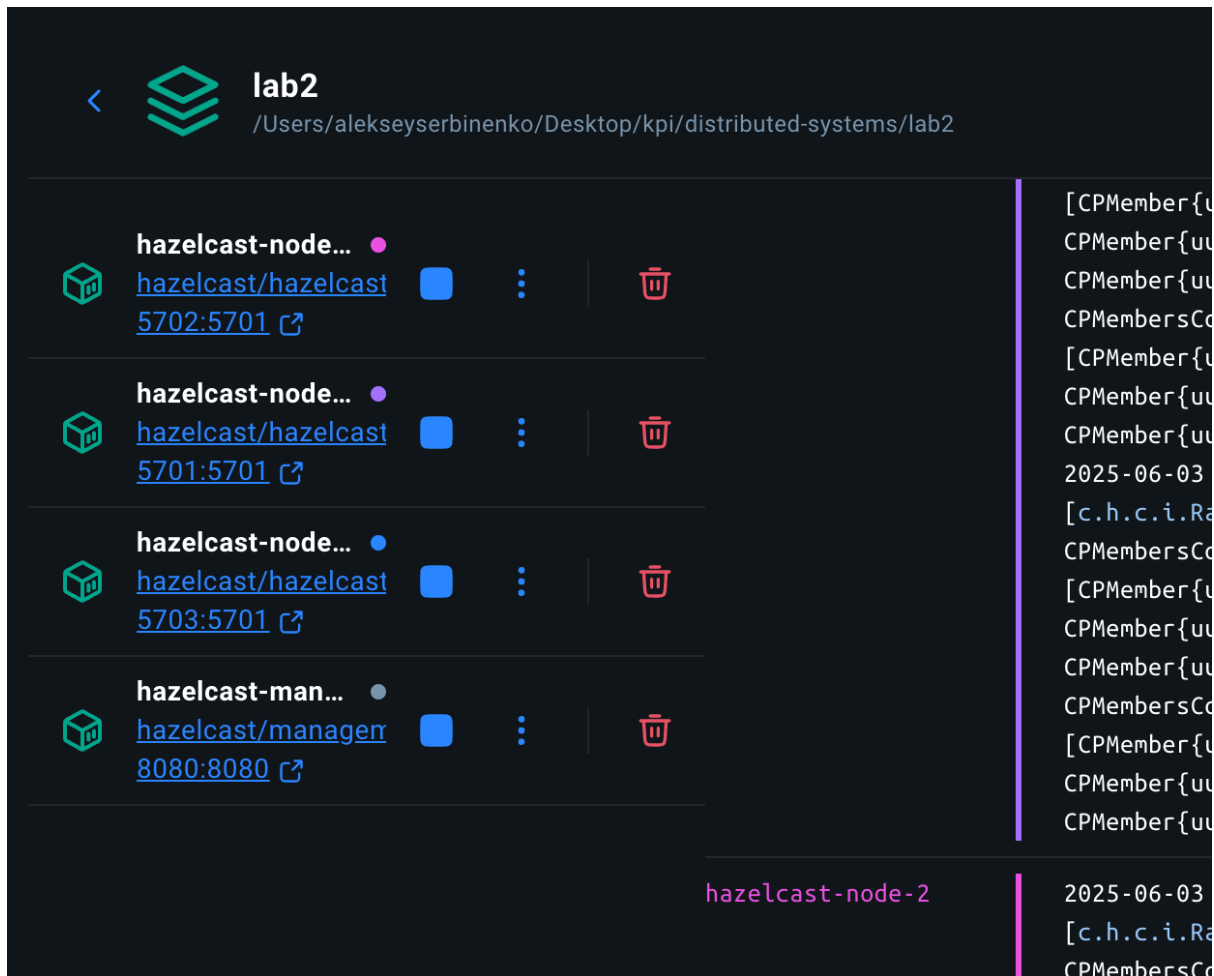
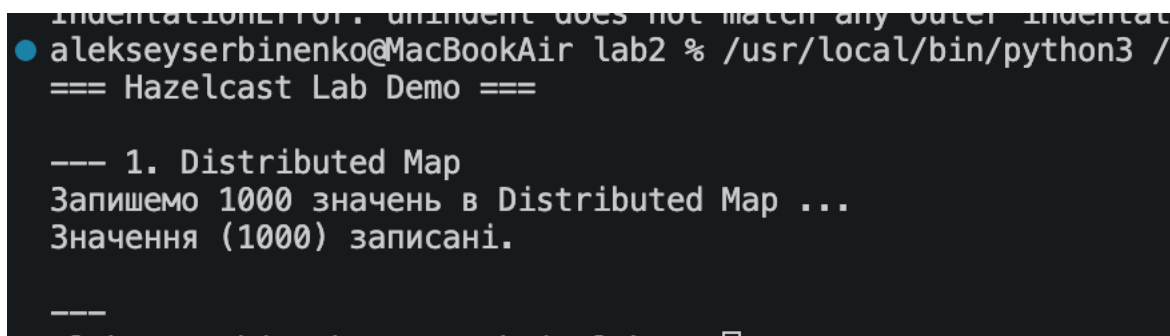


Lab2 Сербіненко Олексій фб-42мп

Запустимо ноди та менедж центр



Distributed maps



Map Statistics (In-Memory Format: BINARY)

RESET TIME

1 minute ago → now

Default View

Member ^	^ Entries	^ Gets	^ Puts	^ Removals	^ Sets	^
host.docker.inte...	339	0	0	0	0	
host.docker.inte...	333	0	1 000	0	0	
host.docker.inte...	328	0	0	0	0	
TOTAL	1 000	0	1 000	0	0	

1 – 3 of 3 Rows 10

Відключу 1 ноду

5702:5701 ↗

hazelcast-nod... ○

hazelcast/hazelcast

5701:5701

▶

⋮

🗑️

hazelcast-node... ●

hazelcast/hazelcast

▶

⋮

🗑️

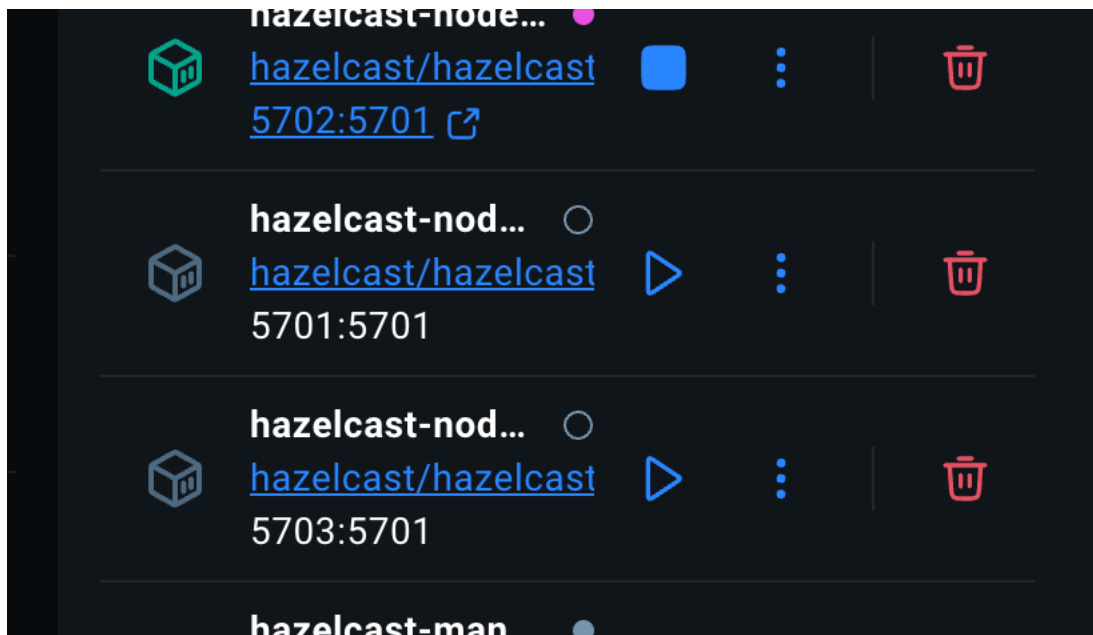
Значення збережені

Map Statistics (In-Memory Format: BINARY)

Member ^	^ Entries	^ Gets
host.docker.inte...	510	0
host.docker.inte...	490	0
TOTAL	1 000	0

1 – 2 of 2 Rows

Відключив ще одну ноду



Member ^	Entries ^	Gets ^	Puts ^
host.docker.inte...	1 000	0	1 000
TOTAL	1 000	0	1 000

1 - 1 of 1 Rows 10 ▾

Ввімкнув ноди  
та земулював падіння серверів

```
alekseyserbinenko@MacBookAir lab2 % docker kill lab2-hazelcast-node-2-1 lab2-hazelcast-node-1-1  
lab2-hazelcast-node-2-1  
lab2-hazelcast-node-1-1
```

Бачу що залишилось 653

## Map Statistics (In-Memory Format: BINARY)

Member ^	^ Entries	^ Gets
host.docker.inte...	653	0
TOTAL	653	0

1 – 1 c

Запустив знову але дані не повернулися

Map Statistics (In-Memory Format: BINARY)

Member ^	^ Entries	
host.docker.inte...	154	
host.docker.inte...	316	
host.docker.inte...	183	
TOTAL	653	

2.

в No lock ми отримали очікувану втрату даних

Для оптимістичного та песимістичного блокування втрат даних вже немає. Оптимістичне блокування спрацювало швидше.

--- 2. Демонстрація конкурентного інкремента в Distributed Map

Start test: No lock

Результат теста 'No lock': фінальне значення = 12065, час виконання = 12.07 секунд

Очікувана втрата даних

Start test: Optimistic lock

Результат теста 'Optimistic lock': фінальне значення = 30000, час виконання = 34.28 секунд

Дані коректні.

```
Start test: Pessimistic lock
Результат теста 'Pessimistic lock': фінальне значення = 30000, час виконання = 69.51 секунд
Дані коректні.
```

## Bounded queue

### --- 3. Bounded Queue

Продюсер: починаємо вставляти значення от 1 до 100...

Продюсер намагається додати: 1

Консюмер 1: запускає читання з черги...

Консюмер 2: запускає читання з черги...

Продюсер додав: 1

Консюмер 1 отримав: 1

Продюсер намагається додати: 2

Консюмер 2 отримав: 2

Продюсер додав: 98

Консюмер 2 отримав: 98

Продюсер намагається додати: 99

Продюсер додав: 99

Консюмер 1 отримав: 99

Продюсер намагається додати: 100

Продюсер додав: 100

Консюмер 2 отримав: 100

Продюсер: завершив вставку значень.

Консюмер 2: черга порожня, завершую роботу.

Консюмер 1: черга порожня, завершую роботу.

Демонстрація Bounded Queue завершена.