

MOBILE APPLICATION

Pertemuan 6

Collections & Documents

- Firebase dan beberapa DBMS lainnya menerapkan sistem collections sebagai model penyimpanan datanya
- Berbeda dengan konsep biasanya yang menggunakan baris dan kolom, model ini meniru teknik dokumen yang biasa dibuat oleh orang

Lanjutan

- Setiap dokumen terdiri dari satu set pasangan *key-value*. Sehingga Firebase dioptimalkan untuk menyimpan koleksi yang sangat banyak dari dokumen-dokumen kecil
- Ilustrasi -> Slide Selanjutnya



This Photo by Unknown author is licensed under CC BY.

1 Dokumen = 1 Buku

1 Koleksi = 1 Rak

1 Database = 1 Perpustakaan

Dokumen

- Satuan terkecil dalam database NoSQL/Firebase
- Berisi kolom, yang dipetakan ke nilai. Setiap dokumen diidentifikasi dengan nama.

 alovelace

first : "Ada"

last : "Lovelace"

born : 1815

 alovelace

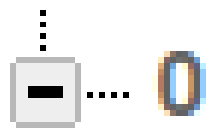
name :

first : "Ada"

last : "Lovelace"

born : 1815

Contoh: Firebase Console



```
.....  
..... email: "maulanahirzan@gmail.com"  
.....  
..... nama: "Alauddin Maulana Hirzan"  
.....  
..... password: "123"
```

JSON?

- Ada yang mungkin memperhatikan bahwa dokumen sangat mirip dengan JSON.
- Faktanya, memang dasarnya adalah JSOPN.
- Ada beberapa perbedaan (misalnya, dokumen mendukung tipe data tambahan dan ukurannya dibatasi hingga 1 MB), tetapi secara umum, dokumen adalah JSON versi ringan.

Koleksi

- Dokumen disimpan dalam koleksi, yang hanya merupakan wadah untuk dokumen. Misalnya, pengembang dapat memiliki koleksi *user* untuk menampung berbagai informasi *user*, di masing-masing dokumen:

```
users
  alovelace
    first : "Ada"
    last  : "Lovelace"
    born  : 1815
  aturing
    first : "Alan"
    last  : "Turing"
    born  : 1912
```


Skema Koleksi dan Dokumen

- Cloud Firestore tidak memiliki skema, sehingga pengembang memiliki kebebasan penuh atas jenis data yang dimasukkan ke dalam setiap dokumen.
- Namun, sebaiknya gunakan field dan tipe data yang sama di beberapa dokumen, sehingga dapat membuat kueri dokumen dengan lebih mudah.

Lanjutan

- Nama dokumen dalam koleksi bersifat unik. programmer dapat memberikan kunci sendiri, seperti ID pengguna, atau Cloud Firestore membuat ID acak untuk secara otomatis.
- Programmer tidak perlu "membuat" atau "menghapus" koleksi. Setelah membuat dokumen pertama dalam koleksi, koleksi akan ada. Jika menghapus semua dokumen dalam koleksi, itu tidak ada lagi.

Teknik Query

- Untuk melakukan perubahan data atau menghapus nya, bisa dilakukan dengan cara referensi dokumen yang mau diubah.

```
val alovelaceDocumentRef = db.collection("users").document("alovelace")
```

DocSnippets.kt [↗](#)

```
val usersCollectionRef = db.collection("users")
```

DocSnippets.kt [↗](#)

Data Hierarki

- Coba perhatikan gambar berikut:
- Bagaimana cara menyimpan pesan?.
- Pesan sebaiknya tidak disimpan di dokumen, karena dokumen harus sederhana

 rooms

 roomA

name : "my chat room"

 roomB

...

Subkoleksi

- Untuk mengatasi hal itu, subkoleksi bisa dibuat untuk menyimpan pesan-pesan tadi.
- Di dalam subkoleksi pesan-pesan berisikan dokumen yang berisikan pesan singkat

rooms → Koleksi Utama

roomA → Dokumen 1

name : "my chat room"

messages → SubKoleksi Pesan

message1 → Pesan

from : "alex"

msg : "Hello World!"

message2

...

roomB

Referensi Database

- Sehingga jika direferensikan dokumen tersebut menjadi

```
val messageRef = db
    .collection("rooms").document("roomA")
    .collection("messages").document("message1")
```

DocSnippets.kt [↗](#)

Terima Kasih
