

Install Android Studio

Apa itu android.

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler seperti ponsel pintar dan tablet. Salah satu cara untuk membuat aplikasi Android adalah dengan menggunakan Android Studio, sebuah lingkungan pengembangan terpadu (IDE) yang disediakan oleh Google. Untuk menggunakan Android Studio, kita perlu menginstal beberapa modul yang diperlukan, seperti SDK Tools, SDK Platform-Tools, dan SDK Build-Tools.

Pengenalan Android Studio

Android Studio adalah lingkungan pengembangan terpadu (IDE) untuk membuat aplikasi Android. Android Studio menyediakan berbagai fitur dan alat yang membantu pengembang membangun, menguji, dan mendistribusikan aplikasi Android dengan mudah dan efisien. Beberapa fitur utama Android Studio adalah:

- **Editor kode** yang mendukung bahasa pemrograman Java, Kotlin, dan C/C++.
- **Template proyek dan wizard** yang memudahkan pembuatan aplikasi baru atau impor proyek yang ada.
- **Gradle**, sistem otomatisasi bangunan yang mengintegrasikan dependensi, konfigurasi, dan proses kompilasi.
- **Emulator Android**, perangkat lunak yang meniru perangkat Android nyata di komputer pengembang.
- **Android Debug Bridge (ADB)**, antarmuka perintah baris yang memungkinkan pengembang berkomunikasi dengan perangkat atau emulator Android.
- **Android Profiler**, alat yang membantu pengembang menganalisis kinerja aplikasi mereka dalam hal CPU, memori, jaringan, dan baterai.
- **Firebase**, platform yang menyediakan layanan backend seperti analitik, autentikasi, penyimpanan data, notifikasi push, dan lainnya.

Persyaratan Sistem

Untuk menginstal Android Studio, Anda perlu memenuhi beberapa persyaratan sistem yang diperlukan. Persyaratan sistem ini meliputi spesifikasi minimum dan rekomendasi untuk hardware, software, dan lingkungan pengembangan. Berikut ini adalah persyaratan sistem untuk menginstall android studio.

Windows

| Requirement | Minimum | Recommended |
|-------------------|---|--------------------------------------|
| OS | 64-bit Microsoft Windows 8 | Latest 64-bit version of Windows |
| RAM | 8 GB RAM | 16 GB RAM or more |
| CPU | x86_64 CPU architecture; 2nd generation Intel Core or newer, or AMD CPU with support for a Windows Hypervisor Framework . | Latest Intel Core processor |
| Disk space | 8 GB (IDE and Android SDK and Emulator) | Solid state drive with 16 GB or more |
| Screen resolution | 1280 x 800 | 1920 x 1080 |

Mac

| Requirement | Minimum | Recommended |
|-------------------|--|--------------------------------------|
| OS | MacOS 10.14 (Mojave) | Latest version of MacOS |
| RAM | 8 GB RAM | 16 GB RAM or more |
| CPU | Apple M1 chip, or 2nd generation Intel Core or newer with support for Hypervisor Framework . | Latest Apple Silicon chip |
| Disk space | 8 GB (IDE and Android SDK and Emulator) | Solid state drive with 16 GB or more |
| Screen resolution | 1280 x 800 | 1920 x 1080 |

Mengunduh dan Menginstal Android Studio

Untuk memulai pengembangan aplikasi Android, Anda perlu mengunduh dan menginstal Android Studio, lingkungan pengembangan terpadu (IDE) resmi dari Google. Android Studio menyediakan alat-alat yang diperlukan untuk membuat aplikasi Android berkualitas tinggi, seperti editor kode, emulator, debugger, dan template proyek. Berikut adalah langkah-langkah untuk mengunduh dan menginstal Android Studio pada komputer Anda:

1. Kunjungi situs web resmi Android Studio di <https://developer.android.com/studio>
2. Pilih sistem operasi Anda (Windows, Mac, atau Linux) dan klik tombol Download Android Studio.
3. Setelah file unduhan selesai, jalankan file tersebut dan ikuti petunjuk instalasi di layar.
4. Setelah instalasi selesai, jalankan Android Studio dan ikuti petunjuk konfigurasi awal untuk menyiapkan SDK Android dan komponen lainnya yang diperlukan.
5. Selamat, Anda telah berhasil mengunduh dan menginstal Android Studio!

Install SDK

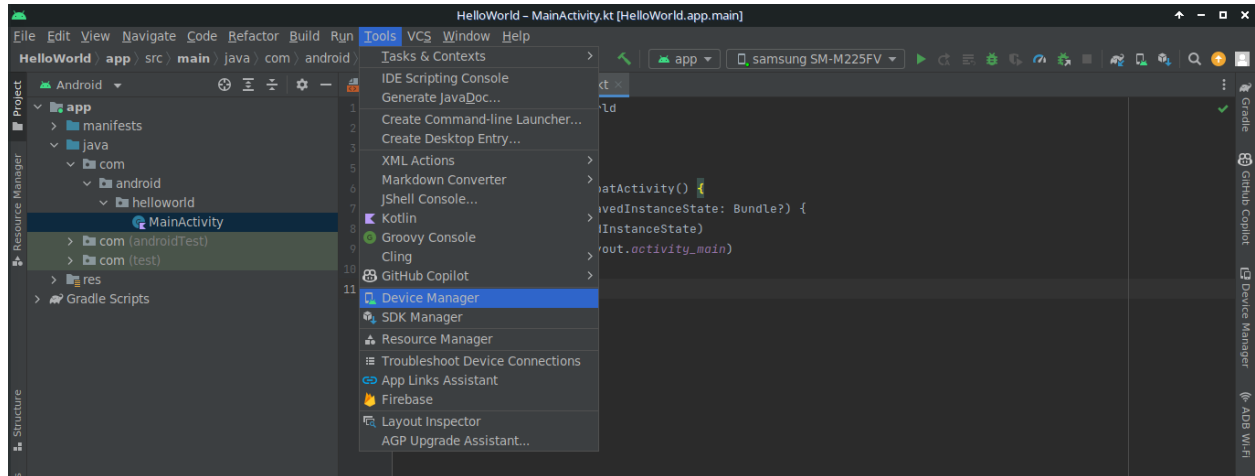
SDK adalah kumpulan perangkat lunak yang digunakan untuk mengembangkan aplikasi android. SDK berisi berbagai komponen, seperti library, emulator, debugger, dan dokumentasi. Untuk install SDK, ikuti langkah-langkah berikut:

1. Buka android studio dan pilih menu Tools > SDK Manager.
2. Pada jendela SDK Manager, pilih tab SDK Platforms dan centang platform android yang ingin diinstall. Misalnya, android 11 (R).
3. Pilih tab SDK Tools dan centang tool yang dibutuhkan, seperti Android SDK Build-Tools, Android Emulator, Android SDK Platform-Tools, dan lain-lain.
4. Klik OK dan tunggu proses download dan install selesai.
5. Setelah selesai, klik Finish dan restart android studio jika diminta.

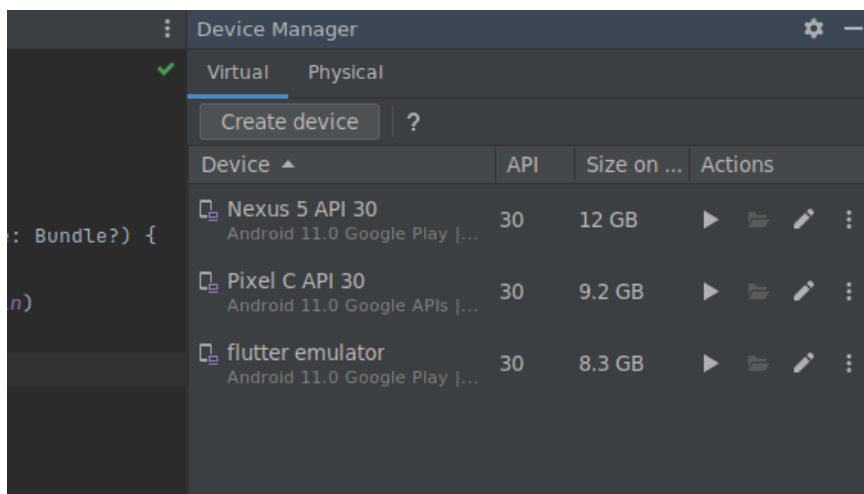
Mengatur Android Virtual Device (Emulator)

Salah satu langkah penting dalam memulai pengembangan aplikasi Android adalah mengatur Android Virtual Device (AVD). AVD adalah emulator yang memungkinkan Anda menjalankan dan menguji aplikasi Android di komputer Anda tanpa perlu menggunakan perangkat fisik. Untuk mengatur AVD, Anda perlu menginstal Android Studio, yang merupakan lingkungan pengembangan terpadu (IDE) untuk Android. Berikut adalah poin-poin yang harus Anda lakukan untuk mengatur AVD:

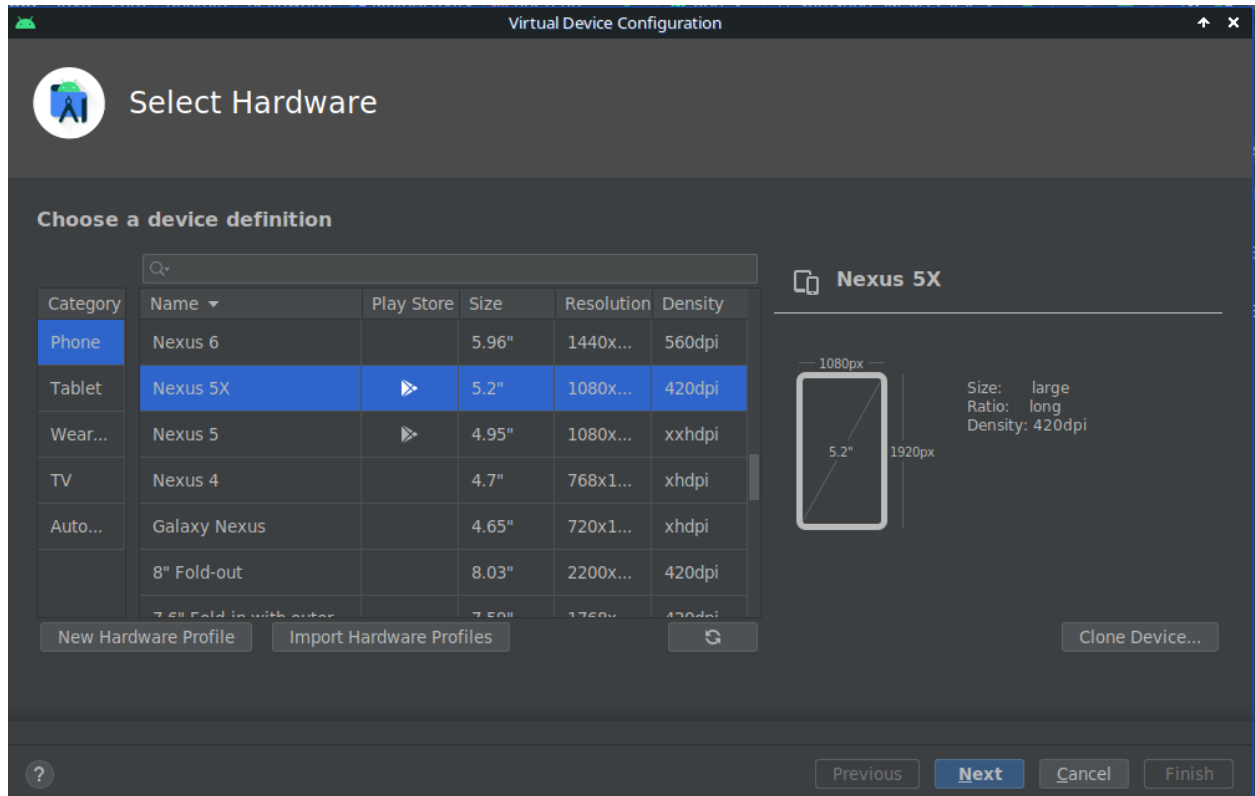
1. Buka Android Studio dan klik menu Tools > Device Manager.



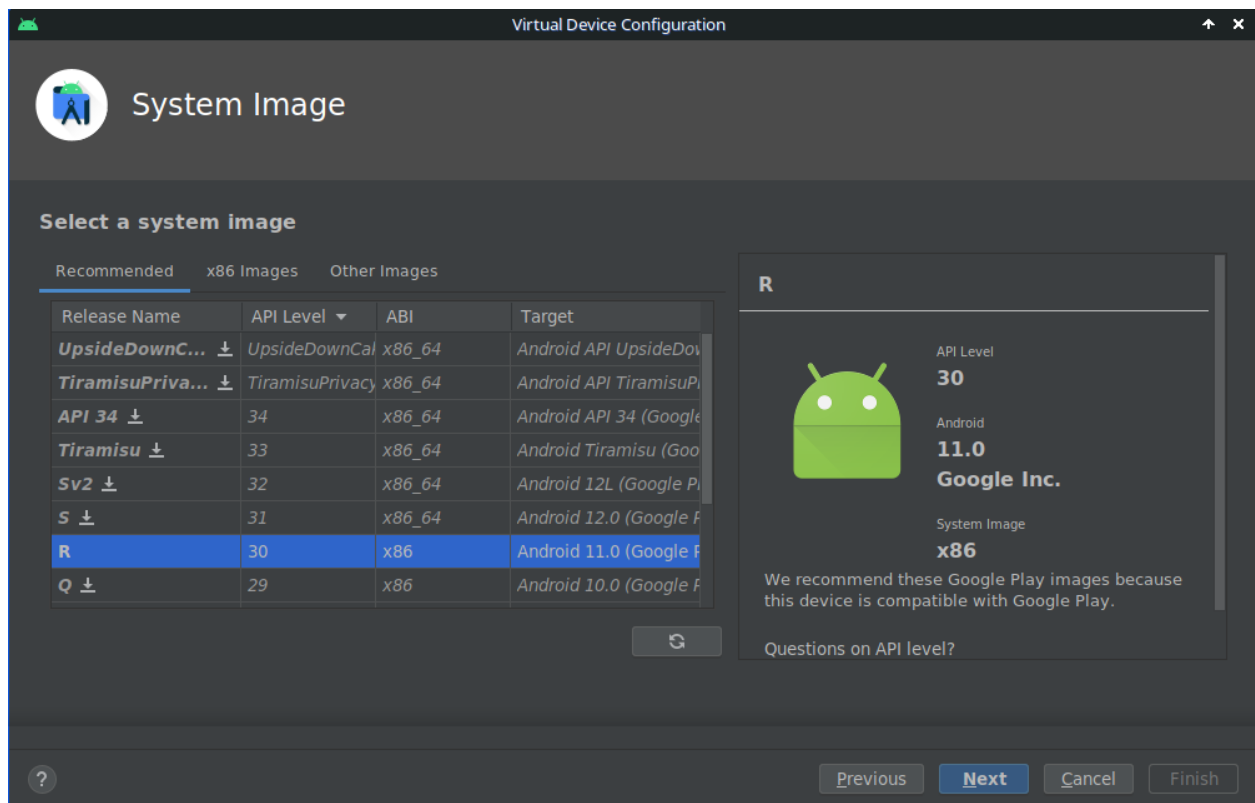
2. Jika sudah maka akan muncul tampilan sebagai berikut, kemudian pilih “Create device”



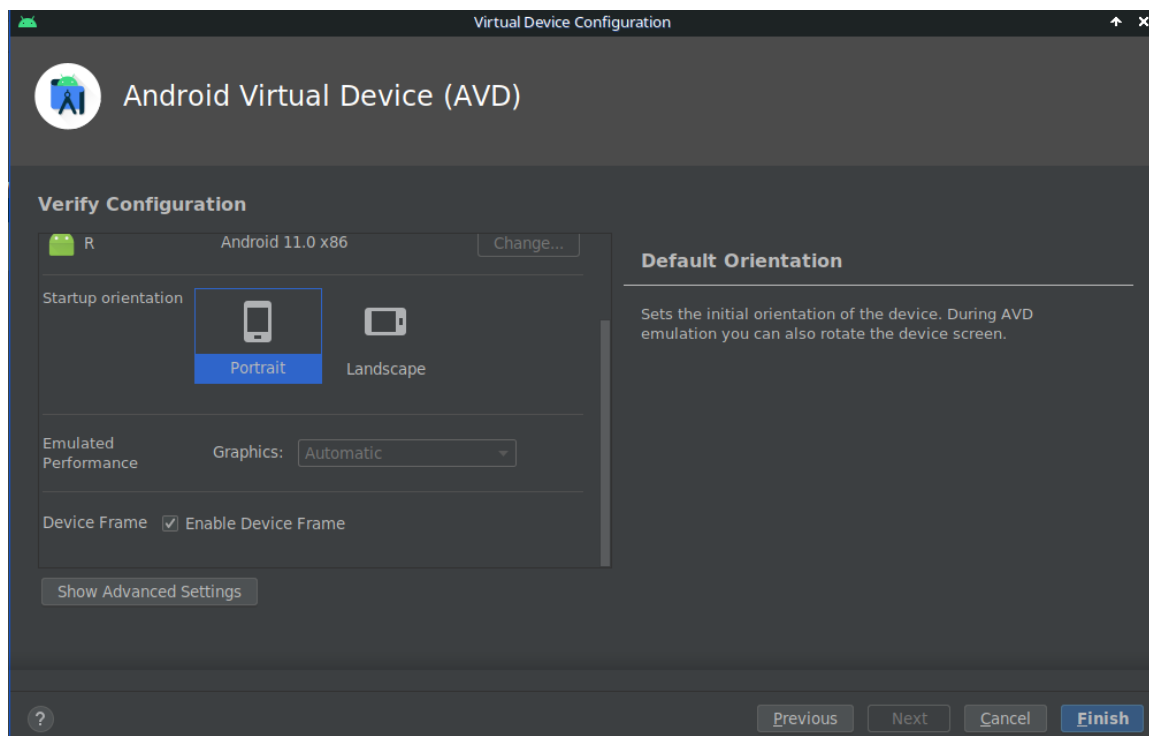
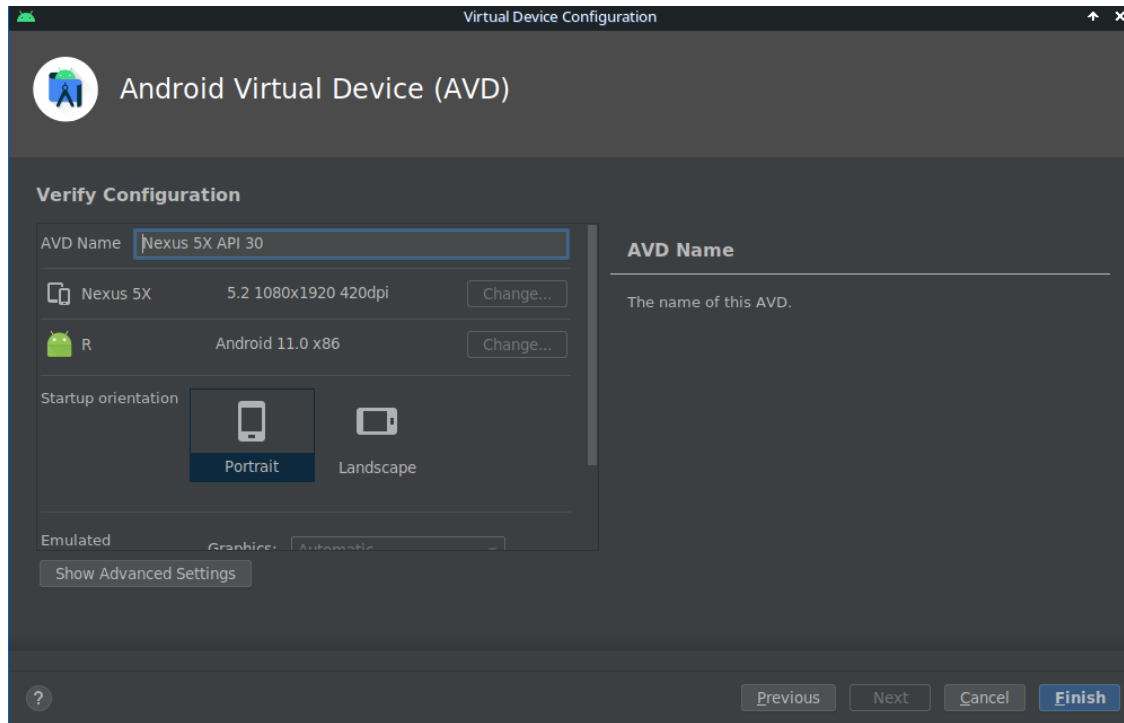
3. Pilih jenis perangkat yang ingin Anda emulasikan, misalnya Nexus 5X, dan klik Next.



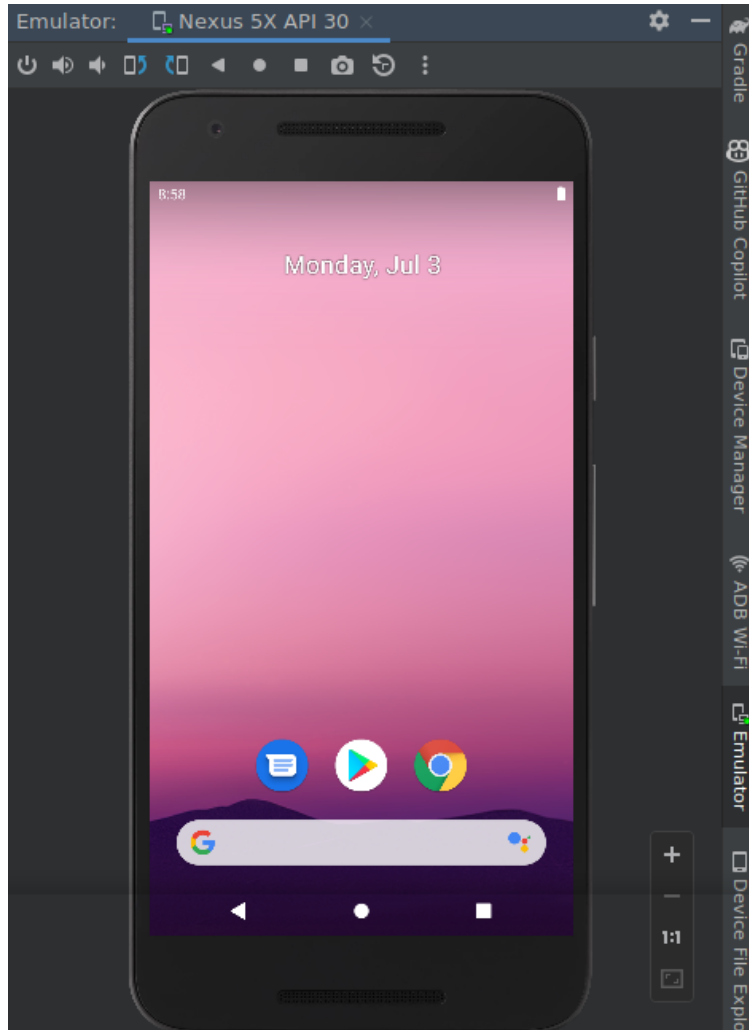
- Pilih versi sistem operasi Android yang ingin Anda gunakan, misalnya Android R, dan klik Next.



5. Beri nama AVD Anda dan sesuaikan opsi konfigurasi sesuai kebutuhan Anda, seperti orientasi layar, ukuran RAM, kamera, dll. Kemudian klik Finish.



6. AVD Anda sekarang siap digunakan. Anda dapat menjalankannya dengan klik tombol Play di kolom Actions jendela Device Manager.



Alternatif AVD: menggunakan physical device

Physical device adalah perangkat nyata yang dapat dihubungkan ke komputer Anda melalui kabel USB atau nirkabel. Berikut adalah langkah-langkah untuk menggunakan physical device:

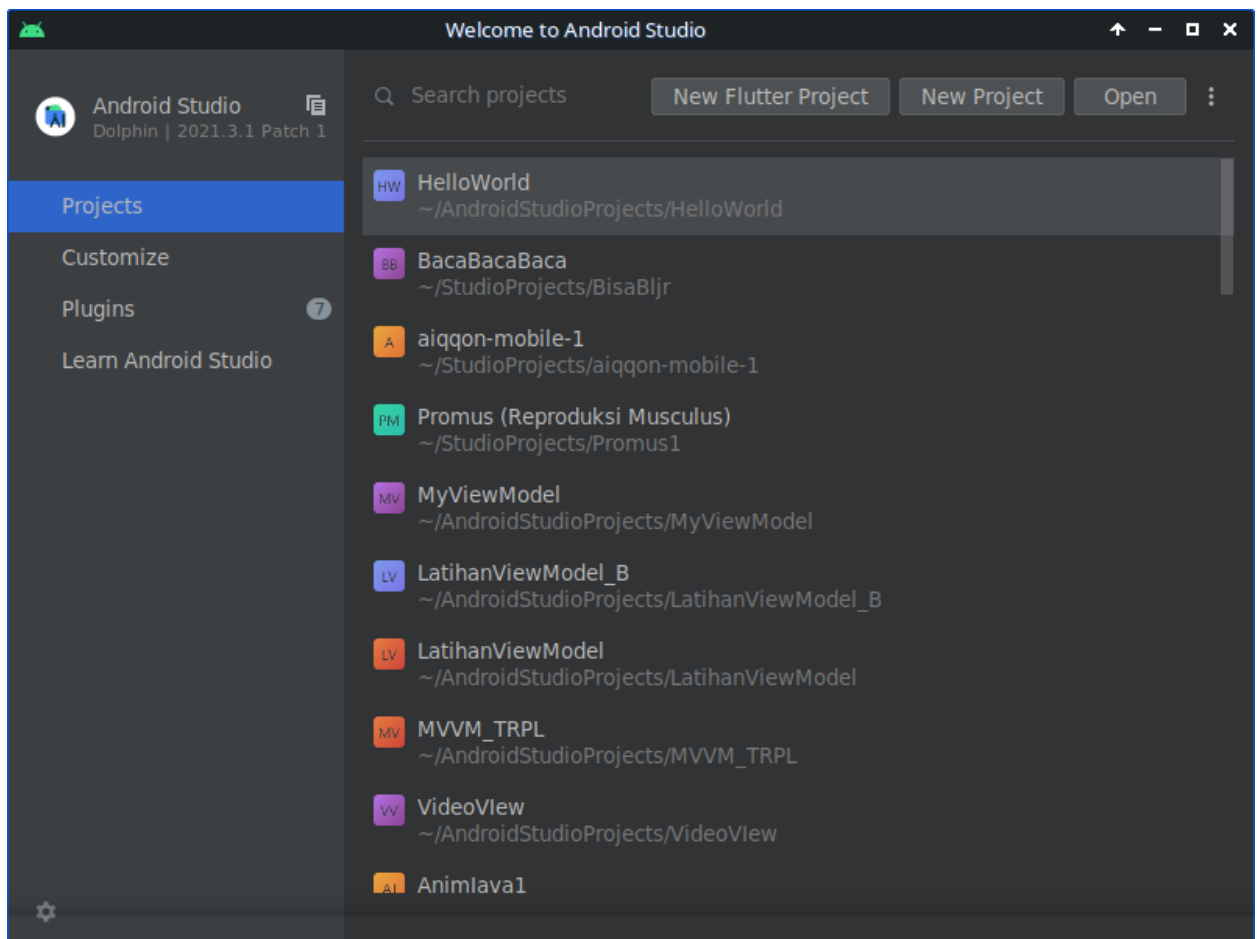
1. Pastikan perangkat Anda memiliki sistem operasi Android 4.1 (Jelly Bean) atau lebih tinggi.
2. Aktifkan mode pengembang dan USB debugging pada perangkat Anda. Caranya bervariasi tergantung pada model dan versi Android Anda, tetapi biasanya dapat ditemukan di menu Settings > About phone > Build number. Ketuk Build number tujuh kali untuk mengaktifkan mode pengembang, lalu kembali ke menu Settings > Developer options dan centang USB debugging.
3. Hubungkan perangkat Anda ke komputer Anda melalui kabel USB atau nirkabel. Jika menggunakan kabel USB, pastikan kabelnya berkualitas baik dan terhubung dengan kuat. Jika menggunakan nirkabel, pastikan perangkat Anda dan komputer Anda terhubung ke jaringan Wi-Fi yang sama.

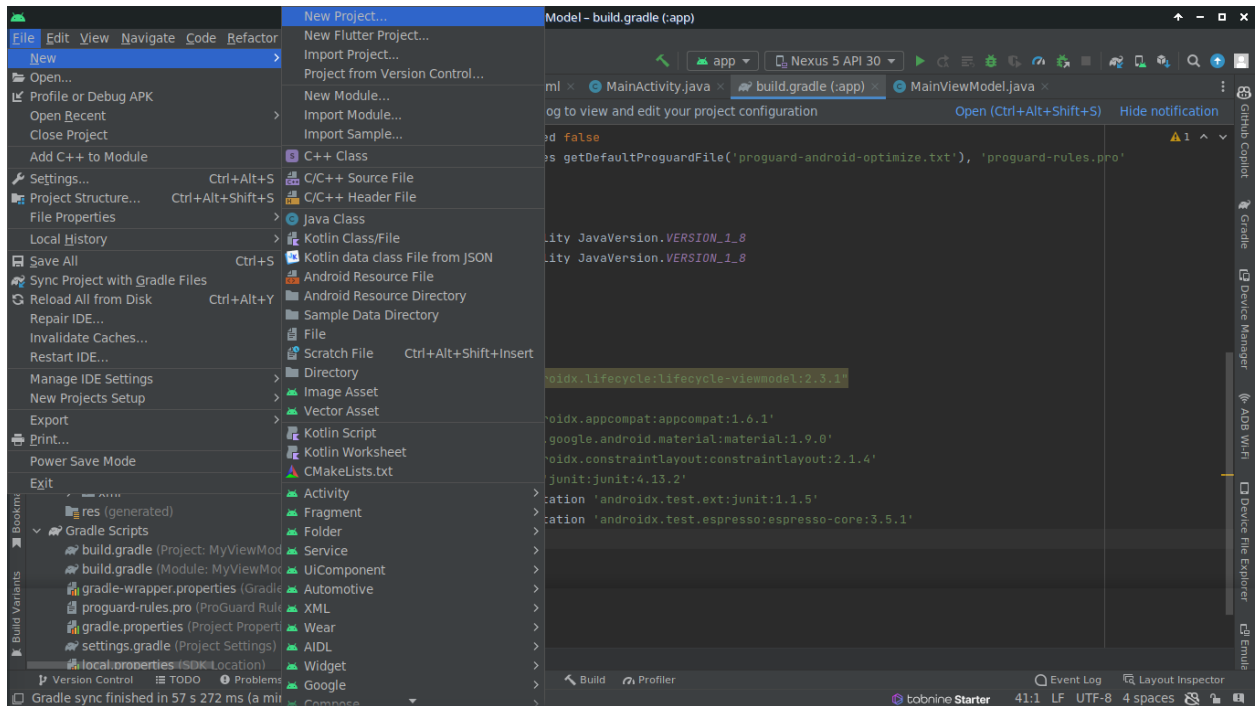
4. Buka android studio dan buat atau buka proyek yang ingin Anda jalankan pada perangkat Anda.
5. Pada toolbar android studio, pilih perangkat Anda dari daftar drop-down yang berisi avd dan perangkat yang tersedia. Jika perangkat Anda tidak muncul, coba cabut dan pasang kembali kabel USB atau matikan dan nyalakan kembali mode nirkabel.
6. Klik tombol Run (ikon segitiga hijau) untuk memulai instalasi dan menjalankan aplikasi Anda pada perangkat Anda.

Membuat Proyek Hello World!

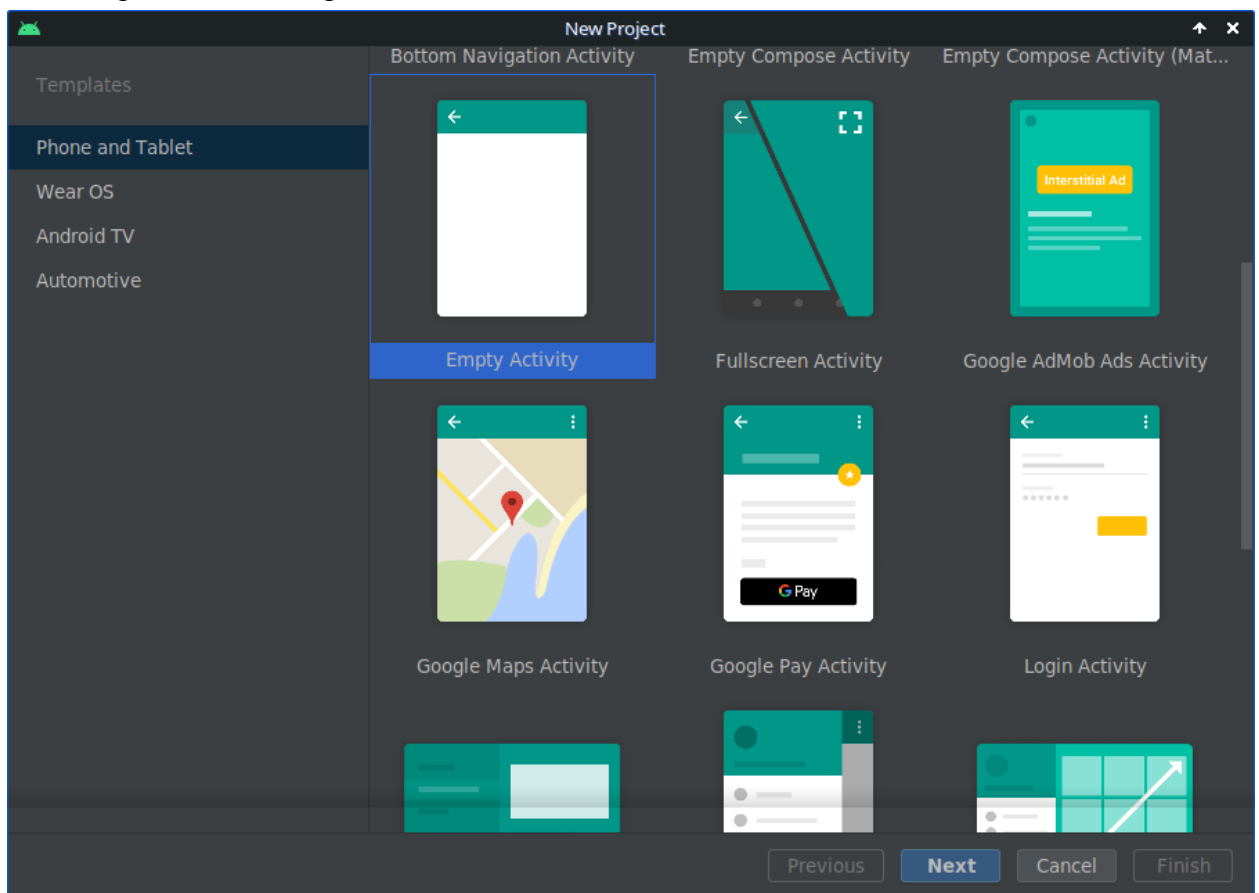
Setelah menginstal Android Studio, langkah selanjutnya adalah membuat proyek Hello World! Proyek ini akan menampilkan pesan "Hello World!" di layar perangkat atau emulator. Berikut adalah cara membuat proyek di Android Studio:

1. Pertama, ketika belum ada proyek yang dibuka klik **“New Project”** jika sudah membuka proyek pilih **“File>New>New Project”**



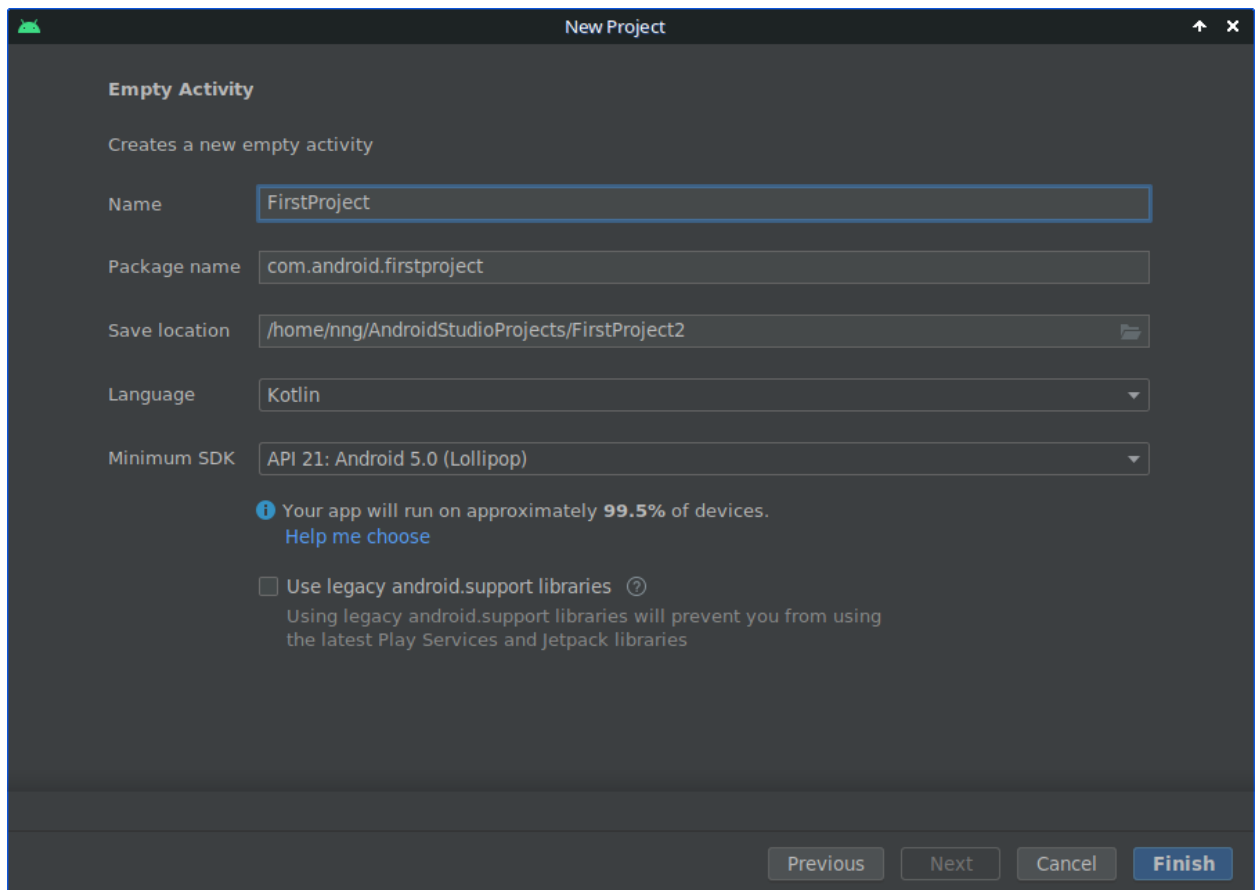


2. Pilih template sesuai dengan kebutuhan anda.

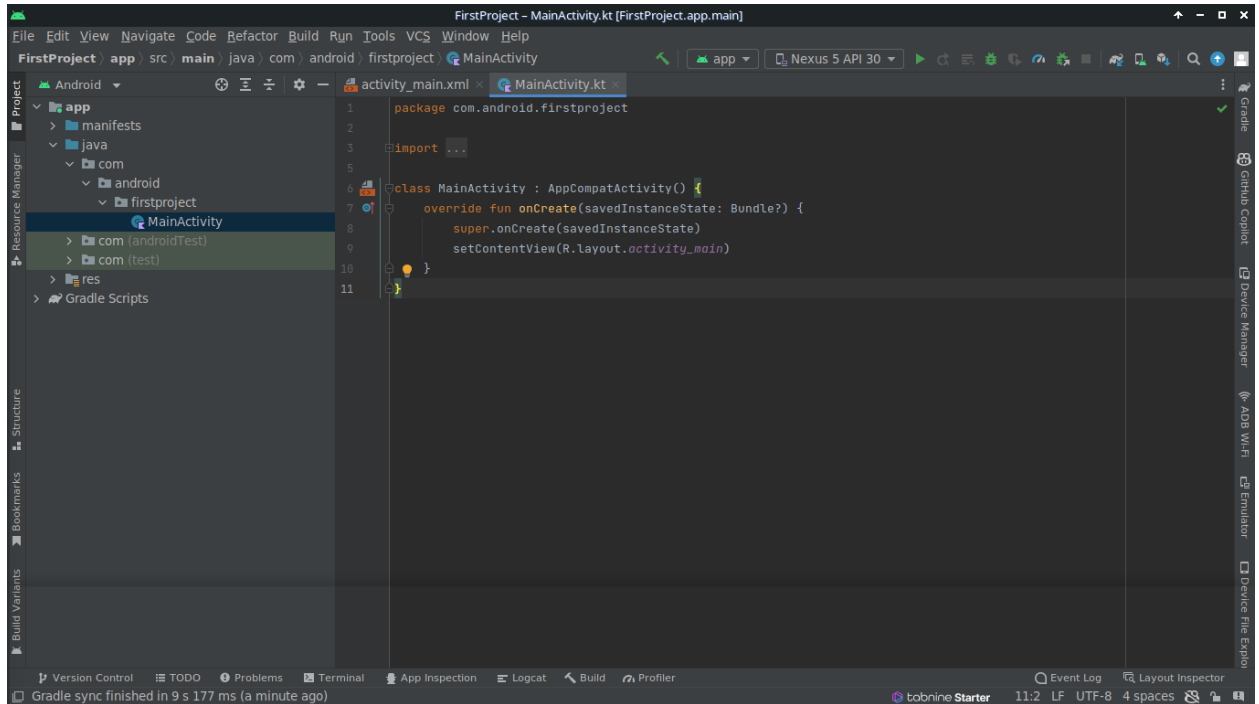


3. Lengkapi beberapa keterangan seperti Name, Package Name, Save location, Language, Minimum SDK.
 - **Name:** Nama aplikasi Anda yang akan ditampilkan di layar perangkat.
 - **Package Name:** Identitas unik aplikasi Anda yang digunakan oleh sistem operasi dan Google Play Store.
 - **Save location:** Lokasi di komputer Anda di mana proyek Anda akan disimpan.
 - **Language:** Bahasa pemrograman yang akan Anda gunakan, yaitu Java atau Kotlin.
 - **Minimum SDK:** Versi minimum sistem operasi Android yang didukung oleh aplikasi Anda.

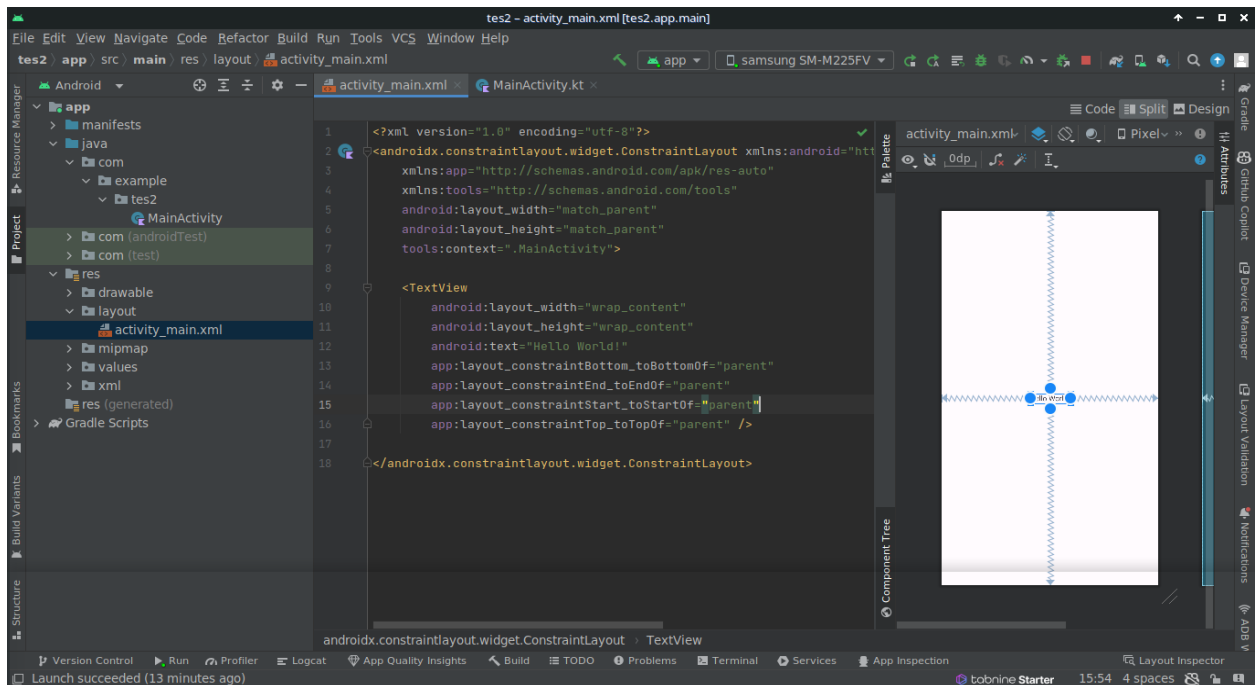
Setelah mengisi semua keterangan tersebut, klik Finish untuk membuat proyek Anda. Proses ini mungkin membutuhkan beberapa waktu tergantung pada kecepatan komputer Anda.



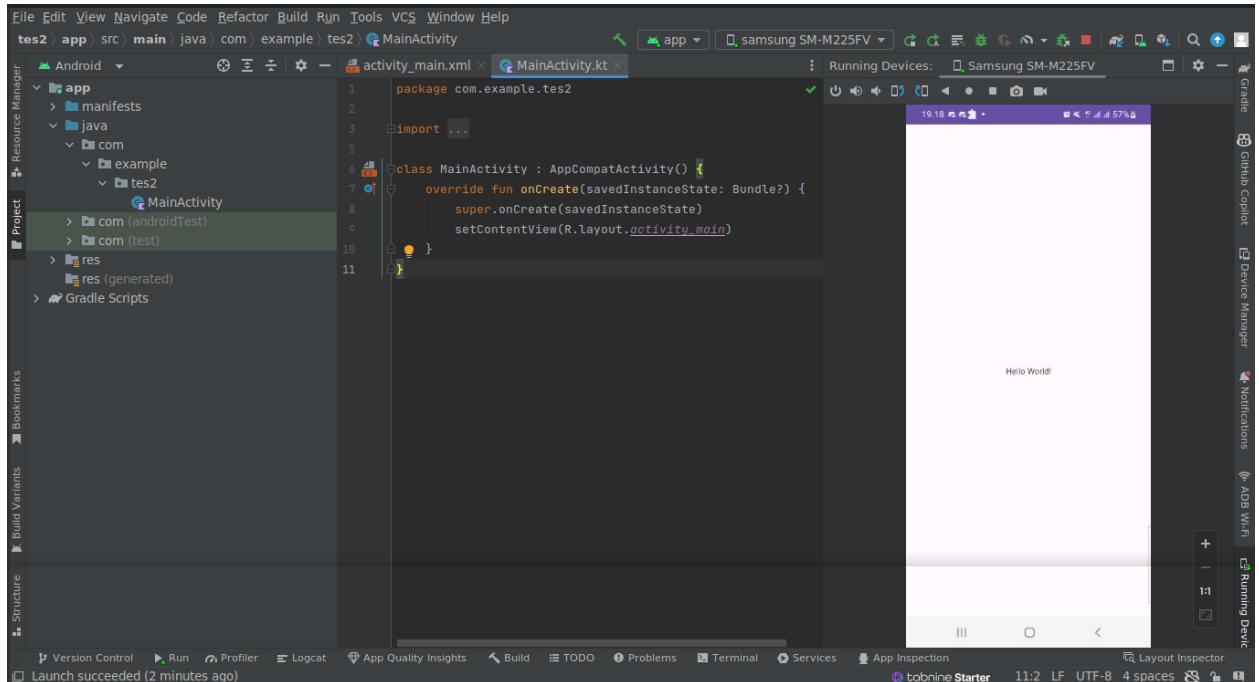
4. Anda akan melihat tampilan editor kode dengan file MainActivity.java atau MainActivity.kt terbuka. File ini berisi kelas utama yang mengontrol perilaku aplikasi. Berikut ini tampilan setelah proyek selesai dibuat.



- Di panel Project, buka app > res > layout > activity_main.xml. File ini berisi layout yang menentukan tampilan antarmuka aplikasi. Anda akan melihat sebuah TextView dengan teks "Hello World!" di tengah layar.



- Untuk menjalankan aplikasi, klik tombol Run di toolbar dan pilih perangkat atau emulator yang tersedia. Setelah aplikasi berhasil dijalankan, Anda akan melihat pesan "Hello World!" di layar perangkat atau emulator Anda.



Debugging dan Logcat

Debugging dan Logcat adalah fitur penting yang disediakan oleh Android Studio untuk membantu pengembang aplikasi menguji, menganalisis, dan memperbaiki kode mereka. Debugging memungkinkan pengembang untuk menghentikan sementara eksekusi aplikasi pada titik tertentu (breakpoint) dan memeriksa nilai variabel, ekspresi, atau kondisi. Logcat adalah alat yang menampilkan pesan log yang dihasilkan oleh aplikasi atau sistem Android. Pesan log dapat membantu pengembang melacak kesalahan, peringatan, informasi, atau kejadian lain yang terjadi selama pengembangan atau pengujian aplikasi.

Salah satu cara untuk melakukan debugging pada aplikasi Android adalah dengan menggunakan log. Log adalah pesan yang ditampilkan di jendela Logcat pada Android Studio. Log dapat memiliki berbagai level, seperti debug, error, info, dan warning. Untuk menulis log, kita bisa menggunakan kelas Log dan metode statisnya, seperti Log.d, Log.e, Log.i, dan Log.w. Metode-metode ini menerima dua parameter: tag dan pesan. Tag adalah string yang mengidentifikasi sumber log, biasanya nama kelas atau aktivitas. Pesan adalah string yang berisi informasi yang ingin kita tampilkan.

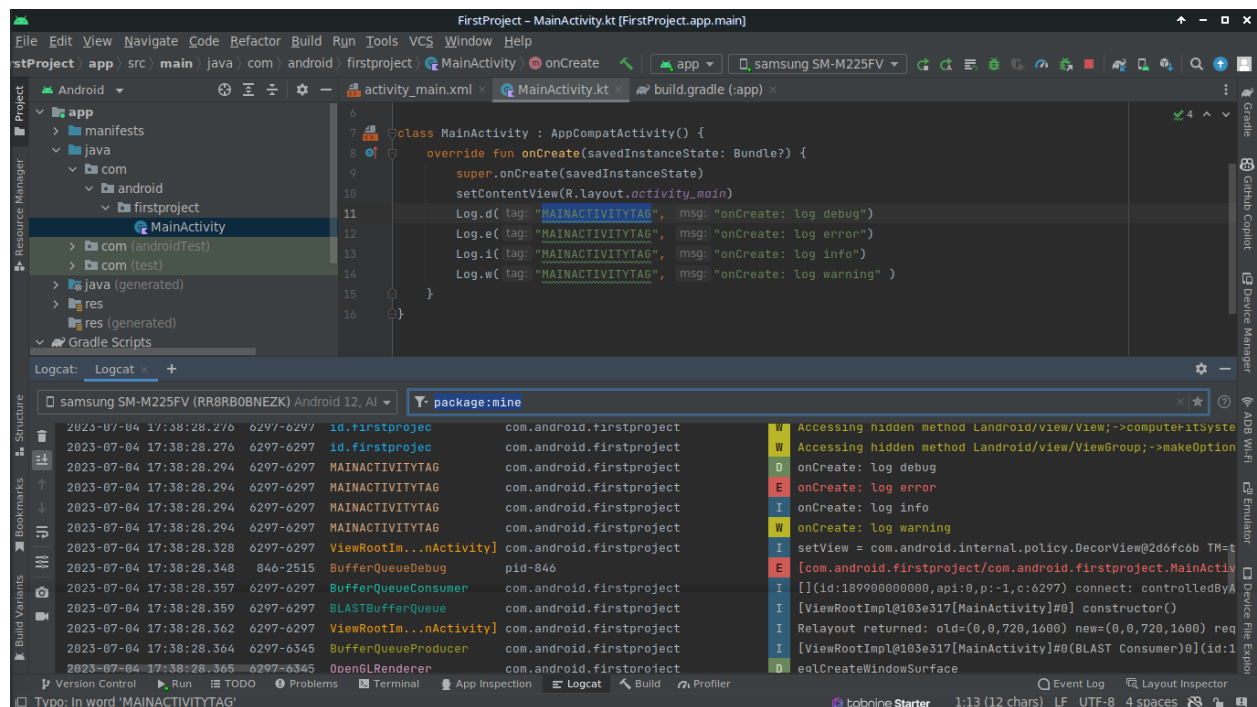
Contohnya, jika kita ingin menulis log pada metode onCreate dari MainActivity, kita bisa menulis kode berikut:

```
Log.d("MAINACTIVITYTAG", "onCreate: log debug")
Log.e("MAINACTIVITYTAG", "onCreate: log error")
Log.i("MAINACTIVITYTAG", "onCreate: log info")
Log.w("MAINACTIVITYTAG", "onCreate: log warning" )
```

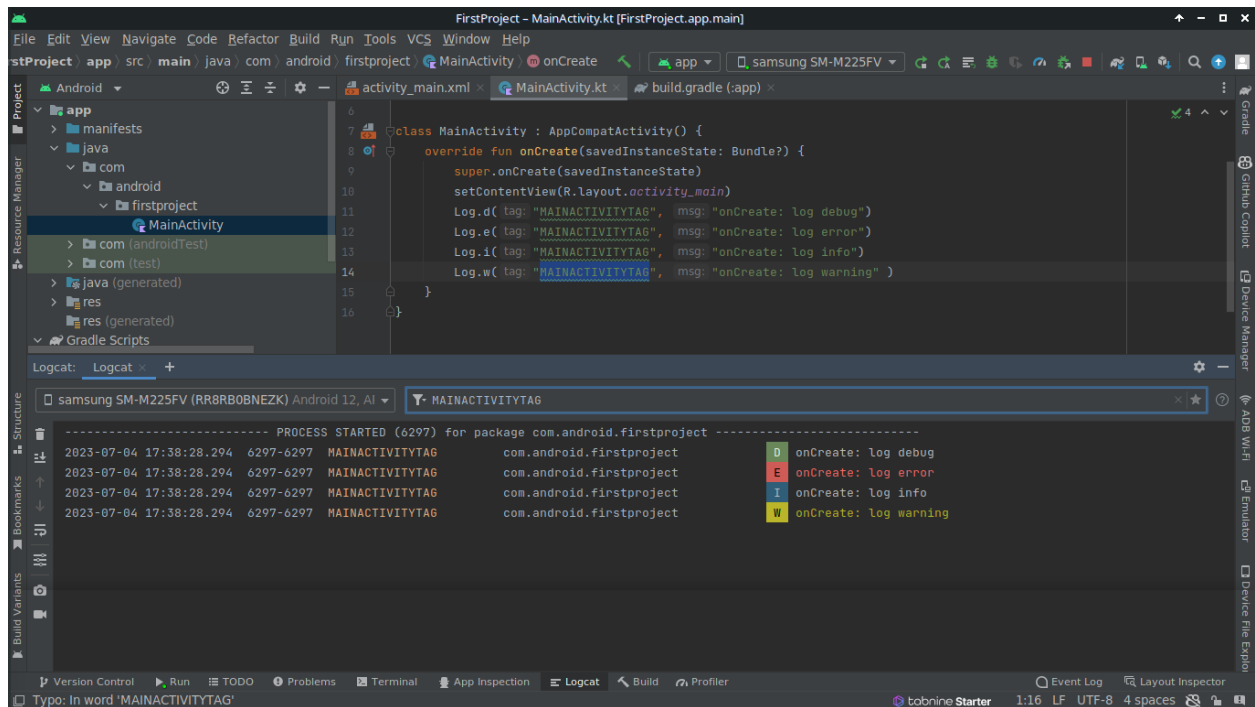
Penjelasan Code:

- **Log.d:** log debug, digunakan untuk menampilkan informasi detail tentang proses aplikasi, seperti nilai variabel atau parameter.
- **Log.e:** log error, digunakan untuk menampilkan informasi tentang kesalahan yang terjadi pada aplikasi, seperti exception atau crash.
- **Log.i:** log info, digunakan untuk menampilkan informasi umum tentang aplikasi, seperti status atau hasil operasi.
- **Log.w:** log warning, digunakan untuk menampilkan informasi tentang situasi yang berpotensi bermasalah pada aplikasi, seperti resource yang tidak tersedia atau input yang tidak valid.

Jika sudah, jalankan kembali proyek anda kemudian klik **“Logcat”** pada bagian bawah android studio.



Logcat di android studio juga dapat difilter. Berikut ini adalah contoh dari filter logcat dengan menggunakan tag “**MAINACTIVITYTAG**”



Sumber Refrensi:

- Google Developers. (2021). Android Studio overview. Diakses dari <https://developer.android.com/studio/intro>