



kafka



# KAFKA X GOLANG

## DISTRIBUTED STREAMING



# Apa itu Apache Kafka?

Apache Kafka adalah platform streaming terdistribusi yang dapat memproses stream data secara real-time. Kafka umumnya digunakan untuk membangun real-time data pipelines dan streaming apps.

## Kelebihan Kafka Apaan ya?

1. Mampu menangani jutaan pesan per detik.
2. Menyimpan data secara replication untuk memastikan keandalan.
3. Bersifat Distribusi.
4. Dengan cepat menambah broker untuk menangani beban lebih besar.

# Konsep pada Kafka

**Producer:** Aplikasi yang berfungsi untuk mengirimkan pesan atau data pada kafka (publisher).

**Consumer:** Aplikasi yang berfungsi untuk menerima atau membaca pesan atau data dari Kafka (subscribe).

**Topic:** Topik sendiri adalah tempat pesan ataupun data yang akan di publish

# Implementasi Kafka dengan GIN

Arsitektur implementasi Kafka dalam aplikasi Gin akan seperti berikut:

Step 1

Buatkan service untuk A yang berperan sebagai Producer:

- Memproduksi atau mengirim pesan ke kafka

Step 2

Buatkan service untuk B yang berperan sebagai Consumer 1:

- Menerima atau mengolah pesan dari kafka

Step 3

Buatkan service untuk C yang berperan sebagai Consumer 2:

- Menerima atau mengolah pesan dari kafka

Step 4

Buatkan service untuk D yang berperan sebagai Consumer 3:

- Menerima atau mengolah pesan dari kafka

GO-KAFKA/

```
|─ kafka-consumer-1/
|   |─ Dockerfile
|   |─ go.mod
|   |─ go.sum
|   └─ main.go
|─ kafka-consumer-2/
|   |─ Dockerfile
|   |─ go.mod
|   |─ go.sum
|   └─ main.go
|─ kafka-consumer-3/
|   |─ Dockerfile
|   |─ go.mod
|   |─ go.sum
|   └─ main.go
|─ kafka-producer/
|   |─ Dockerfile
|   |─ go.mod
|   |─ go.sum
|   └─ main.go
|─ docker-compose.yaml
└─ README.md
```

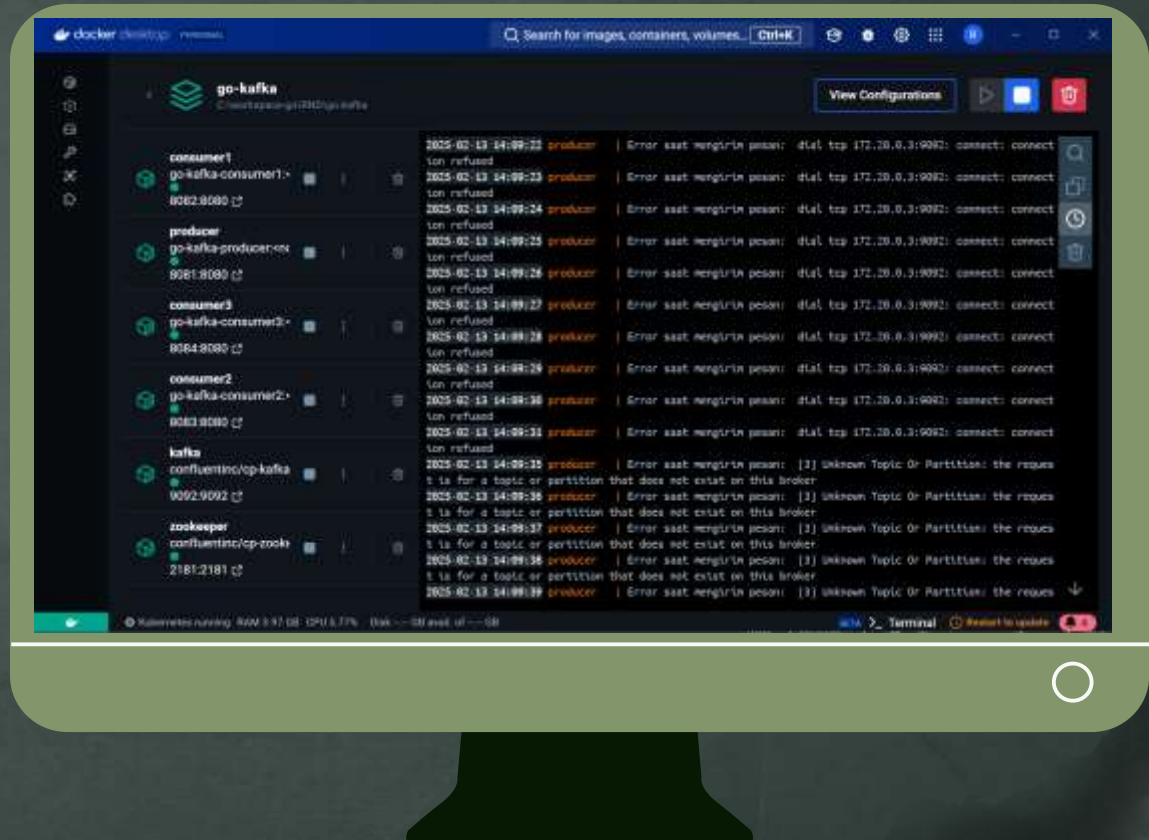


# Integrasi dengan Docker

Jika Kalian ingin menjalankan Kafka, Producer, dan Consumer secara *containerized*, Kalian bisa menggunakan **docker-compose** dengan contoh struktur berikut:



# Berikut hasil yang telah dibaca oleh Docker desktop.



# Source Code :

<https://bit.ly/bws-kafka-golang-gin>



**SEE YOU ...**

