

1. Activity merupakan sebuah komponen di Android yang berfungsi untuk menampilkan user *interface* ke layar handset Android pengguna. Ini seperti pada saat Anda melihat

daftar percakapan pada aplikasi *chat* atau daftar *email* pada aplikasi Gmail di ponsel Android Anda.

2. Umumnya dalam sebuah aplikasi terdapat lebih dari satu activity yang saling terhubung dengan tugas yang berbeda-beda.
3. Activity merupakan salah satu komponen penting Android yang memiliki daur hidup (life cycle) dalam sebuah *stack* pada *virtual sandbox* yang disiapkan oleh *Dalvik Virtual Machine* (DVM) atau *Android Runtime* (ART) yang bersifat *last in first out*.
4. Pada implementasinya, activity selalu memiliki satu layout *user interface* dalam bentuk berkas xml.
5. Suatu aplikasi Android bisa memiliki lebih dari satu activity dan harus terdaftar di berkas **AndroidManifest.xml** sebagai sub aplikasi.
6. Sebuah class Java dinyatakan sebuah activity jika mewarisi (*extends*) superclass **Activity** atau turunannya seperti **AppCompatActivity** dan **FragmentActivity**.

Jenis-jenis Activity pada Android Studio :

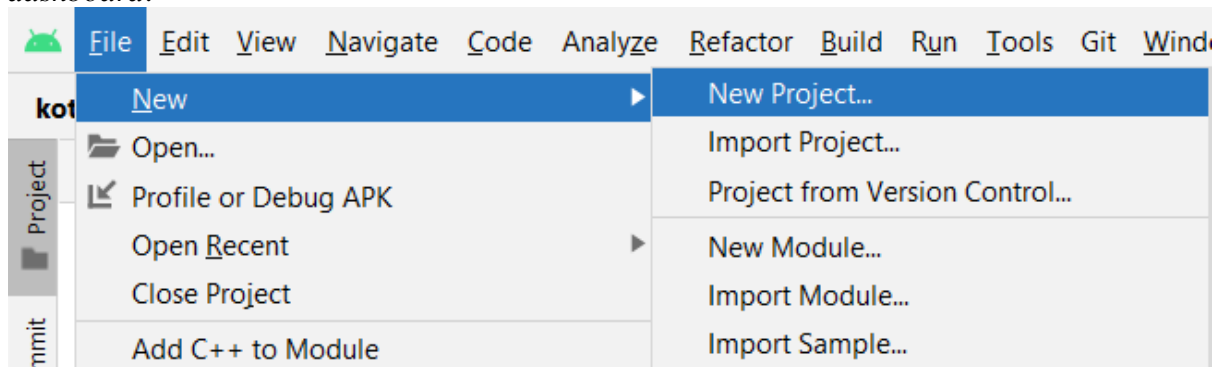
Jenis-jenis template Activity	Fungsinya
Add No Activity	Tidak ada Activity yang ditambahkan
Basic Activity	Activity dengan <i>template</i> komponen material design seperti <i>FloatingActionButton</i>
Bottom Navigation Activity	Activity dengan tampilan side bar menu di bagian bawah
Empty Activity	Activity dalam bentuk yang paling dasar
Fragment + ViewModel	Activity dengan menerapkan <i>architecture component</i>
Fullscreen Activity	Activity <i>fullscreen</i> tanpa status bar
Google AdMob Ads Activity	Activity dengan konfigurasi <i>default</i> iklan Admob
Google Maps Activity	Activity dengan menyediakan konfigurasi dasar Google Maps
Login Activity	Activity untuk halaman <i>login</i>
Master / Detail Flow	Activity yang diperuntukan untuk alur aplikasi <i>master detail</i> pada peranti tablet
Navigation Drawer Activity	Activity dengan tampilan side bar menu
Scrolling Activity	Activity dengan kemampuan <i>scroll</i> konten didalamnya secara vertikal
Settings Activity	Activity yang diperuntukan untuk konfigurasi aplikasi
Tabbed Activity	Activity yang diperuntukan untuk menampilkan lebih dari satu tampilan, dapat digeser ke kanan dan ke kiri (<i>swipe</i>) dengan menggunakan komponen <i>ViewPager</i>

C. Praktikum

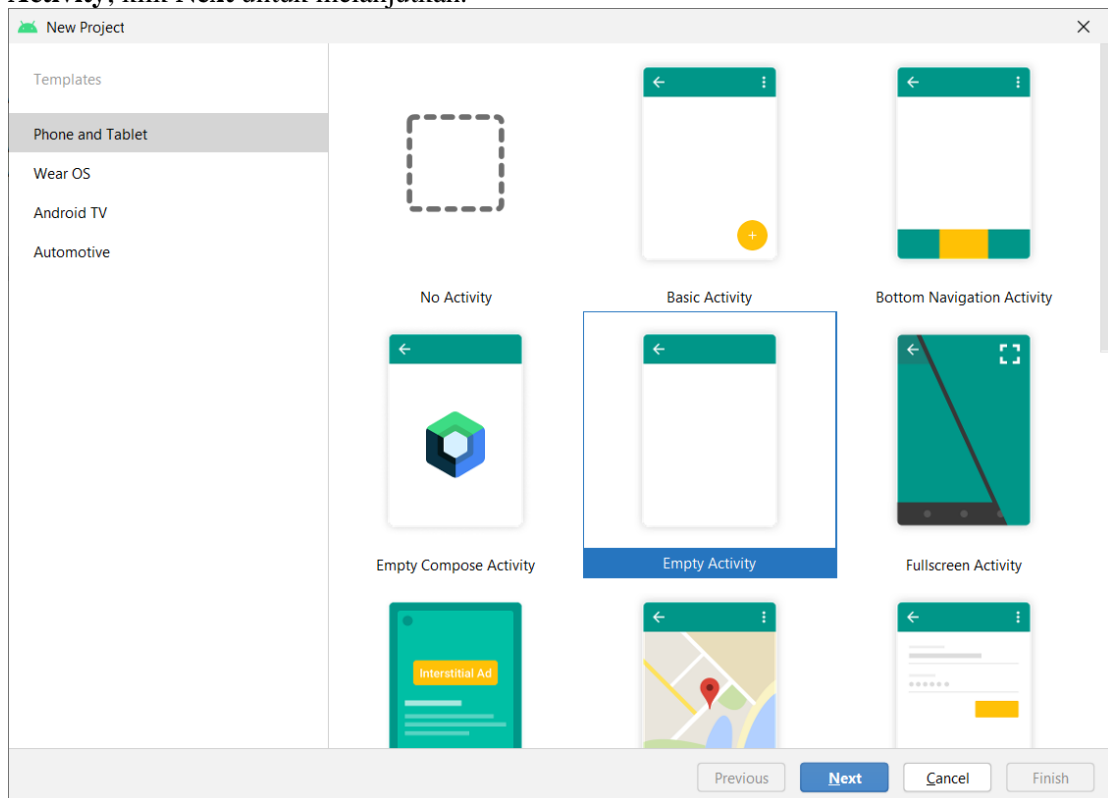
Codelab Membuat Proyek Baru

Ikuti alur berikut untuk mengerjakan latihan ini:

1. Buat proyek baru dengan klik **File** → **New** → **New Project** pada Android Studio Anda atau Anda bisa memilih **Start a new Android Studio project** di bagian *dashboard*.

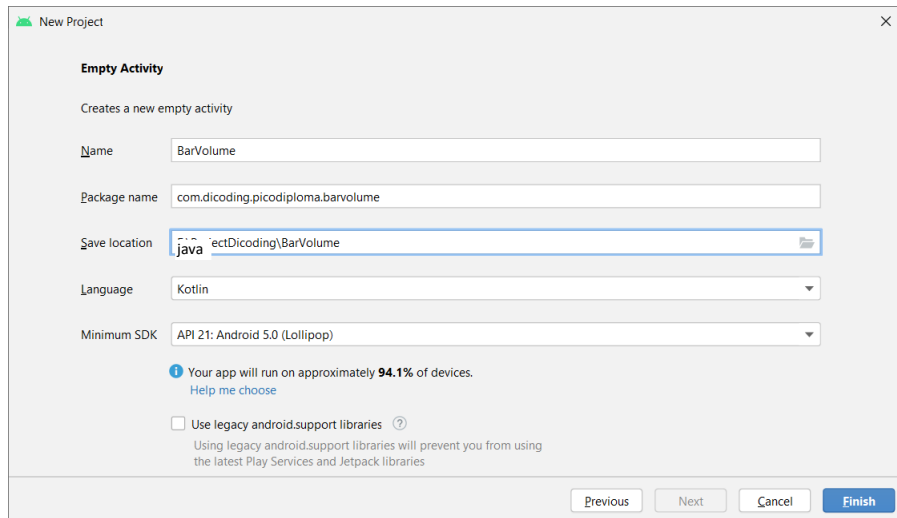


2. Pada bagian ini kita akan memilih tipe Activity awal dari *template* yang telah disediakan. Saat ini Android Studio sudah menyediakan berbagai macam *template* Activity dari yang paling sederhana hingga yang paling kompleks.
3. Selain itu, Anda juga bisa memilih target *device* mana yang akan Anda buat seperti Phone and Tablet, Wear OS, TV, Android Auto atau Android Things. Saat ini kita pilih tipe **Empty Activity**, klik **Next** untuk melanjutkan.

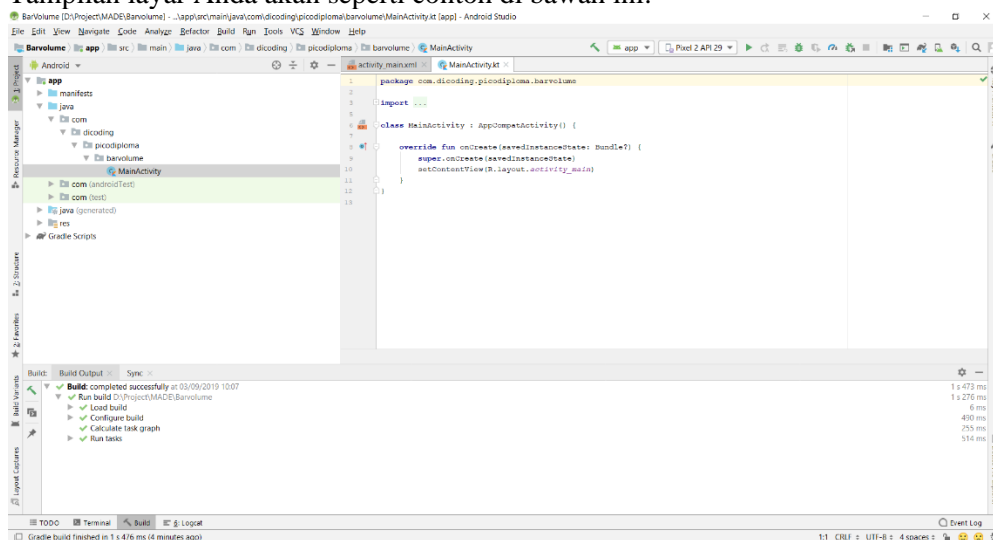


4. Selanjutnya masukkan **nama aplikasi** dan nama *package* aplikasi Anda. Sebaiknya jangan sama dengan apa yang ada di contoh, karena ini berfungsi sebagai **id** dari aplikasi yang Anda

buat. Kemudian Anda bisa menentukan lokasi proyek yang akan Anda buat. Setelah itu pilih tipe gawai/peranti (*device*) untuk aplikasi beserta target minimum BarBarVolumem SDK yang akan digunakan. Pilihan target Android SDK akan mempengaruhi banyaknya peranti yang dapat menggunakan aplikasi. Di sini kita memilih nilai minimum SDK kita pasang ke Level 21 (Lollipop). Klik **Finish** untuk melanjutkan.



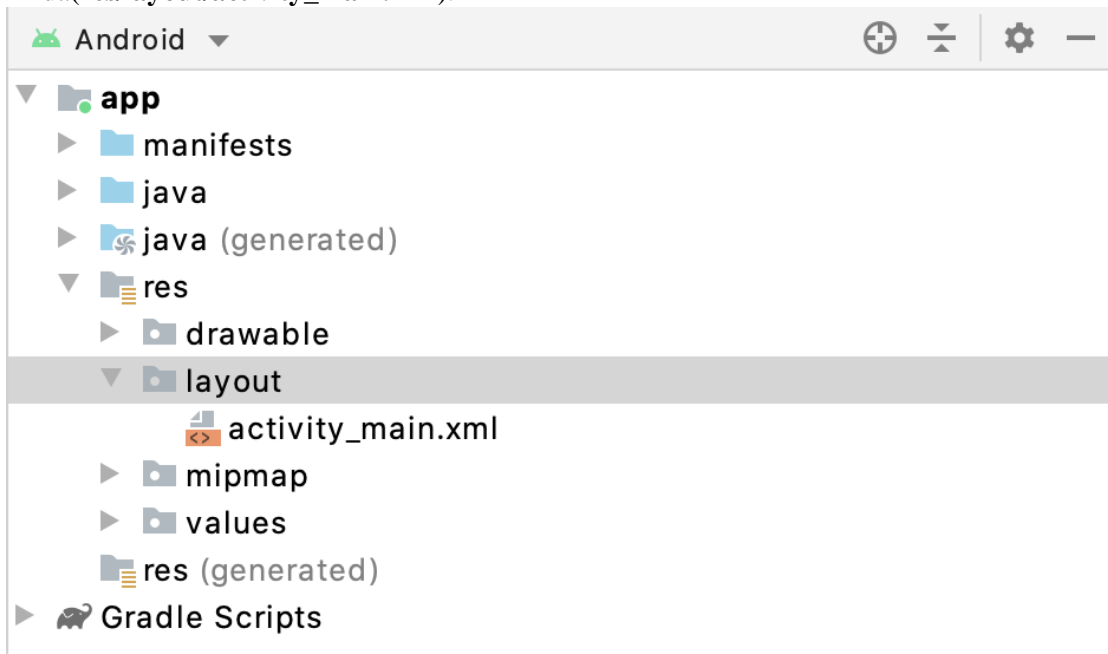
5. Tampilan layar Anda akan seperti contoh di bawah ini:



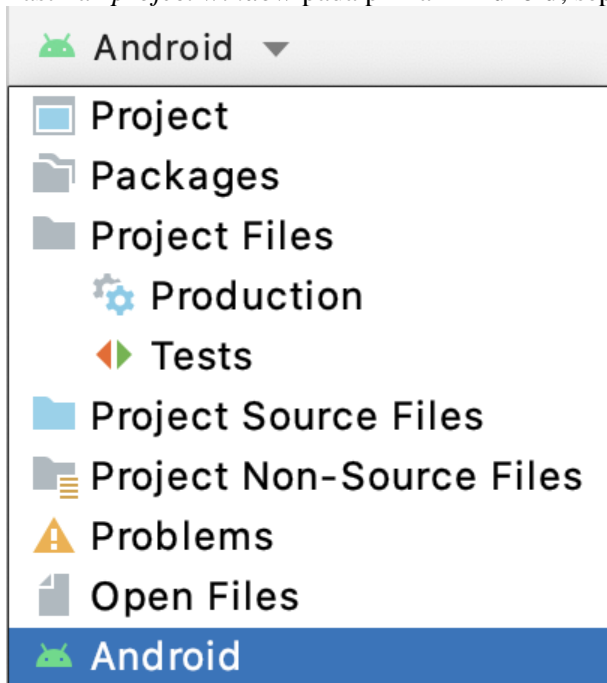
6. Di sebelah kanan Anda adalah *workspace* di mana Activity anda berada dan bernama **MainActivity** dengan *layout*-nya **activity_main.xml**. Di sebelah kiri Anda terdapat struktur proyek, di mana nanti kita akan banyak menambahkan berbagai komponen baru, *asset*, dan *library*. Untuk lebih mengenal Android Studio lebih dalam silakan baca materi di [sini](#).

Codelab Layouting Tampilan

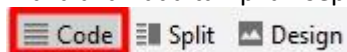
1. Silakan pilih *tab* berkas **activity_main.xml** pada workspace Anda(res/layout/activity_main.xml).



2. Pastikan *project window* pada pilihan **Android**, seperti di bawah ini:



3. Maka akan ada tampilan seperti ini, kemudian pilih *tab* **Code** di sebelah pojok kanan atas.



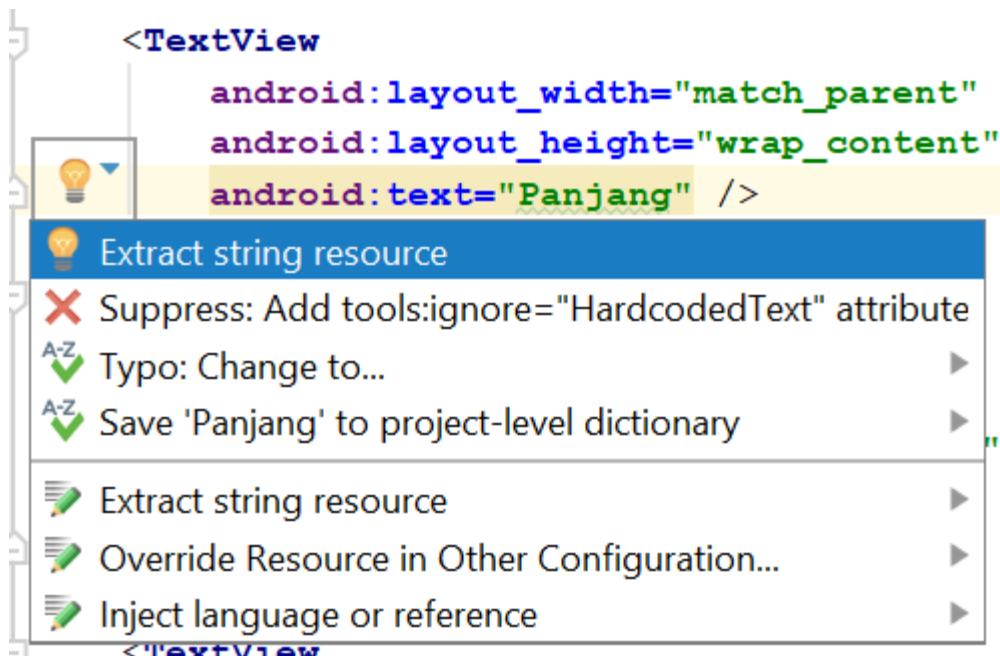
4. Ubah layout dasar dari ConstraintLayout menjadi LinearLayout seperti berikut dan menambahkan baris TextView dan ListView seperti kode di bawah ini :
 1. `<?xml version="1.0" encoding="utf-8"?>`
 2. `<LinearLayout`
`xmlns:android="http://schemas.android.com/apk/res/android"`
 3. `xmlns:tools="http://schemas.android.com/tools"`
 4. `android:layout_width="match_parent"`
 5. `android:layout_height="match_parent"`
 6. `android:padding="16dp"`
 7. `android:orientation="vertical">`

```

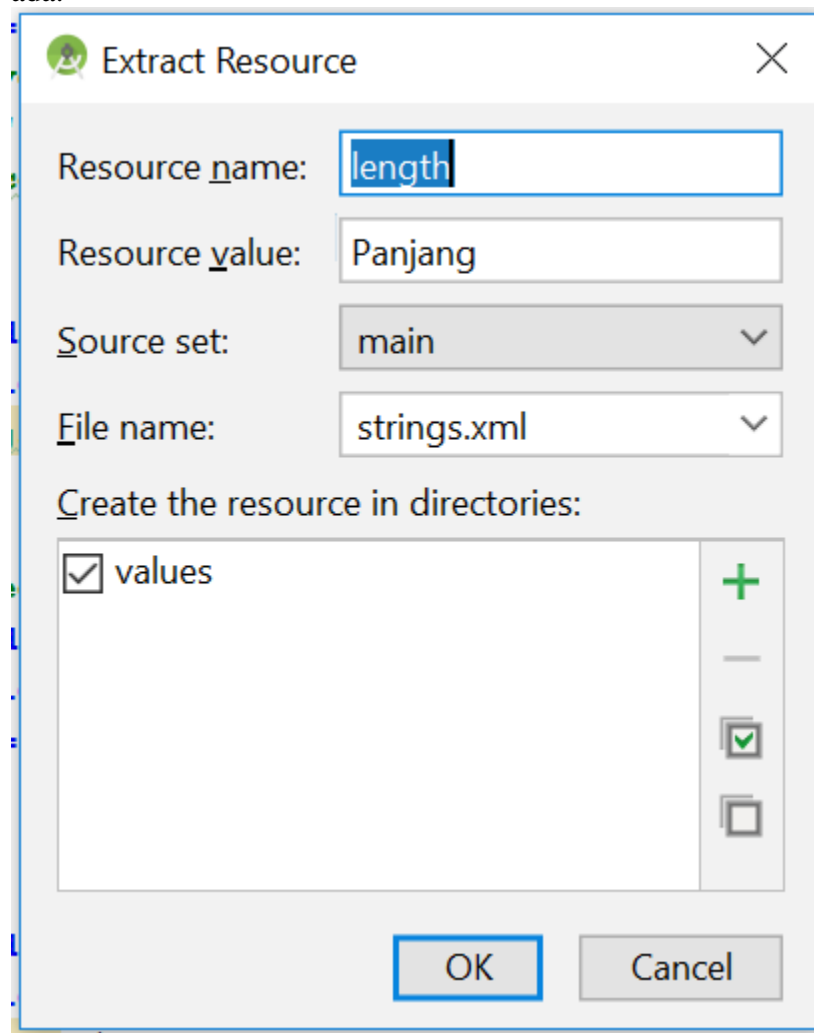
8.
9.     <TextView
10.         android:layout_width="match_parent"
11.         android:layout_height="wrap_content"
12.         android:text="Panjang" />
13.
14.     <EditText
15.         android:id="@+id/edt_length"
16.         android:layout_width="match_parent"
17.         android:layout_height="wrap_content"
18.         android:inputType="numberDecimal"
19.         android:lines="1" />
20.
21.     <TextView
22.         android:layout_width="match_parent"
23.         android:layout_height="wrap_content"
24.         android:text="Lebar" />
25.
26.     <EditText
27.         android:id="@+id/edt_width"
28.         android:layout_width="match_parent"
29.         android:layout_height="wrap_content"
30.         android:inputType="numberDecimal"
31.         android:lines="1" />
32.
33.     <TextView
34.         android:layout_width="match_parent"
35.         android:layout_height="wrap_content"
36.         android:text="Tinggi" />
37.
38.     <EditText
39.         android:id="@+id/edt_height"
40.         android:layout_width="match_parent"
41.         android:layout_height="wrap_content"
42.         android:inputType="numberDecimal"
43.         android:lines="1" />
44.
45.     <Button
46.         android:id="@+id/btn_calculate"
47.         android:layout_width="match_parent"
48.         android:layout_height="wrap_content"
49.         android:text="Hitung" />
50.
51.     <TextView
52.         android:id="@+id/tv_result"
53.         android:layout_width="match_parent"
54.         android:layout_height="wrap_content"
55.         android:gravity="center"
56.         android:text="Hasil"
57.         android:textSize="24sp"
58.         android:textStyle="bold" />
59. </LinearLayout>

```

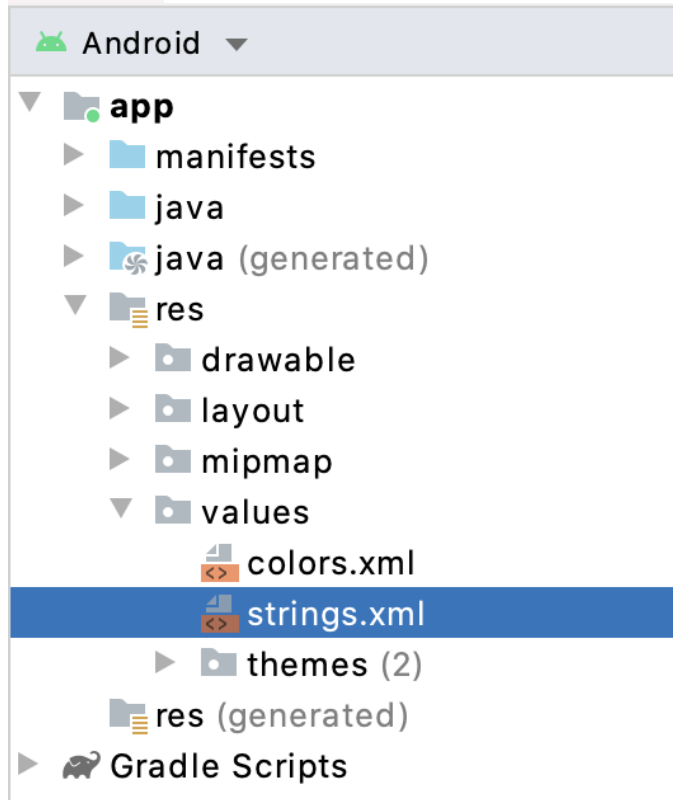
5. Kemudian akan muncul *warning* pada atribut `android:text` pada layout tersebut. Ini karena kita melakukan *hardcoding* pada nilai string-nya. Mari kita hilangkan *code warning* tersebut dengan menekan **Alt+Enter** (**option + return** pada Mac) atau menekan lampu kuning yang muncul pada *attribute* `android:text`. Akan muncul dialog seperti ini, pilih **extract string resource**.



6. Kemudian akan muncul *dialog* seperti di bawah ini. Sesuaikan dengan nama yang ada.



7. Fungsi **extract string resource** akan secara otomatis menambahkan nilai dari **android:text** ke dalam berkas **res** → **values** → **strings.xml**.



8. Lakukan hal yang sama pada komponen view lainnya hingga tidak ada *warning* lagi. Jika kita buka berkas **strings.xml**, maka isinya akan menjadi seperti ini:

```
1. <resources>
2.     <string name="app_name">BarVolume</string>
3.     <string name="width">Lebar</string>
4.     <string name="height">Tinggi</string>
5.     <string name="calculate">Hitung</string>
6.     <string name="result">Hasil</string>
7.     <string name="length">Panjang</string>
8. </resources>
```

9. Maka kode di dalam **activity_main.xml** akan menjadi seperti ini:

```
1. <?xml version="1.0" encoding="utf-8"?>
2. <LinearLayout
3.     xmlns:android="http://schemas.android.com/apk/res/android"
4.     xmlns:tools="http://schemas.android.com/tools"
5.     android:layout_width="match_parent"
6.     android:layout_height="match_parent"
7.     android:padding="16dp"
8.     android:orientation="vertical">
9.     <TextView
10.         android:layout_width="match_parent"
11.         android:layout_height="wrap_content"
12.         android:text="@string/length" />
13.     <EditText
14.         android:id="@+id/edt_length"
15.         android:layout_width="match_parent"
16.         android:layout_height="wrap_content"
17.         android:inputType="numberDecimal"
18.         android:lines="1" />
```



```

20.
21.     <TextView
22.         android:layout_width="match_parent"
23.         android:layout_height="wrap_content"
24.         android:text="@string/width" />
25.
26.     <EditText
27.         android:id="@+id/edt_width"
28.         android:layout_width="match_parent"
29.         android:layout_height="wrap_content"
30.         android:inputType="numberDecimal"
31.         android:lines="1" />
32.
33.     <TextView
34.         android:layout_width="match_parent"
35.         android:layout_height="wrap_content"
36.         android:text="@string/height" />
37.
38.     <EditText
39.         android:id="@+id/edt_height"
40.         android:layout_width="match_parent"
41.         android:layout_height="wrap_content"
42.         android:inputType="numberDecimal"
43.         android:lines="1" />
44.
45.     <Button
46.         android:id="@+id/btn_calculate"
47.         android:layout_width="match_parent"
48.         android:layout_height="wrap_content"
49.         android:text="@string/calculate" />
50.
51.     <TextView
52.         android:id="@+id/tv_result"
53.         android:layout_width="match_parent"
54.         android:layout_height="wrap_content"
55.         android:gravity="center"
56.         android:text="@string/result"
57.         android:textSize="24sp"
58.         android:textStyle="bold" />
59. </LinearLayout>

```

10. Jika Anda perhatikan, hasil layout sementara akan menjadi seperti ini:

The screenshot shows a mobile application interface with a light blue background. It contains three input fields stacked vertically, each with a label above it: 'Panjang' (Length), 'Lebar' (Width), and 'Tinggi' (Height). Below these fields is a red button with the text 'HITUNG' (Calculate). At the bottom of the screen, the word 'Hasil' (Result) is displayed in a large, bold, black font.

Codelab Kode Logika pada MainActivity

1. Selanjutnya setelah selesai, lanjutkan dengan membuka berkas **MainActivity** dan lanjutkan ketik baris-baris di bawah ini.

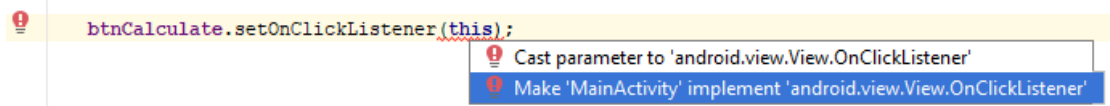
Tambahkan beberapa variabel yang akan digunakan untuk menampung view.

```
1. private EditText edtWidth;
2. private EditText edtHeight;
3. private EditText edtLength;
4. private Button btnCalculate;
5. private TextView tvResult;
```

2. Kemudian inisiasi variabel yang telah kita buat dengan menambahkan kode berikut di dalam metode **onCreate**.

```
1. @Override
2. protected void onCreate(Bundle savedInstanceState) {
3.     super.onCreate(savedInstanceState);
4.     setContentView(R.layout.activity_main);
5.
6.     edtWidth = findViewById(R.id.edt_width);
7.     edtHeight = findViewById(R.id.edt_height);
8.     edtLength = findViewById(R.id.edt_length);
9.     btnCalculate = findViewById(R.id.btn_calculate);
10.    tvResult = findViewById(R.id.tv_result);
11.
12.    btnCalculate.setOnClickListener(this);
13.
14. }
```

3. Akan muncul baris merah pada kata **this**. Hal ini karena kita belum menambahkan interface di kelas **MainActivity**. Silakan klik di atas baris merah tersebut, kemudian tekan tombol **Alt + Enter** (option + return pada Mac) atau menekan lampu merah yang muncul lalu pilih aksi berikut untuk implement interface.



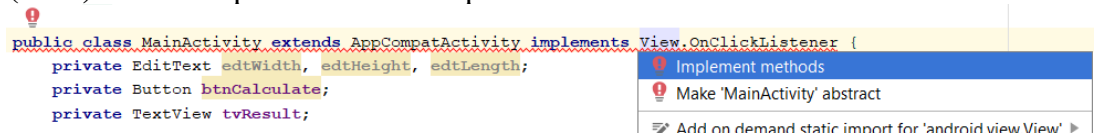
4. Maka secara otomatis akan ada penambahan kode pada kelas **MainActivity** seperti berikut ini:

```
1. public class MainActivity extends AppCompatActivity implements
   View.OnClickListener {
2.
3.     ...
4.
5. }
```

5. Jika terdapat baris merah seperti ini:

class MainActivity

Jangan khawatir! Silakan klik di atas baris merah tersebut, kemudian tekan tombol **Alt + Enter** (option + return pada Mac) atau menekan lampu merah yang muncul lalu pilih **implement members (Kotlin)** atau **implement methods (Java)**. Jika terdapat baris merah seperti ini:



6. Maka secara otomatis akan ada penambahan metode **onClick** di kelas **MainActivity**. Setelah itu, tambahkan kode berikut ke dalam metode **onClick**:

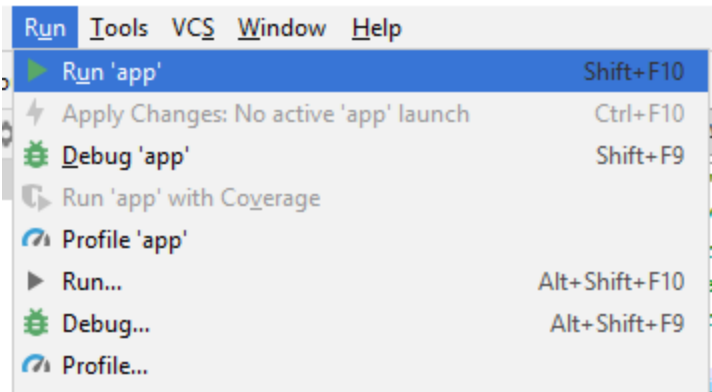
```
1. @Override
2. protected void onCreate(Bundle savedInstanceState) {
3.     super.onCreate(savedInstanceState);
4.     setContentView(R.layout.activity_main);
5.
6.     ...
7.
8. }
9.
10.     @Override
11.     public void onClick(View v) {
12.         if (v.getId() == R.id.btn_calculate) {
13.             String inputLength =
14.                 edtLength.getText().toString().trim();
15.             String inputWidth =
16.                 edtWidth.getText().toString().trim();
17.             String inputHeight =
18.                 edtHeight.getText().toString().trim();
19.
20.             Double volume = Double.parseDouble(inputLength) *
21.                 Double.parseDouble(inputWidth) * Double.parseDouble(inputHeight);
22.             tvResult.setText(String.valueOf(volume));
23.         }
24.     }
```

7. Akhirnya kelas **MainActivity** akan memiliki kode seperti berikut ini:

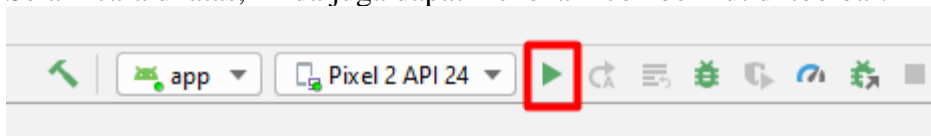
```
1. public class MainActivity extends AppCompatActivity implements
2.     View.OnClickListener {
3.     private EditText edtWidth, edtHeight, edtLength;
4.     private Button btnCalculate;
5.     private TextView tvResult;
6.
7.     @Override
8.     protected void onCreate(Bundle savedInstanceState) {
9.         super.onCreate(savedInstanceState);
10.        setContentView(R.layout.activity_main);
11.
12.        edtWidth = findViewById(R.id.edt_width);
13.        edtHeight = findViewById(R.id.edt_height);
14.        edtLength = findViewById(R.id.edt_length);
15.        btnCalculate = findViewById(R.id.btn_calculate);
16.        tvResult = findViewById(R.id.tv_result);
17.
18.        btnCalculate.setOnClickListener(this);
19.    }
20.
21.    @Override
22.    public void onClick(View v) {
23.        if (v.getId() == R.id.btn_calculate) {
24.            String inputLength =
25.                edtLength.getText().toString().trim();
26.            String inputWidth =
27.                edtWidth.getText().toString().trim();
28.            String inputHeight =
29.                edtHeight.getText().toString().trim();
30.
31.            Double volume = Double.parseDouble(inputLength)
32.                * Double.parseDouble(inputWidth) *
33.                Double.parseDouble(inputHeight);
34.            tvResult.setText(String.valueOf(volume));
35.        }
36.    }
```

```
29.         }  
30.     }  
31. }
```

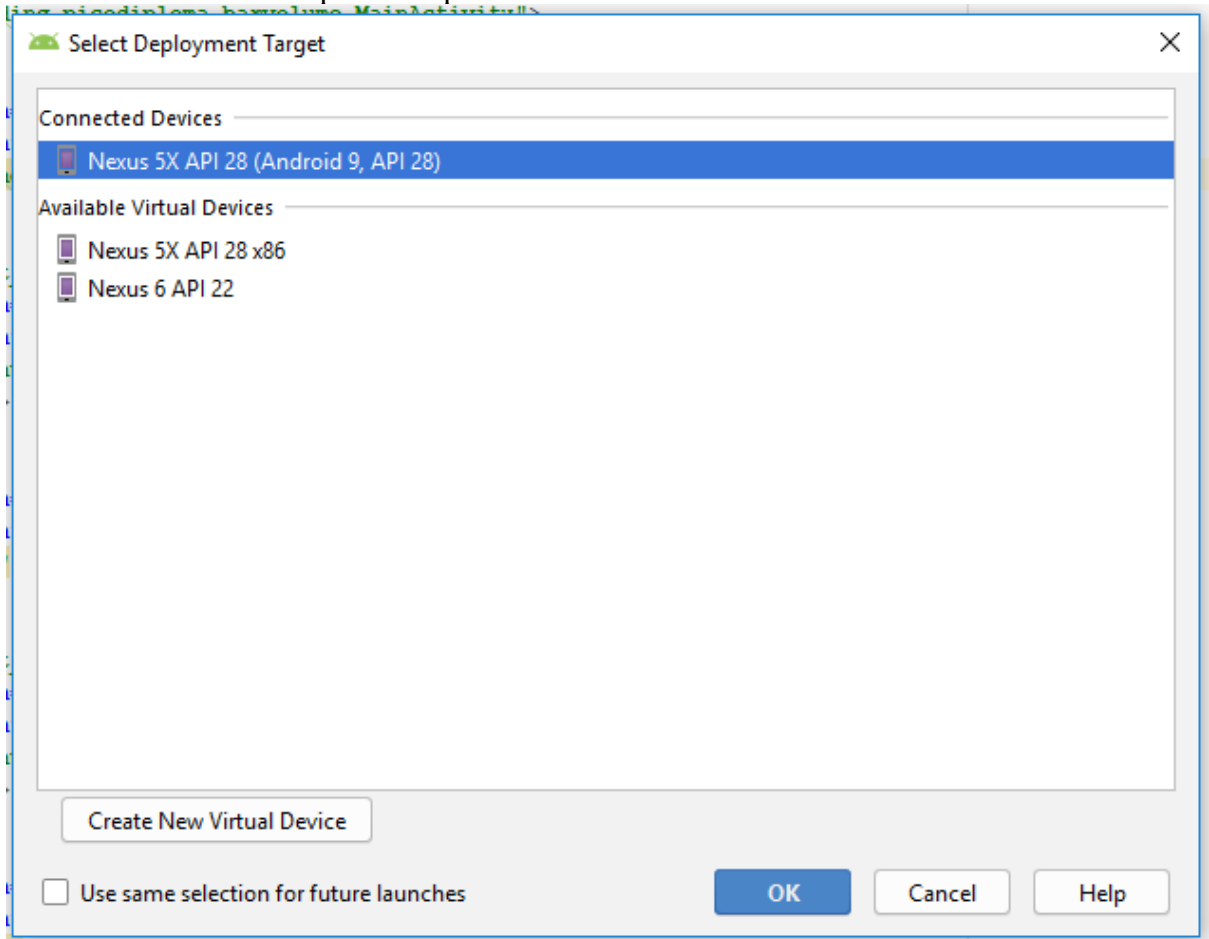
8. Setelah selesai, silakan jalankan aplikasi dengan memilih menu **Run** → **Run 'app'** dari *menu bar*.



9. Selain cara di atas, Anda juga dapat menekan icon berikut di toolbar:



10. Kemudian akan muncul pilihan seperti ini:



11. Pilih **OK** untuk menjalankan dan tunggu hingga proses *building* dan instalasi APK selesai.
12. Silakan masukkan nilai panjang, lebar, dan tinggi kemudian tekan tombol **Hitung** dan hasilnya akan ditampilkan di objek textview **tvResult**. Namun masih ada sedikit masalah di sini, yaitu Anda tetap melakukan proses perhitungan walaupun salah satu nilainya kosong. Hal ini akan menyebabkan aplikasi *force close* karena perhitungan tidak dapat diproses. Maka untuk mengatasinya Anda akan menggunakan percabangan untuk mengecek apakah masing-masing EditText kosong atau tidak.
Silakan buka kembali kelas **MainActivity**. Tambahkan kode berikut ke dalam metode **onClick** sebelum melakukan perhitungan.

```
1.  @Override
2.  public void onClick(View v) {
3.      if (v.getId() == R.id.btn_calculate) {
4.          String inputLength =
5.          edtLength.getText().toString().trim();
6.          String inputWidth =
7.          edtWidth.getText().toString().trim();
8.          String inputHeight =
9.          edtHeight.getText().toString().trim();
10.
11.         boolean isEmptyFields = false;
12.
13.         if (TextUtils.isEmpty(inputLength)) {
14.             isEmptyFields = true;
15.             edtLength.setError("Field ini tidak boleh kosong");
16.         }
17.
18.         if (TextUtils.isEmpty(inputWidth)) {
19.             isEmptyFields = true;
20.             edtWidth.setError("Field ini tidak boleh kosong");
21.         }
22.
23.         if (TextUtils.isEmpty(inputHeight)) {
24.             isEmptyFields = true;
25.             edtHeight.setError("Field ini tidak boleh kosong");
26.         }
27.
28.         if (!isEmptyFields) {
29.             Double volume = Double.parseDouble(inputLength) *
30.             Double.parseDouble(inputWidth) *
31.             Double.parseDouble(inputHeight);
32.             tvResult.setText(String.valueOf(volume));
33.         }
34.     }
35. }
```

13. Akhirnya kelas **MainActivity** akan memiliki kode seperti berikut ini:

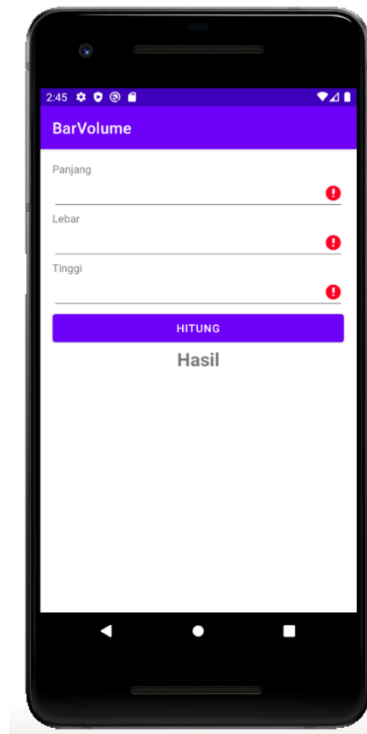
```
1.  public class MainActivity extends AppCompatActivity implements
2.  View.OnClickListener {
3.      private EditText edtWidth, edtHeight, edtLength;
4.      private Button btnCalculate;
5.      private TextView tvResult;
6.
7.      @Override
8.      protected void onCreate(Bundle savedInstanceState) {
9.          super.onCreate(savedInstanceState);
10.         setContentView(R.layout.activity_main);
11.     }
```

```

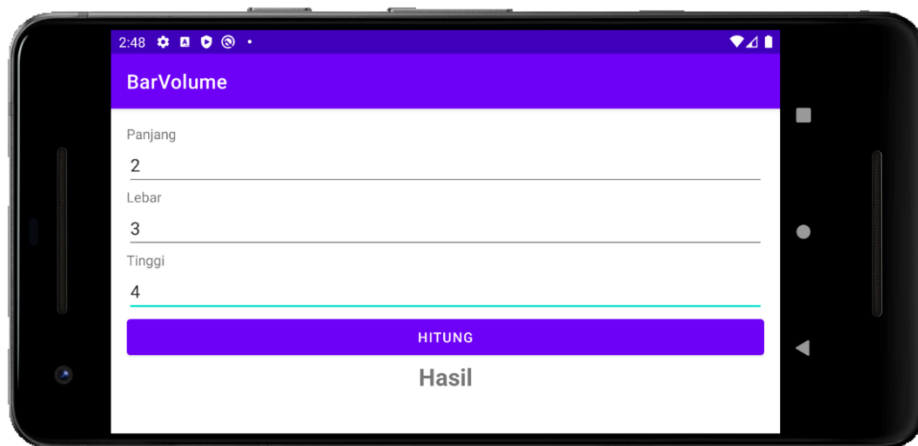
11.         edtWidth = findViewById(R.id.edt_width);
12.         edtHeight = findViewById(R.id.edt_height);
13.         edtLength = findViewById(R.id.edt_length);
14.         btnCalculate = findViewById(R.id.btn_calculate);
15.         tvResult = findViewById(R.id.tv_result);
16.
17.         btnCalculate.setOnClickListener(this);
18.     }
19.
20.     @Override
21.     public void onClick(View v) {
22.         if (v.getId() == R.id.btn_calculate) {
23.             String inputLength =
24. edtLength.getText().toString().trim();
25.             String inputWidth =
26. edtWidth.getText().toString().trim();
27.             String inputHeight =
28. edtHeight.getText().toString().trim();
29.
30.             boolean isEmptyFields = false;
31.
32.             if (TextUtils.isEmpty(inputLength)) {
33.                 isEmptyFields = true;
34.                 edtLength.setError("Field ini tidak boleh
35. kosong");
36.             }
37.
38.             if (TextUtils.isEmpty(inputWidth)) {
39.                 isEmptyFields = true;
40.                 edtWidth.setError("Field ini tidak boleh
41. kosong");
42.             }
43.
44.             if (TextUtils.isEmpty(inputHeight)) {
45.                 isEmptyFields = true;
46.                 edtHeight.setError("Field ini tidak boleh
47. kosong");
48.             }
49.
50.             if (!isEmptyFields) {
51.                 Double volume = Double.parseDouble(inputLength)
52. * Double.parseDouble(inputWidth) *
53. Double.parseDouble(inputHeight);
54.                 tvResult.setText(String.valueOf(volume));
55.             }
56.         }
57.     }
58. }
59.
60. }
61.
62. }
63.
64. }
65.
66. }
67.
68. }
69.
70. }
71.
72. }
73.
74. }
75.
76. }
77.
78. }
79.
80. }
81.
82. }
83.
84. }
85.
86. }
87.
88. }
89.
90. }
91.
92. }
93.
94. }
95.
96. }
97.
98. }
99.
100. }

```

14. Jalan kembali aplikasi Anda dengan memilih menu **Run** → **Run 'app'** atau shortcut **Shift + F10**. Cobalah langsung menekan tombol **HITUNG** tanpa mengisi EditText, maka aplikasi Anda tidak akan force close dan akan muncul peringatan bahwa **"Field ini tidak boleh kosong"**.



15. Apakah kita sudah selesai? Belum! Masih ada yang kurang. Ketika nilai *volume* sudah dihitung dan kemudian terjadi pergantian orientasi (*portrait-landscape*) pada peranti, maka hasil perhitungan tadi akan hilang. Hal ini karena di dalam Android, jika melakukan pergantian orientasi, Android akan memanggil fungsi `onCreate` kembali, sehingga data akan kembali menjadi seperti semula.



16. Untuk mengatasinya, tambahkan metode `onSaveInstanceState()` pada **MainActivity** dan sesuaikan seperti berikut:

```

1. private static final String STATE_RESULT = "state_result";
2.
3. ...
4.
5. @Override
6. protected void onSaveInstanceState(Bundle outState) {
7.     super.onSaveInstanceState(outState);
8.     outState.putString(STATE_RESULT,
9.         tvResult.getText().toString());
10. }

```

17. Kemudian tambahkan juga beberapa baris berikut pada baris terakhir metode `onCreate`.

```

1.  @Override
2.  protected void onCreate(Bundle savedInstanceState) {
3.      super.onCreate(savedInstanceState);
4.      setContentView(R.layout.activity_main);
5.
6.      ...
7.
8.      if (savedInstanceState != null) {
9.          String result =
savedInstanceState.getString(STATE_RESULT);
10.         tvResult.setText(result);
11.     }
12. }

```

18. Silakan jalankan kembali aplikasinya. Ulangi proses perhitungan seperti sebelumnya. Kemudian ganti orientasi peranti Anda. Jika sudah benar maka hasil perhitungan tidak akan hilang.

Daftar Pustaka

Kelas Online, Kelas Fundamental Android Developer Expert, Dicoding.