



Toast dan Snackbar

Gold - Chapter 4 - Topic 1

Selamat datang di **Chapter 4 Topic 1** online course
Android Developer dari Binar Academy!





Welcome to Gold - Chapter 4 ✨

Pada chapter 3, kita udah melewati banyak fitur yang berkaitan dengan Android Studio, mulai dari Activity, Fragment, hingga Navigasi antar screen.

Di chapter ini, kita akan gali lebih dalam fitur-fitur di Android Studio seputar deklarasi pesan, recycler view, hingga penyimpanan data.

Pertama bin utama, kita akan bahas apa yang dimaksud dengan **Toast and Snack Bar**. Yuk, lanjut!

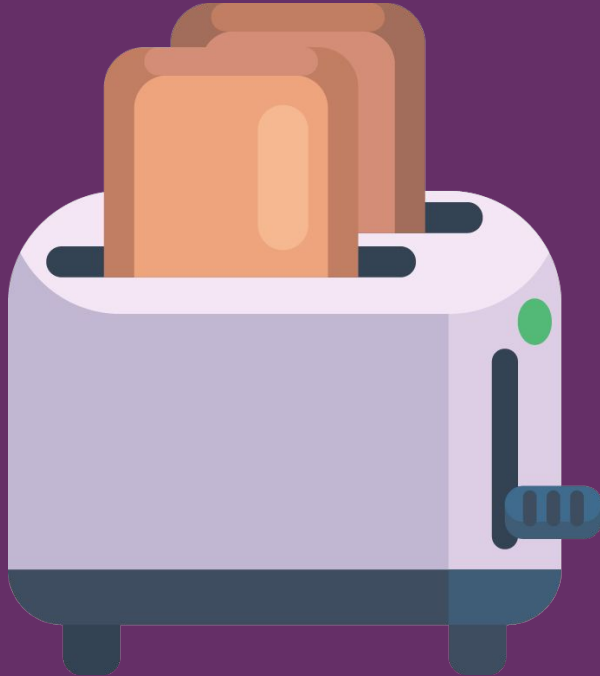




Detailnya, kita bakal bahas hal-hal berikut ini:

- Toast
- Snackbar
- Perbedaan Toast dan Snackbar





Pertama kita akan mempelajari tentang **Toast**.

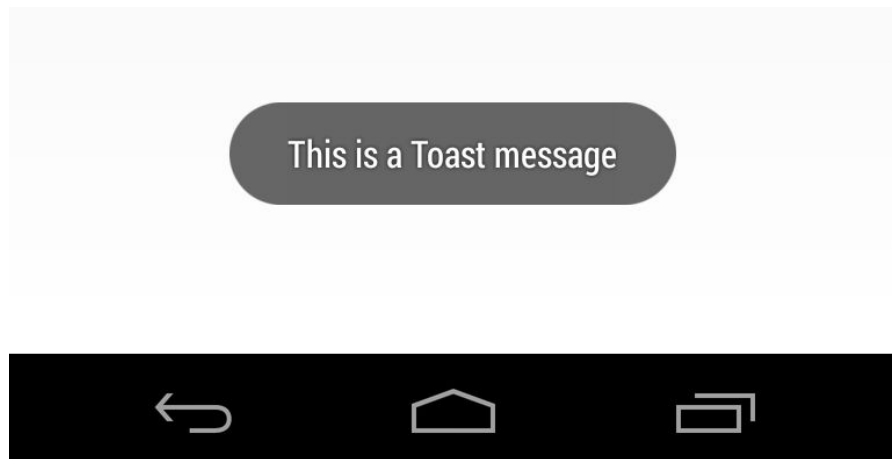
Apa sih Toast itu?

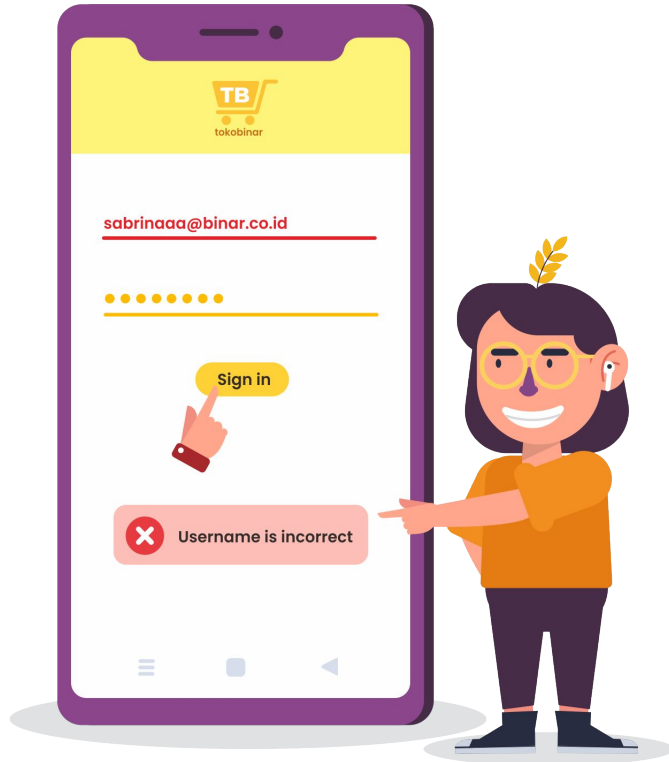
Apakah sama kayak roti panggang yang biasa kita makan?



Hmm.. biar gampang, ambil contoh di aplikasi yang sering kita pake aja ya!

Coba deh.. biasanya kamu liat pesen kayak gini dimana? Yang muncul sekilas, terus tuh munculnya ditengah layar gitu. Nah itu adalah Toast.



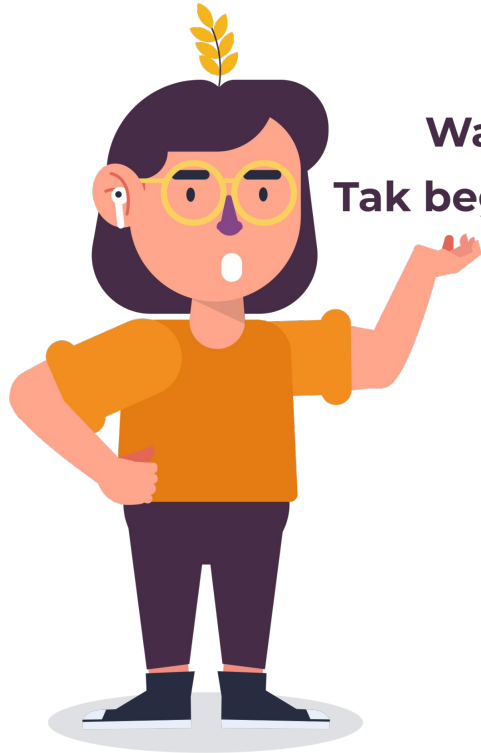


Apa itu Toast?

Toast merupakan pesan informasi yang **tampil secara sementara**.

Biasanya tampil sebagai pesan informasi yang mengambang contohnya:

- Kalo pas SMS terkirim,
- Koneksi gagal, dan
- Pesan informasi lainnya yang sifatnya nggak terlalu penting.



Waktu Singkat 
Tak begitu penting 

Suatu informasi bisa disebut Toast kalo kriteria-nya kayak gini :

- Menampilkan pesan informasi yang singkat, tidak terlalu panjang. Pesan yang disampaikan dapat dibaca user dari 2 sampai 3.5 Detik.
- Pesan pemberitahuan memiliki prioritas yang tidak terlalu penting.



Tampil Sementara Informasi yang kontekstual

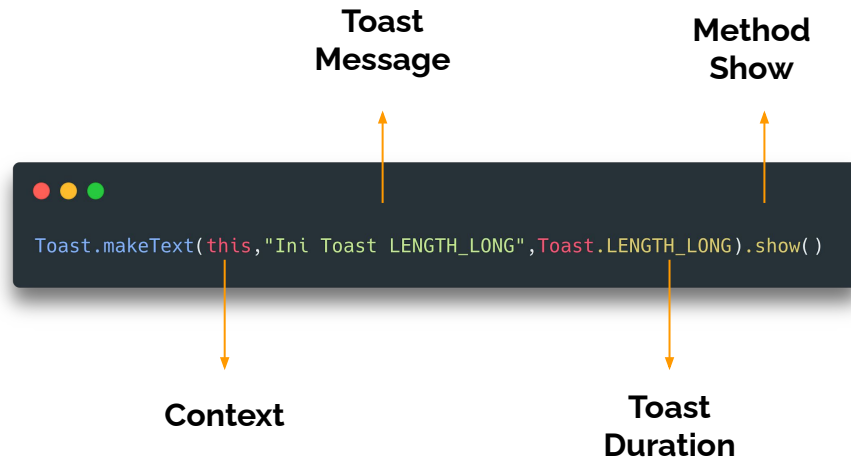


- Toast akan tampil sementara. Toast akan hilang secara otomatis.
- Toast menampilkan informasi yang kontekstual. Pesan Toast yang ditampilkan sesuai dengan apa yang tampil pada layar.



Gengs.. sekarang kalo kita liat ada pop up info SMS kekirim, nyebutnya Toast aja yaa hihi 😏

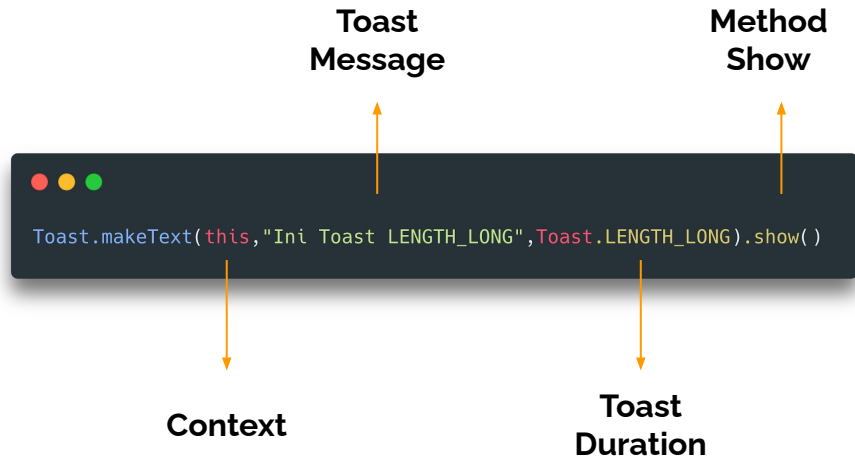
Lanjut nih, gimana sih caranya bikin **Toast** begitu?



Membuat Toast ternyata mudah!

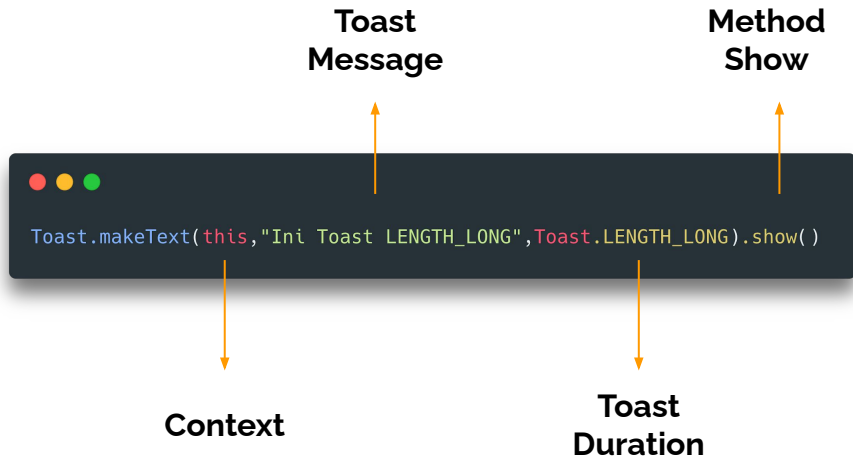
Easy peasy lemon squeezy, untuk membuat Toast, kita cukup menggunakan method **makeText** dari Class **Toast** yang udah ada, tanpa menambahkan persiapan apapun sebelumnya.

Supaya bisa menampilkan sebuah Toast, kamu bisa pake code kayak gambar di samping.



Kalo kamu perhatiin code-nya, ada beberapa bagian yang perlu dipahami, yaitu :

- **Context** : Merupakan sebuah Object context yang ada pada tampilan Activity maupun Fragment.
 - Jika pemanggilan pada Activity, cukup menggunakan syntax **this**.
 - Jika pemanggilan pada Fragment dapat menggunakan syntax **requireContext()**.



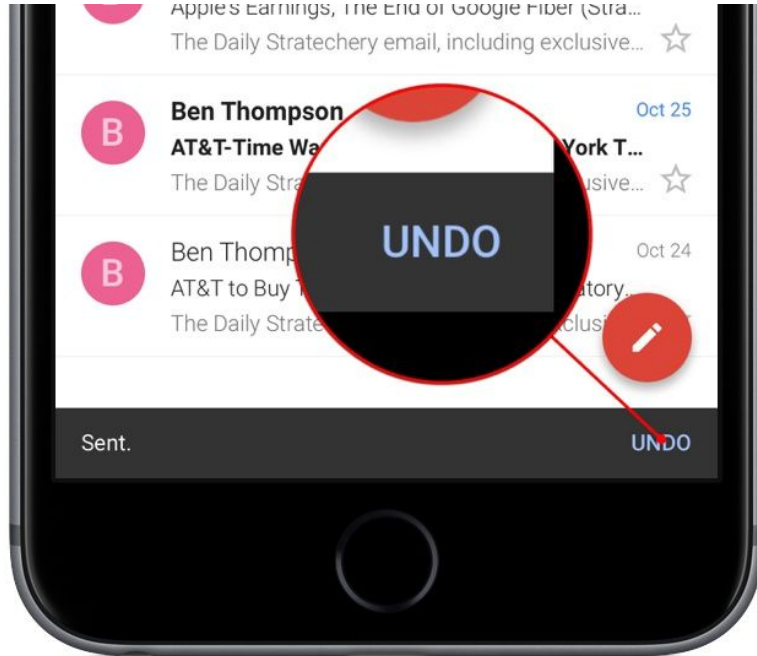
- **Toast Message** : String yang akan ditampilkan pada Toast).
- **Toast Duration** : Durasi tampilnya Toast. Ada dua pilihan, yaitu :
 - ***Toast.LENGTH_LONG*** berdurasi 3.5 detik,
 - ***Toast.LENGTH_SHORT*** berdurasi 2 detik.
- **Method Show** : Method akhir untuk menampilkan Toast.



Gimana? Simple kan bikinnya? Kaya manggang roti pake toaster itu~

Karena kayaknya pada bisa bikin Toast sambil merem, kita lanjut aja ya ke bahasan selanjutnya, yaitu ..

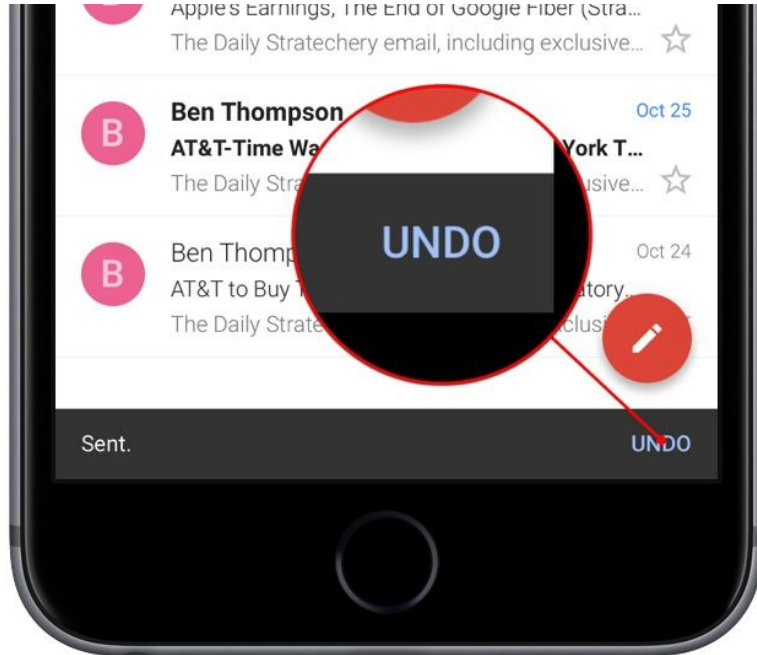
✨ **Snackbar** ✨



Jadi, apa itu Snackbar?

Snackbar itu hampir mirip kayak Toast, yaitu untuk **menampilkan informasi yang bisa hilang secara otomatis**.

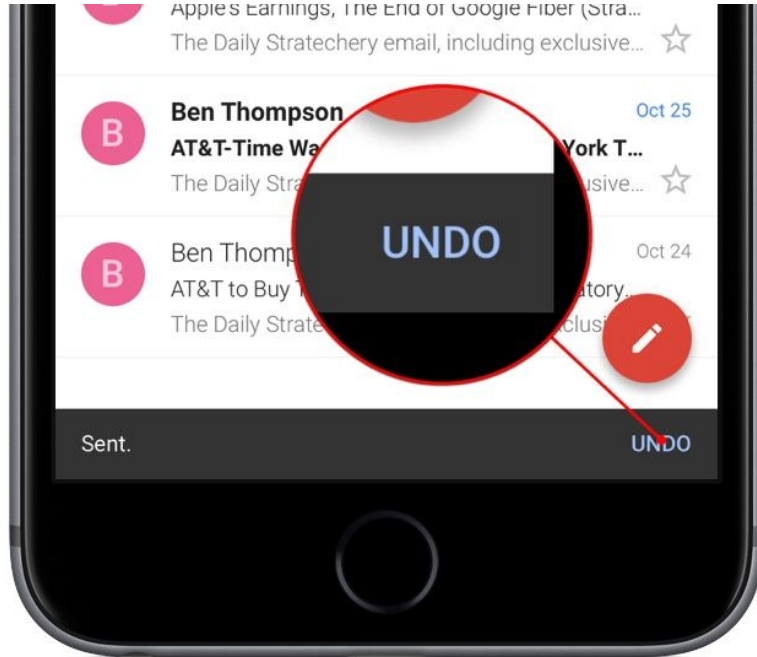
Contoh dari Snackbar yang sering kita temui adalah pada aplikasi Gmail. Misalnya, pas kita ngehapus email, mengirim email, ataupun mengarsipkan email, biasanya akan muncul Snackbar.



Nah kan sama-sama informasi yang bisa hilang secara otomatis tuh, terus bedanya sama Toast apa?

Ini dia yang kalian tunggu, suatu informasi bisa disebut Snackbar kalo punya kriteria gini :

- Menampilkan pesan informasi yang singkat, tidak terlalu panjang. Pesan yang disampaikan dapat dibaca user selama 2 sampai 3.5 Detik.



- Snackbar bisa tetap tampil **selamanya**.
- Snackbar bisa memiliki maksimal dua **tombol Action**.
- Snackbar biasanya menampilkan informasi yang kontekstual. Pesan Snackbar yang ditampilkan sesuai dengan apa yang tampil pada layar.



Udah keliatan bedanya kan?

Walaupun sama sama sementara, si SnackBar ini stay lebih lama. Kalo yang muncul terus ngilang gitu aja namanya si Toast. #sedih

Terus gimana nih caranya **membuat SnackBar** ini?



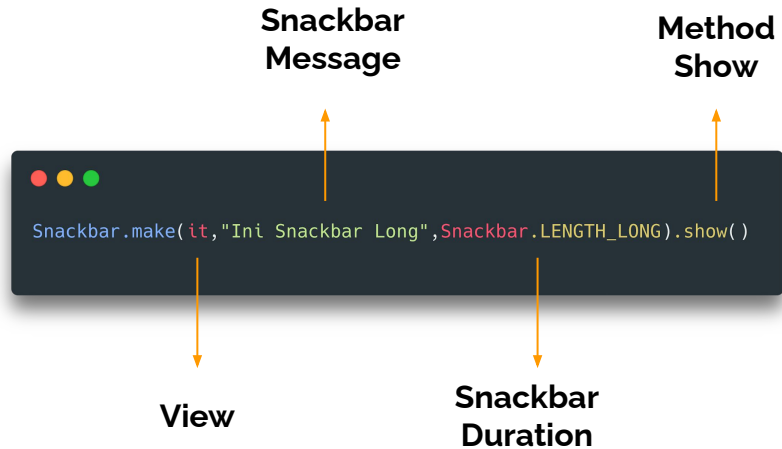
```
dependencies {  
    //other dependencies  
    implementation 'com.google.android.material:material:1.2.1'  
}
```

0) Tambahkan dependency dulu!

Snackbar itu bagian dari Library Material Design. Sedangkan dependency material design itu nggak otomatis ditambahkan ke build.gradle level app kita.

Jadi kita perlu nih nambahin **Library Material Design** pada build.gradle level module:app untuk menggunakan Snackbar ini.

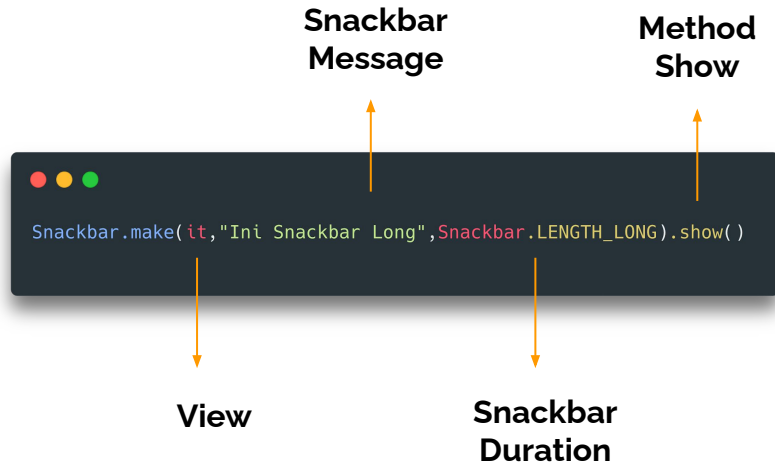
Kayak apa implementasinya? bisa diliat di gambar yes~



1) Ayo buat Snackbar!

Setelah kita nambahin dependency Library Material Design tadi, baru deh kita bisa buat si Snackbar ini!

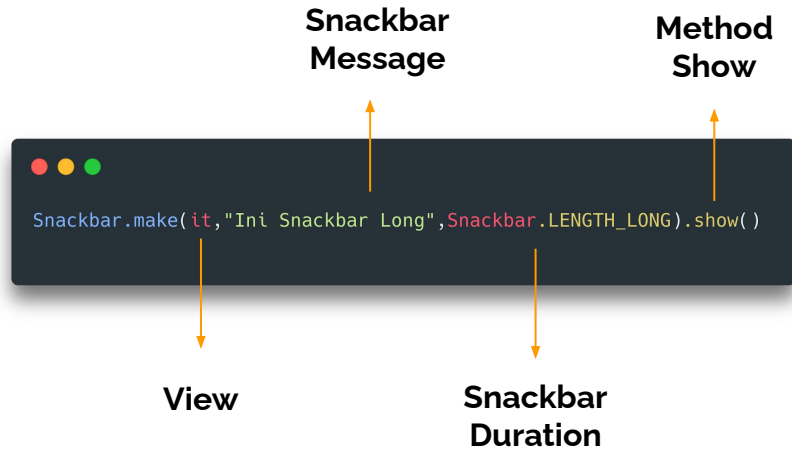
Kita bisa membuat Snackbar menggunakan method **make** dari class **Snackbar**.



Yepp, Snackbar ini juga punya beberapa bagian dalam method make yang perlu dipahami, yaitu:

- **View :** Object dari komponen view.

Kalo Snackbar dipanggil dari method `setOnClickListener`, bisa pake object view yang udah disediakan. Jadinya kita tinggal panggil **it**.



- **Snackbar Message** : Pesan yang mau ditampilkan.
- **Snackbar Duration** : Panjang durasi snackbar tampil. Ada beberapa pilihan.
 - ***Snackbar.LENGTH_SHORT*** = 2 Detik
 - ***Snackbar.LENGTH_LONG*** = 3.5 Detik
 - ***Snackbar.LENGTH_INDEFINITE*** = Snackbar akan tampil hingga menekan tombol Aksi atau ada Snackbar lain yang tampil.
- **Method Show**. Method Akhir untuk menampilkan Snackbar



2) Buat Snackbar dengan Tombol Aksi yang menampilkan Toast!

Nah kayak yang tadi kita bahas, Snackbar bisa memiliki Tombol Aksi. Dimana ketika ditekan akan menjalankan code yang bisa kita tentukan.

Contohnya ketika ditekan, akan menampilkan pesan Toast!

Lebih lengkapnya bisa cuss lihat di Gambar~

```
Snackbar.make(it,"Ini Snackbar Long dengan Action",Snackbar.LENGTH_LONG)
    .setAction("Text Action") {
        Toast.makeText(this, "Toast dari Action Snackbar",
            Toast.LENGTH_LONG).show()
    }
    .show()
```



```
val snackBar = SnackBar.make(it,"Ini SnackBar  
Indefinite. Gak bakal pergi kalo gak klik Dismiss /  
Ada SnackBar lainnya", SnackBar.LENGTH_INDEFINITE)  
  
snackBar.setAction("Dismiss"){  
    snackBar.dismiss()  
}  
  
snackBar.show()
```

3) Membuat SnackBar dengan Tombol Aksi Dismiss!

Mirip kayak membuat SnackBar di slide sebelumnya, bedanya di sini kita perlu menampung object SnackBar supaya bisa menjalankan method **dismiss**.

Untuk lebih lengkapnya bisa lihat di gambar yes~



Gampang kan implementasinya?

Kalau kamu masih bingung, bisa lihat contoh repository tentang cara implementasi Toast dan Snackbar di link ini ya :

<https://github.com/sennohananto/ToastSnackbar>



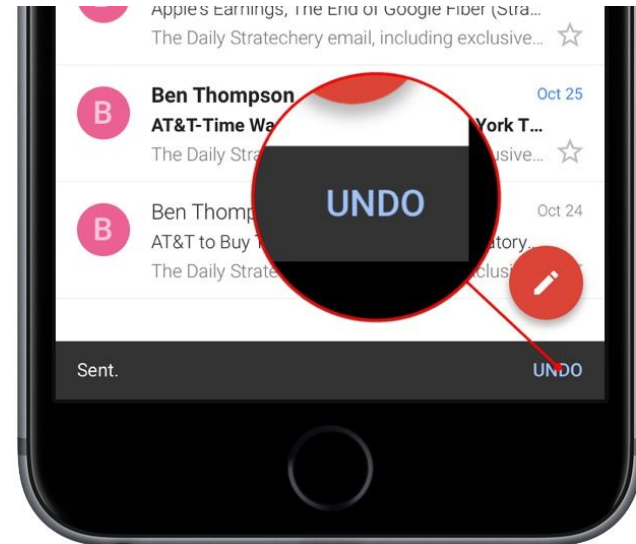
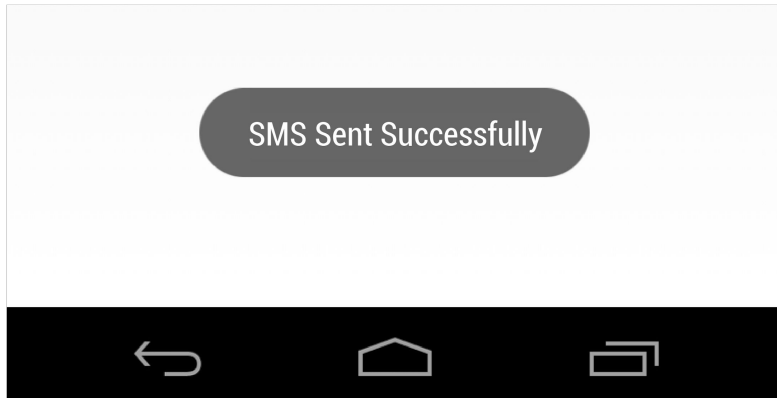
Selamat! Kamu udah berhasil membuat Toast dan Snackbar 🎉

Yippiee.. prakteknya udah, sekarang kita coba gali lebih dalam yuk **perbedaan antara Toast dan Snackbar**, supaya tau nih kapan pake Toast dan kapan pake Snackbar!



Kebayang Bedanya Toast dan Snackbar?

Sempet kita bahas diawal, walaupun sama-sama pesan sementara, keduanya punya perbedaan penting. Bedanya bisa dilihat dari segi tampilan maupun ketersediaan button.





Fitur	Toast	Snackbar
Fungsi	Menampilkan pesan informasi sementara	Menampilkan pesan informasi sementara
Durasi	LENGTH_SHORT LENGTH_LONG	LENGTH_SHORT LENGTH_LONG LENGTH_INDEFINITE
Tombol Aksi	Tidak mendukung Tombol aksi	Bisa mendukung Tombol aksi maksimal 2 Tombol



Dari tabel perbandingan gambar tadi, bisa disimpulkan kayak gini nih:

- Toast maupun Snackbar sama-sama bertujuan untuk **menampilkan pesan informasi yang sifatnya sementara**/tidak sangat penting dan sesuai dengan konteksnya.
- Kalo kita mau menampilkan pesan informasi tanpa membutuhkan aksi, mendingan pake Toast aja. Sebaliknya, kalo membutuhkan tombol aksi maka lebih baik menggunakan Snackbar.
- Kalo pesan informasi cukup panjang atau kalo dibaca kira-kira butuh waktu lebih dari 3.5 Detik, maka lebih baik menggunakan Snackbar dengan duration `LENGTH_INDEFINITE`.





Biar paham, yuk ikut Challenge berikut!

Buatlah suatu Aplikasi dengan Kriteria :

- Memiliki Satu EditText untuk mengisikan Nama
- Ada tombol untuk menampilkan toast pesan selamat datang disertai Nama dengan durasi 3.5 Detik.
- Ada tombol untuk menampilkan Snackbar dengan Text bebas, disertai dengan **dua tombol aksi** yang berbeda.

Pengumpulan tugas diserahkan kepada mentor.

Kita akan membahas bersama-sama di pertemuan berikutnya.

Saatnya kita
Quiz!





1. Berikut ini adalah komponen untuk menampilkan pesan yang berisi informasi yang akan tampil sementara, kecuali?

- A. Toast
- B. Snackbar
- C. Logcat



1. Berikut ini adalah komponen untuk menampilkan pesan yang berisi informasi yang akan tampil sementara, kecuali?

- A. Toast
- B. Snackbar
- C. Logcat

Logcat nggak akan menampilkan apa-apa ke user dan yang bisa menampilkan pesan sementara adalah Toast dan Snackbar!



2. Jika menggunakan `LENGTH_SHORT`, berapa durasi yang akan ditampilkan?

- A. 1 detik
- B. 2 detik
- C. 3,5 detik



2. Jika menggunakan `LENGTH_SHORT`, berapa durasi yang akan ditampilkan?

- A. 1 detik
- B. 2 detik
- C. 3,5 detik

Betul banget! Panjang durasi dari `LENGTH_SHORT` adalah selama 2 detik. Pastikan pesan yang kamu tampilkan bisa dibaca hanya dalam waktu 2 detik yaa~



3. Berikut ini adalah beberapa contoh penggunaan Toast dalam aplikasi, kecuali

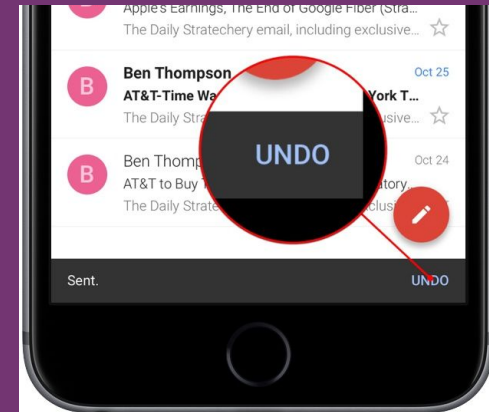
- A. Jika SMS terkirim,
- B. Koneksi gagal, dan
- C. Setelah mengirim pesan dalam Gmail



3. Berikut ini adalah beberapa contoh penggunaan Toast dalam aplikasi, kecuali

- A. Jika SMS terkirim,
- B. Koneksi gagal, dan
- C. Setelah mengirim pesan dalam Gmail

Kalau kamu ngirim pesan lewat Gmail di HP Android, pesan yang muncul itu Snackbar yah, bukan Toast. Contohnya digambar berikut.





4. Berikut ini adalah beberapa hal yang dimiliki oleh Snackbar, namun tidak dimiliki Toast, kecuali?

- A. LENGTH_LONG
- B. Tombol Aksi
- C. LENGTH_INDEFINITE



4. Berikut ini adalah beberapa hal yang dimiliki oleh Snackbar, namun tidak dimiliki Toast, kecuali?

- A. LENGTH_LONG
- B. Tombol Aksi
- C. LENGTH_INDEFINITE

Betuls! Durasi LENGTH_LONG dimiliki oleh Toast maupun Snackbar. Sementara fitur Tombol aksi dan Durasi LENGTH_INDEFINITE hanya dimiliki oleh Snackbar saja!



5. Apa yang terjadi jika Toast atau Snackbar tidak diakhiri dengan `method show()` ?

- A. Toast/Snackbar tidak tampil
- B. Toast/Snackbar jadi transparan
- C. Aplikasi Error



5. Apa yang terjadi jika Toast atau Snackbar tidak diakhiri dengan `method show()` ?

- A. Toast/Snackbar tidak tampil
- B. Toast/Snackbar jadi transparan
- C. Aplikasi Error

Bener bingits! Jika kita tidak memanggil method `show()`, maka Toast maupun Snackbar tidak akan tampil. Nggak bikin aplikasi error kok :)



Referensi dan bacaan lebih lanjut~

1. [Snackbars - Material Design](#)
2. [Toasts overview | Android Developers](#)
3. [Snackbar | Android Developers](#)





Nah, selesai sudah pembahasan kita di Chapter 4 Topic 1 ini.

Selanjutnya, kita bakal belajar tata cara **pembuatan Dialog di Android.**

Penasaran kayak gimana? Cus langsung ke topik selanjutnya~



Terima Kasih!



Next Topic

loading...