

LAPORAN TUGAS
BAHASA PEMROGRAMAN 3
(Dosen : Dede Husen, M.Kom.)

Modul 1



Nama : Muhammad Rizal Nurfirdaus
NIM : 20230810088
Kelas : TINFC-2023-04

TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS KUNINGAN

PRE-TEST

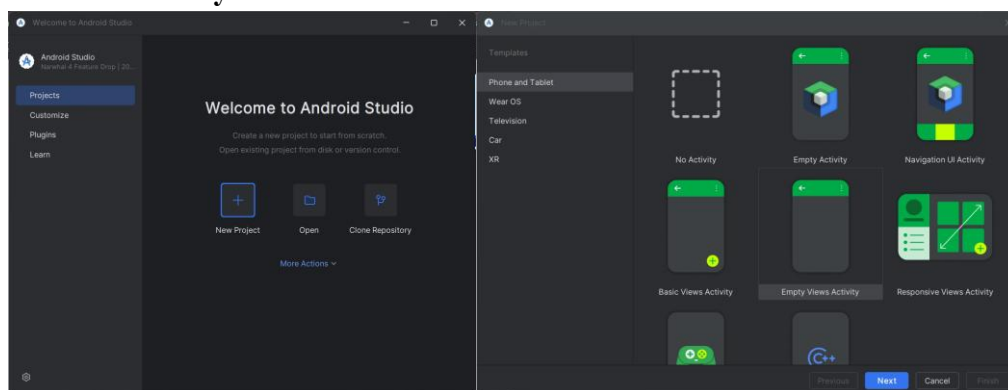
1. Untuk mempersiapkan pengembangan Aplikasi Android software apa saja yang harus diinstall?
 - **Android Studio:** IDE resmi dari Google untuk membuat, mengedit, menjalankan, dan menguji aplikasi Android.
 - **Java Development Kit (JDK):** Diperlukan untuk menjalankan dan meng-compile kode program berbasis Java di Android Studio.
 - **Android SDK (Software Development Kit):** Berisi tools, library, dan API yang dibutuhkan untuk membangun aplikasi Android serta berkomunikasi dengan sistem Android.
2. Apa fungsi dari Android Virtual Device?

Android Virtual Device berfungsi sebagai **emulator perangkat Android** yang memungkinkan pengembang menjalankan dan menguji aplikasi pada berbagai versi Android dan ukuran layar **tanpa memerlukan perangkat fisik**.
3. Apa kegunaan dari Android SDK?

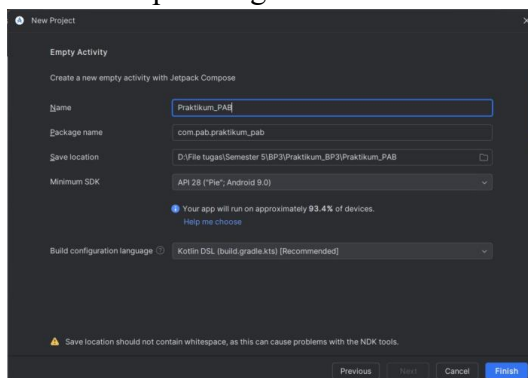
Android SDK digunakan untuk **mengembangkan dan menguji aplikasi Android**. SDK ini berisi **tools, library, dan API** yang memudahkan proses **kompilasi, debugging, serta pengujian aplikasi** agar sesuai dengan sistem operasi Android.

PRAKTIKUM

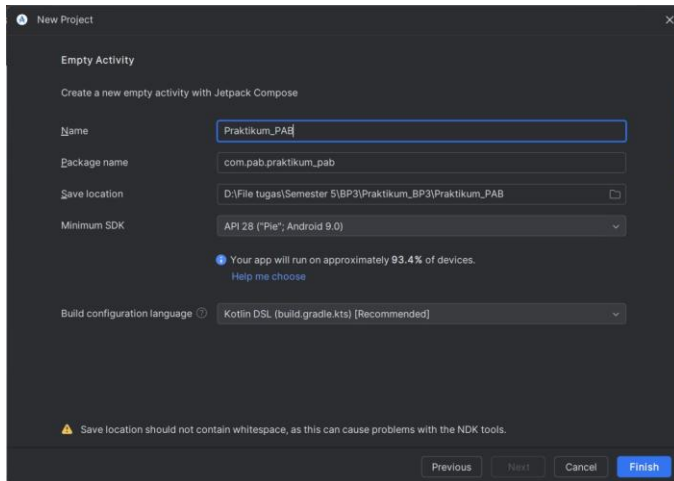
1. Buka Aplikasi IDE Android studio, kemudian klik new project Selanjutnya tentukan jenis perangkat aplikasi yang akan dibuat, lalu pilih template yang tersedia, dalam praktikum ini kita pilih template untuk **phone and tablet** lalu pilih template **Empty View Activity** kemudian klik next



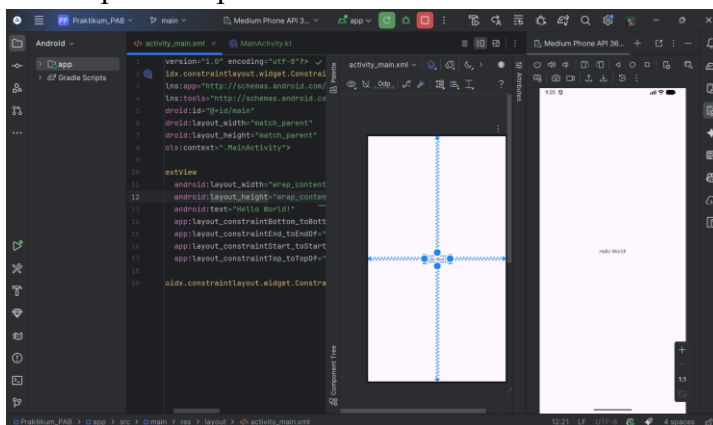
2. Setelah itu pada bagian ini silakan isi nama proyek, misal nama proyek kita isi dengan:



3. Kemudian klik finish, tunggu hingga jendela IDE terbuka. Pada tahap ini kita akan memerlukan **koneksi internet** untuk mensinkronkan **gradle** dan mendownload file penting kita untuk pertama kalinya.



4. Setelah proses setup project berhasil, maka Anda sudah bisa memodifikasi proyek yang ada baik dari file logic, layout xml, asset gambar dan lain sebagainya seperti pada tampilan dibawah ini:

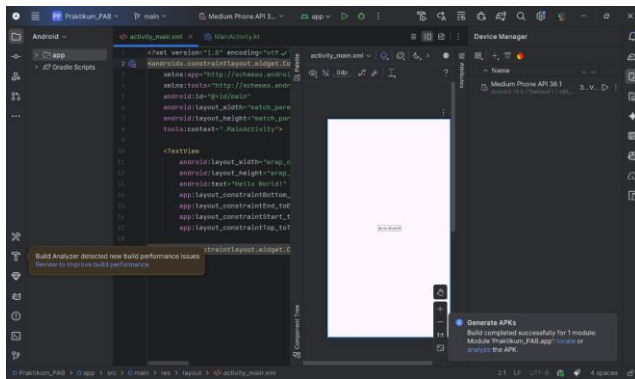


Praktikum - Build File APK (debug-mode)

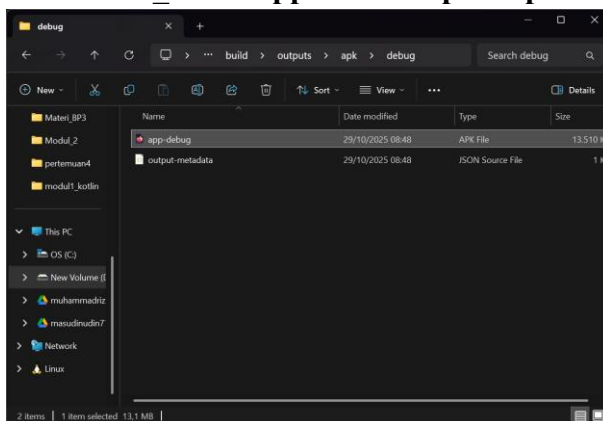
Setelah praktikum 5 dapat dijalankan di virtual device, kita juga dapat membuat file APK yang dapat diinstall di ponsel yang lain. Terdapat 2 jenis APK yang dapat di generate, yang pertama adalah file APK yang digunakan untuk keperluan debugging atau testing di device kita, yang kedua adalah file APK Signed (*release*) yang akan digunakan untuk diupload ke playstore. Berikut langkah-langkahnya

Generate APK (debug mode)

1. Buka kembali proyek praktikum 5
2. Kemudian klik menu Build -> Build App Bundle(s)/Apk(s)
3. Setelah melakukan langkah-langkah diatas, di bagian pojok kanan bawah ada notifikasi bahwa file APK telah di generate, klik bagian tulisan biru **locate** untuk membuka dimana lokasi file apk berada.



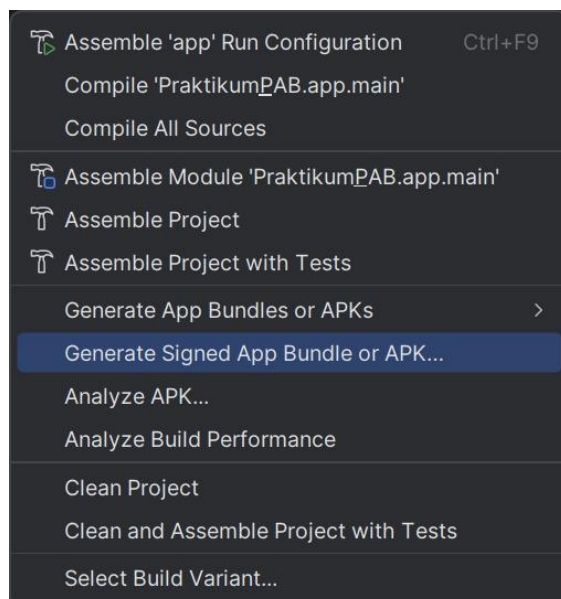
4. File dengan nama app-debug.apk akan ada di lokasi path sesuai dengan proyek ini dibuat, pada kasus ini kita bisa melihat file tersebut ada di D:\PAB_MID3\app\build\outputs\apk\debug\ seperti dibawah ini.



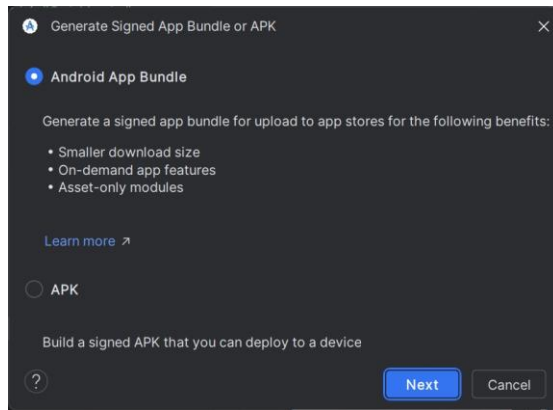
5. File APK siap di install di perangkat Android.

Praktikum – Build Signed Apk (Release Mode)

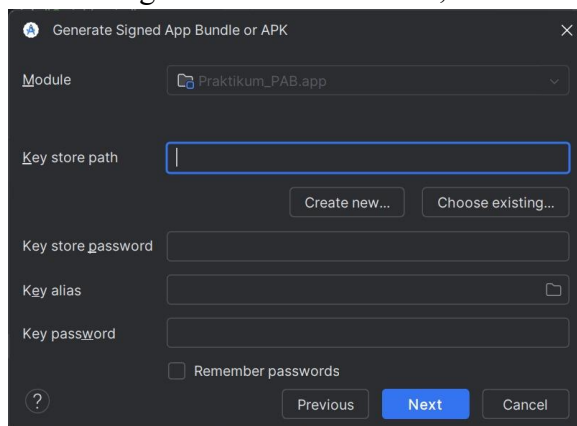
1. Buka kembali file proyek praktikum ke-5
2. Klik menu build -> Build-> Generate Signed App Bundle/ APK



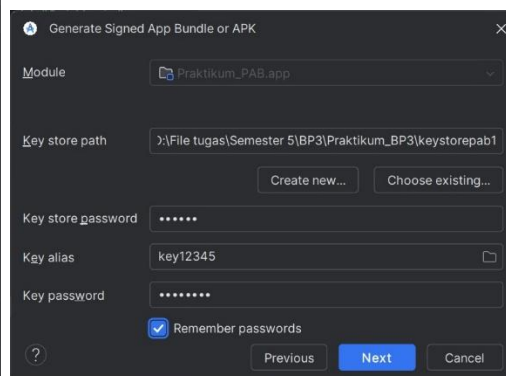
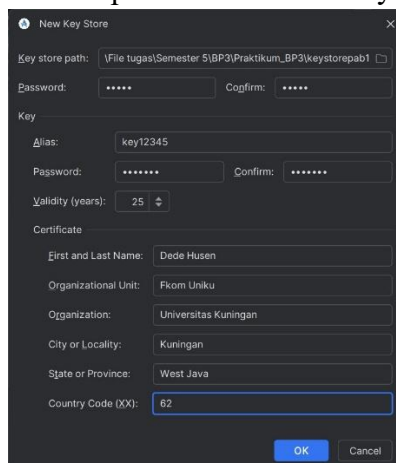
3. Kemudian akan ada dua pilihan untuk men generate APK, pilih yang pilihan **Android App Bundle** kemudian klik Next untuk melanjutkan



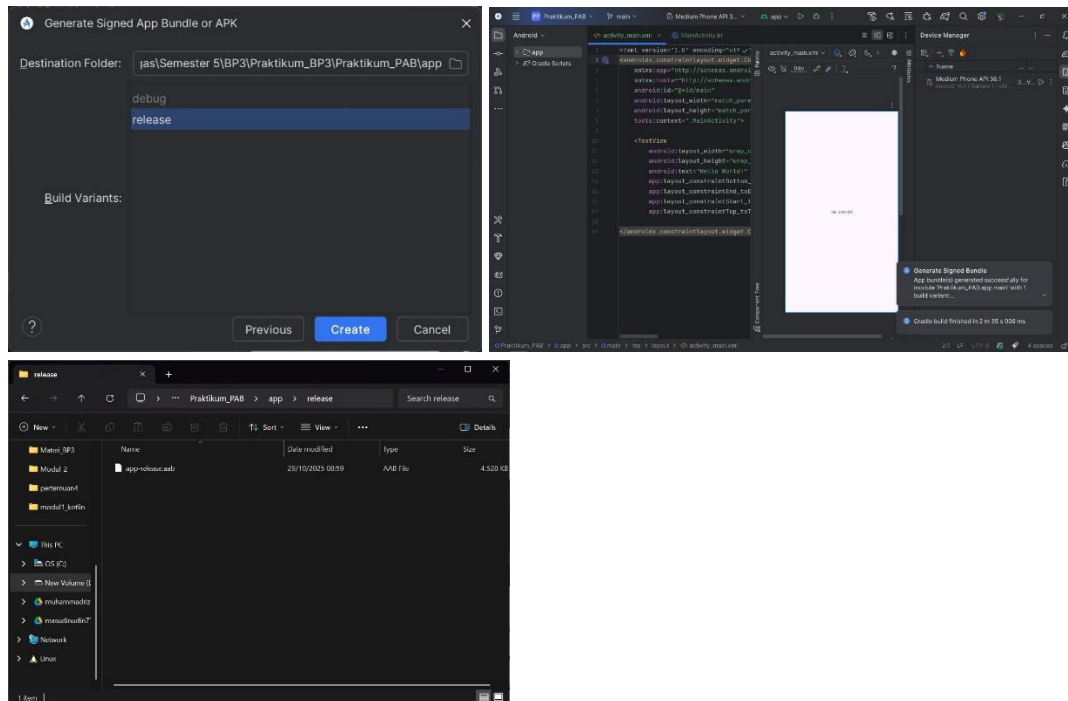
4. Kemudian akan muncul permintaan keystore, silakan klik **create new** dan isikan sesuai dengan contoh dibawah ini, kemudian klik ok



5. Data keystore baru telah terisi selanjutnya klik next, dan pada dialog selanjutnya kita Silakan pilih **build variants** yang *release*, lalu klik *create*



6. Jika berhasil maka akan ada notif dibagian bawah kanan, klik link *locate* maka anda akan diarahkan ke halaman **apk-release.aab** di generate. File ini yang dapat anda upload ketika anda akan memasarkan aplikasi ke google playstore



F. POST-TEST

1. Sebutkan apa saja yang harus dipersiapkan/diinstall ketika akan membuat sebuah aplikasi Android?

- **Android Studio** – sebagai IDE utama untuk menulis dan mengelola proyek aplikasi Android.
- **Java Development Kit (JDK)** – untuk menjalankan dan meng-compile kode program berbasis Java.
- **Android SDK (Software Development Kit)** – berisi tools, library, dan API untuk membangun aplikasi Android.
- **Android Virtual Device (AVD)** – emulator untuk menguji aplikasi tanpa perangkat fisik.

2. Jelaskan fungsi dan kegunaan dari kebutuhan dari software yang diinstal tersebut?

- **Android Studio:** tempat membuat, mengedit, mendesain, dan menjalankan aplikasi Android.
- **JDK:** menyediakan compiler dan runtime agar kode Java bisa dijalankan di Android Studio.
- **Android SDK:** menyediakan komponen penting seperti API dan library agar aplikasi dapat berjalan di sistem Android.
- **AVD:** digunakan untuk menguji aplikasi di berbagai tipe perangkat dan versi Android secara virtual.

3. Dari hasil praktikum diatas, jelaskan perbedaan file yang dihasilkan dengan build debug dan build release untuk ke playstore?

➤ Build Debug:

- Digunakan untuk **pengujian dan pengembangan**.

- Mengandung **debug information** dan **belum dioptimalkan**.
- Tidak bisa diunggah ke Play Store.

➤ **Build Release:**

- Digunakan untuk **rilis resmi ke Play Store**.
- Sudah **dioptimalkan dan ditandatangani (signed)** dengan **release key**.
- Tidak mengandung informasi debug dan performanya lebih baik.

G. LATIHAN/TUGAS

Buatlah proyek baru dengan nama berbeda dari praktikum sebelumnya, kemudian silahkan tambahkan komponen teks pada file xml layoutnya, hasilnya aplikasi dapat menampilkan teks nama, usia, prodi, fakultas dan universitas seperti dibawah ini .

Tips: Gunakan layout dasar linearLayout dengan orientasi vertikal



Source Code :

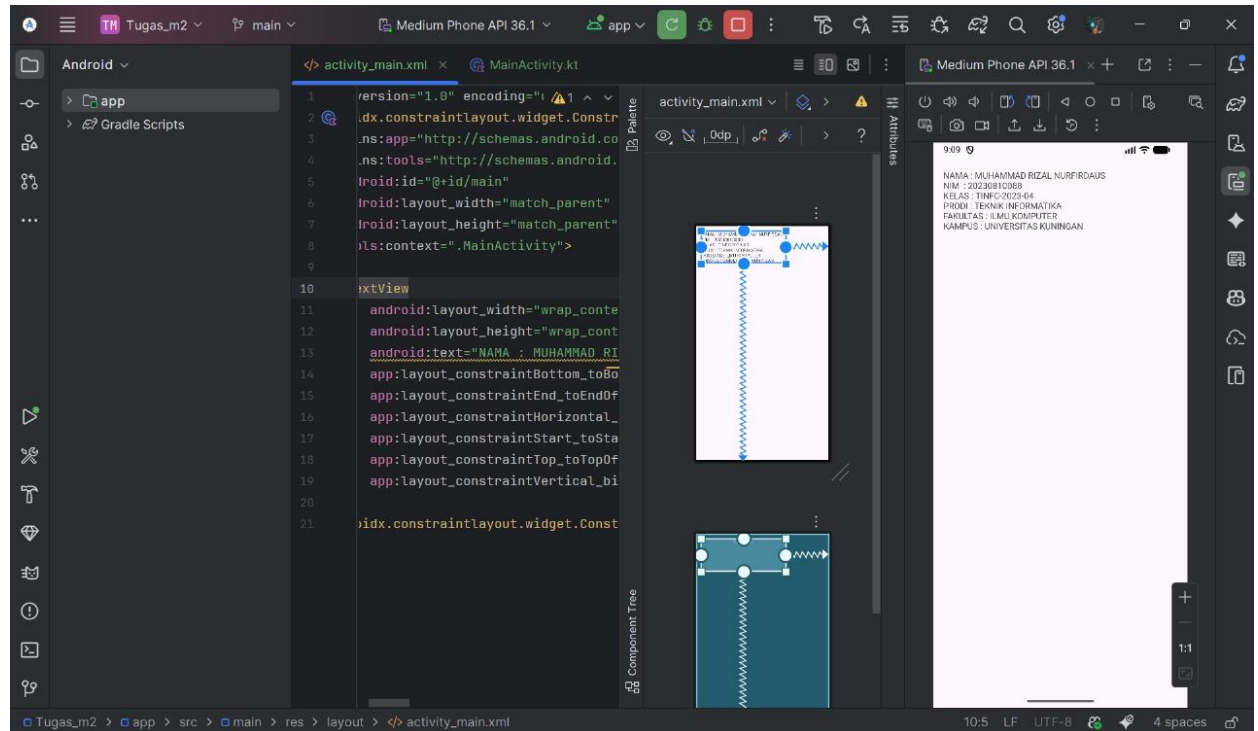
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="NAMA : MUHAMMAD RIZAL NURFIRDAUS\nNIM :
20230810088\nKELAS : TINFC-2023-04\nPRODI : TEKNIK
INFORMATIKA\nFAKULTAS : ILMU KOMPUTER\nKAMPUS :
UNIVERSITAS KUNINGAN"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.107"
        app:layout_constraintStart_toStartOf="parent"
    />
</TextView>
</androidx.constraintlayout.widget.ConstraintLayout>
```

```
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.025" />
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

Hasil Run :



Analisis : Kode XML di atas merupakan **tampilan layout utama (activity_main.xml)** dalam proyek Android Studio yang menggunakan **ConstraintLayout** sebagai layout utama. Layout ini berfungsi untuk mengatur posisi komponen secara fleksibel dengan memberikan batasan (constraint) pada setiap sisi elemen terhadap parent atau elemen lainnya.

Pada kode tersebut, hanya terdapat satu komponen yaitu **TextView**, yang digunakan untuk menampilkan informasi identitas seperti **nama, NIM, kelas, prodi, fakultas, dan kampus**. Atribut `android:text` berisi teks yang ditulis dengan pemisah baris `\n`, agar setiap informasi tampil pada baris baru.

TextView ini diatur agar **posisinya berada di bagian atas dan agak ke kiri layar**, dengan menggunakan atribut `app:layout_constraintTop_toTopOf="parent"` dan `app:layout_constraintStart_toStartOf="parent"`. Nilai `layout_constraintHorizontal_bias="0.107"` dan `layout_constraintVertical_bias="0.025"` digunakan untuk mengatur jarak relatif terhadap tengah layar, sehingga teks tidak tepat di tengah tetapi sedikit ke kiri atas.

Secara keseluruhan, layout ini sederhana dan berfungsi menampilkan data teks statis dengan tata letak yang rapi dan mudah dipahami. Penggunaan **ConstraintLayout** juga memudahkan pengaturan posisi komponen jika nantinya ingin menambahkan elemen lain ke dalam tampilan.

