

Lập trình trên thiết bị di động

GIỚI THIỆU LẬP TRÌNH VỚI ANDROID STUDIO

GV: Nguyễn Huy Cường

Email: nh.cuong@hutech.edu.vn

Nội dung

1. Giới Thiệu Về Android

- Tổng quan về android
- Kiến trúc
- Môi trường phát triển

2. Sử dụng Android Studio

- Tạo project trên Android Studio
- Cấu trúc project
- Biên dịch và gỡ lỗi ứng dụng

GIỚI THIỆU VỀ ANDROID

Tổng quan về Android

❖ Hệ điều hành Android

- Là hệ điều hành nguồn mở dựa trên **Linux** được phát triển bởi Tổng công ty Android. **Google** mua lại (2005) và ra mắt vào 2007.
- Phát triển cho các thiết bị có màn hình cảm ứng chủ yếu dành cho điện thoại, máy tính bảng...
 - ❑ Ngôn ngữ Java (2011: Kotlin)
 - ❑ Sử dụng Bộ phát triển phần mềm Android (SDK).



Tổng quan về Android

❖ Lịch sử phát triển của Android

- Phiên bản hiện tại:

Android 14 (Android 14 Developer Preview)

Android 13 August 15, 2022

- Các phiên bản trước đó: 2008-2021

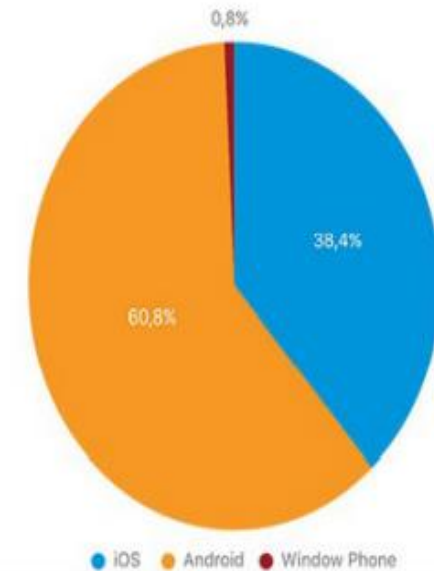


Tổng quan về Android

❖ Thị phần các hệ điều hành trên smartphone



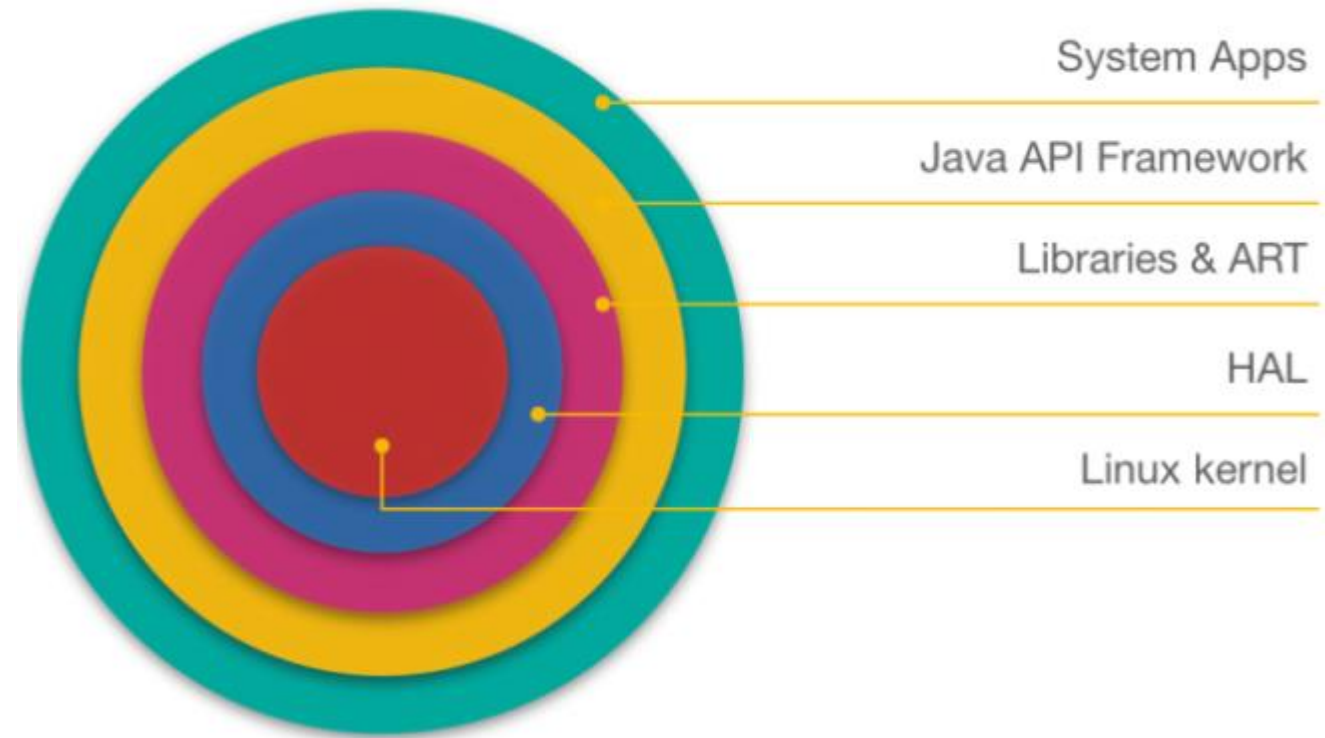
Thị phần theo hệ điều hành Smartphone
tháng 6/2020



Tổng quan về Android

❖ Android platform

- Là môi trường phần mềm được xây dựng cho thiết bị di động Android bao gồm:
 - Hệ điều hành dựa trên nhân Linux
 - Giao diện người dùng
 - Ứng dụng người dùng cuối
 - Thư viện code
 - Application framework

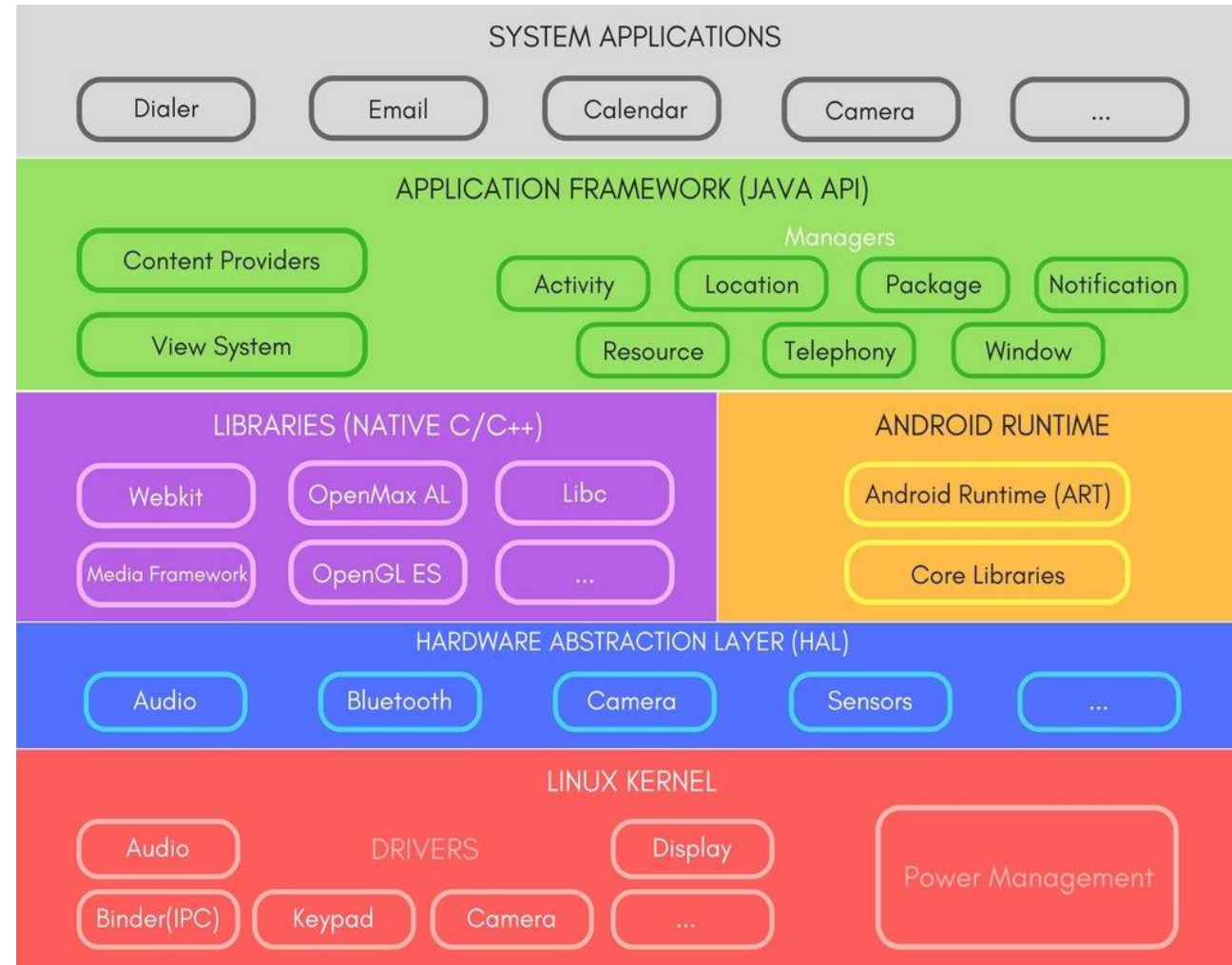


Tổng quan về Android

❖ Kiến trúc Android

- Chia thành 6 thành phần:

1. Nhân Linux
2. Lớp trừu tượng phần cứng
3. Runtime
4. Thư viện c/c++
5. Java API Framework
6. Ứng dụng



Kiến trúc Android

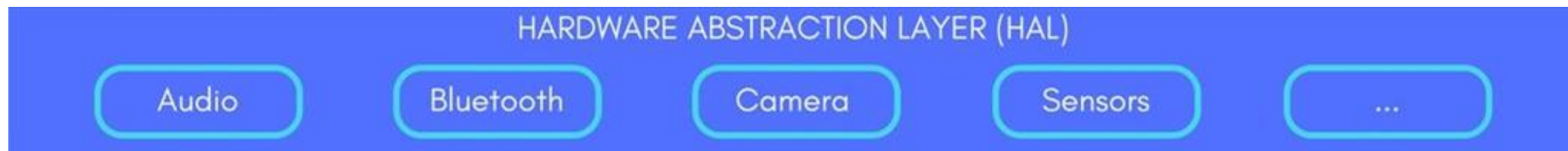
1- Nhân Linux

- Nhân Linux là tầng trừu tượng giữa phần cứng và software. Nó chịu trách nhiệm quản lý bộ nhớ, quản lý nguồn, quản lý thiết bị, truy cập tài nguyên... giúp giảm bớt khó khăn khi giao tiếp với phần cứng ngoại vi.
- Có thể thay thế kernel trên điện thoại bằng quyền truy cập root của thiết bị



2- Lớp trừu tượng phần cứng

- Lớp trừu tượng phần cứng (Hardware Abstraction Layer – HAL) là tầng thứ hai tính từ dưới lên trong kiến trúc của hệ điều hành Android. Tầng này sử dụng khung API (Application Programming Interface) Java cấp cao hơn để cung cấp các giao diện tiêu chuẩn, thực hiện các khả năng của phần cứng thiết bị.
- HAL bao gồm nhiều mô-đun, mỗi mô-đun sẽ thực hiện một giao diện cho một loại thành phần phần cứng cụ thể, ví dụ như mô-đun camera hoặc mô-đun bluetooth.
- Khi API thực hiện việc truy cập vào phần cứng của thiết bị, hệ điều hành Android sẽ tải mô-đun cho thành phần của phần cứng đó.



3- Android Runtime

- ART là engine cung cấp sức mạnh kết nối đến thư viện và lớp ứng dụng của hệ điều hành
- ART là một phần quan trọng của Android chứa các thành phần như: các thư viện Core và máy ảo Dalvik (Dalvik virtual machine).
 - ❑ Máy ảo Dalvik: biên dịch sang mã máy, khả năng chạy nhiều thread (chạy song song nhiều luồng)...
 - ❑ Java Core: Tập hợp các thư viện

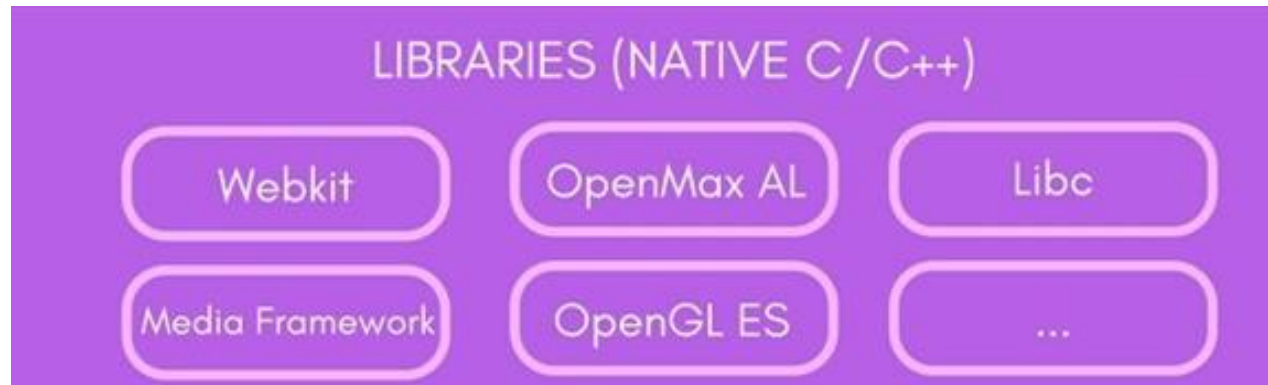


4- Native C/C++ Libraries

- Những thành phần và dịch vụ hệ thống cốt lõi như ART và HAL được xây dựng từ mã gốc, được viết bằng ngôn ngữ C và C++. Android cung cấp các khung API Java để hiển thị chức năng của một số thư viện C/C++ gốc (Native C/C++ Libraries) cho các ứng dụng.

Ví dụ: Chúng ta có thể thêm thao tác hỗ trợ vẽ và đồ họa 2D và 3D cho ứng dụng bằng cách truy cập OpenGL ES (Open Graphics Library for Embedded System) thông qua API Java OpenGL.

- Nếu bạn đang phát triển một ứng dụng yêu cầu mã C/C++, có thể sử dụng Android NDK (Native Development Kit) để truy cập trực tiếp vào một số thư viện C/C++ gốc từ mã gốc của mình.



5- Java API Framework



- Java API Framework cung cấp nhiều dịch vụ cấp cao hơn dưới dạng các **lớp Java** dành cho các ứng dụng.
- Các dịch vụ chính của Java API Framework bao gồm:
 - ❑ Trình quản lý hoạt động (Activity Manager): Kiểm soát tất cả các khía cạnh trong vòng đời của ứng dụng và activity stack.
 - ❑ Nhà cung cấp nội dung (Content Providers): Cho phép ứng dụng xuất bản và chia sẻ dữ liệu với các ứng dụng khác.
 - ❑ Trình quản lý tài nguyên (Resource Manager): Cung cấp quyền truy cập vào các tài nguyên không code như chuỗi, cài đặt màu, bố cục giao diện người dùng.
 - ❑ Trình quản lý thông báo (Notifications Manager): Cho phép ứng dụng hiển thị những cảnh báo và thông báo đến người dùng.
 - ❑ Hệ thống xem (View System): Tập hợp nhiều dạng khung hình có thể mở rộng, sử dụng để tạo giao diện cho người dùng ứng dụng.

6- Tầng ứng dụng

- Lớp trên cùng của kiến trúc android. Các ứng dụng nguyên thủy và ứng dụng của bên thứ ba như danh bạ, email, nhạc, thư viện, đồng hồ, trò chơi, v...v... bất cứ thứ gì ta sẽ xây dựng thì chúng sẽ chỉ được cài đặt tại lớp này
- Lớp ứng dụng chạy run-time trong Android bằng cách sử dụng các lớp (classes) và các dịch vụ (services) được tạo sẵn từ khung ứng dụng (application framework)



Môi trường phát triển

Top Android Development IDE



Android Studio



Visual Studio



Eclipse



IntelliJ IDEA



Cordova



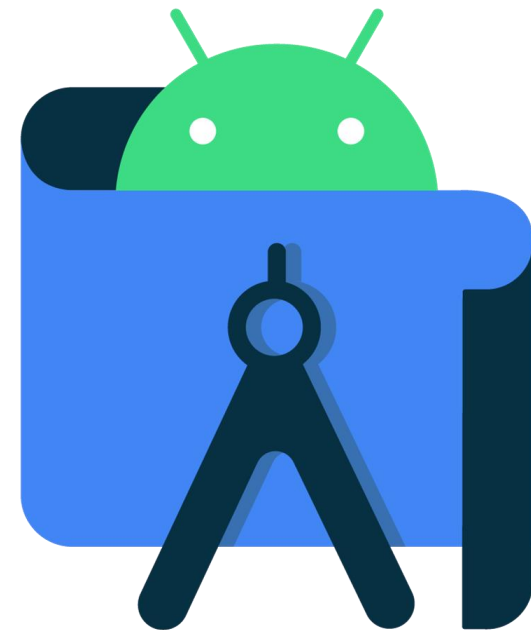
NetBeans



AIDE

Môi trường phát triển – Android Studio

- Android Studio là IDE (môi trường phát triển tích hợp - Integrated Development Environment).
 - ❑ Miễn phí
 - ❑ Phiên bản: Windows, Mac OS X và Linux.
 - ❑ Tích hợp Android SDK hỗ trợ cho việc lập trình trên nền tảng Android như trình biên dịch, các thư viện, máy ảo, tài liệu, source code mẫu



Môi trường phát triển Android Studio

- Yêu cầu hệ thống
 - ❑ 8 GB RAM or more.
 - ❑ 8 GB of available disk space minimum (IDE + Android SDK + Android Emulator)
 - ❑ 1280 x 800 minimum screen resolution.
- Tải và cài đặt Android Studio
<https://developer.android.com/studio/install.html>
- Phiên bản
 - Giraffe** (8-2023)
 - Flamigo** (4-2023)
 - Electric Eel (1-2023), Dolphin (9-2022)...



May 2023: Flamingo | 2022.2.1



Jan 2023: Electric Eel | 2022.1.1

Môi trường phát triển

Cài đặt Android Studio

Thực hiện theo hướng dẫn của bài 01 – lab 01



May 2023: Flamingo | 2022.2.1

SỬ DỤNG ANDROID STUDIO

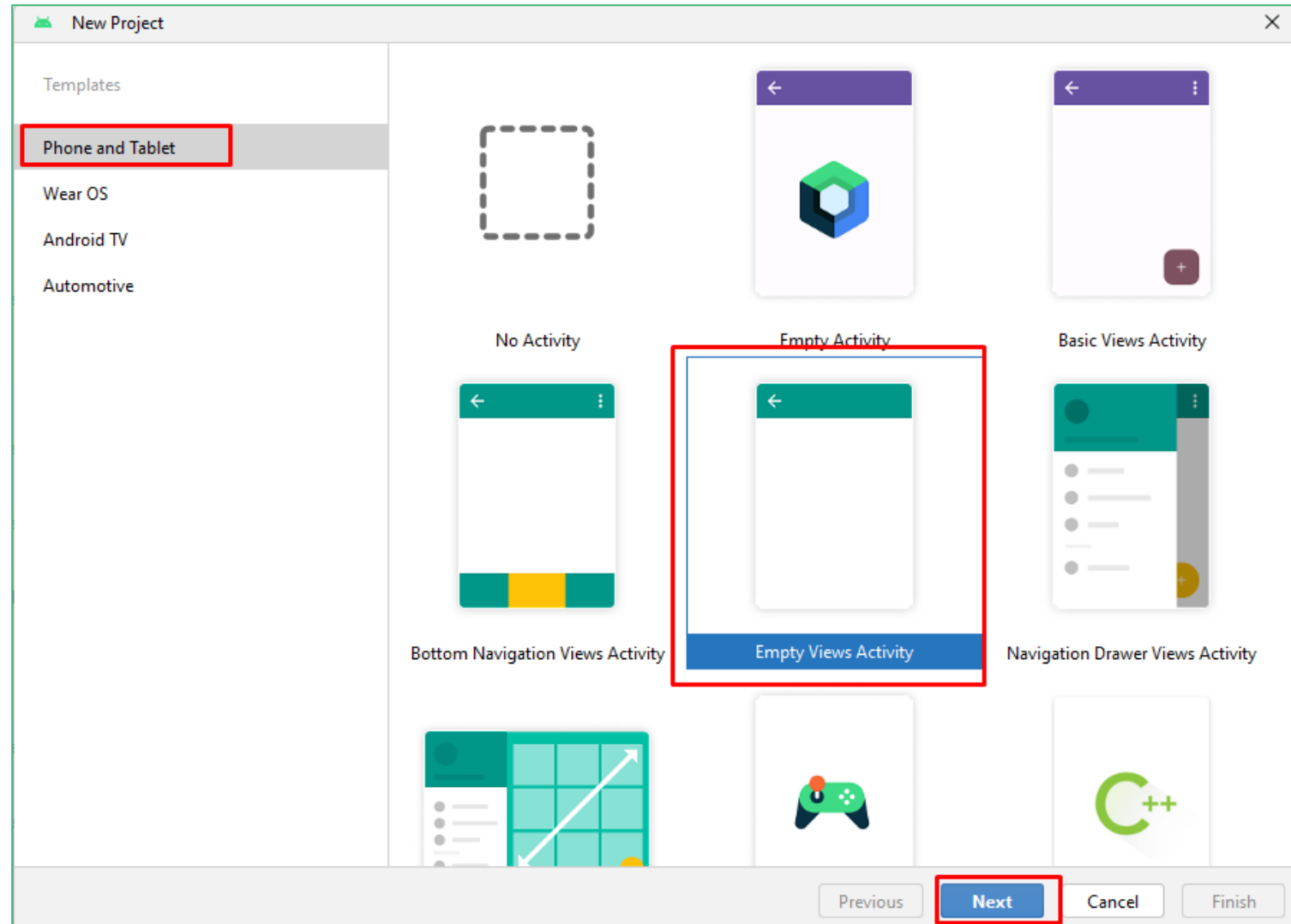
Tạo project trên Android studio

B1. Tạo project “Empty Activity”

- Mở android studio
- Chọn File>New>**New Project**
- Chọn **Phones and Tablet**
- Chọn **Empty Views Activity**

ban đầu của ứng dụng
(ở các phiên bản cũ: Empty Activity)

Next.



Tạo project trên Android studio

B2. Cấu hình cho Project

- Nhập Application Name

Package name

Save location

Language (**Java**/ Kotlin)

min SDK: ≥ 8.0

Finish



New Project

Empty Views Activity

Creates a new empty activity

Name: DemoLab01

Package name: com.example.demolab01

Save location: D:\MonHoc\LapTrinhAndroid\Lab\DemoLab01

Language: Java

Minimum SDK: API 26: Android 8.0 (Oreo)

i Your app will run on approximately 92.4% of devices.
[Help me choose](#)

Previous Next Cancel **Finish**

Cấu trúc project trên Android Studio

❑ **Gradle Scripts:** build các cấu hình cho

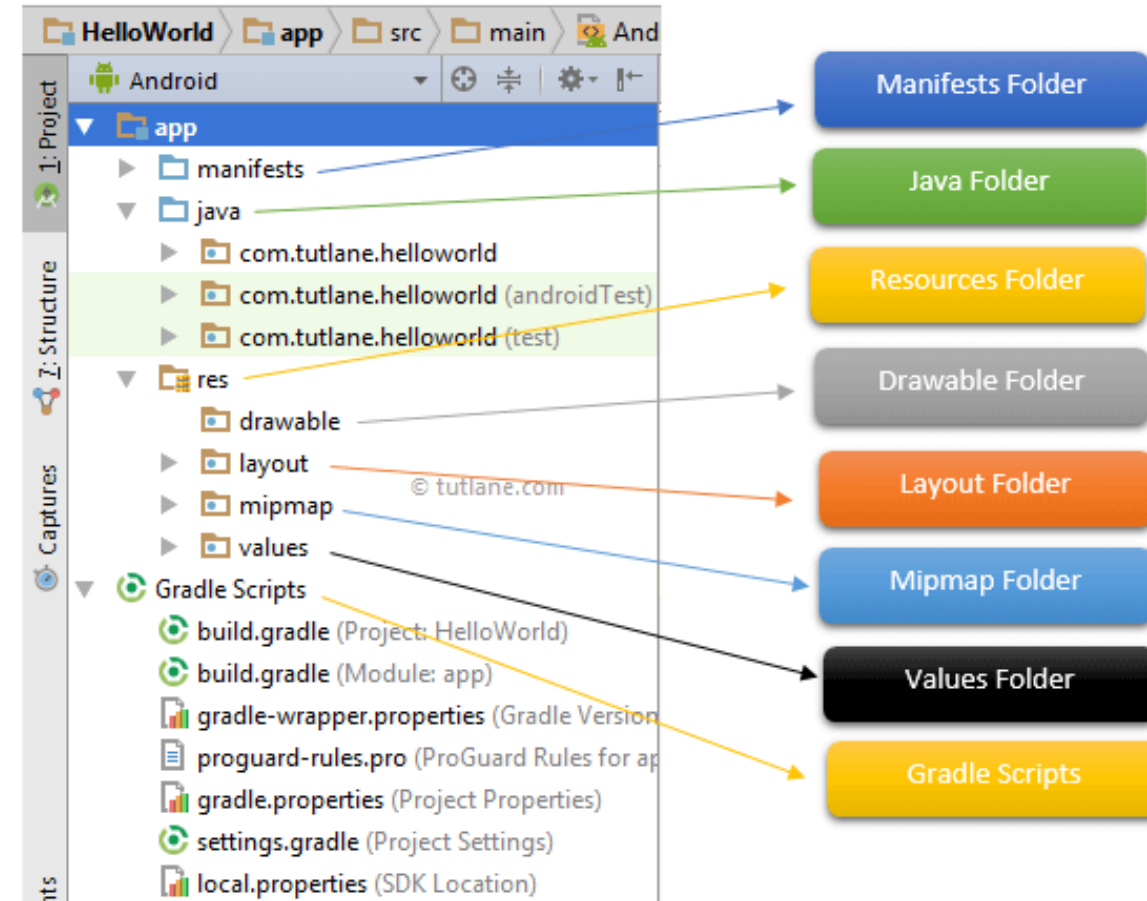
- Tất cả các module build.gradle (Project)
- Riêng cho một module build.gradle (Module)

❑ **app**

File Android Layout (Activity_main.xml)

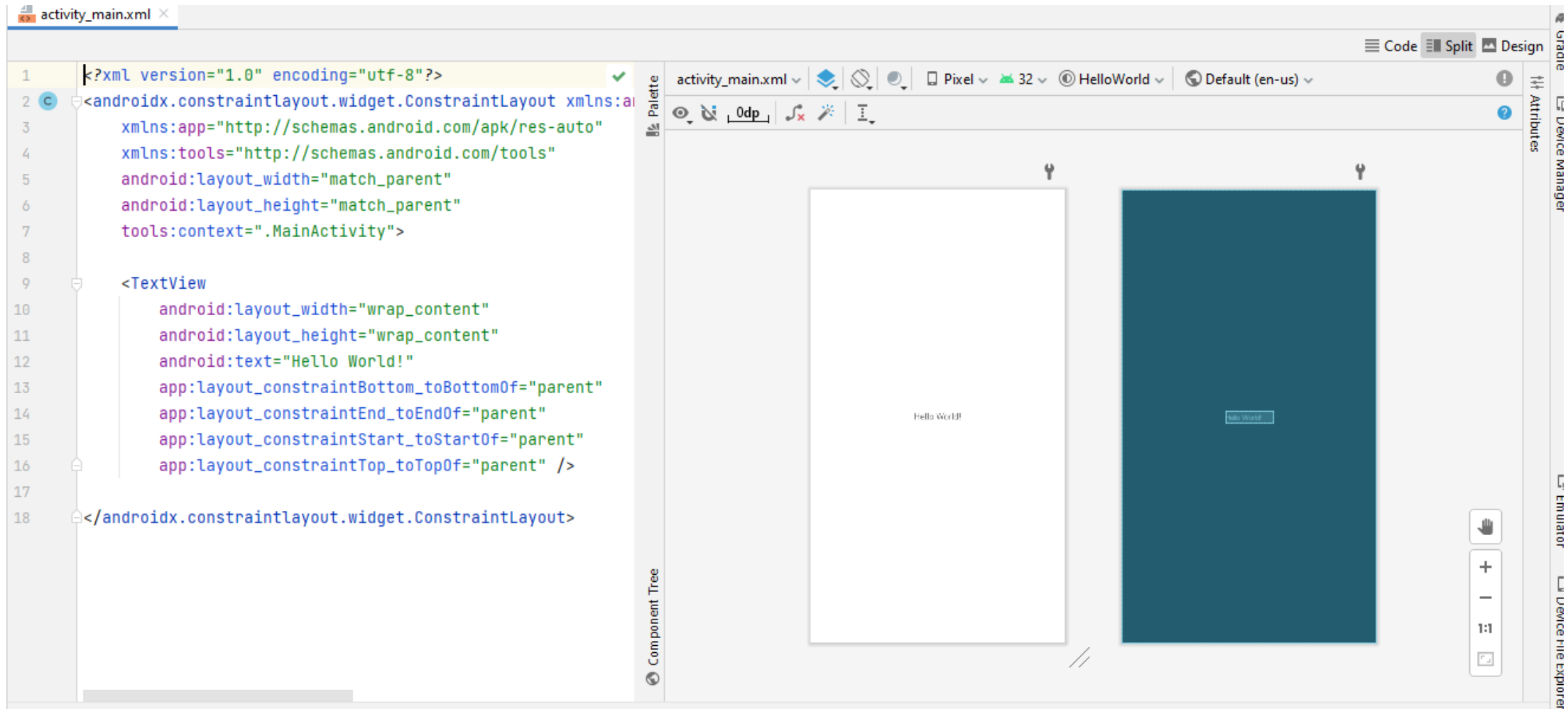
File Android Main Activity (MainActivity.java)

...



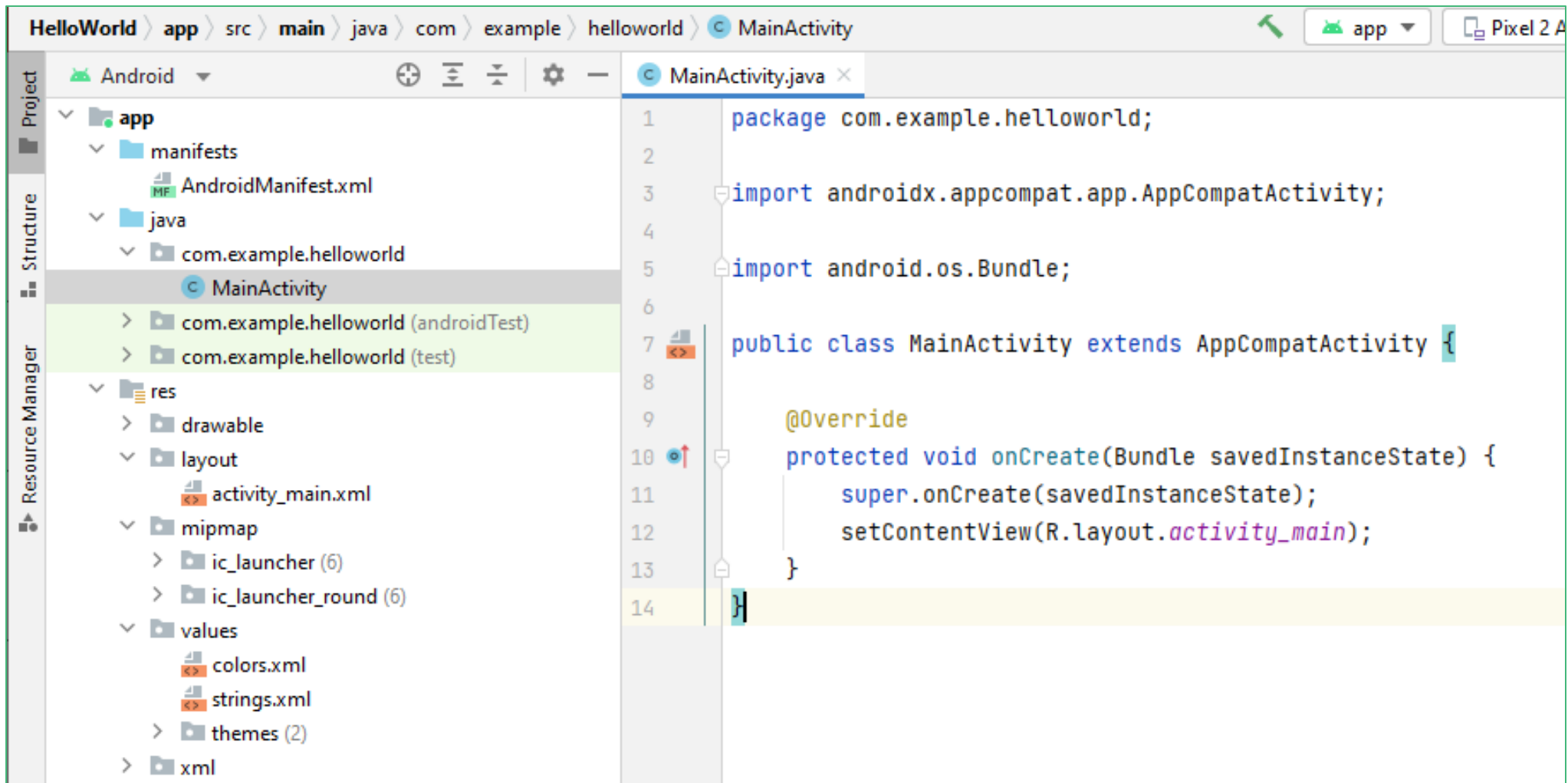
Giao diện người dùng

- File giao diện người dùng `res/layout/main_activity`



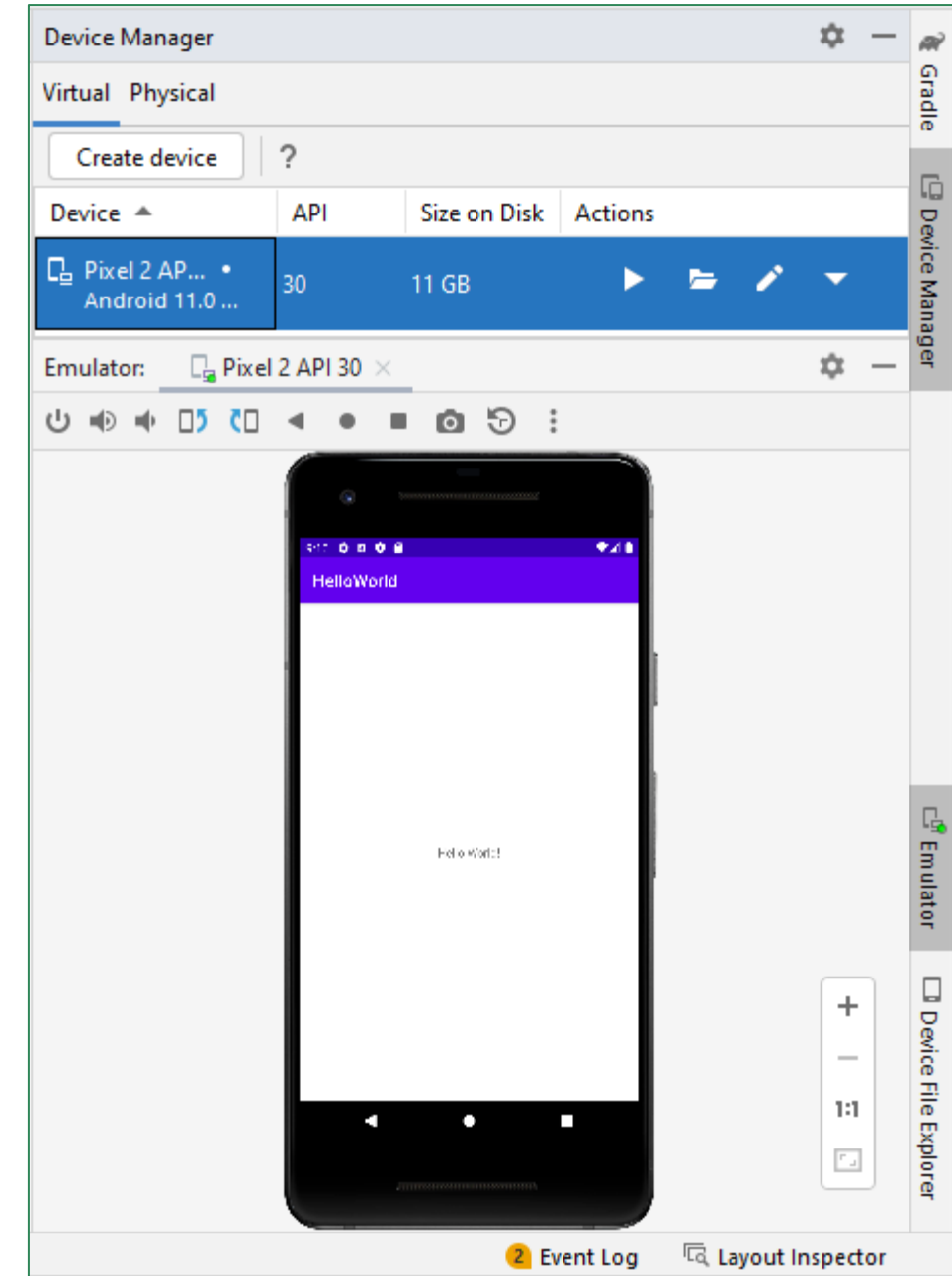
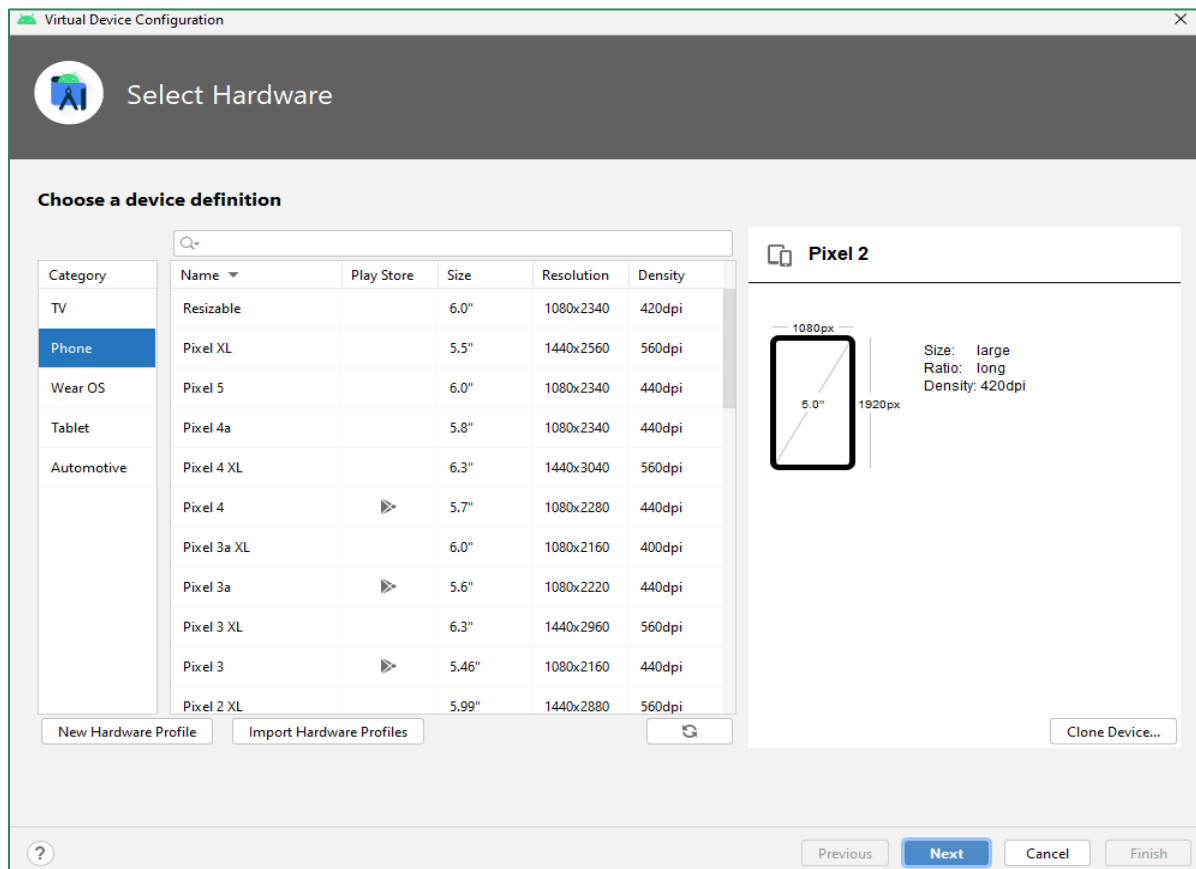
Mã nguồn java

- Mã nguồn java (java/packageName/MainActivity.java)



Chạy ứng dụng trên Android Emulator

- Mở Android Virtual Device (AVD)
 - Tạo Emulator tương ứng





❑ Tạo ứng dụng “Demo” đầu tiên

❑ Tìm hiểu về **resources**

1. Thay đổi kích thước text Size “**50sp**”. Tìm hiểu về “**sp**” và “**dp**”.

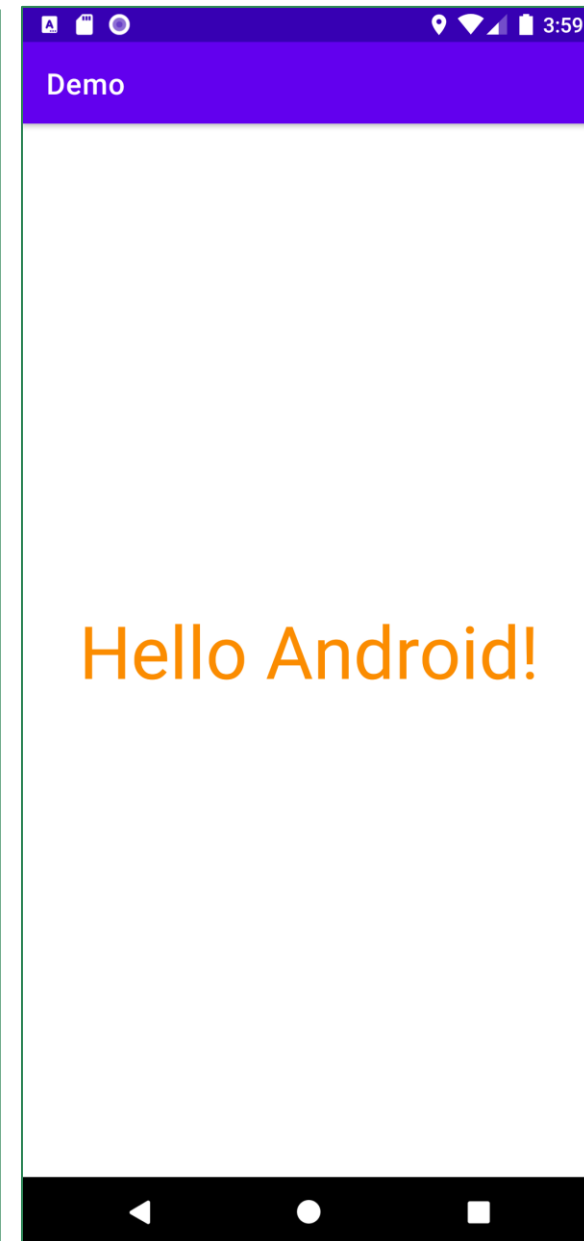
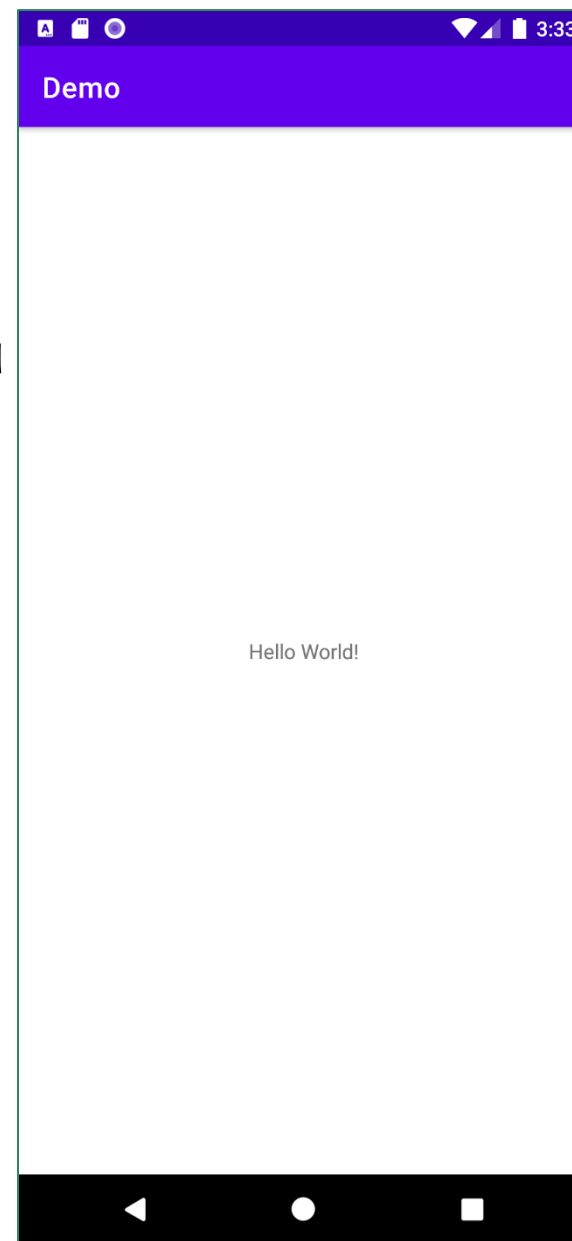
2. Thay đổi nội dung của **TextView** là

“Hello Android!” lấy từ *resources/strings.xml*

android:text = “@string/MSG_HELLO”

3. Đổi màu chữ (có mã màu #FB8C00) và được định nghĩa từ *resources/colors.xml*

android:textColor = “@color/COLOR_TEXT”





❑ Tìm hiểu một số thuộc tính view

4. Tìm hiểu về *layout_width*, *layout_height*

với các giá trị

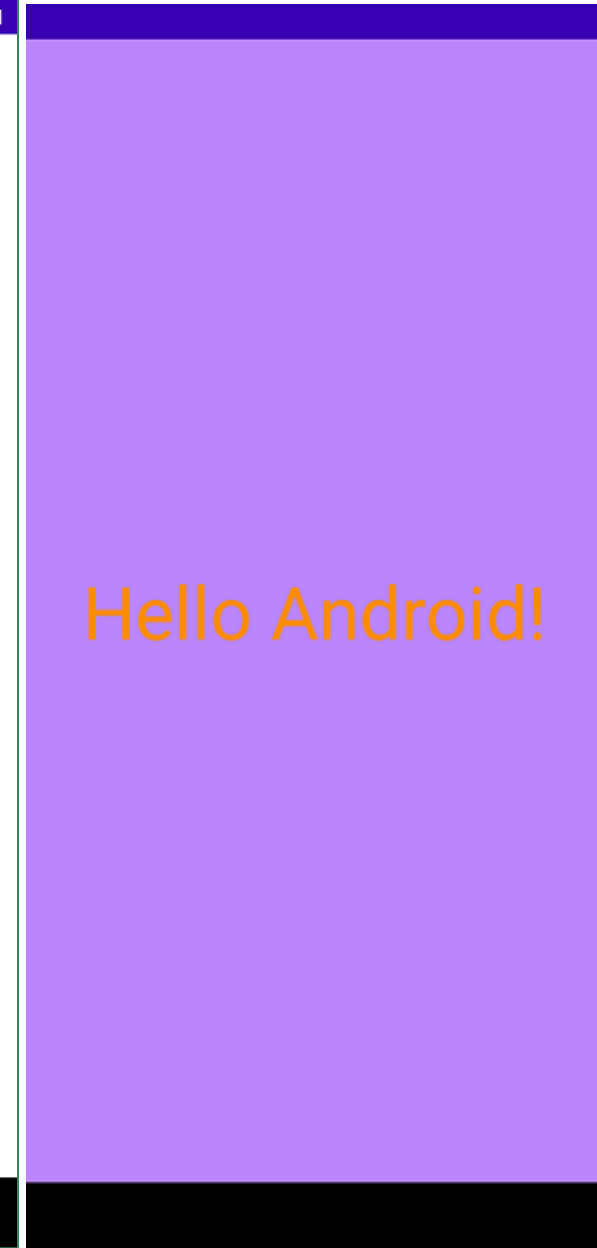
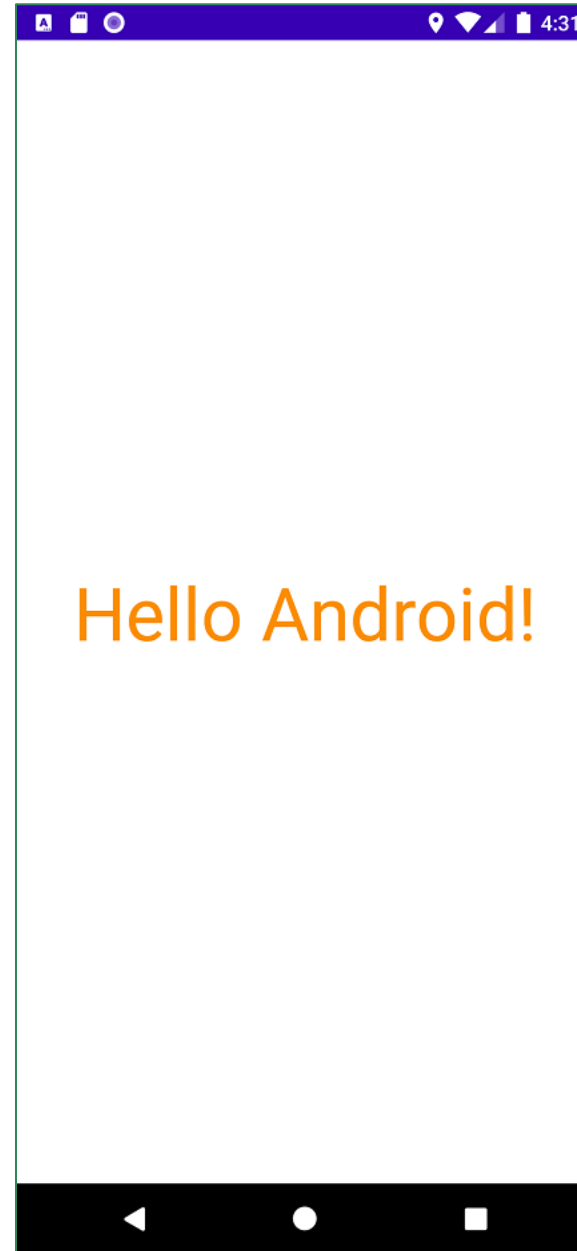
“wrap_content”

“match_parent”.

5. Các giá trị “padding” “margin”

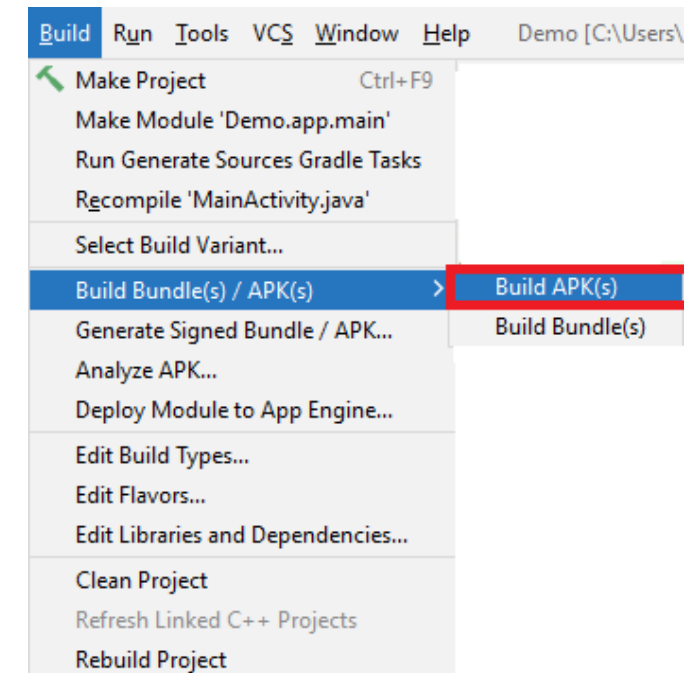
❑ Đổi màu nền của màn hình chính

`android:background="@color/purple_200"`



Biên dịch ứng dụng

- Trong tiến trình biên dịch, android project được biên dịch và đóng gói thành file có đuôi **.apk**
- Tập tin **.apk** chứa các thông tin cho chương trình android:
 - ☐ Dex: Các lớp Java được biên dịch sẽ được chạy trên thiết bị (tệp .DEX)
 - ☐ META-INF: Chứa tệp kê khai, chữ ký và danh sách các tài nguyên trong kho lưu trữ
 - ☐ ...
- Cách tạo **.apk**: .../**Build APK(s)**
Lấy file .apk: [ProjectName]/**build/apk/**
- Đưa ứng dụng lên **Google Play**



Tổng kết

- Hệ điều hành android là hệ điều hành gì?
- Ứng dụng android có thể tạo ra cho thiết bị nào ?
- Kiến trúc Android Android là một hệ điều hành mã nguồn mở dựa trên nền tảng Linux, gồm mấy tầng?
- Linux Kernel là gì ?
- Hardware Abstraction Layer – HAL là gì ?
- Android runtime là gì?
- Java API Framework là gì ?
- Android Development IDE bao gồm ?
- Thư mục Manifest chứa gì trong đó ?
- Gradle Scripts là gì?
- Các file *.xml được lưu ở đâu ?