

ДОКУМЕНТАЦИЈА

Домашна задача 1 – Apache Flink и Apache Kafka

1. Опис на задачата

Целта на домашната задача е да се развие Apache Flink апликација која во реално време ги процесира пораките испратени на Apache Kafka topic-от **sensors**. Секоја порака е JSON објект што содржи **key**, **value** и **timestamp**. Задачата бара дефинирање на временски прозорци и испраќање агрегирани резултати на два други Kafka topics: **results1** и **results2**.

2. Функционалност на Flink апликацијата

Flink апликацијата ги чита пораките од **sensors** и ги парсира во објекти со key, value и time. Потоа се применува групирање по key и дефинирање временски прозорци со големина 8000 ms и лизгање 6000 ms и во двете задачи(во мојот случај користам sliding windows).

2.1 Резултати во results1

За секој **key** во одреден **временски прозорец** се испраќа бројот на пораки што пристигнале.

Формат:

```
[ "window_start": "window_end", Num ("key") == "count"]
```

2.2 Резултати на results2

На вториот топик се испраќаат агрегирани статистики: минимум, максимум, просек и број на елементи.

Формат:

```
{ "key": "A", "window_start":1764555456000, "window_end":1764555464000, "min_value":191, "count":3, "average":341.00, "max_value": 626 }
```

3. Тестирање

На крај, со Python скрипта consume-messages.py која успешно се поврзува на results1 и results2 се прикажуваат резултатите во реално време.

На следните слики може да се види:

3.2 Задача 1 - Consumer (consume-messages.py)

```
Received message: [ 1764554148000 : 1764554156000 ], Num(C) = 2
Received message: [ 1764554148000 : 1764554156000 ], Num(A) = 2
Received message: [ 1764554148000 : 1764554156000 ], Num(D) = 1
Received message: [ 1764554148000 : 1764554156000 ], Num(B) = 2
Received message: [ 1764554154000 : 1764554162000 ], Num(B) = 2
Received message: [ 1764554154000 : 1764554162000 ], Num(C) = 1
Received message: [ 1764554154000 : 1764554162000 ], Num(D) = 2
Received message: [ 1764554154000 : 1764554162000 ], Num(A) = 2
Received message: [ 1764554160000 : 1764554168000 ], Num(A) = 3
Received message: [ 1764554160000 : 1764554168000 ], Num(D) = 2
Received message: [ 1764554160000 : 1764554168000 ], Num(C) = 2
Received message: [ 1764554166000 : 1764554174000 ], Num(D) = 3
Received message: [ 1764554166000 : 1764554174000 ], Num(A) = 1
Received message: [ 1764554166000 : 1764554174000 ], Num(C) = 1
Received message: [ 1764554166000 : 1764554174000 ], Num(B) = 1
Received message: [ 1764554172000 : 1764554180000 ], Num(A) = 3
Received message: [ 1764554172000 : 1764554180000 ], Num(D) = 1
Received message: [ 1764554172000 : 1764554180000 ], Num(C) = 1
Received message: [ 1764554172000 : 1764554180000 ], Num(B) = 1
Received message: [ 1764554178000 : 1764554186000 ], Num(B) = 3
Received message: [ 1764554178000 : 1764554186000 ], Num(A) = 2
Received message: [ 1764554178000 : 1764554186000 ], Num(D) = 1
Received message: [ 1764554184000 : 1764554192000 ], Num(B) = 2
Received message: [ 1764554184000 : 1764554192000 ], Num(C) = 1
Received message: [ 1764554184000 : 1764554192000 ], Num(A) = 1
Received message: [ 1764554184000 : 1764554192000 ], Num(D) = 1
Received message: [ 1764554190000 : 1764554198000 ], Num(A) = 2
Received message: [ 1764554190000 : 1764554198000 ], Num(C) = 4
Received message: [ 1764554196000 : 1764554204000 ], Num(C) = 1
Received message: [ 1764554196000 : 1764554204000 ], Num(D) = 2
Received message: [ 1764554196000 : 1764554204000 ], Num(B) = 2
```

3.3 Задача 2 - Consumer (consume-messages.py)

```
Received message: { "key":"A", "window_start":1764555450000, "window_end":1764555458000, "min_value":861, "count":1, "average":861.00, "max_value":861 }
Received message: { "key":"D", "window_start":1764555450000, "window_end":1764555458000, "min_value":485, "count":2, "average":554.00, "max_value":623 }
Received message: { "key":"B", "window_start":1764555450000, "window_end":1764555458000, "min_value":817, "count":1, "average":817.00, "max_value":817 }
Received message: { "key":"C", "window_start":1764555450000, "window_end":1764555458000, "min_value":48, "count":2, "average":440.50, "max_value":833 }
Received message: { "key":"C", "window_start":1764555456000, "window_end":1764555464000, "min_value":182, "count":1, "average":182.00, "max_value":182 }
Received message: { "key":"D", "window_start":1764555456000, "window_end":1764555464000, "min_value":56, "count":3, "average":317.00, "max_value":485 }
Received message: { "key":"A", "window_start":1764555456000, "window_end":1764555464000, "min_value":191, "count":3, "average":341.00, "max_value":626 }
Received message: { "key":"A", "window_start":1764555462000, "window_end":1764555470000, "min_value":289, "count":3, "average":557.33, "max_value":755 }
Received message: { "key":"C", "window_start":1764555462000, "window_end":1764555470000, "min_value":182, "count":1, "average":182.00, "max_value":182 }
Received message: { "key":"B", "window_start":1764555462000, "window_end":1764555470000, "min_value":347, "count":2, "average":439.50, "max_value":532 }
Received message: { "key":"B", "window_start":1764555468000, "window_end":1764555476000, "min_value":275, "count":3, "average":421.67, "max_value":643 }
Received message: { "key":"C", "window_start":1764555468000, "window_end":1764555476000, "min_value":80, "count":2, "average":438.00, "max_value":796 }
Received message: { "key":"A", "window_start":1764555468000, "window_end":1764555476000, "min_value":755, "count":1, "average":755.00, "max_value":755 }
```

На оваа слика се прикажани резултатите што ги прима Python consumer-от од Flink апликацијата преку Kafka topics results1 и results2.

Изработила: Симона Ристовска 221003