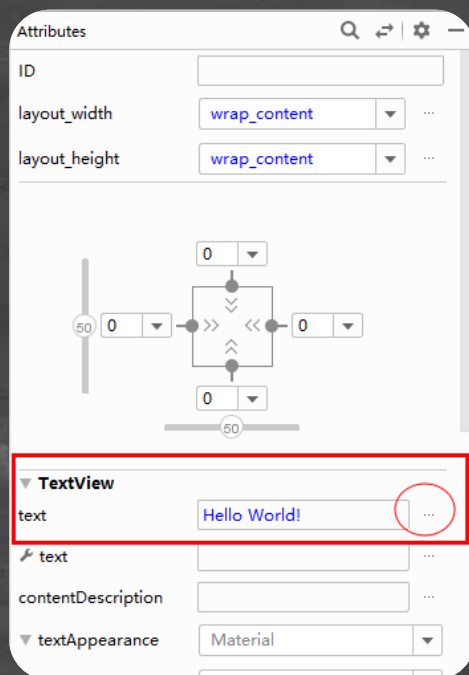
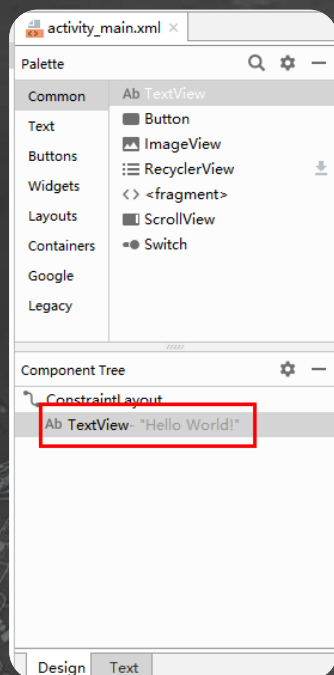




4 第一个App

▶ 3. 开发App：更改文本框内容

在Design界面选中文本框，在属性页修改text，选择字符资源





4 第一个App

▶ 3. 开发App: 更改文本框内容

本质为将`android:text`属性指向`@string/info`

The screenshot shows the Android Studio interface. On the left, the 'Attributes' panel shows the 'text' attribute of a 'TextView' widget set to '@string/info'. In the center, the 'activity_main.xml' file is open, showing the XML code for the layout. A red box highlights the `android:text="@string/info"` attribute in the `<TextView>` tag. On the right, the 'resources' section shows the definition of the string resource: `<string name="info">SunYu 34022621</string>`. A red arrow points from this resource definition to the attribute in the XML code.

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/info"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>
```

```
<resources>
    <string name="app_name">Hello World</string>
    <string name="info">SunYu 34022621</string>
</resources>
```

不推荐硬编码字符

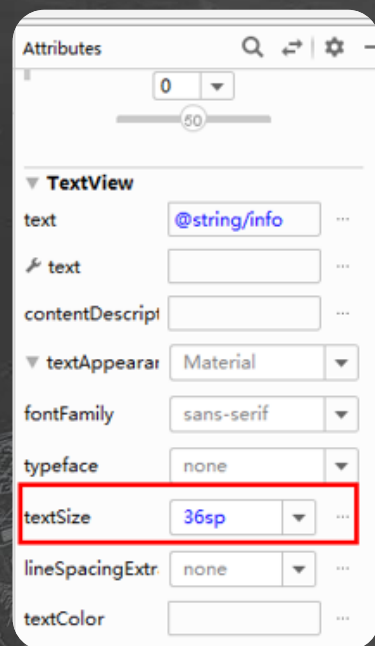
使用字符资源, 方便统一修改和国际化



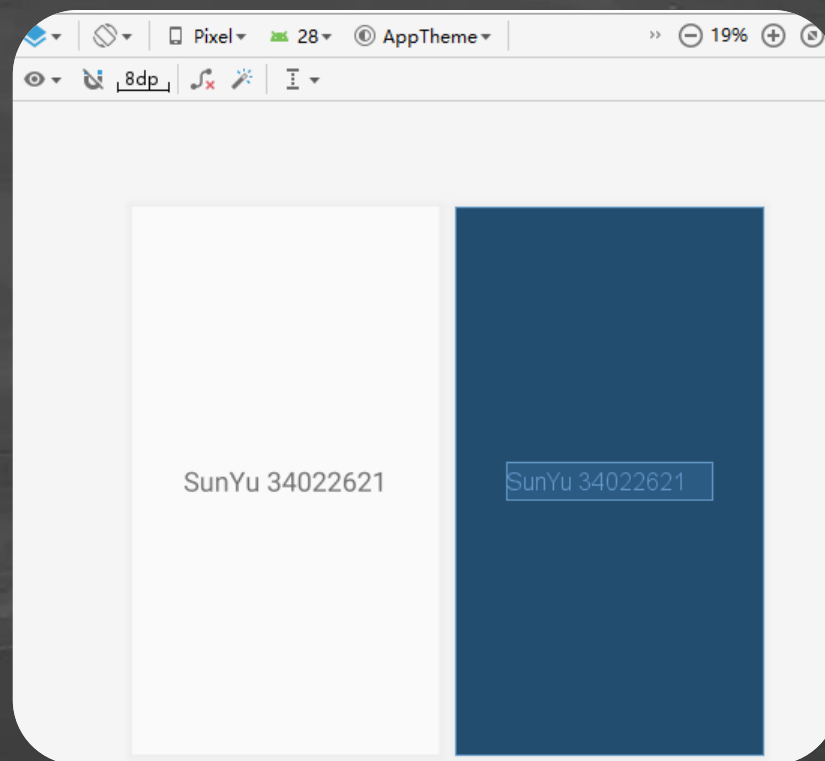
4 第一个App

▶ 3. 开发App：更改文本框内容

指增大字号：textSize属性修改为36sp，观察预览和蓝图



```
<TextView  
    android:layout_width="wrap_c  
    android:layout_height="wrap  
    android:text="@string/info"  
    android:textSize="36sp"  
    app:layout_constraintBottom_  
    app:layout_constraintLeft_toL  
    app:layout_constraintRight_to  
    app:layout_constraintTop_toT
```



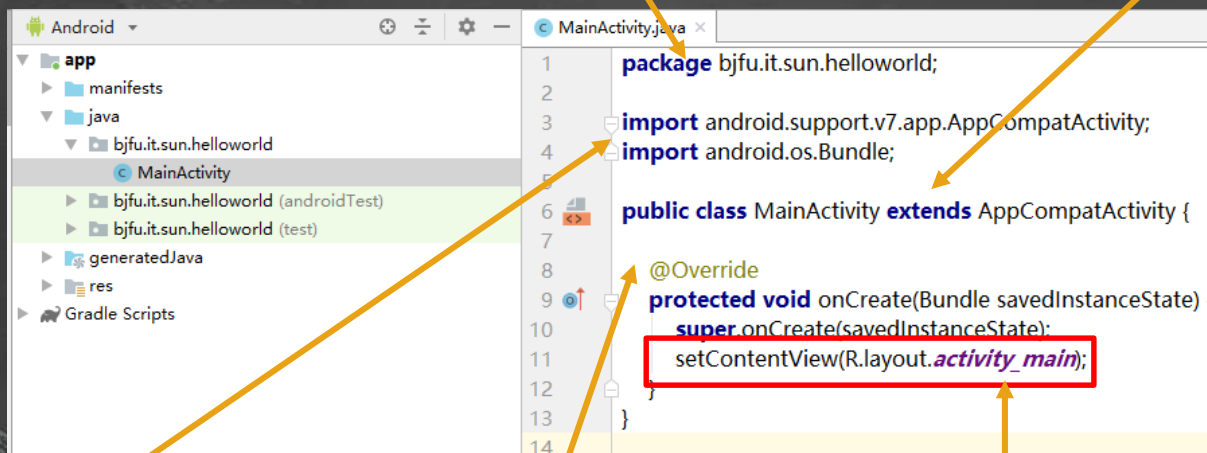


4 第一个App

3. 开发App: 认识Activity

包名区别各人的同名App

继承自框架层Activity类



导入框架层
Android Java API

覆盖override
父类onCreate方法

指定布局
(res内的xml资源文件编译为R.java)

