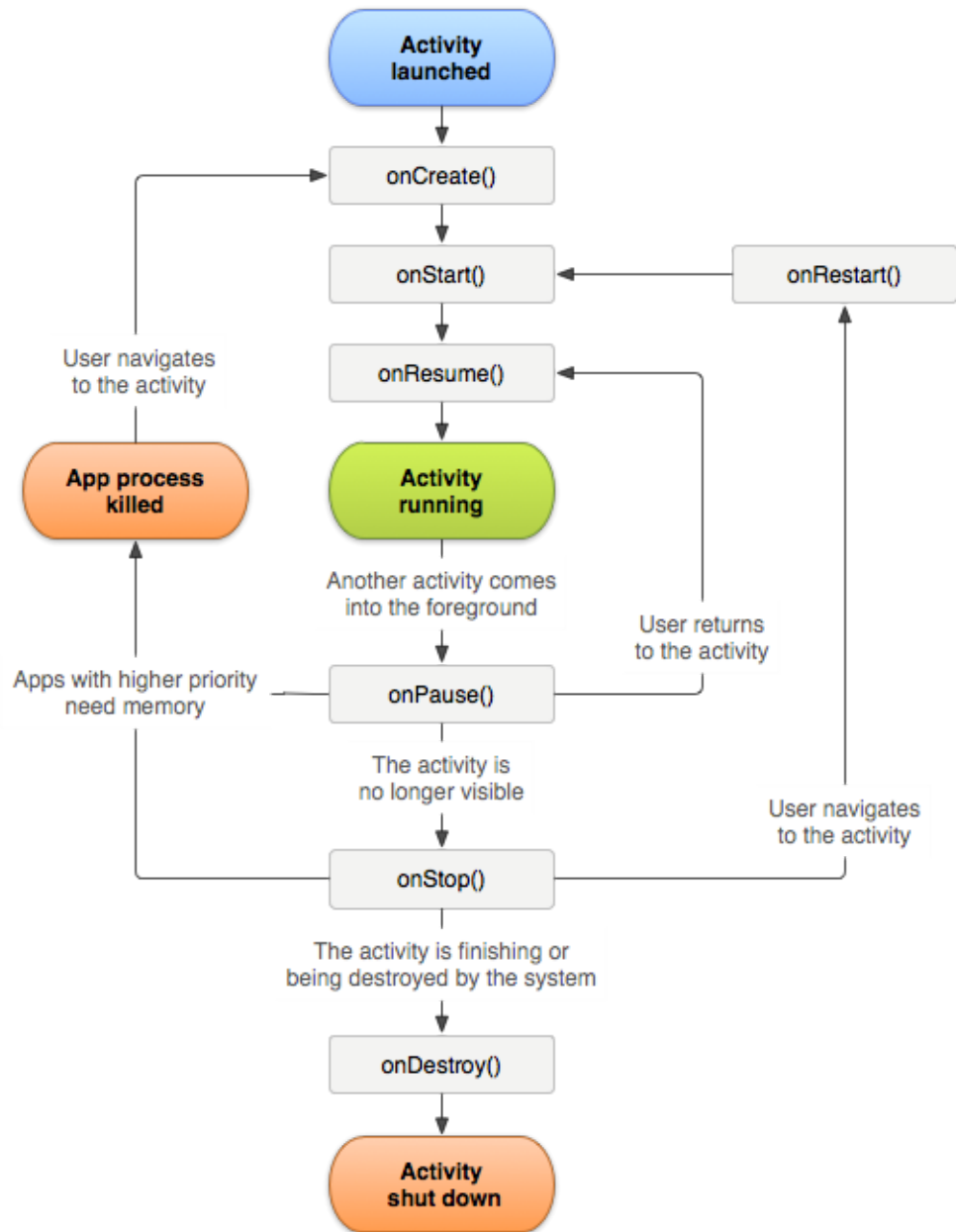


Modul 1 (Activity)

- Activity merupakan salah satu komponen penting Android yang berfungsi untuk menampilkan user interface ke layar pengguna.
- ketika Anda membuat project baru di Android Studio, biasanya akan ada dua berkas yang sudah tercipta, yaitu MainActivity dan activity_main.xml.
- MainActivity ini disebut sebagai class Activity karena mewarisi (extends) superclass Activity. Tugasnya yaitu menampilkan layout activity_main.xml dan mengelola interaksi yang ada di dalamnya.
- Yang perlu diperhatikan yaitu setiap Activity harus terdaftar di **AndroidManifest.xml**.

Activity Lifecycle






- `onCreate()`
Di-method ini Activity sudah dimulai tapi belum terlihat oleh pengguna. Inisialisasi sebagian besar dimulai di sini. Misalnya memanggil `setContentView()` untuk membaca layout, membaca View, dll.
- `onStart()`
Activity sudah terlihat tapi belum bisa berinteraksi. Method ini jarang dipakai, tapi bisa sangat berguna untuk mendaftarkan sebuah BroadcastReceiver untuk mengamati perubahan yang dapat mempengaruhi UI.

- `onResume()`
Activity sudah terlihat dan pengguna sudah dapat berinteraksi. Di sini adalah tempat terbaik untuk menjalankan animasi, membuka akses seperti camera, mengupdate UI, dll.
- `onPause()`
Kebalikan dari `onResume()`. Activity sudah akan bersiap-siap meninggalkan layar (masih terlihat) dan sudah tidak berinteraksi dengan pengguna. Biasanya bila perlu melakukan undo untuk pekerjaan yang dilakukan di `onResume()` kita lakukan di sini.
- `onStop()`
Kebalikan dari `onStart()` Activity sudah tidak terlihat. Biasanya kita melakukan undo untuk pekerjaan yang dilakukan di dalam `onStart()`.
- `onDestroy()`
Kebalikan dari `onCreate()`. Method ini dapat terpanggil karena memanggil method `finish()` atau karena sistem membutuhkan memori lebih. Di dalam `onDestroy()` kita biasanya membersihkan proses-proses yang ada di belakang layar. Misalnya pengunduhan data dari internet yang mungkin masih berjalan jika tidak dihentikan di `onDestroy()`.
- `onRestart()`
Dipanggil saat activity sudah melalui `onStop()` tapi akan diaktifkan lagi. Method ini jarang di implementasi.

Referensi link mengenai Activity https://youtu.be/rucdc_Qv2HM

Jenis-jenis Activity

 Jenis-jenis template Activity	 Fungsinya
<u>Add No Activity</u>	Tidak ada activity yang ditambahkan
<u>Basic Activity</u>	Activity dengan template komponen material design seperti FloatingActionButton

<u>Aa</u> Jenis-jenis template Activity	 Fungsinya
<u>Bottom Navigation Activity.</u>	Activity dengan tampilan side bar menu di bagian bawah
<u>Empty Activity.</u>	Activity dalam bentuk yang paling dasar
<u>Fragment + ViewModel</u>	Activity dengan menerapkan architecture component
<u>Fullscreen Activity.</u>	Activity fullscreen tanpa status bar
<u>Activity fullscreen tanpa status bar</u>	Activity dengan konfigurasi default iklan Admob
<u>Google Maps Activity.</u>	Activity dengan menyediakan konfigurasi dasar Google Maps
<u>Login Activity.</u>	Activity untuk halaman login
<u>Master / Detail Flow</u>	Activity yang diperuntukan untuk alur aplikasi master detail pada peranti tablet
<u>Navigation Drawer Activity.</u>	Activity dengan tampilan side bar menu
<u>Scrolling Activity.</u>	Activity dengan kemampuan scroll konten didalamnya secara vertikal
<u>Settings Activity.</u>	Activity yang diperuntukan untuk konfigurasi aplikasi
<u>Tabbed Activity.</u>	Activity yang diperuntukan untuk menampilkan lebih dari satu tampilan, dapat digeser ke kanan dan ke kiri (swipe) dengan menggunakan komponen ViewPager

Code Lab menghitung volume balok

Untuk Membuat tampilan maka kita akan menggunakan layout paling sederhana yaitu Linear Layout

Panjang

Lebar

Tinggi

HITUNG

Hasil

Diawali dengan membuat activity_main

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
```

```

        android:padding="16dp"
        android:orientation="vertical">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Panjang" />

        <EditText
            android:id="@+id/edt_length"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="numberDecimal"
            android:lines="1" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Lebar" />

        <EditText
            android:id="@+id/edt_width"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="numberDecimal"
            android:lines="1" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Tinggi" />

        <EditText
            android:id="@+id/edt_height"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="numberDecimal"
            android:lines="1" />

        <Button
            android:id="@+id/btn_calculate"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Hitung" />

        <TextView
            android:id="@+id/tv_result"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:text="Hasil"
            android:textSize="24sp"
            android:textStyle="bold" />
    </LinearLayout>

```

Dilanjutkan dengan membuat logikanya

```

class MainActivity : AppCompatActivity(), View.OnClickListener {
    private lateinit var edtWidth: EditText
    private lateinit var edtHeight: EditText
    private lateinit var edtLength: EditText
    private lateinit var btnCalculate: Button
    private lateinit var tvResult: TextView

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        edtWidth = findViewById(R.id.edt_width)
        edtHeight = findViewById(R.id.edt_height)
        edtLength = findViewById(R.id.edt_length)
        btnCalculate = findViewById(R.id.btn_calculate)
        tvResult = findViewById(R.id.tv_result)

        btnCalculate.setOnClickListener(this)
    }

    override fun onClick(v: View?) {
        if (v?.id == R.id.btn_calculate) {
            val inputLength = edtLength.text.toString().trim()
            val inputWidth = edtWidth.text.toString().trim()
            val inputHeight = edtHeight.text.toString().trim()

            val volume = inputLength.toDouble() * inputWidth.toDouble() * inputHeight.toDouble()
            tvResult.text = volume.toString()
        }
    }
}

```

Bedah code

```

private lateinit var edtWidth: EditText
private lateinit var edtHeight: EditText
private lateinit var edtLength: EditText
private lateinit var btnCalculate: Button
private lateinit var tvResult: TextView

```

Pada bagian ini kita melakukan inisialisasi sebuah variabel untuk menampung data yang ada pada view.

```

edtWidth = findViewById(R.id.edt_width)
edtHeight = findViewById(R.id.edt_height)
edtLength = findViewById(R.id.edt_length)
btnCalculate = findViewById(R.id.btn_calculate)
tvResult = findViewById(R.id.tv_result)

```

Pada bagian ini kita menampung variabel yang kita inisialisasi sebelumnya dengan data yang ada pada activity_main

```
btnCalculate.setOnClickListener(this)
```

Pada bagian diatas kita memberi kemampuan agar sebuah button akan memiliki aksi saat di tekan

```
override fun onClick(v: View?) {  
    if (v?.id == R.id.btn_calculate) {  
        val inputLength = edtLength.text.toString().trim()  
        val inputWidth = edtWidth.text.toString().trim()  
        val inputHeight = edtHeight.text.toString().trim()  
  
        val volume = inputLength.toDouble() * inputWidth.toDouble() * inputHeight.toDouble()  
        tvResult.text = volume.toString()  
    }  
}
```

Untuk aksi pada button yang ditekan kita memanggil function onClick yang mana mengubah data dari inputLength, inputWidht dan inputHeight menjadi toString dan di trim agar dap

```
tvResult.text = volume.toString()
```

Diakhiri dengan menampung hasil perkalian volume tadi ke dalam tvResult yang ada pada Activity_main.