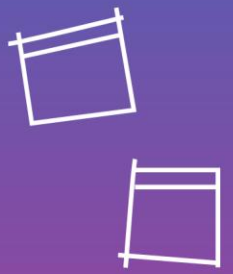
DBC
DIGITAL BUSINESS COMPANY®

VEM SER
DBC



Apache Kafka

Aula 2 – Objetos

Lidando com Objetos

- Para enviar um objeto, basta transformá-lo em JSON

```
public void sendMessage(TesteDTO testeDTO) throws JsonProcessingException {
    String payload = objectMapper.writeValueAsString(testeDTO);
    Message<String> message = MessageBuilder.withPayload(payload)
            .setHeader(KafkaHeaders.TOPIC, topicDTO)
            .setHeader(KafkaHeaders.MESSAGE_KEY, UUID.randomUUID().toString())
            .build();
```

Lidando com Objetos

- Para recuperar um objeto, converter a String em Objeto

```
@KafkaListener(
        topics = "${kafka.topic.dto}",
        groupId = "${kafka.group-id.latest}",
        containerFactory = "listenerContainerFactory")
public void consume(@Payload String message,
                    @Header(KafkaHeaders.RECEIVED_MESSAGE_KEY) String key,
                    @Header(KafkaHeaders.OFFSET) Long offset) throws JsonProcessingException {
    TesteDTO dto = objectMapper.readValue(message, TesteDTO.class);
    log.info("#### offset -> '{}' key -> '{}' -> Consumed Object message -> '{}'  ", offset, key, dto);
}
```

DBC
DIGITAL BUSINESS COMPANY®

show me your code;

VEM SER
DBC

Exercício #1

O objetivo desse exercício é consumir e produzir mensagens de sensores para um veículo que está se locomovendo

- Criar um objeto com os seguintes atributos:
  - **dataLeitura**: LocalDateTime
  - **velocidade**: Double
  - **rotacao**: Integer
  - **sensorFrenagem**: boolean
- Criar um projeto “veiculo-produtor-consumidor” que deve ter um endpoint que produzirá essas mensagens no tópico “sensor-veiculo”
- No mesmo projeto, deve ter um consumidor que irá consumir essa mensagem e salvar no mongoDB a mensagem
- Deve ter um endpoint capaz de retornar todos os dados lidos
- Deve ter um endpoint capaz de retornar:
  - Velocidade média do veículo (soma de todas as velocidades / quantidade total)
  - Rotação média do veículo (soma de todas as rotações / quantidade total)
  - Quantidade de leituras

Homework

O objetivo desse exercício é construir um chat utilizando o apache Kafka, esse chat deve enviar e receber mensagens e também enviar mensagens privadas para todos os participantes do grupo

- Criar um novo projeto com o spring initializr com no nome de “chat-kafka”
- Colocar o projeto em “modulo 4.3”
- Desenvolver um produtor que produza mensagens com objeto contendo os seguintes atributos:
  - usuario: String (usuário que mandou a mensagem)
  - mensagem: String (mensagem a ser enviada ao usuário)
  - dataCriacao: LocalDateTime (data atual)

(TEM QUE TER OS MESMOS NOMES DE VARIÁVEIS E TIPOS DE DADOS)

- O tópico geral será “chat-geral”
- Cada usuário deverá ter seu próprio tópico (para caso queira mandar mensagem privada)
  - Os tópicos privados deverão ser exemplo: “chat-jean”, “chat-bruno”, “chat-maicon"

Homework

- Desenvolver um serviço (API) capaz de produzir mensagens para esses tópicos.
  - Essa api deve ser capaz de enviar mensagens para o geral e para todos os tópicos de usuários
  - Deve ser possível mandar uma mesma mensagem para quantos usuários quiser em uma MESMA REQUEST...
- Desenvolver 2 consumidores sendo:
  - um consumidor para o tópico “chat-geral”
    - Esse consumidor deve receber o objeto e imprimir em formato
      “DIA/MÊS/ANO HORA:MINUTO:SEGUNDO [USUARIO]: MENSAGEM”
  - um consumidor para o tópico “chat-seunome”
    - Esse consumidor deve receber o objeto e imprimir em formato
      “DIA/MÊS/ANO HORA:MINUTO:SEGUNDO [USUARIO] (privada): MENSAGEM”
  - O **group-id** e **clientId** será o seu nome para ambos os tópicos
- Os tópicos deverão ser criados em: **vemser-chat.servebeer.com:9092**