

# Membuat dan menjalankan aplikasi Android pertama Anda

## 1. Pendahuluan

Dalam codelab ini, Anda akan membuat aplikasi Android pertama (bernama Happy Birthday) yang dimulai dari template untuk aplikasi dasar yang disediakan oleh Android Studio. Anda juga akan mempelajari seperti apa project Android itu, serta cara menggunakan jendela yang berbeda di Android Studio.

### Prasyarat

- Pemahaman tentang penyiapan, konfigurasi, dan penggunaan aplikasi, seperti pengolah kata atau spreadsheet

### Yang akan Anda pelajari

- Cara membuat project Android Studio untuk aplikasi Android menggunakan template

### Yang akan Anda build

- Aplikasi Android dasar dari template

### Yang Anda perlukan

- Komputer yang dilengkapi Android Studio

## 2. Membuat aplikasi pertama Anda

Dalam tugas ini, Anda akan membuat aplikasi Android menggunakan template project yang disediakan oleh Android Studio.

### Template project

Di Android Studio, template project adalah aplikasi Android yang memiliki semua bagian yang diperlukan, tetapi fungsinya terbatas. Tujuannya adalah untuk membantu Anda memulai lebih cepat dan mempercepat pekerjaan. Beberapa contoh template di Android Studio adalah aplikasi dengan peta, dan aplikasi dengan beberapa layar.

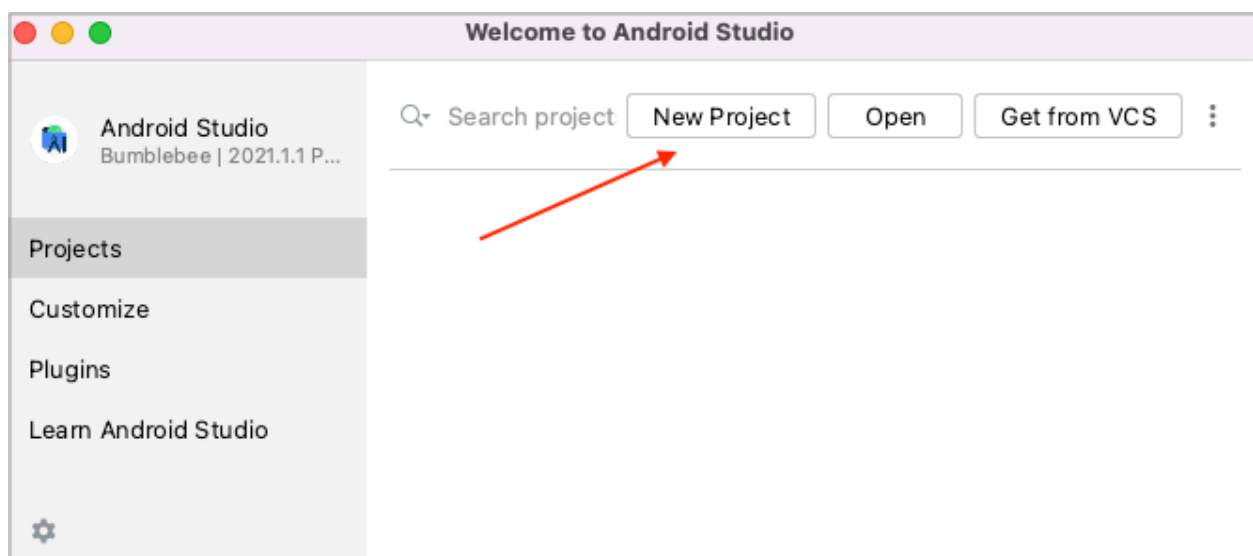
### Membuat project Empty Activity

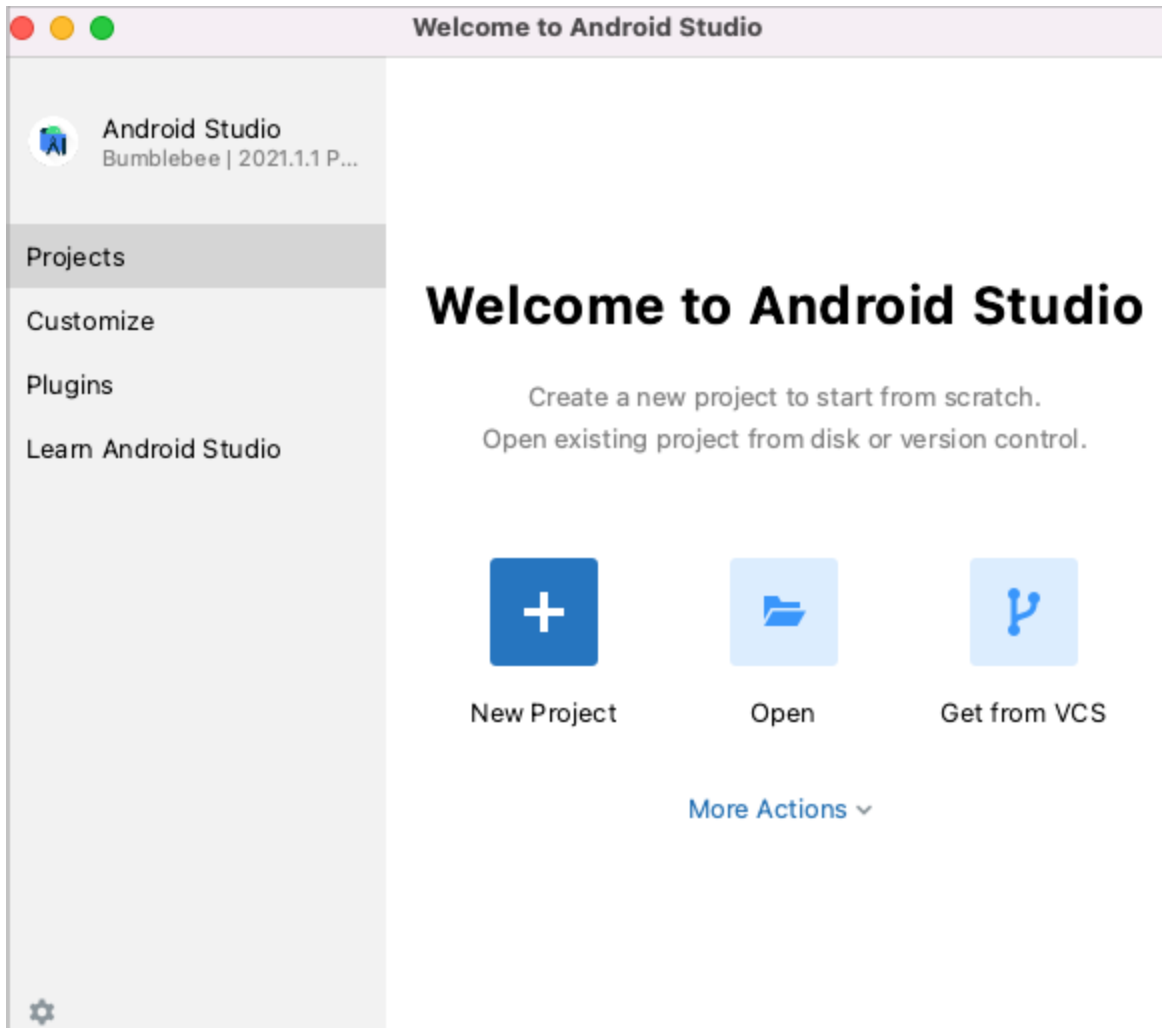
Dalam langkah ini, Anda akan membuat project Android Studio baru menggunakan template project Empty Activity untuk aplikasi baru.

1. Mulai Android Studio (jika belum terbuka) dengan mengklik ikon Android Studio:

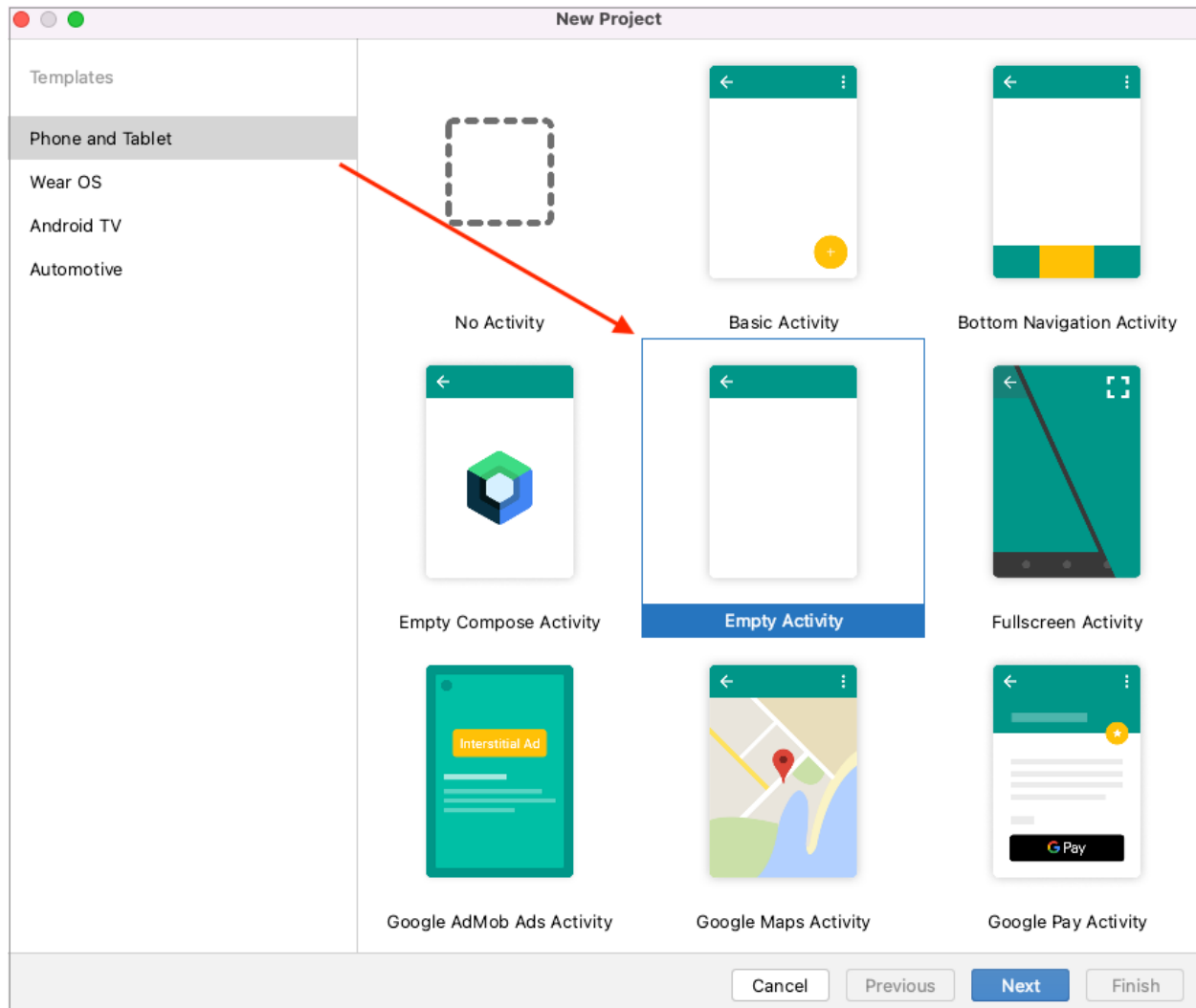


2. Jendela Welcome to Android Studio akan terbuka. Klik New Project.





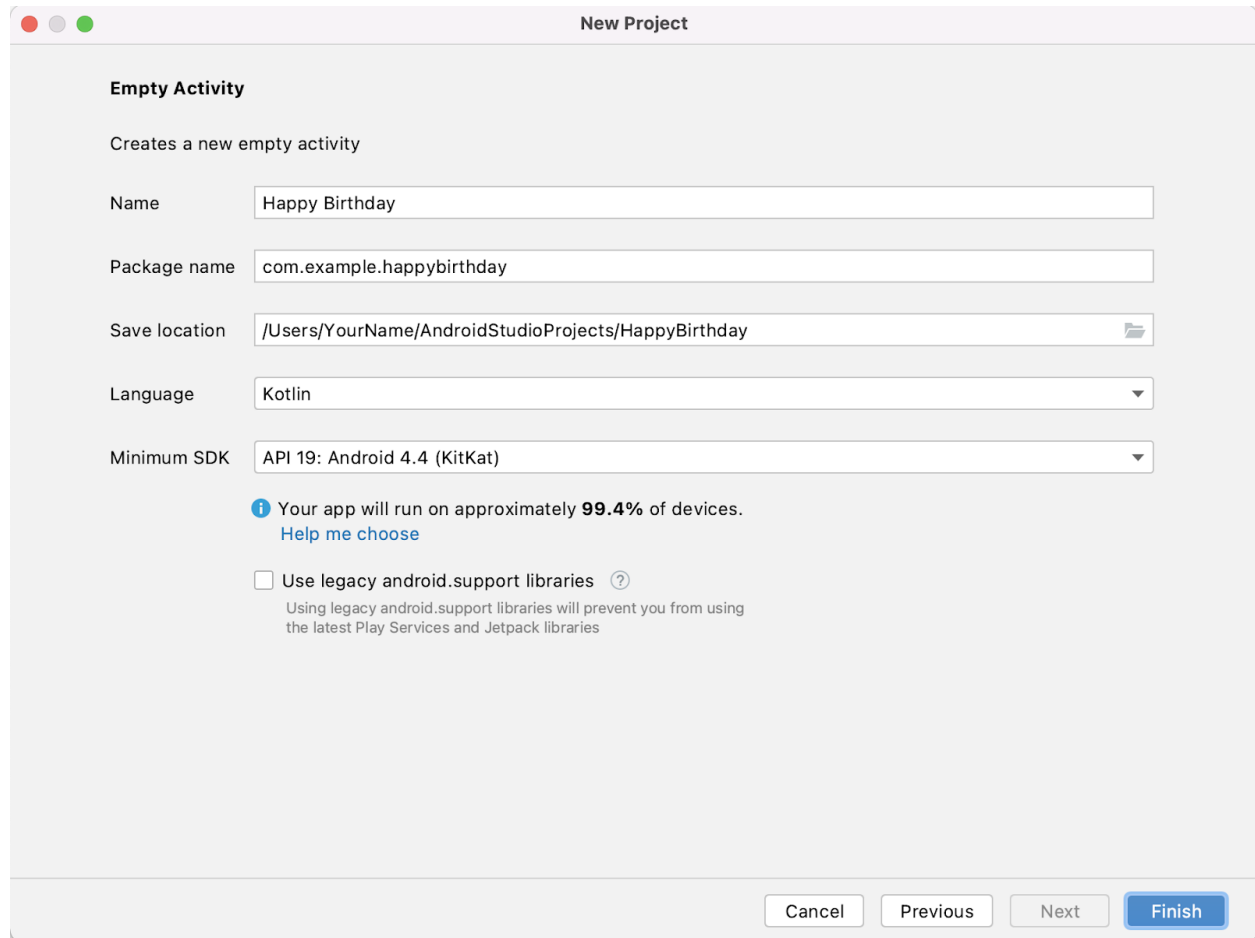
3. Jendela New Project akan terbuka dengan daftar template yang disediakan oleh Android Studio.



4. Klik tab di bagian atas, lalu scroll template untuk mendapatkan gambaran tentang hal yang dapat Anda lakukan. Terdapat template untuk berbagai jenis perangkat (seperti ponsel, tablet, dan smartwatch), serta berbagai jenis aplikasi (aplikasi dengan layar yang dapat di-scroll, aplikasi dengan peta, dan aplikasi dengan navigasi mewah).
5. Di kiri atas jendela, klik tab Phone and Tablet.
6. Di baris atas, klik template Empty Activity untuk memilihnya sebagai template untuk project Anda.

Template Empty Activity adalah template paling sederhana yang dapat digunakan untuk membuat aplikasi. Template ini memiliki satu layar dan menampilkan pesan "Hello World!" yang sederhana.

7. Di bagian bawah jendela, klik Next. Dialog New Project akan terbuka.



Konfigurasi project Anda sebagai berikut:

8. Name adalah nama aplikasi Anda. Pada kotak di bawah Name, masukkan **Happy Birthday** sebagai nama project Anda.
9. Package name adalah nama yang digunakan oleh sistem Android untuk mengidentifikasi aplikasi Anda secara unik. Biasanya, nama ini akan ditetapkan secara default ke nama organisasi Anda diikuti dengan nama aplikasi, semuanya dalam huruf kecil (dalam hal ini **"com.example.happybirthday"**).
10. Save location adalah lokasi tempat semua file yang terkait dengan project Anda disimpan. Catat lokasi tersebut di komputer, agar Anda dapat menemukan file Anda. Anda juga dapat keluar dari lokasi penyimpanan untuk saat ini.
11. Language menentukan bahasa pemrograman yang ingin Anda gunakan untuk project. Pastikan Language adalah **Kotlin**.
12. Minimum SDK menunjukkan versi minimum Android yang dapat dijalankan oleh aplikasi Anda. Pilih **API 19: Android 4.4 (KitKat)** dari menu drop-down.

Catatan: Ada berbagai macam versi sistem operasi Android, yang masing-masing diberi nama dalam urutan abjad saat dirilis.

13. Di bawah Minimum SDK, perhatikan catatan informasi tentang jumlah perangkat yang dapat menjalankan aplikasi Anda dengan API level yang dipilih. Jika ingin tahu, klik link Help me choose untuk menampilkan daftar versi Android yang berbeda, sebagaimana ditampilkan di bawah ini. Kemudian, kembali ke jendela New Project.

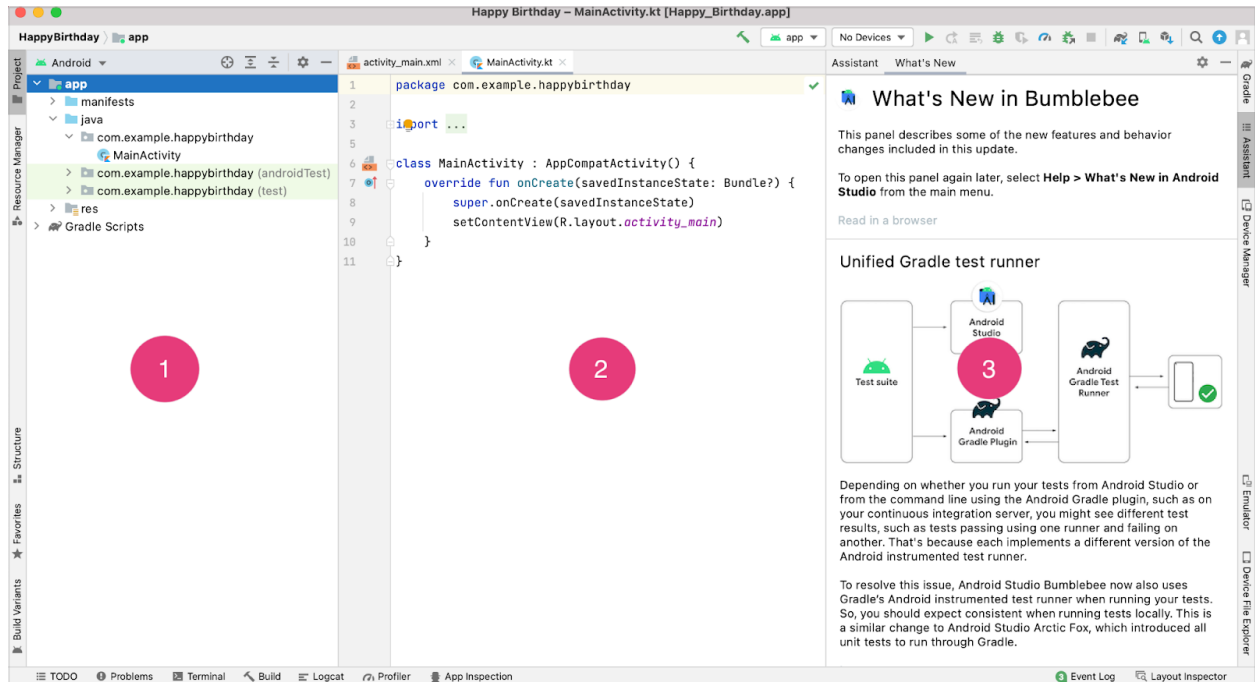
| Android Platform/API Version Distribution |           |                         |  |  |
|---|-----------|-------------------------|--|--|
| ANDROID PLATFORM VERSION                  | API LEVEL | CUMULATIVE DISTRIBUTION |  |  |
| 4.1 Jelly Bean                            | 16        |                         | <b>KitKat</b><br>Printing Framework<br>Print generic content<br>Print images<br>OEM print services<br>SMS Provider<br>Read and write SMS and MMS messages<br>Select default SMS app<br>Wireless and Connectivity<br>Host NFC card emulation<br>NFC reader mode<br>Infrared support<br>Multimedia<br>Adaptive video playback<br>On-demand audio timestamps<br>Surface image reader<br>Peak and RMS audio measurements<br>Loudness enhancer<br>Remote controllers<br>Closed captions<br>Animation and Graphics<br>Scenes and transitions<br>Animator pausing<br>Reusable bitmaps | User Content<br>Storage access framework<br>External storage access<br>Sync adapters<br>User Input<br>New sensor types, including step detector<br>Batched sensor events<br>Controller identities<br>User Interface<br>Immersive full-screen mode<br>Translucent system bars<br>Enhanced notification listener<br>Live regions for accessibility |
| 4.2 Jelly Bean                            | 17        | 99.8%                   |  |  |
| 4.3 Jelly Bean                            | 18        | 99.5%                   |  |  |
| 4.4 KitKat                                | 19        | 99.4%                   |  |  |
| 5.0 Lollipop                              | 21        | 98.0%                   |  |  |
| 5.1 Lollipop                              | 22        | 97.3%                   |  |  |
| 6.0 Marshmallow                           | 23        | 94.1%                   |  |  |
| 7.0 Nougat                                | 24        | 89.0%                   |  |  |
| 7.1 Nougat                                | 25        | 85.6%                   |  |  |
| 8.0 Oreo                                  | 26        | 82.7%                   |  |  |
| 8.1 Oreo                                  | 27        | 78.7%                   |  |  |
| 9.0 Pie                                   | 28        | 69.0%                   |  |  |
| 10. Q                                     | 29        | 50.8%                   |  |  |
| 11. R                                     | 30        | 24.3%                   |  |  |

<https://developer.android.com/about/versions/android-4.4.html>

Cancel OK

14. Di jendela New Project, pastikan Use legacy android.support libraries tidak dicentang. Klik tanda tanya jika Anda ingin mengetahui lebih lanjut tentang pilihan ini.
15. Klik Finish.

Android Studio akan membuka project tersebut dan semua filenya.



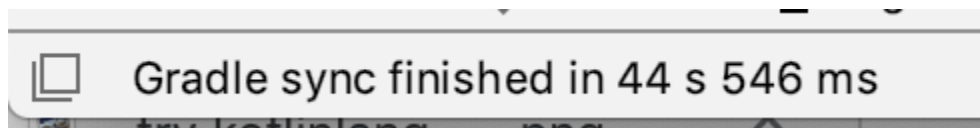
Saat pertama kali membuka Android Studio, Anda akan melihat tiga jendela:

- (1) Jendela Project menampilkan file dan folder project Anda.
- (2) Jendela Editing adalah untuk mengedit kode.
- (3) Jendela What's New menampilkan berita dan tips bermanfaat.

Di pojok kanan bawah Android Studio, status progres atau pesan menunjukkan apakah Android Studio masih bekerja untuk menyiapkan project Anda. Contoh:



16. Tunggu hingga Android Studio menyelesaikan penyiapan project Anda. Pesan di pojok kiri bawah, seperti yang ditampilkan di bawah ini, akan memberi tahu Anda saat project selesai.



### 3. Menjalankan aplikasi di perangkat virtual (emulator)

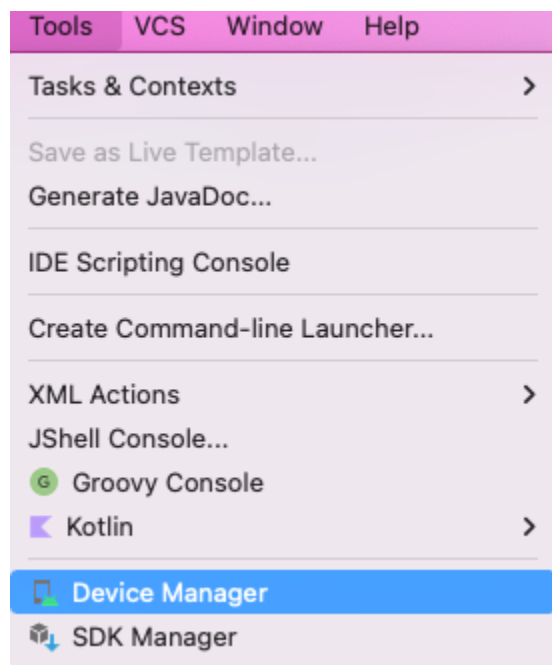
Dalam tugas ini, Anda akan menggunakan [Pengelola Perangkat](#) untuk membuat versi software (emulator) dari perangkat seluler dan menjalankannya di komputer. Perangkat virtual, atau emulator, menyimulasikan konfigurasi untuk jenis perangkat Android tertentu, seperti ponsel. Perangkat dapat berupa ponsel atau tablet yang menjalankan versi sistem Android pilihan Anda. Kemudian, Anda akan menggunakan perangkat virtual tersebut untuk menjalankan aplikasi yang dibuat dengan template Empty Activity.


Catatan: Android Emulator adalah aplikasi independen yang digunakan untuk menyiapkan perangkat virtual, dan memiliki persyaratan sistem sendiri. Perangkat virtual dapat menggunakan banyak ruang disk. Jika Anda mengalami masalah, lihat [Menjalankan aplikasi di Android Emulator](#).

### Membuat perangkat virtual Android (AVD)

Langkah pertama untuk menjalankan emulator di komputer adalah membuat konfigurasi untuk perangkat virtual.

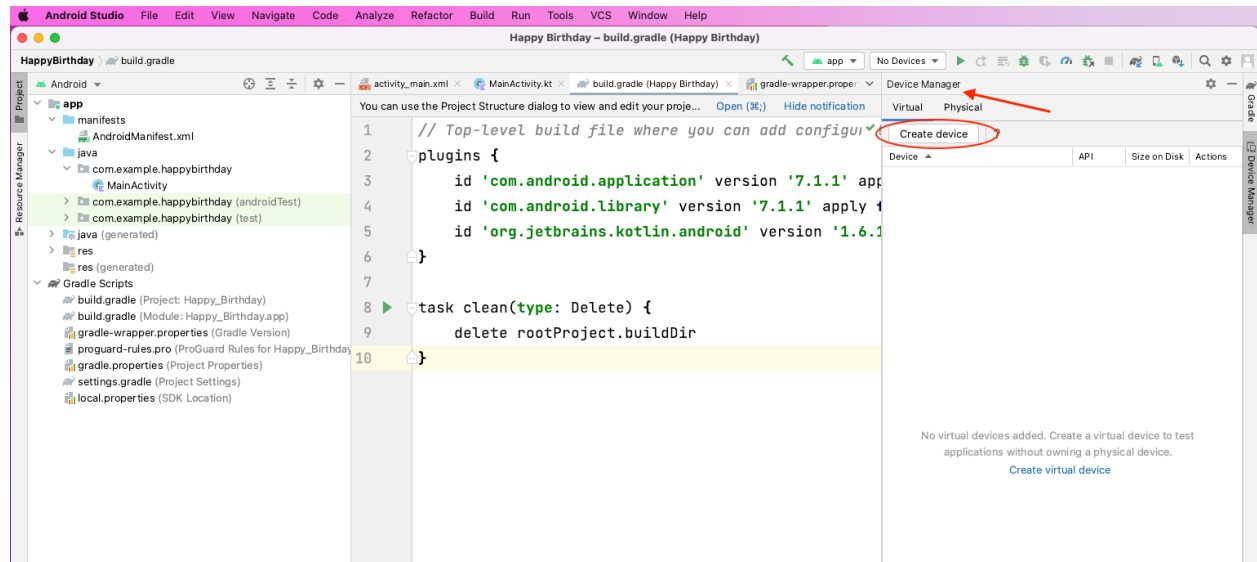
1. Dari panel menu Android Studio, pilih Tools > Device Manager.



Tips: Anda juga dapat membuka Pengelola Perangkat dengan mengklik ikon  di toolbar.

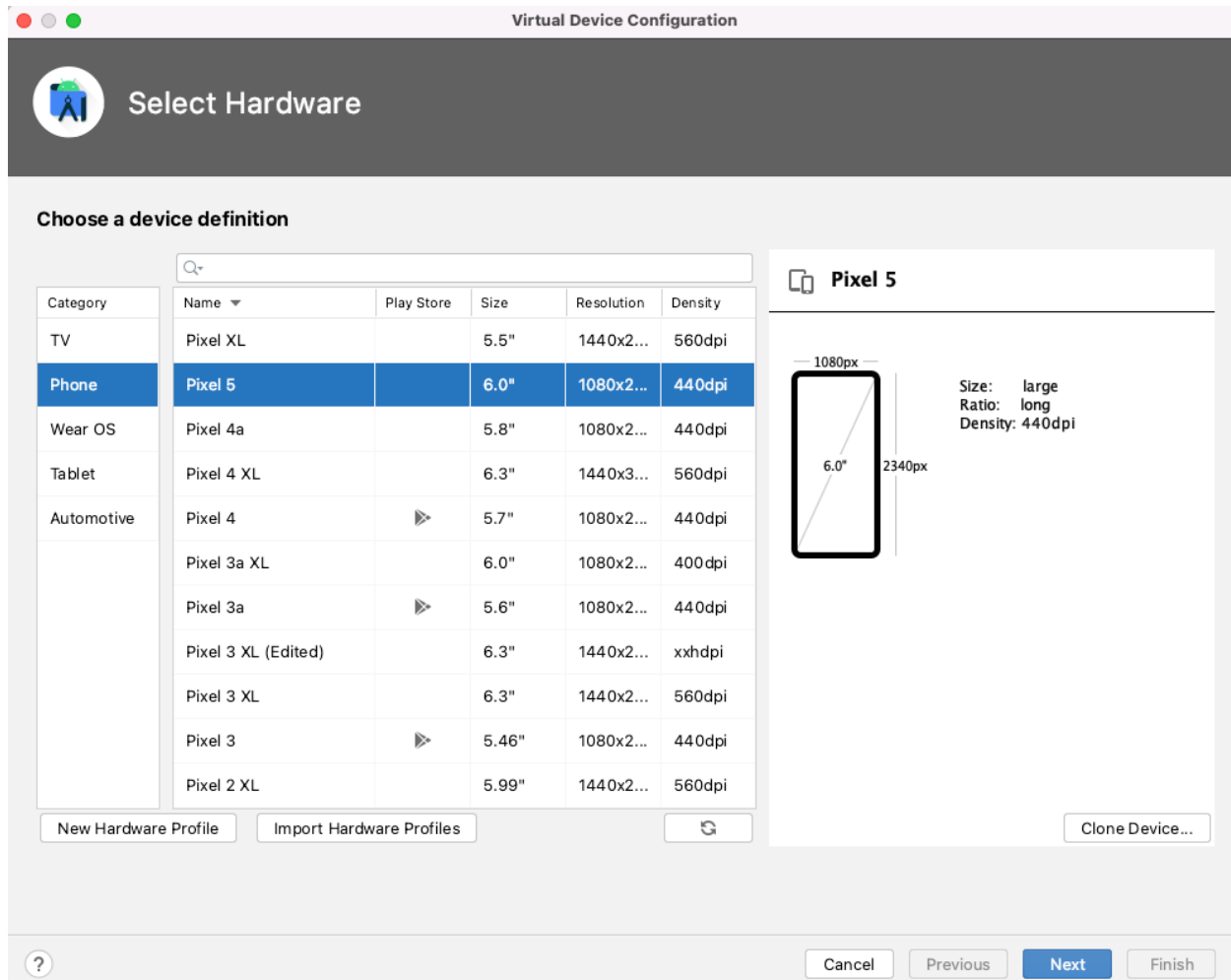
Pengelola Perangkat akan ditampilkan seperti yang ditunjukkan di bawah. (Jika Anda telah membuat perangkat sebelumnya, perangkat tersebut akan dicantumkan di sini).





2. Klik Create device.

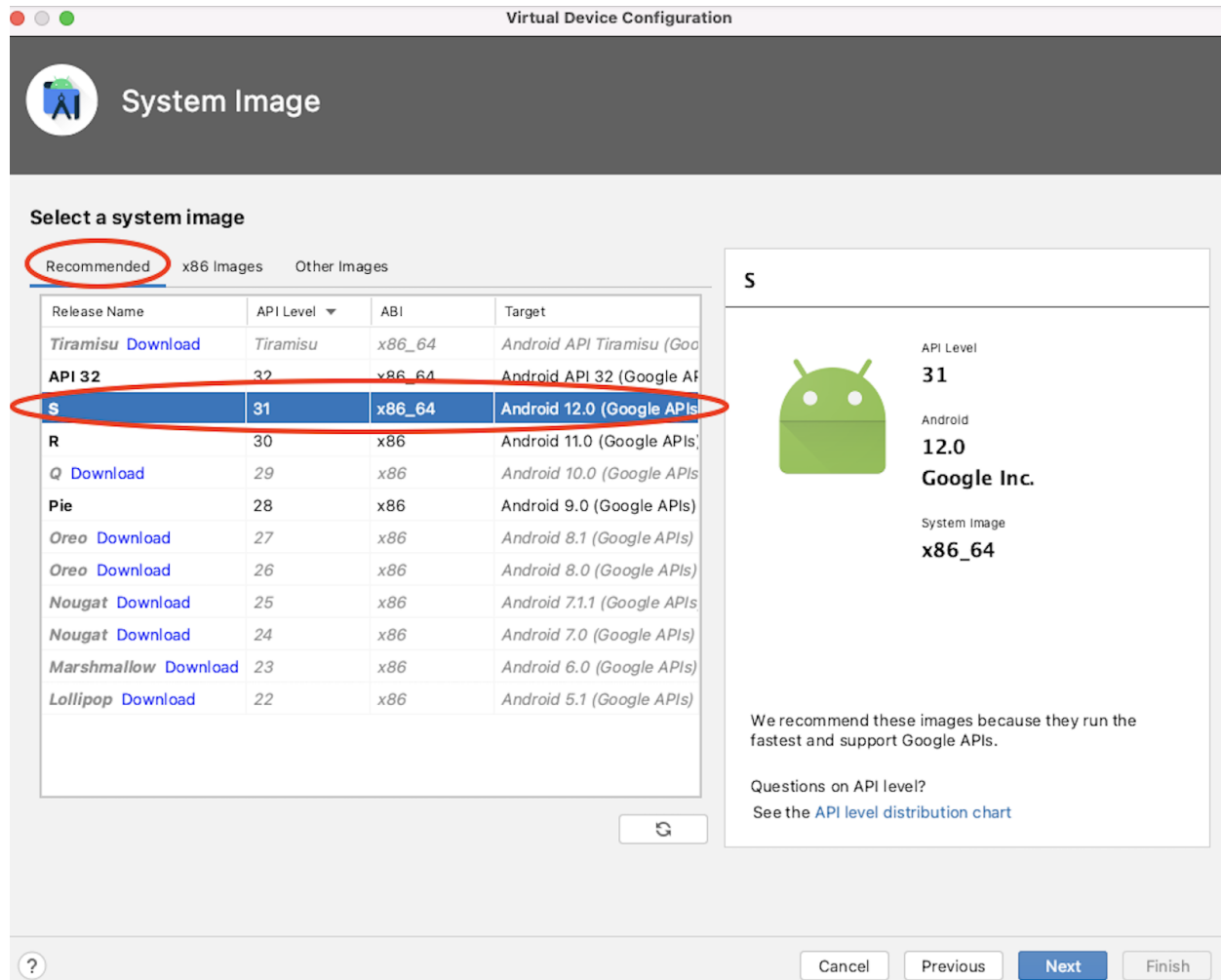
Jendela Select Hardware akan ditampilkan.



Jendela Select Hardware menampilkan daftar perangkat yang telah dikonfigurasi sebelumnya dan disusun berdasarkan Category yang dapat Anda pilih. Untuk setiap perangkat, tabel menyediakan kolom untuk ukuran layarnya (Size), resolusi layar dalam piksel (Resolution), dan kepadatan piksel (Density).

- Pilih Phone sebagai kategori.
- Pilih ponsel (misalnya, Pixel 5), lalu klik Next. Anda dapat memilih ponsel apa pun, tetapi untuk codelab ini, pilih perangkat yang lebih baru.

Jendela System Image akan ditampilkan. Dari sini, Anda dapat memilih versi sistem Android yang akan dijalankan di perangkat virtual. Proses ini memungkinkan Anda menguji aplikasi pada versi sistem Android yang berbeda.



5. Dari tab Recommended, pilih S sebagai versi sistem Android yang akan dijalankan di perangkat virtual. Pada saat artikel ini ditulis, versi ini merupakan versi Android terbaru, tetapi Anda dapat memilih versi stabil yang lebih baru. Lihat [di sini](#) untuk daftar versi stabil.

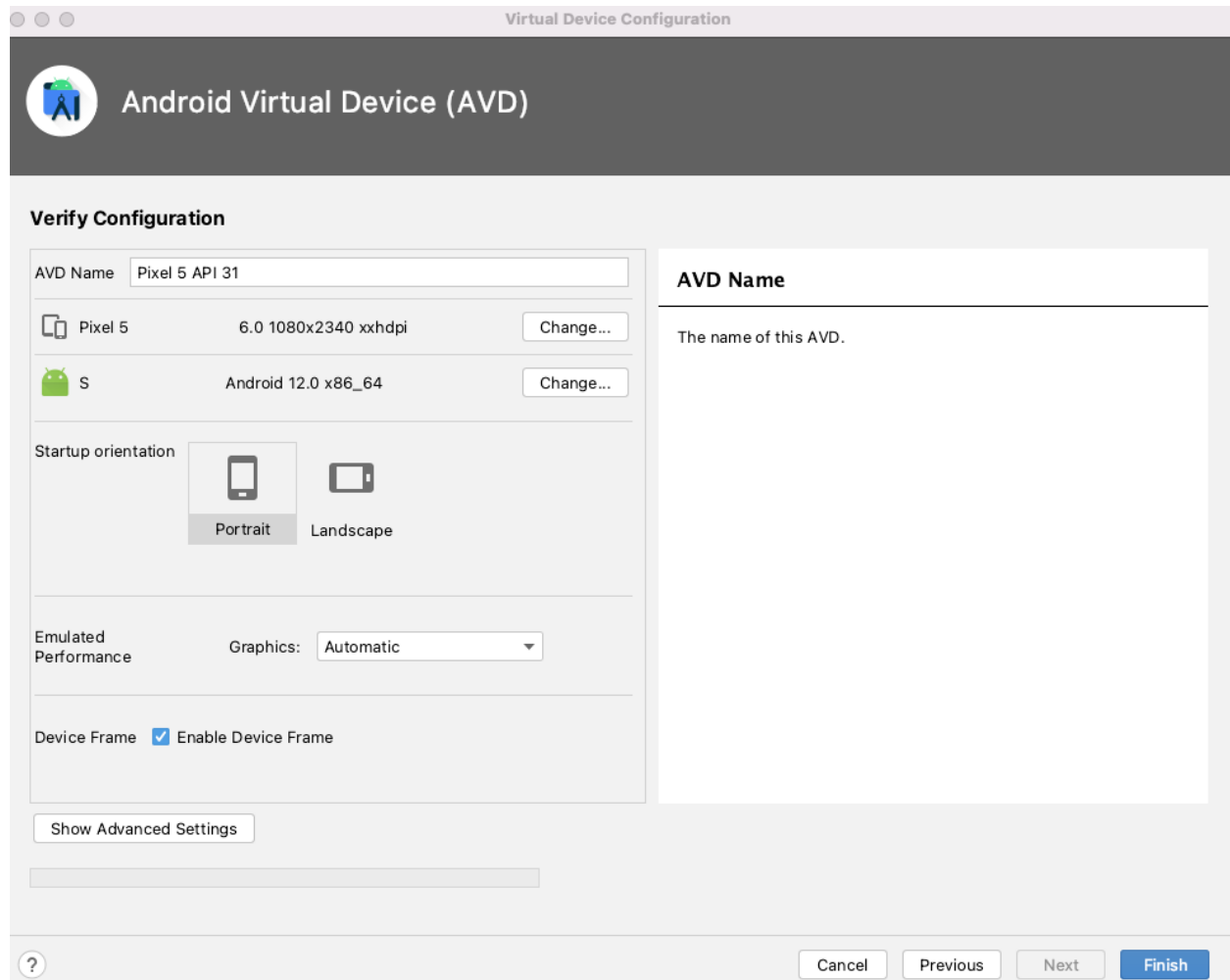
Catatan: Jika link Download terlihat di samping image sistem yang ingin Anda gunakan, image tersebut tidak akan diinstal di komputer. Anda harus menginstal image tersebut sebelum perangkat virtual dapat dikonfigurasi.

Untuk menginstal image sistem, klik link Download. Perlu diperhatikan bahwa proses download dapat memerlukan waktu lama, bergantung pada koneksi internet Anda. Setelah download selesai, klik Finish.

Penting: Image sistem Android ini menggunakan banyak ruang disk, jadi hanya sedikit yang merupakan bagian dari penginstalan asli. Ada lebih banyak versi sistem Android yang tersedia dibandingkan yang ditampilkan di tab Recommended. Untuk melihatnya, buka tab x86 Images dan Other Images.

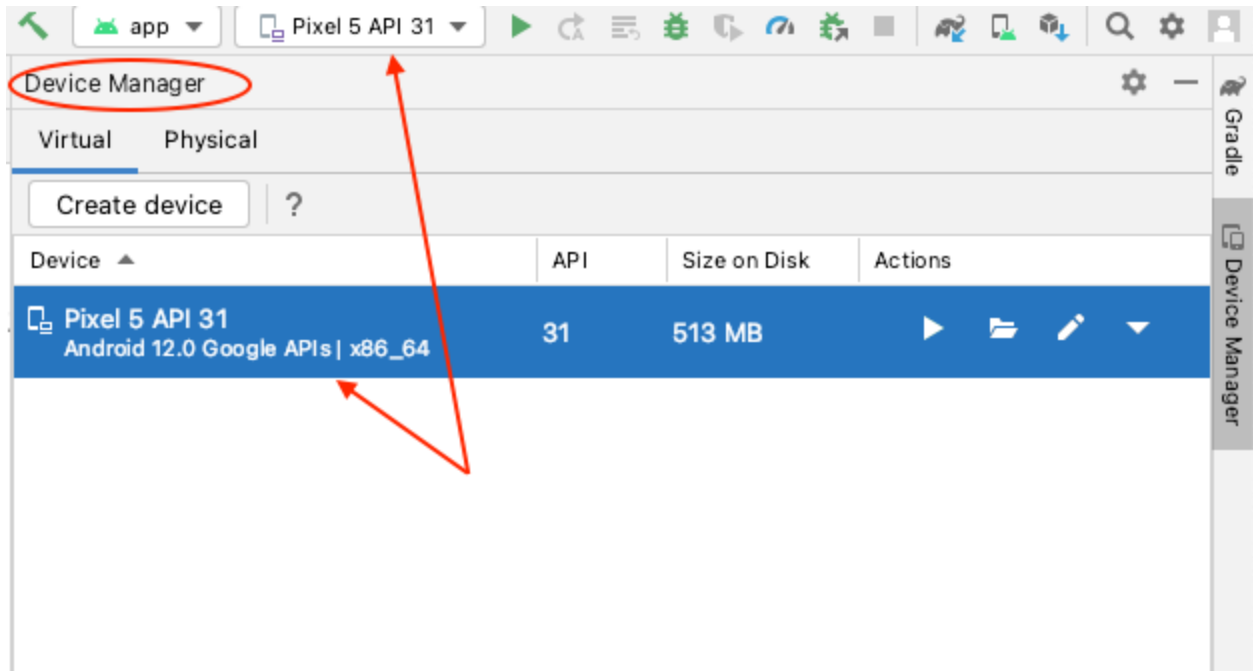
6. Klik Next.

Jendela Android Virtual Device (AVD) akan ditampilkan, tempat Anda dapat memilih detail konfigurasi tambahan untuk perangkat.



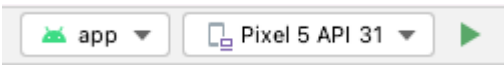
7. Di kolom AVD Name, masukkan nama untuk Perangkat Virtual Android Anda. Kolom lainnya tidak perlu diubah.
8. Klik Finish.

Perangkat virtual baru akan ditampilkan di jendela Device Manager dan siap digunakan.



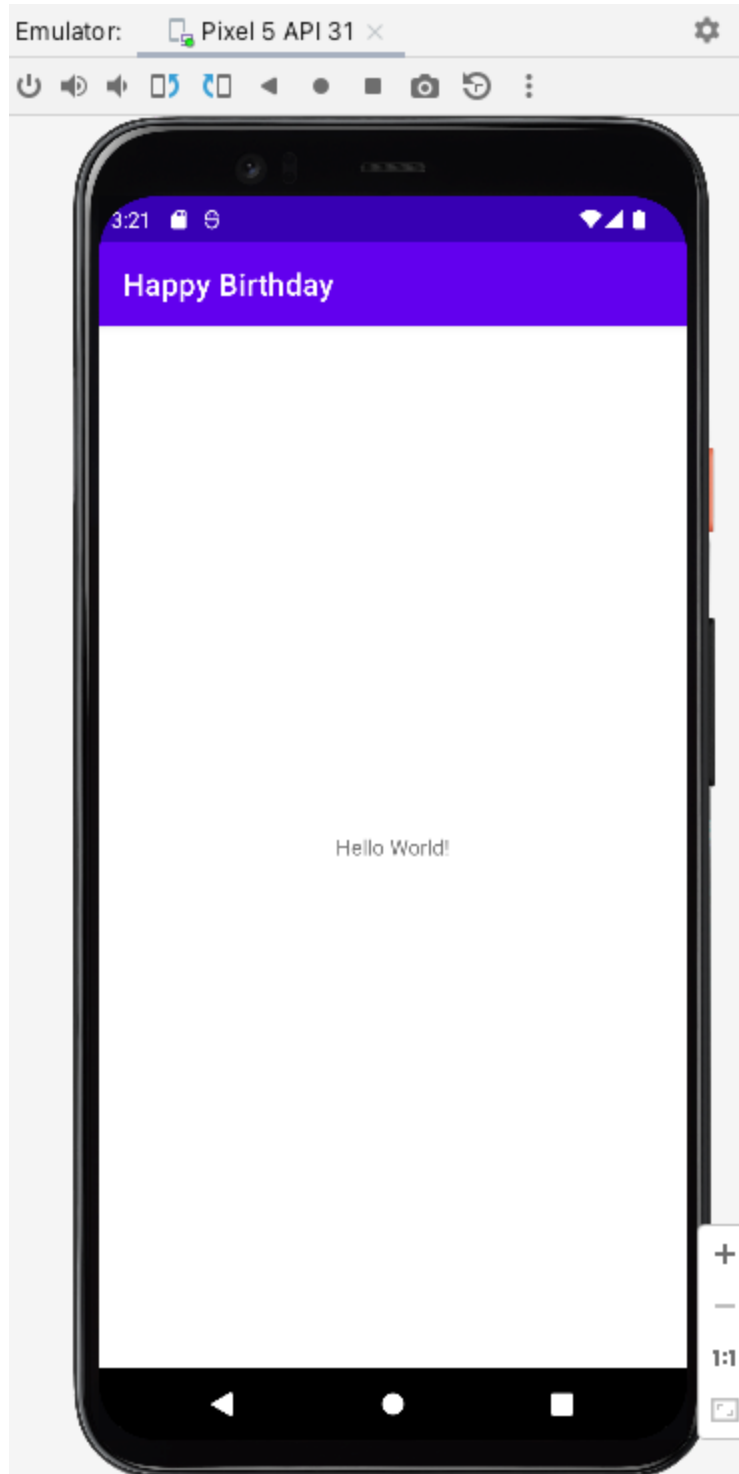
Menjalankan aplikasi di perangkat virtual

1. Alihkan Android Studio ke latar depan, jika Anda belum melakukannya.
2. Di Android Studio, dalam toolbar, temukan menu dropdown perangkat virtual (tampilannya akan mirip dengan contoh berikut), lalu pilih perangkat virtual yang Anda buat dari daftar

dropdown.  (Atau, klik Run > Select Device, lalu pilih perangkat virtual Anda dari perangkat yang tersedia di pop-up).

3. Di Android Studio, pilih Run > Run app, atau klik ikon Run di toolbar. Perangkat virtual akan dimulai dan di-boot, seperti perangkat fisik. Bergantung pada kecepatan komputer Anda, proses ini mungkin memerlukan sedikit waktu.

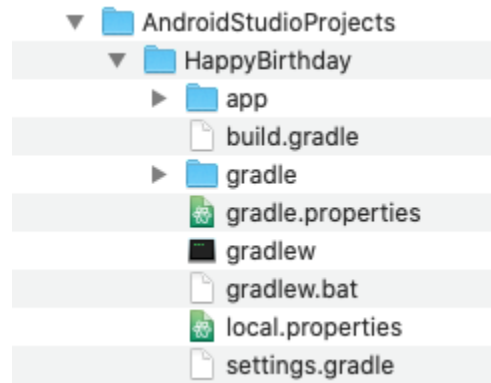
Jika sudah siap, aplikasi akan terbuka di perangkat virtual, seperti yang ditunjukkan di bawah.



Bagus. Perangkat virtual Anda sekarang sudah aktif dan berjalan. Perhatikan bahwa judulnya sekarang adalah "Happy Birthday", dan "Hello World!" ditampilkan di layar.

## 4. Menemukan file project

Saat Anda mengonfigurasi project, Android Studio akan membuat folder di komputer untuk semua project Android Anda yang disebut AndroidStudioProjects. Dalam folder AndroidStudioProjects, Android Studio juga membuat folder dengan nama yang sama seperti aplikasi Anda (dalam kasus ini, HappyBirthday).



Folder HappyBirthday adalah folder project Anda. Android Studio menyimpan file yang Anda buat dan file yang dibuat oleh Android Studio di folder project Anda.

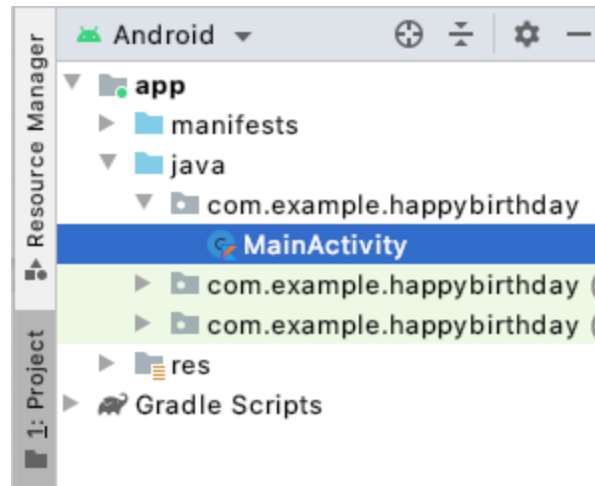
1. Di Android Studio, lihat jendela Project di sebelah kiri. Jendela Project menampilkan file dan folder project Anda.

File di jendela Project diatur untuk mempermudah navigasi di antara file project saat Anda menulis kode. Namun, jika Anda melihat file di file browser, seperti Finder atau Windows Explorer, hierarki file diatur dengan sangat berbeda.

Dalam tugas ini, Anda akan mempelajari dua tampilan hierarki folder project yang berbeda ini.

2. Di Android Studio, di jendela Project, pilih Android dari menu drop-down di pojok kiri atas.

Anda akan melihat listingan file yang mirip seperti berikut ini:

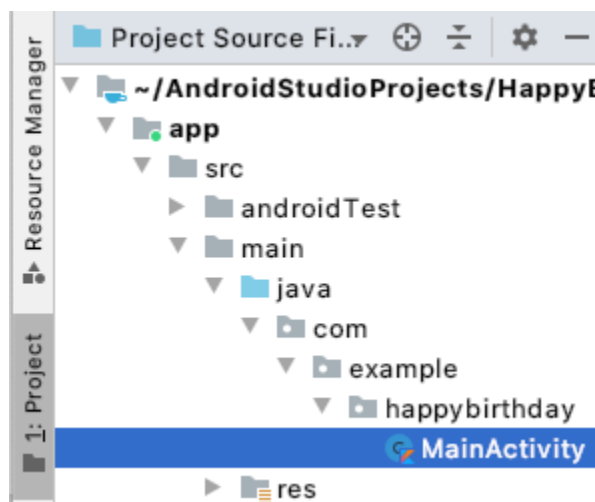


Tampilan dan pengaturan file ini berguna saat mengerjakan penulisan kode untuk project Anda.

Anda juga dapat melihat file sebagaimana ditampilkan di file browser, seperti Finder (untuk macOS) atau Explorer (untuk Windows).

3. Di jendela Project, pilih Project Source Files dari menu drop-down.

Perhatikan bahwa judul berubah menjadi folder tempat file project disimpan.




Sekarang Anda dapat menjelajahi file dengan cara yang sama seperti di penjelajah file apa pun.

4. Untuk beralih kembali ke tampilan sebelumnya, di jendela Project, pilih Android lagi.

Bagus. Sekarang Anda dapat membuat dan menjalankan aplikasi dari template, dan Anda dapat menemukan file project.



## 5. Ringkasan

- Untuk membuat project baru, mulai Android Studio, klik + Start a new Android Studio project, beri nama project Anda, pilih template, lalu isi detailnya.
- Untuk membuat perangkat virtual Android (emulator) agar dapat menjalankan aplikasi Anda, pilih Tools > Device Manager, lalu gunakan [Device Manager](#) untuk memilih perangkat hardware dan image sistem.
- Untuk menjalankan aplikasi di perangkat virtual, pastikan Anda telah membuat perangkat, pilih perangkat dari menu drop-down toolbar, lalu jalankan aplikasi dengan mengklik ikon Run  pada toolbar.
- Untuk mencari file project Anda, di jendela Project, pilih Project Source Files dari menu dropdown.

# Menjalankan aplikasi di perangkat seluler

## 1. Sebelum memulai

### Prasyarat

- Pemahaman dasar tentang cara menggunakan Android Studio.
- Kemampuan untuk membuka dan menyesuaikan setelan di perangkat Android.

### Yang akan Anda pelajari

- Cara membuat perangkat Android dapat menjalankan aplikasi dari Android Studio.
- Cara menghubungkan dan menjalankan aplikasi di Android Studio pada perangkat Android fisik.

### Yang Anda perlukan

- Android Studio yang sudah didownload dan diinstal di komputer.
- Project aplikasi yang sudah disiapkan di Android Studio.
- Perangkat Android, seperti ponsel atau tablet.
- Kabel USB untuk menghubungkan perangkat Android ke komputer melalui port USB.

Catatan: Buka <https://en.wikipedia.org/wiki/USB> jika Anda perlu bantuan dalam menentukan jenis port USB di komputer dan perangkat Android, serta kabel yang diperlukan.

## 2. Mengaktifkan Proses debug USB

Agar Android Studio dapat berkomunikasi dengan perangkat Android, Anda harus mengaktifkan Proses debug USB di setelan Opsi developer di perangkat.

Untuk menampilkan opsi developer dan mengaktifkan Proses debug USB:

1. Buka Setelan di perangkat Android, lalu telusuri Tentang ponsel.
2. Ketuk Tentang ponsel, lalu ketuk Nomor versi sebanyak tujuh kali. Masukkan sandi atau pin perangkat, jika diminta.
3. Kembali ke Setelan, lalu ketuk Sistem.

Sekarang Opsi developer akan muncul dalam daftar. Anda mungkin perlu membuka opsi Lanjutan untuk menemukannya.

4. Ketuk Opsi developer, lalu aktifkan Proses debug USB.

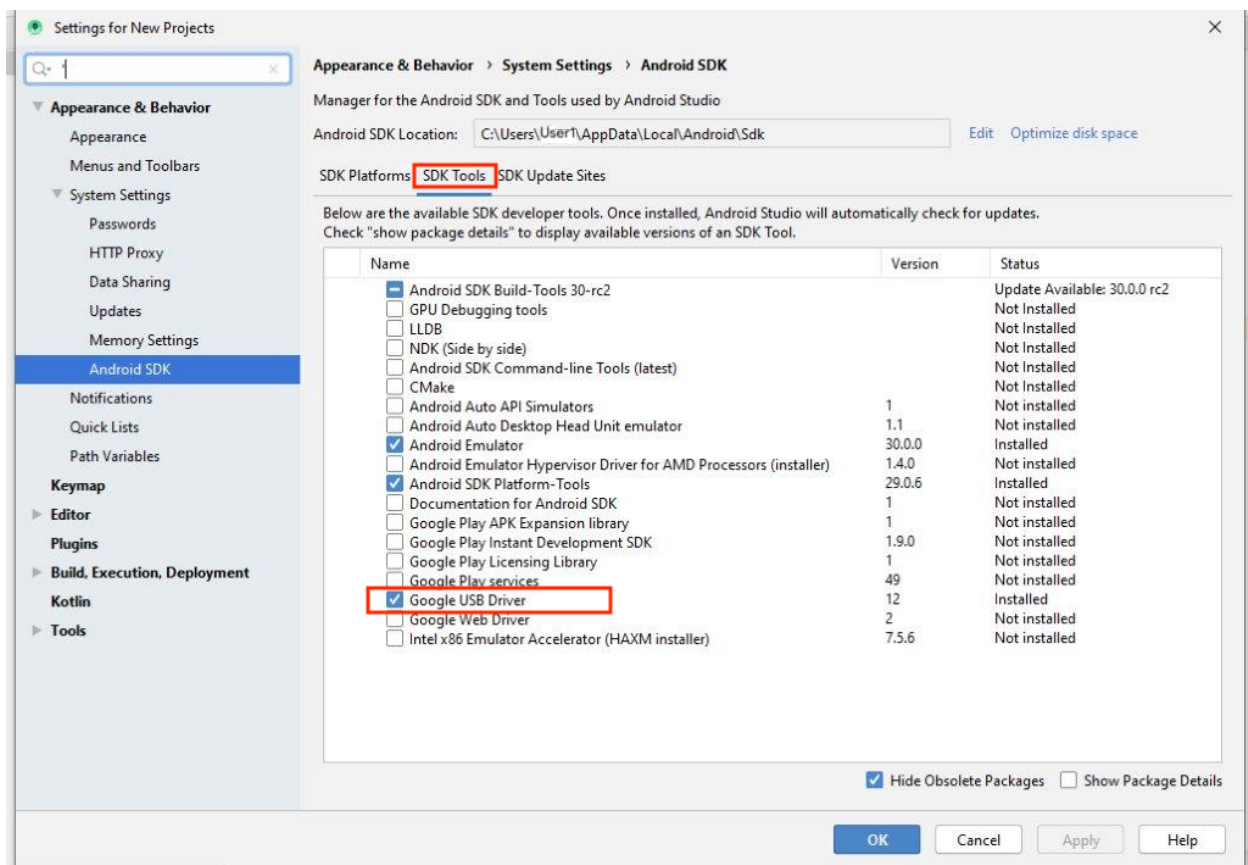
### 3. Menginstal Driver USB Google (khusus Windows)

Jika Android Studio telah diinstal di komputer Windows, Anda harus menginstal driver perangkat USB agar dapat menjalankan aplikasi di perangkat fisik.

Catatan: Untuk Ubuntu Linux, ikuti petunjuk dalam dokumentasi [Menjalankan Aplikasi di Perangkat Hardware](#).

1. Di Android Studio, klik Tools > SDK Manager.

Jendela Settings for New Projects akan ditampilkan.



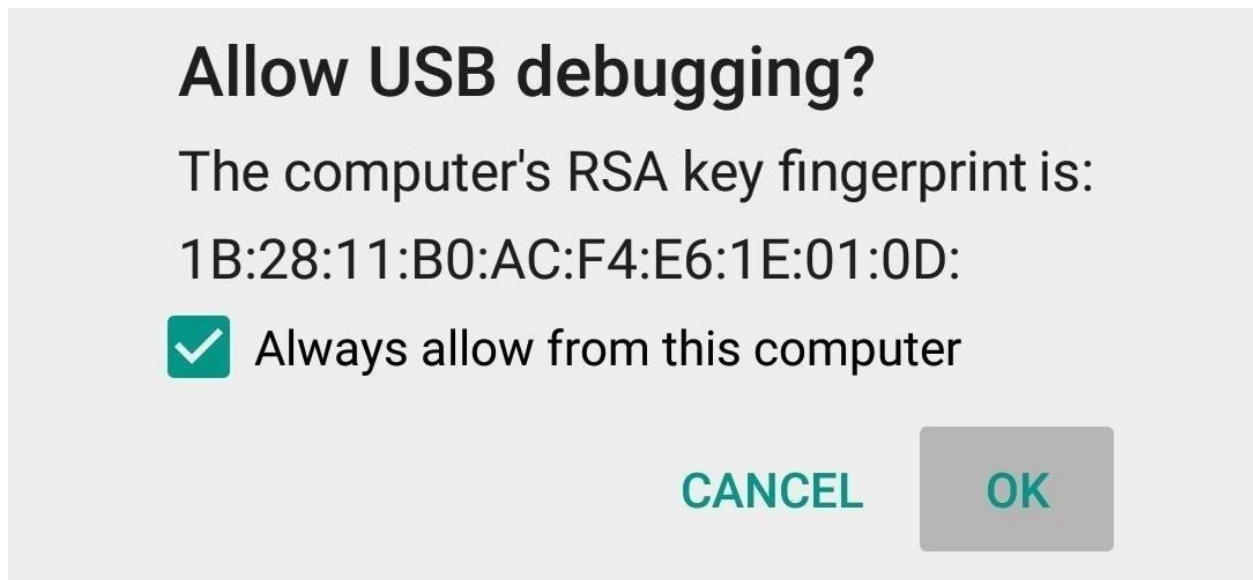
2. Klik tab SDK Tools.
3. Pilih Google USB Driver, lalu klik OK.


Setelah selesai, file driver akan didownload ke direktori *\*android\_sdk\extras\google\usb\_driver\**. Sekarang Anda dapat menghubungkan dan menjalankan aplikasi dari Android Studio.

#### 4. Menjalankan aplikasi di perangkat Android (untuk semua sistem operasi)

Sekarang Anda dapat menghubungkan perangkat dan menjalankan aplikasi dari Android Studio.

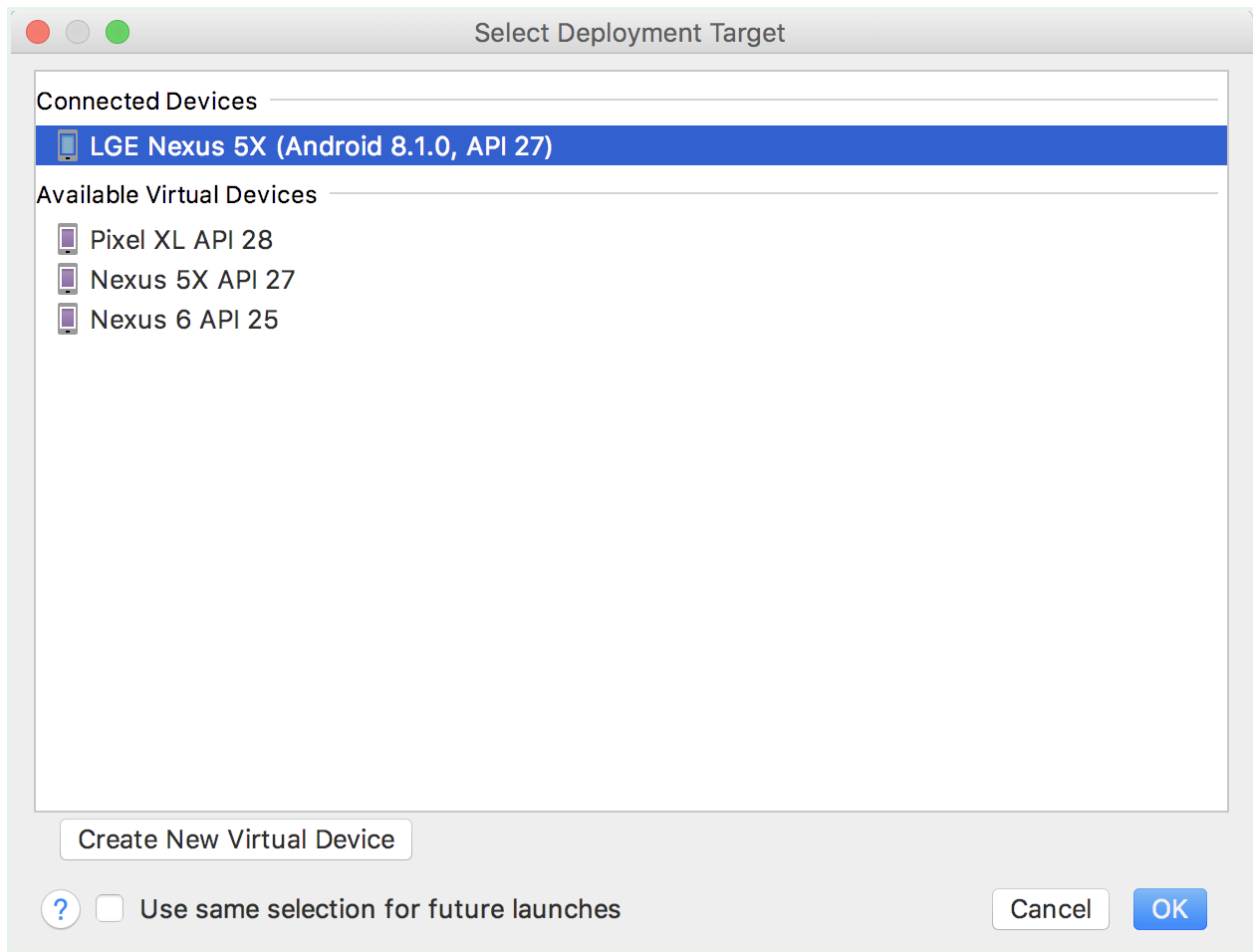
1. Hubungkan perangkat Android ke mesin pengembangan dengan kabel USB. Dialog yang meminta untuk mengizinkan Proses debug USB akan muncul di perangkat.



2. Pilih opsi Selalu izinkan untuk mengingat komputer ini. Ketuk Oke. 3 Di komputer, di toolbar Android Studio, klik tombol Run .

Dialog Select Deployment Target akan terbuka dengan daftar emulator yang tersedia dan perangkat yang terhubung. Anda akan melihat perangkat fisik Anda beserta semua emulator.

Catatan: Untuk Android Studio 3.6 dan versi yang lebih baru, perangkat fisik akan otomatis dipilih saat terhubung dengan proses debug yang aktif.



4. Pilih perangkat, lalu klik OK.

Android Studio akan menginstal aplikasi di perangkat, lalu menjalankannya.

Catatan: Jika perangkat menjalankan platform Android yang tidak diinstal di Android Studio, mungkin akan muncul pesan yang menanyakan apakah Anda ingin menginstal platform yang diperlukan. Klik Install and Continue, lalu klik Finish saat prosesnya selesai.

## 5. Pemecahan masalah

- Jika komputer Anda menjalankan Linux atau Windows, langkah tambahan mungkin perlu dilakukan agar dapat menjalankan aplikasi di perangkat Android fisik. Lihat dokumentasi [Menjalankan Aplikasi di Perangkat Hardware](#).
- Di Windows, Anda mungkin perlu menginstal driver USB yang sesuai dengan perangkat. Lihat [Driver USB OEM](#).

Jika Android Studio tidak mengenali perangkat Anda, coba langkah berikut:

1. Lepas kabel USB, lalu sambungkan kembali.
2. Mulai ulang Android Studio.

Jika komputer masih tidak menemukan perangkat atau menyatakannya sebagai "unauthorized", ikuti langkah-langkah berikut:

1. Lepas kabel USB.
2. Di perangkat, buka Opsi developer di Setelan.
3. Ketuk Cabut otorisasi debug USB.
4. Hubungkan kembali perangkat ke komputer.
5. Jika diminta, berikan otorisasi.