

Membuat apa saja di Android

- Membuat mobile apps
- Uji aplikasi anda
- Berikan pedoman berkualitas dari aplikasi anda
- Distribusikan di Google Play

Mudah!...

Apapun Platformnya

PLATFORM

Wear OS



SELENGKAPNYA

PLATFORM

TV



SELENGKAPNYA

PLATFORM

Mobil



SELENGKAPNYA

PLATFORM

Chrome OS



PLATFORM

Objek



FITUR

Android NDK



Mengenal Android Studio

Android Studio adalah Lingkungan Pengembangan Terpadu (Integrated Development Environment/IDE) resmi untuk pengembangan aplikasi Android, yang didasarkan pada IntelliJ IDEA. Selain sebagai editor kode dan fitur developer IntelliJ yang andal, Android Studio menawarkan banyak fitur yang meningkatkan produktivitas Anda dalam membuat aplikasi Android, seperti:

- Sistem build berbasis Gradle yang fleksibel
- Emulator yang cepat dan kaya fitur
- Lingkungan terpadu tempat Anda bisa mengembangkan aplikasi untuk semua perangkat Android
- Terapkan Perubahan untuk melakukan push pada perubahan kode dan resource ke aplikasi yang sedang berjalan tanpa memulai ulang aplikasi
- Template kode dan integrasi GitHub untuk membantu Anda membuat fitur aplikasi umum dan mengimpor kode sampel
- Framework dan fitur pengujian yang lengkap
- Fitur lint untuk merekam performa, kegunaan, kompatibilitas versi, dan masalah lainnya
- Dukungan C++ dan NDK
- Dukungan bawaan untuk Google Cloud Platform, yang memudahkan integrasi Google Cloud Messaging dan App Engine

Download android studio : <https://developer.android.com/studio>

Menginstal Android Studio

Windows

Untuk menginstal Android Studio di Windows, lakukan langkah berikut:

1. Jika Anda sudah mendownload file `.exe` (direkomendasikan), klik dua kali untuk meluncurkannya.

Jika Anda sudah mendownload file `.zip`, ekstrak ZIP, salin folder **android-studio** ke folder **Program Files**, lalu buka folder **studio-android > bin** dan jalankan `studio64.exe` (untuk komputer 64 bit) atau `studio.exe` (untuk komputer 32 bit).

2. Ikuti wizard penyiapan di Android Studio dan instal paket SDK apa pun yang direkomendasikan.

Mac

Untuk menginstal Android Studio di Mac, lakukan langkah berikut:

1. Luncurkan file Android Studio DMG.
2. Tarik lalu lepas Android Studio ke dalam folder Application, lalu luncurkan Android Studio.
3. Pilih apakah Anda ingin mengimpor setelan Android Studio sebelumnya atau tidak, lalu klik **OK**.
4. Wizard Penyiapan Android Studio akan memandu Anda menyelesaikan langkah penyiapan selanjutnya, termasuk mendownload komponen Android SDK yang diperlukan untuk pengembangan.

Ringkasan project

Sebuah *project* di Android Studio berisi semua hal yang mendefinisikan ruang kerja Anda untuk aplikasi, mulai dari kode sumber dan aset hingga kode pengujian dan konfigurasi build. Saat Anda memulai project baru, Android Studio akan membuat struktur yang diperlukan untuk semua file dan membuatnya terlihat di jendela **Project** di sebelah kiri IDE (klik **Lihat > Jendela Fitur > Project**). Halaman ini menyediakan ringkasan komponen utama di dalam project Anda.

Modul

Modul adalah kumpulan file sumber dan setelan build yang memungkinkan Anda membagi project ke dalam beberapa unit fungsionalitas diskret. Project Anda dapat memiliki satu atau beberapa modul dan satu modul dapat menggunakan modul lain sebagai dependensi. Setiap modul dapat dibuat, diuji, dan di-debug secara independen.

Modul tambahan sering kali berguna saat membuat library kode dalam project Anda sendiri atau jika Anda ingin membuat set kode dan resource berbeda untuk beragam jenis perangkat, seperti ponsel dan perangkat wearable, tetapi menyimpan semua file cakupan dalam project yang sama dan berbagi sebagian kode.

Anda dapat menambahkan modul baru ke project Anda dengan mengklik **File > Baru > Modul Baru**.

Membuat project

Android Studio mempermudah pembuatan aplikasi Android untuk beragam faktor bentuk, seperti handset, tablet, TV, dan perangkat Wear. Halaman ini menunjukkan cara memulai project aplikasi Android baru atau mengimpor project yang sudah ada.

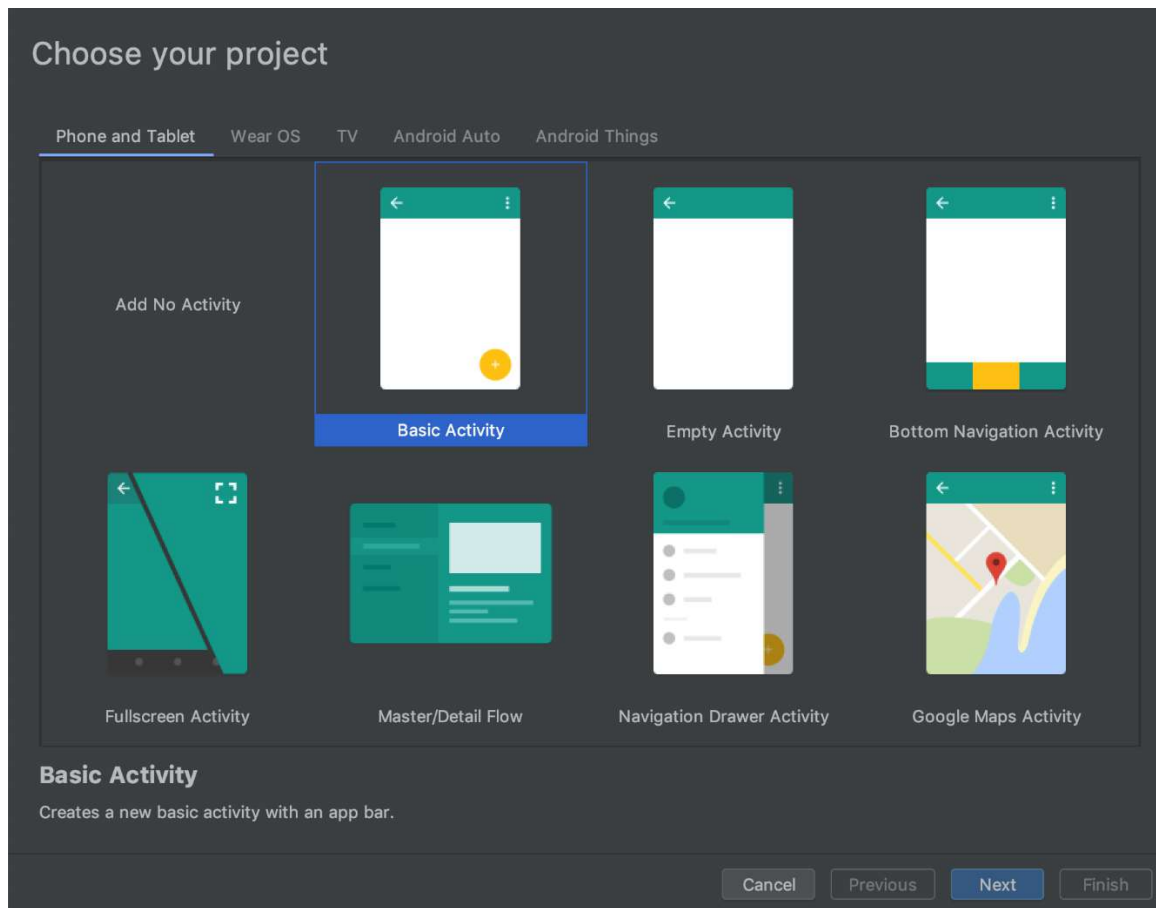
Jika Anda belum membuka project, Android Studio akan menampilkan layar Selamat Datang, di mana Anda dapat membuat project baru dengan mengklik **Start a new Android Studio project**.

Atau, jika sudah ada project yang terbuka, Anda dapat mulai membuat project baru dengan memilih **File > New > New Project** dari menu utama.

Selanjutnya Anda akan melihat wizard **Create New Project**, yang memungkinkan Anda memilih jenis project yang ingin dibuat dan diisi dengan kode dan resource untuk memulai. Halaman ini akan memandu Anda untuk membuat project baru menggunakan wizard **Create New Project**.

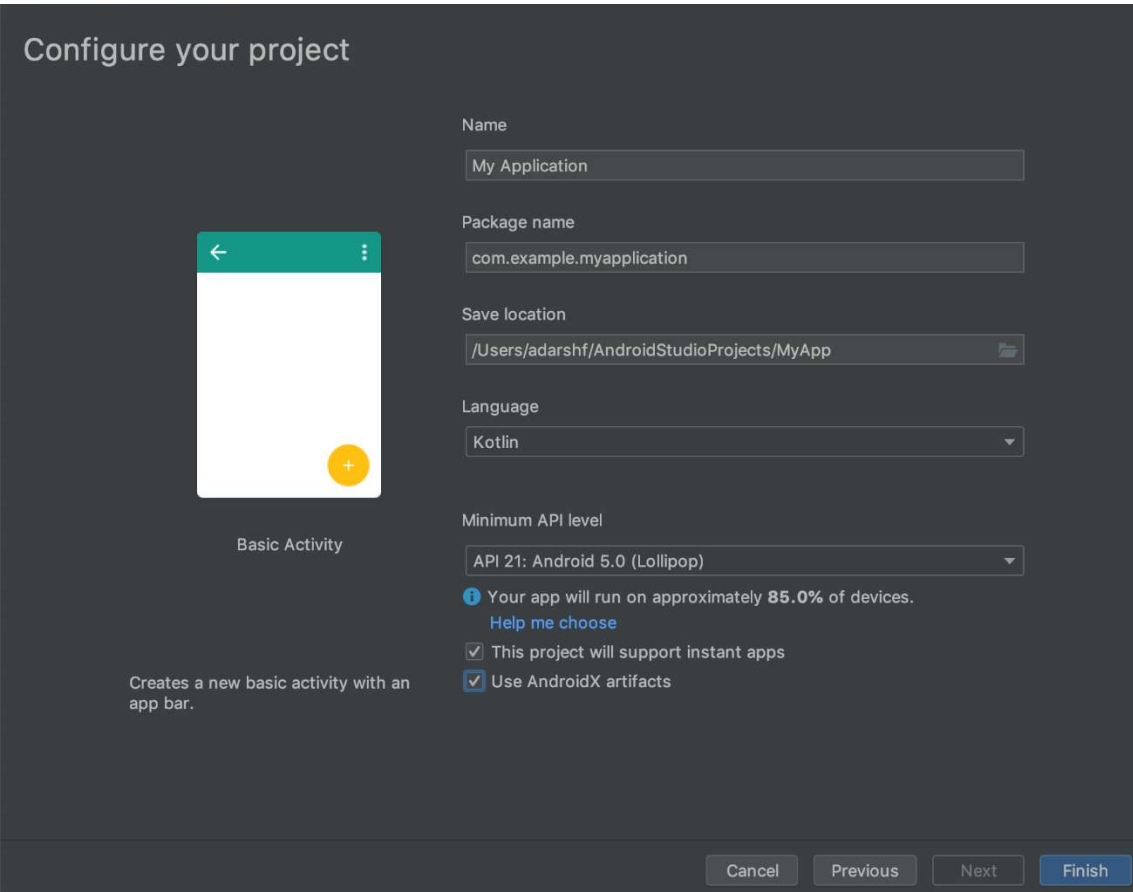
Memilih project Anda

Di layar **Choose your project** yang muncul, Anda dapat memilih jenis project yang ingin dibuat dari kategori faktor bentuk perangkat, yang ditampilkan sebagai tab di dekat bagian atas wizard. Misalnya, gambar 1 menunjukkan project dengan Aktivitas Android dasar untuk ponsel dan tablet yang dipilih.



Mengonfigurasi project Anda

Langkah selanjutnya adalah mengonfigurasi beberapa setelan dan membuat project baru seperti yang dijelaskan di bawah ini dan ditunjukkan pada gambar 2. Jika Anda membuat project **Native C++**, Anda dapat mempelajari opsi yang perlu dikonfigurasi lebih lanjut dengan membaca [Membuat project baru dengan dukungan C/C++](#).



Configure your project

Basic Activity

Creates a new basic activity with an app bar.

Name: My Application

Package name: com.example.myapplication

Save location: /Users/adarshf/AndroidStudioProjects/MyApp

Language: Kotlin

Minimum API level: API 21: Android 5.0 (Lollipop)

Help Your app will run on approximately 85.0% of devices. [Help me choose](#)

☒ This project will support instant apps

☒ Use AndroidX artifacts

Cancel Previous Next Finish

1. Tentukan **Name** project Anda.
2. Tentukan **Package name**. Secara default, nama paket ini juga menjadi **ID aplikasi** Anda, yang dapat diubah di lain waktu.
3. Tentukan **Save location** tempat Anda ingin menyimpan project secara lokal di perangkat yang Anda gunakan.
4. Pilih **Language** yang akan digunakan oleh Android Studio saat membuat kode contoh untuk project baru Anda. Ingat, Anda tidak dibatasi untuk hanya menggunakan bahasa yang dipakai dalam membuat project.
5. Pilih **Minimum API level** yang perlu didukung aplikasi Anda. Ketika memilih API level yang lebih rendah, aplikasi Anda dapat menggunakan API Android modern dalam jumlah yang lebih sedikit. Namun, sebagian besar perangkat Android dapat menjalankan aplikasi Anda. Hal sebaliknya akan berlaku jika Anda memilih API level yang lebih tinggi. Jika ingin melihat lebih banyak data untuk membantu menentukan keputusan, klik **Help me choose**.
6. Jika jenis project yang dipilih mendukung pengalaman instan melalui [Google Play Instan](#), dan Anda ingin mengaktifkannya untuk aplikasi Anda, centang kotak di samping **This project will support instant apps**. Untuk mempelajari lebih lanjut, baca [Membuat aplikasi instan pertama Anda](#).
7. Jika ingin project Anda menggunakan library AndroidX secara default, yang merupakan pengganti yang lebih baik dari Android Support Library, centang kotak di sebelah **Use AndroidX artifacts**. Untuk mempelajari lebih lanjut, baca [Ringkasan AndroidX](#).
8. Jika Anda sudah siap membuat project, klik **Finish**.

Android Studio akan membuat project baru Anda dengan beberapa kode dasar dan resource untuk memulai. Jika nanti Anda memutuskan untuk menambah dukungan bagi faktor bentuk perangkat lain, Anda dapat [menambahkan modul](#) ke project Anda pada lain waktu. Selain itu, jika ingin berbagi kode dan resource antar-modul, Anda dapat melakukannya dengan membuat [library Android](#).

Membuat aplikasi pertama Anda

Bagian ini menjelaskan cara membuat aplikasi Android sederhana. Pertama, Anda akan mempelajari cara membuat project "Hello, World!" dengan Android Studio lalu menjalankannya. Kemudian, Anda membuat antarmuka baru untuk aplikasi yang menerima input pengguna dan beralih ke layar baru di aplikasi untuk menampilkannya.

Sebelum memulai, ada dua konsep dasar tentang aplikasi Android yang harus Anda pahami: bagaimana aplikasi tersebut menyediakan beberapa titik masuk, dan bagaimana aplikasi tersebut beradaptasi dengan berbagai perangkat.

Aplikasi menyediakan beberapa titik masuk

Aplikasi Android dirancang sebagai kombinasi dari beragam komponen yang bisa dipanggil satu per satu. Misalnya, *aktivitas* adalah jenis komponen aplikasi yang menyediakan antarmuka pengguna (UI).

Aktivitas "utama" dimulai saat pengguna mengetuk ikon aplikasi. Anda juga dapat mengarahkan pengguna ke aktivitas dari tempat lain, seperti dari notifikasi atau bahkan dari aplikasi yang berbeda.

Komponen lain, seperti *penerima siaran* dan *layanan*, memungkinkan aplikasi Anda untuk melakukan tugas latar belakang tanpa UI.

Setelah membuat aplikasi pertama, Anda dapat mempelajari komponen aplikasi lainnya lebih lanjut pada

Aplikasi beradaptasi dengan berbagai perangkat

Android memungkinkan Anda menyediakan resource yang berbeda untuk perangkat yang berbeda. Misalnya, Anda bisa membuat tata letak yang berbeda untuk ukuran layar yang berbeda. Sistem akan menentukan tata letak mana yang akan digunakan berdasarkan ukuran layar perangkat yang sedang digunakan.

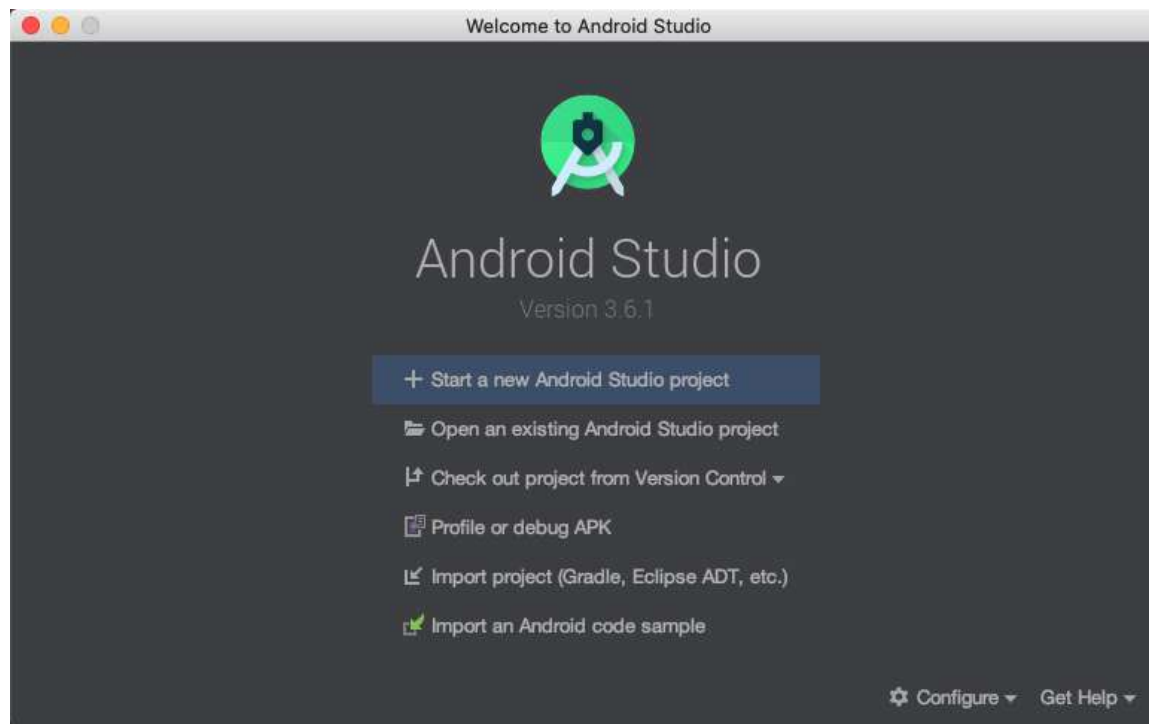
Jika salah satu fitur aplikasi Anda memerlukan hardware tertentu, seperti kamera, Anda dapat mengkueri pada saat runtime terlepas dari apakah perangkat memiliki hardware itu atau tidak, lalu menonaktifkan fitur tersebut jika tidak. Anda dapat menentukan bahwa aplikasi Anda memerlukan hardware tertentu sehingga Google Play tidak akan mengizinkan aplikasi untuk diinstal pada perangkat tanpa hardware tersebut.

Membuat project Android

Tutorial ini menunjukkan cara membuat project Android baru dengan Android Studio, dan menjelaskan beberapa file dalam project tersebut.

Untuk membuat project Android baru, ikuti langkah-langkah berikut:

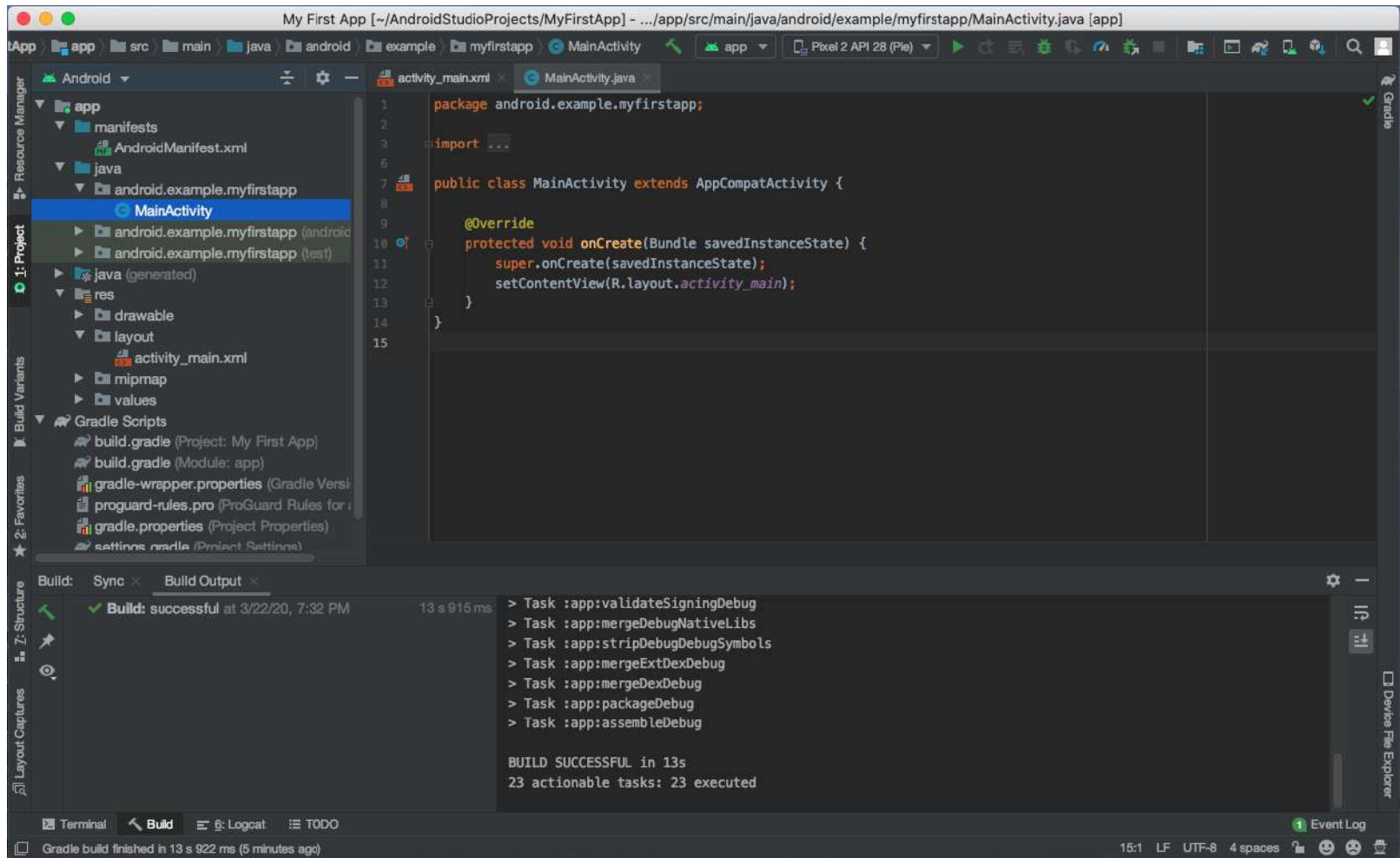
1. Instal [Android Studio](#)  versi terbaru.
2. Di jendela **Welcome to Android Studio**, klik **Start a new Android Studio project**.



3. Di jendela **Choose your project**, pilih **Empty Activity** dan klik **Next**.
4. Di jendela **Configure your project**, lakukan langkah-langkah berikut:
 - Masukkan "My First App" pada kolom **Name**.
 - Masukkan "com.example.myfirstapp" pada kolom **Package name**.
 - Jika ingin menempatkan project di folder lain, ubah lokasi **Save**-nya.
 - Pilih **Java** atau **Kotlin** dari menu drop-down **Language**.
 - Centang kotak di samping **Use androidx.* artifacts**.
 - Biarkan opsi lain sebagaimana adanya.
5. Klik **Finish**.

Sekarang, gunakan waktu Anda untuk meninjau file-file yang paling penting.

Pertama, pastikan jendela **Project** terbuka (pilih **View > Tool Windows > Project**) dan tampilan Android dipilih dari menu drop-down di bagian atas jendela tersebut. Kemudian, Anda bisa melihat file berikut:



app > java > com.example.myfirstapp > MainActivity

Ini adalah aktivitas utama. Ini adalah titik masuk untuk aplikasi Anda. Saat Anda merancang dan menjalankan aplikasi, sistem akan meluncurkan sebuah instance dari [Activity](#) dan memuat tata letaknya.

app > res > layout > activity_main.xml

File XML ini mendefinisikan tata letak untuk antarmuka pengguna (UI) aktivitas. File ini berisi elemen [TextView](#) dengan teks "Hello, World!"

app > manifests > AndroidManifest.xml

[File manifest](#) menjelaskan karakteristik dasar aplikasi dan mendefinisikan masing-masing komponennya.

Gradle Scripts > build.gradle

Ada dua file dengan nama berikut: satu untuk project, "Project: My First App", dan satu lagi untuk modul aplikasi, "Module: app". Setiap modul memiliki file [build.gradle](#) sendiri, tetapi untuk sekarang project ini hanya memiliki satu modul. Gunakan setiap [build.file](#) modul untuk mengontrol cara [plugin Gradle](#) merancang aplikasi Anda. Untuk informasi selengkapnya tentang file ini, lihat [Mengonfigurasi perancangan Anda](#).

Menjalankan aplikasi Anda

Pada [tutorial sebelumnya](#), Anda membuat project Android yang menampilkan tulisan "Hello World!" Sekarang Anda bisa menjalankan aplikasi tersebut di perangkat sungguhan atau emulator.

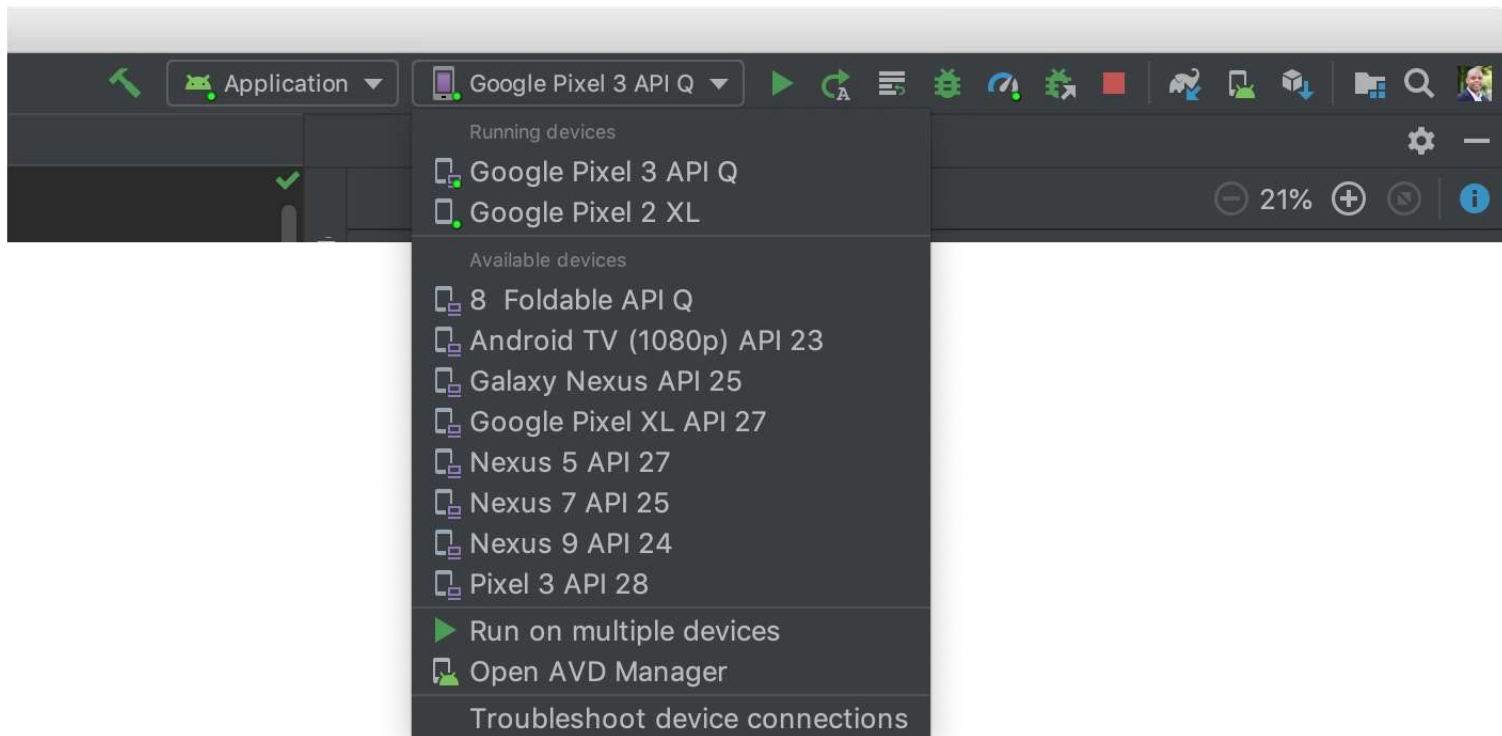
Menjalankan di perangkat sungguhan

Persiapkan perangkat Anda sebagai berikut:

1. Hubungkan perangkat Anda ke komputer pengembangan dengan kabel USB. Jika melakukan pengembangan pada Windows, Anda mungkin perlu [menginstal driver USB yang tepat](#) untuk perangkat Anda.
2. Lakukan langkah-langkah berikut untuk mengaktifkan **proses debug USB** di jendela **Opsi developer**:
 - a. Buka aplikasi **Setelan**.
 - b. Jika perangkat Anda menggunakan Android v8.0 atau lebih tinggi, pilih **Sistem**. Jika tidak, lanjutkan ke langkah berikutnya.
 - c. Scroll ke bagian bawah, lalu pilih **Tentang ponsel**.
 - d. Scroll ke bagian bawah, lalu ketuk **Nomor build** tujuh kali.
 - e. Kembali ke layar sebelumnya, scroll ke bagian bawah, dan ketuk **Opsi developer**.
 - f. Di jendela **Opsi developer**, scroll ke bawah untuk menemukan dan mengaktifkan **proses debug USB**.

Jalankan aplikasi di perangkat Anda sebagai berikut:

1. Di Android Studio, pilih aplikasi Anda dari menu drop-down konfigurasi run/debug di toolbar.
2. Di toolbar, pilih perangkat tempat Anda ingin menjalankan aplikasi dari menu drop-down perangkat target.



Klik **Run** .

Android Studio akan menginstal aplikasi Anda di perangkat yang terhubung dan menjalankannya. Setelah itu, Anda akan melihat "Hello, World!" ditampilkan dalam aplikasi di perangkat Anda.

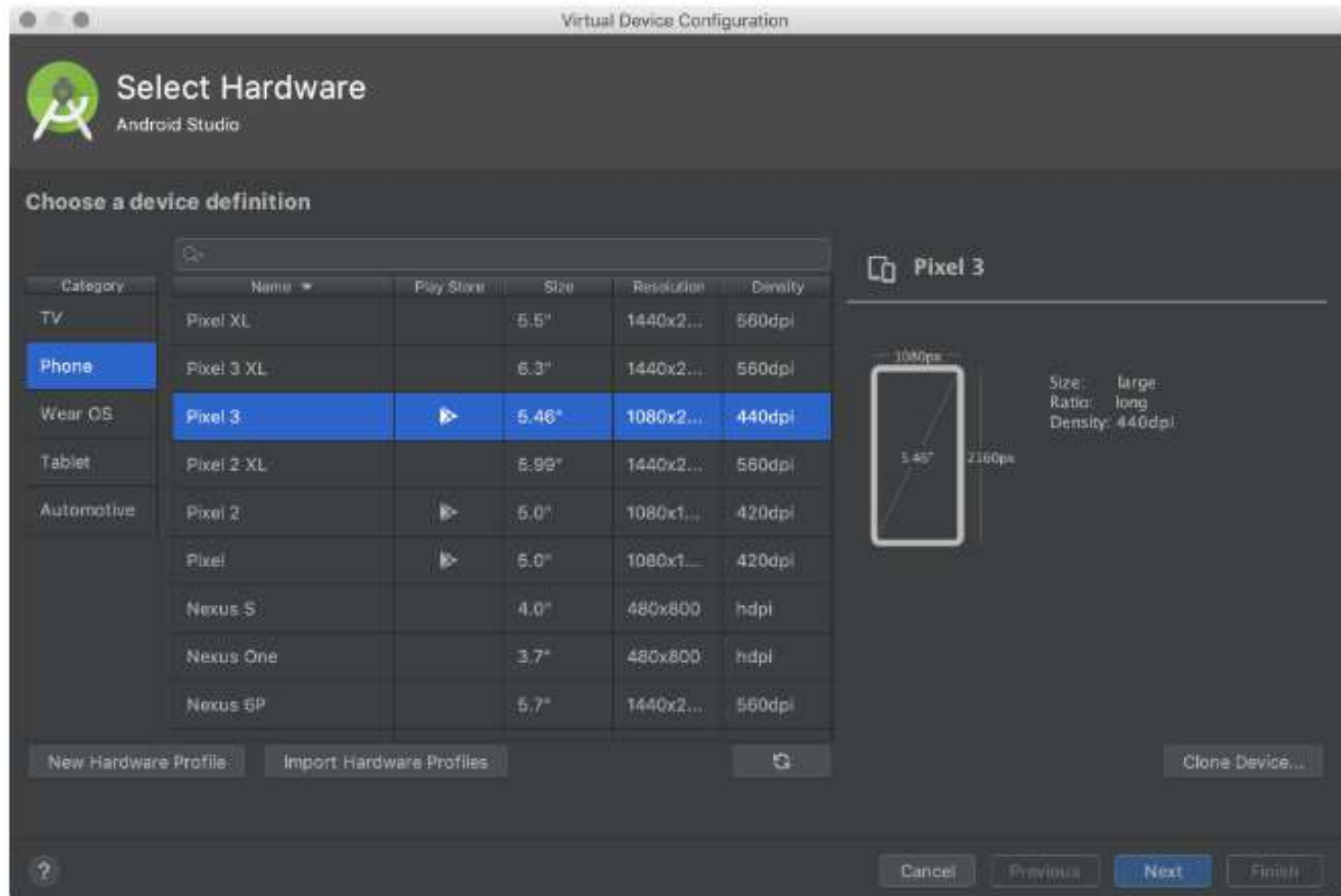
Menjalankan aplikasi di emulator

1. Buka AVD Manager dengan mengklik **Tools > AVD Manager**.



2. Klik **Create Virtual Device**, di bagian bawah dialog AVD Manager.

Halaman **Select Hardware** akan muncul.

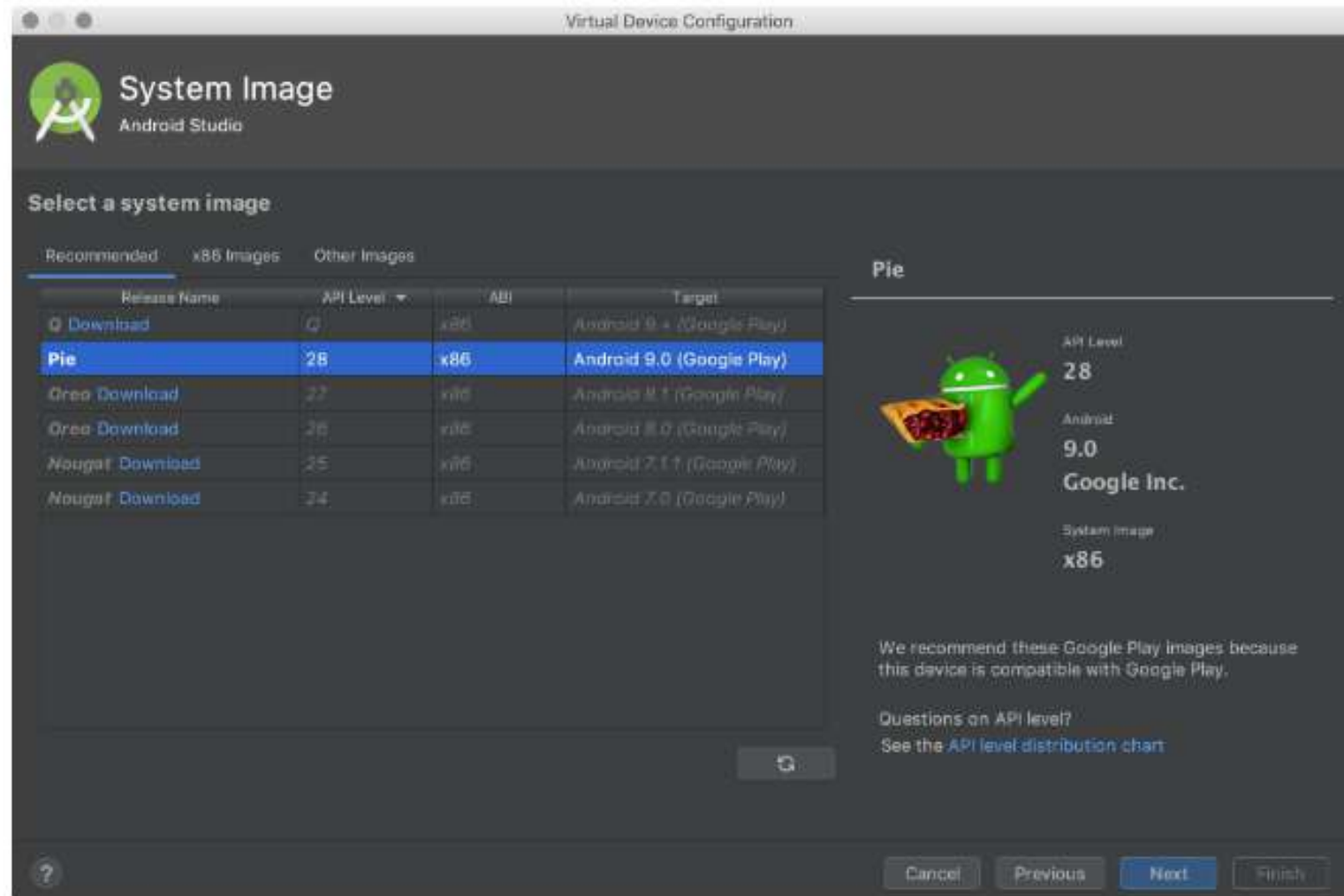


Perhatikan bahwa hanya beberapa profil hardware yang diindikasikan menyertakan **Play Store**. Hal ini mengindikasikan bahwa profil-profil ini sepenuhnya mematuhi **CTS** dan dapat menggunakan image sistem yang menyertakan aplikasi Play Store.

3. Pilih profil hardware, lalu klik **Next**.

Jika tidak melihat profil hardware yang diinginkan, Anda dapat [membuat](#) atau [mengimpor](#) profil hardware.

Halaman **System Image** akan muncul.



4. Pilih image sistem untuk API level tertentu, lalu klik **Next**.

Tab **Recommended** akan mencantumkan image sistem yang disarankan. Tab lainnya mencantumkan daftar yang lebih lengkap. Panel kanan menunjukkan image sistem yang dipilih. Image x86 dijalankan paling cepat dalam emulator.

Jika melihat **Download** di sebelah image sistem, Anda perlu mengkliknya untuk mendownload image sistem. Anda harus terhubung ke internet untuk mendownloadnya.

API level perangkat target cukup penting karena aplikasi Anda tidak dapat berjalan pada image sistem dengan API level yang lebih rendah dari yang diwajibkan oleh aplikasi, seperti yang ditetapkan dalam atribut `minSdkVersion` file manifest aplikasi. Untuk informasi selengkapnya tentang hubungan antara API level sistem dan `minSdkVersion`, lihat [Membuat Versi Aplikasi Anda](#).

Jika aplikasi Anda mendeklarasikan elemen `<uses-library>` dalam file manifest, aplikasi memerlukan image sistem tempat library eksternal tersebut berada. Jika Anda ingin menjalankan aplikasi pada emulator, buat AVD yang menyertakan library yang diperlukan. Untuk melakukannya, Anda mungkin perlu menggunakan komponen add-on untuk platform AVD; misalnya, add-on Google API berisi library Google Maps.

Halaman **Verify Configuration** akan muncul.



Android Virtual Device (AVD)

Android Studio

Verify Configuration

AVD Name Pixel 3 API 28



Pixel 3

5.46 1080x2160 xxhdpi

Change...



Pie

Android 9.0 x86

Change...

Startup orientation



Portrait



Landscape

Emulated
Performance

Graphics: Automatic

Device Frame ☒ Enable Device Frame

Show Advanced Settings

AVD Name

The name of this AVD.



Cancel

Previous

Next

Finish

5. Ubah [AVD properties](#) sesuai kebutuhan, lalu klik **Finish**.

Klik **Show Advanced Settings** untuk menampilkan setelan lainnya, seperti skin.

AVD baru akan muncul di halaman **Your Virtual Devices** atau dialog **Select Deployment Target**.

Untuk membuat AVD yang dimulai dengan salinan:

1. Dari halaman [Your Virtual Devices](#) pada AVD Manager, klik kanan salah satu AVD, lalu pilih **Duplicate**.

Atau, klik Menu ▼ dan pilih **Duplicate**.

Halaman [Verify Configuration](#) akan muncul.

2. Klik **Change** atau **Previous** jika Anda perlu melakukan perubahan pada halaman [System Image](#) dan [Select Hardware](#).

3. Buat perubahan, lalu klik **Finish**.

AVD akan muncul di halaman **Your Virtual Devices**.