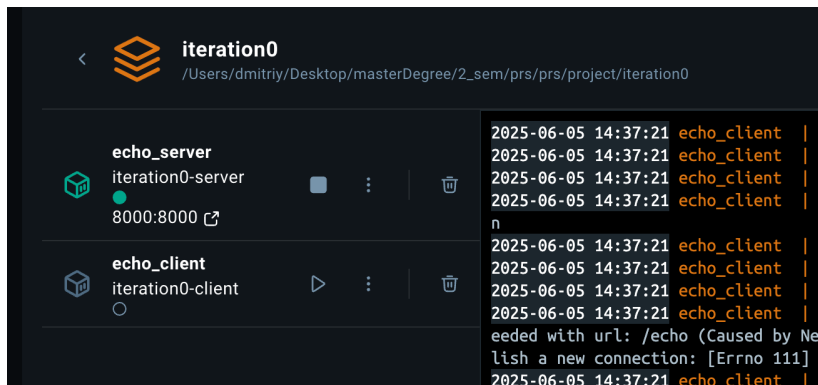


Рябко Дмитро ФБ-42мп

Replicated Log Project

Iteration 0:



```
2025-06-05 14:38:28 echo_client | 2025-06-05 11:38:28,571 - INFO - Sending to http://server:8000/echo: {'message': 'Hello, this is a test echo!', 'timestamp': 1749123508.5715437}
2025-06-05 14:38:28 echo_client | 2025-06-05 11:38:28,589 - INFO - Received response: {'echo': {'message': 'Hello, this is a test echo!', 'timestamp': 1749123508.5715437}}
```

Client – це звичайни python код, який шле запит:

```
logging.basicConfig(level=logging.INFO, format="%(asctime)s - %(levelname)s - %(message)s")
logger = logging.getLogger(__name__)

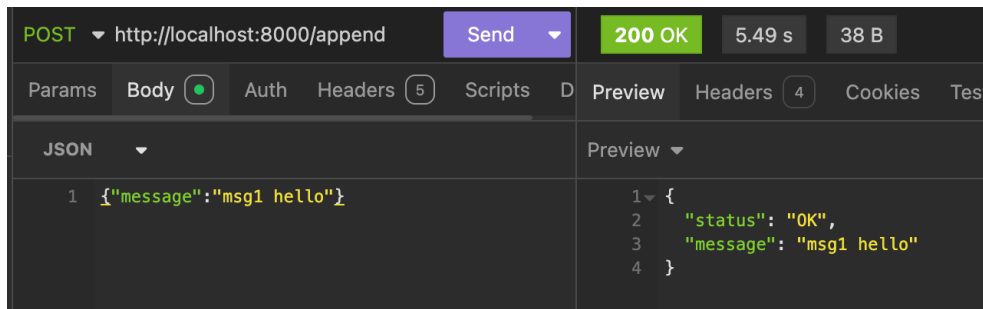
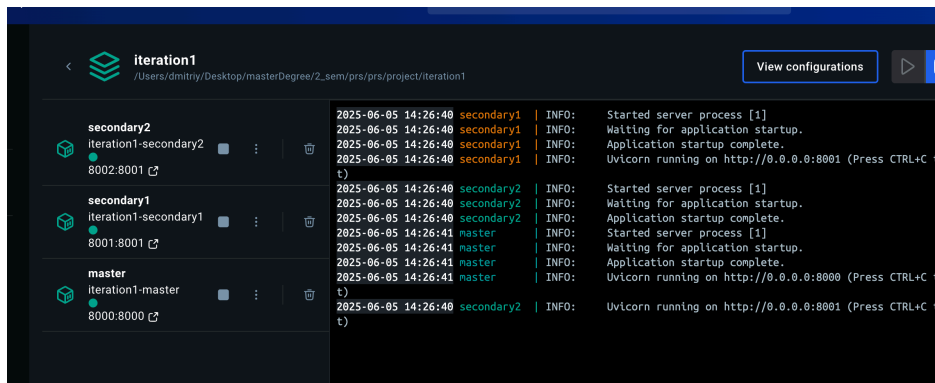
SERVER_URL = os.getenv("SERVER_URL", "http://server:8000")

def main():
    test_data = {"message": "Hello, this is a test echo!", "timestamp": time.time()}
    logger.info(f"Sending to {SERVER_URL}/echo: {test_data}")
    resp = requests.post(f"{SERVER_URL}/echo", json=test_data)
    if resp.ok:
        logger.info(f"Received response: {resp.json()}")
    else:
        logger.error(f"Server error: {resp.status_code} {resp.text}")

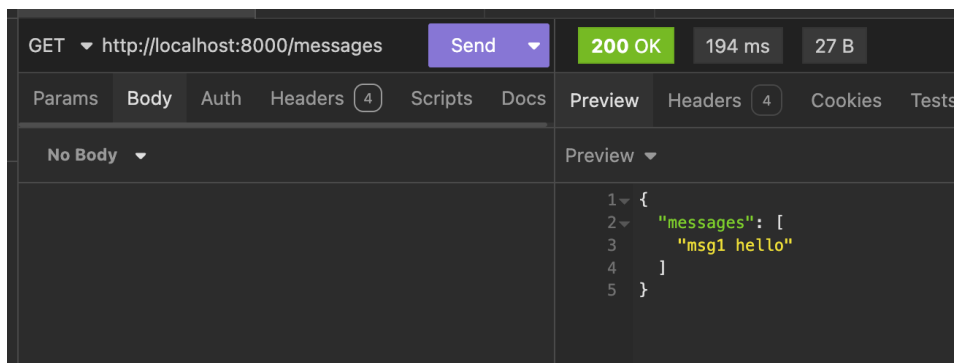
if __name__ == "__main__":
    main()
```

Iteration 1:

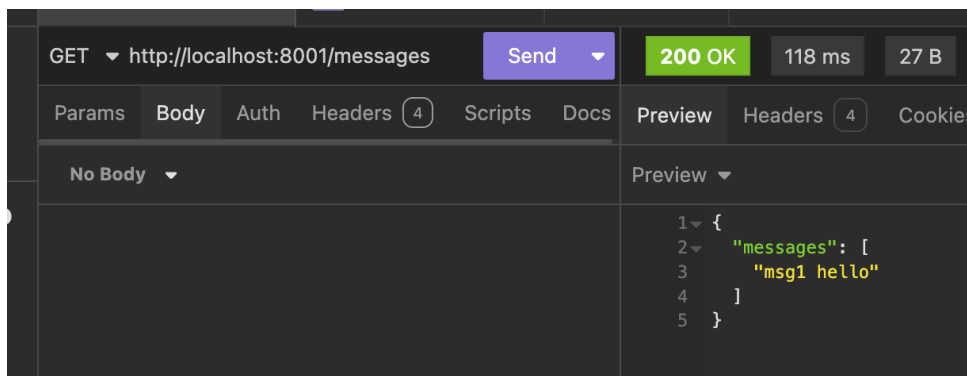
```
dmitriy@dmitriys-MacBook-Air iteration1 % docker-compose up -d
WARN[0000] /Users/dmitriy/Desktop/masterDegree/2_sem/prs/prs/project/iteration1/docker-compose.yml: the attribute `version`
obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Running 0/0
[+] Running 0/2ndary2 Building                                0.1s
[+] Running 0/21s (9/13)                                     0.2s
[+] Building 0.3s (10/13)                                     docker:desktop-linux
[+] Building 7.4s (13/15)                                     docker:desktop-linux
[+] Building 8.8s (19/22)                                     docker:desktop-linux
=> => transferring context: 1.03kB                                0.4s
```



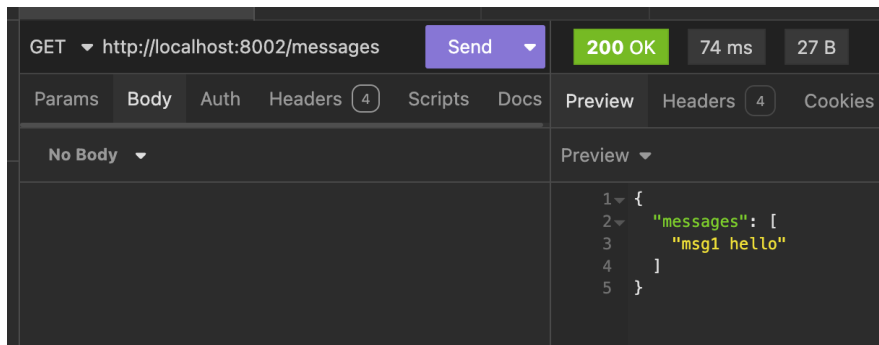
Master:



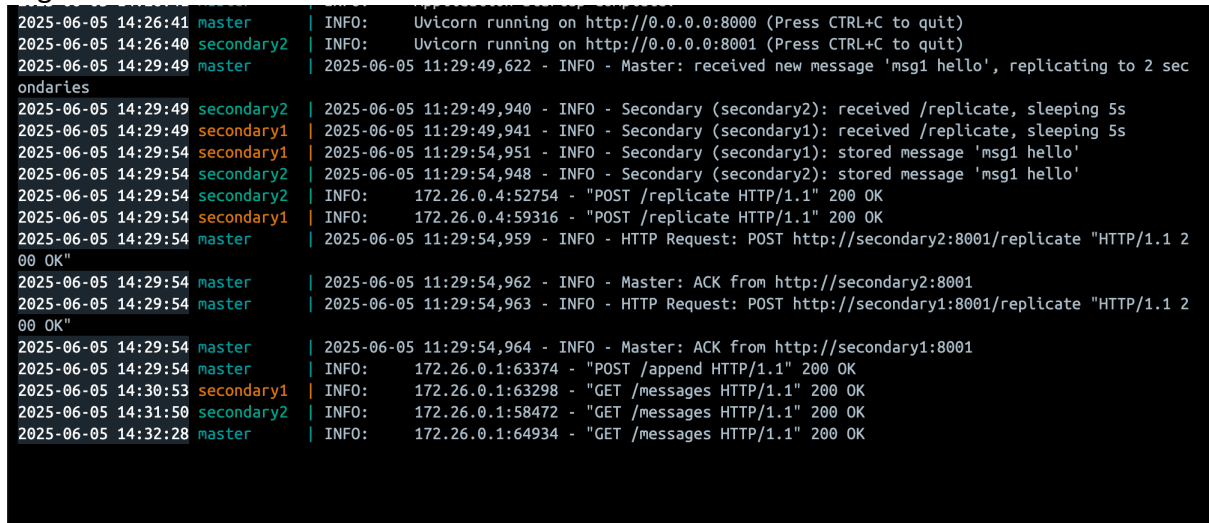
Secondary 1:



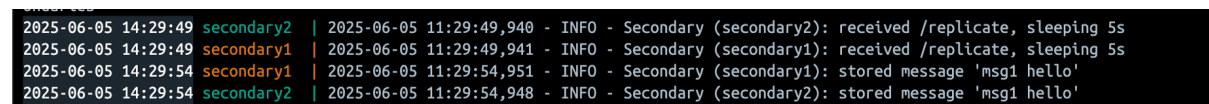
Secondary 2:



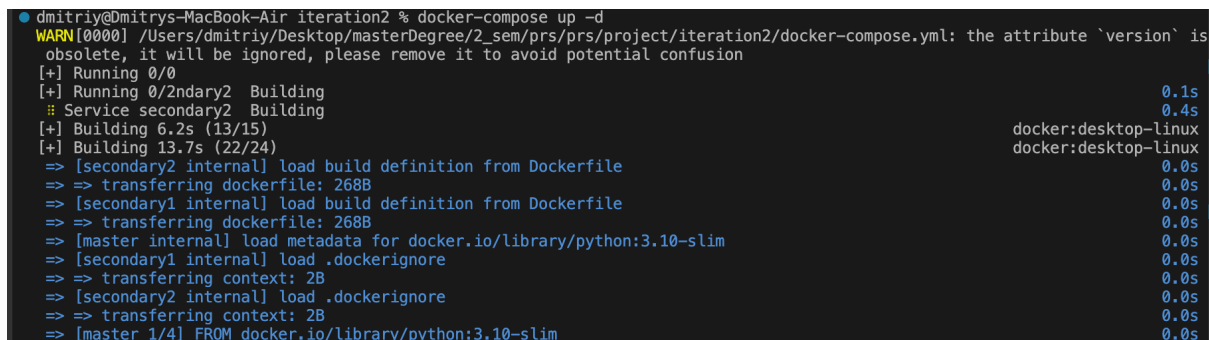
Log:

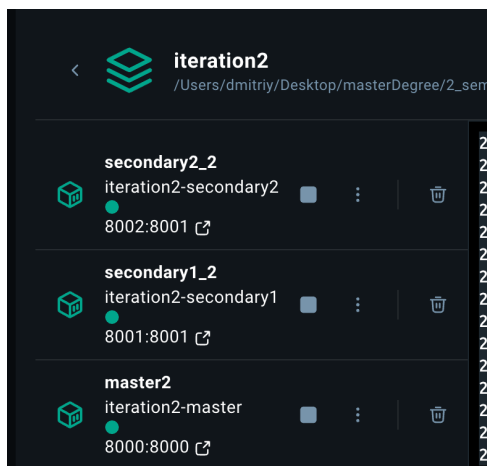


В логах чітко видно, як виконується sleep 5 sec:

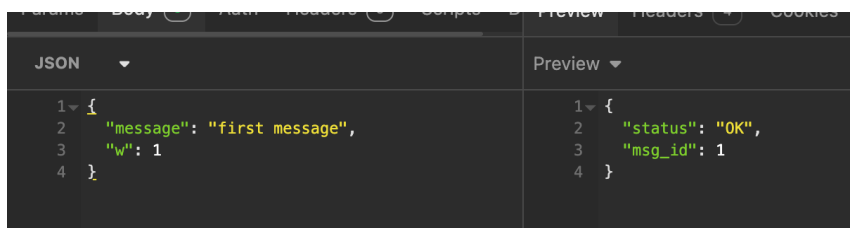


Iteration 2:

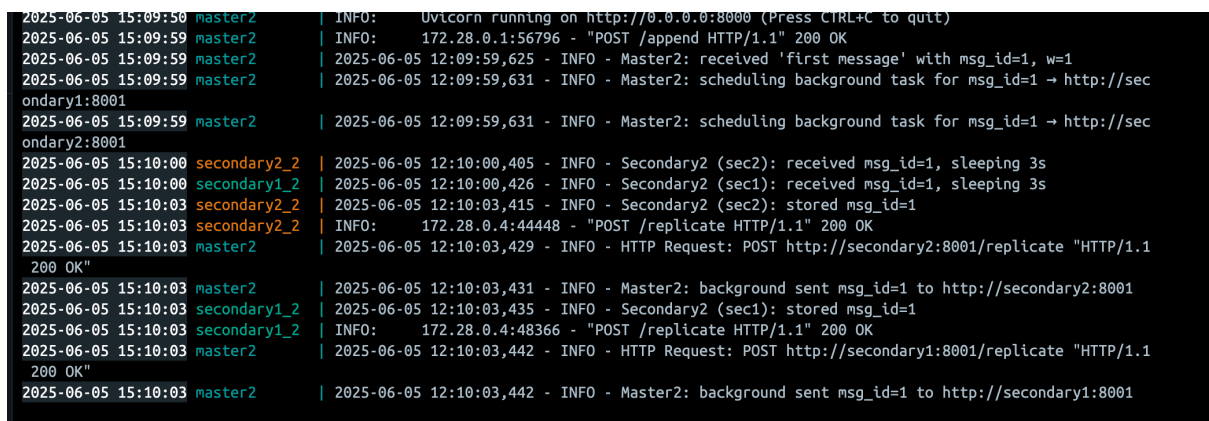




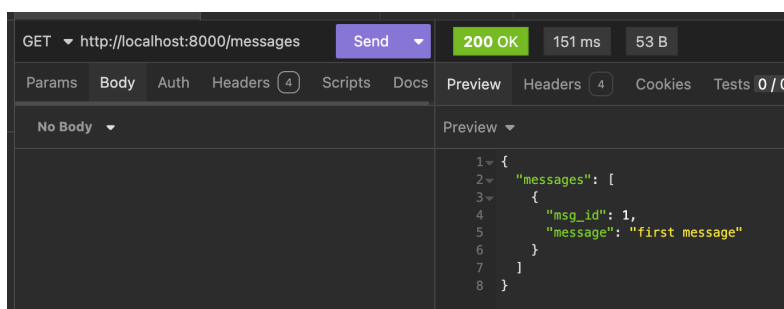
W=1



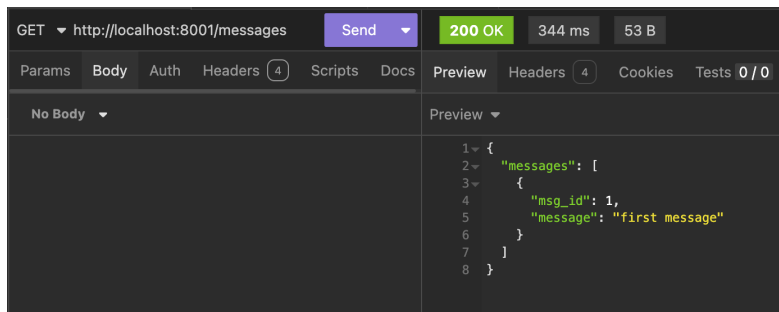
Відповідь була отримана одразу



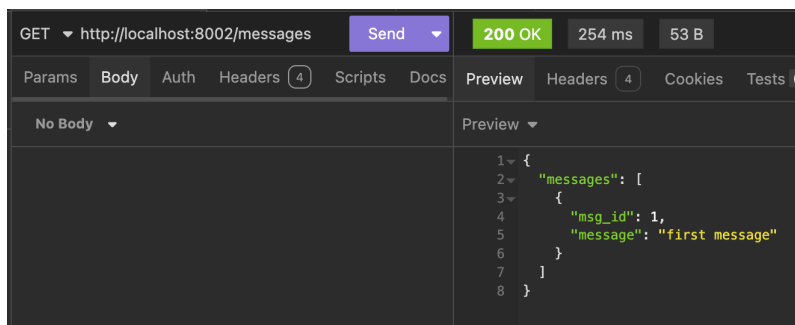
Master:



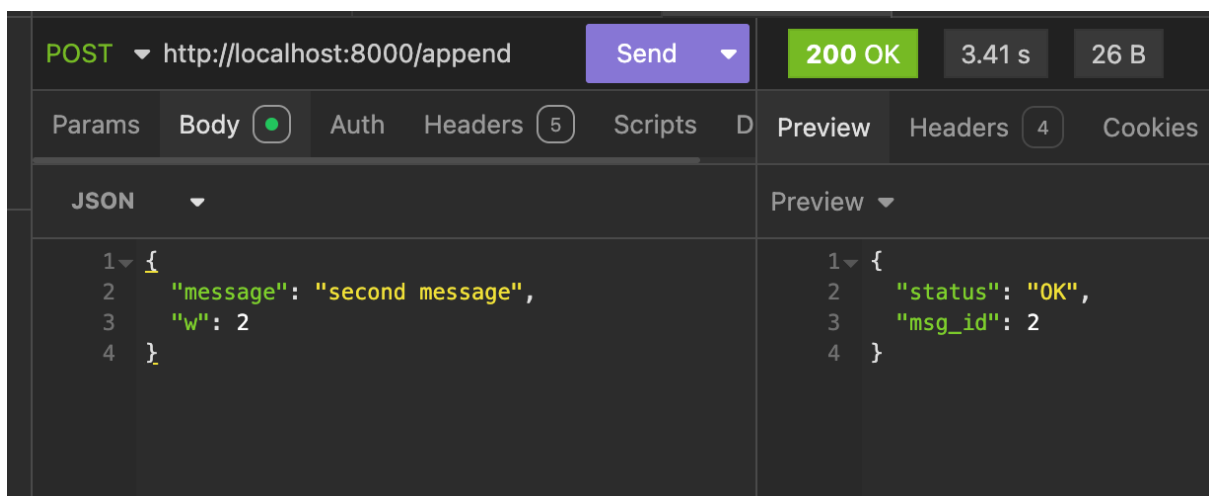
Secondary 1:



Secondary 2:



W=2



Відповідь була отримана з затримкою
Master:

GET http://localhost:8000/messages Send 200 OK 103 ms 93 B

Params Body Auth Headers 4 Scripts Docs Preview Headers 4 Cookies Tests 0/0

No Body

```
1 {
2   "messages": [
3     {
4       "msg_id": 1,
5       "message": "first message"
6     },
7     {
8       "msg_id": 2,
9       "message": "second message"
10    }
11  ]
12 }
```

Secondary 1:

Personal Project / prs_1 001/messages Send 200 OK 34 ms 93 B

Params Body Auth Headers 4 Scripts Docs Preview Headers 4 Cookies Tests 0/0 → Mock

No Body

```
1 {
2   "messages": [
3     {
4       "msg_id": 1,
5       "message": "first message"
6     },
7     {
8       "msg_id": 2,
9       "message": "second message"
10    }
11  ]
12 }
```

Secondary 2:

GET ▼ http://localhost:8002/messages

Send

200 OK

74 ms

93 B

Params
Body
Auth
Headers 4
Scripts
Docs

Preview
Headers 4
Cookies
Tests 0/0
→ Mock

No Body ▼

Preview ▼

```

1 {
2   "messages": [
3     {
4       "msg_id": 1,
5       "message": "first message"
6     },
7     {
8       "msg_id": 2,
9       "message": "second message"
10    }
11  ]
12 }

```

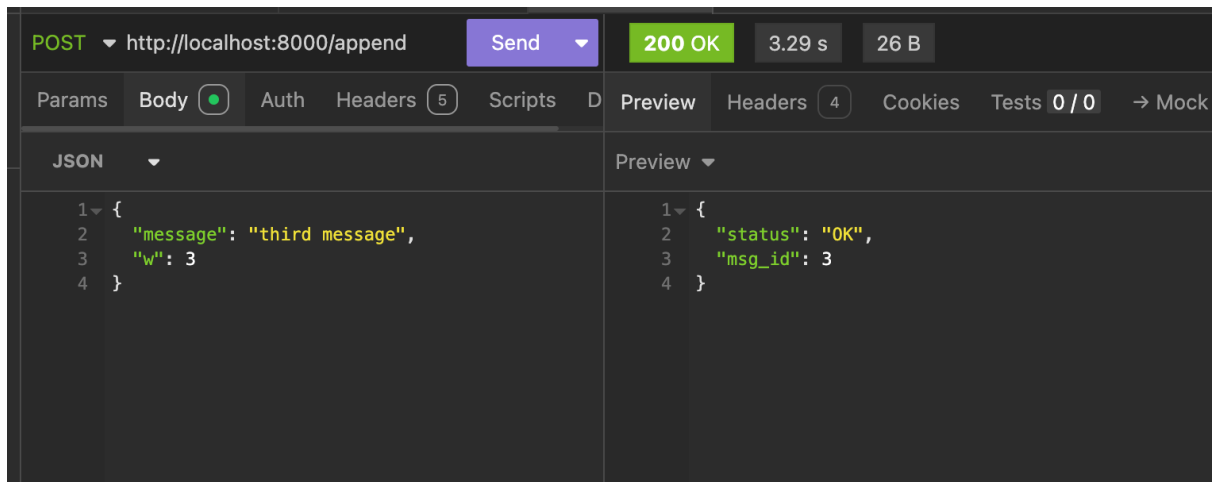
Logs:

```

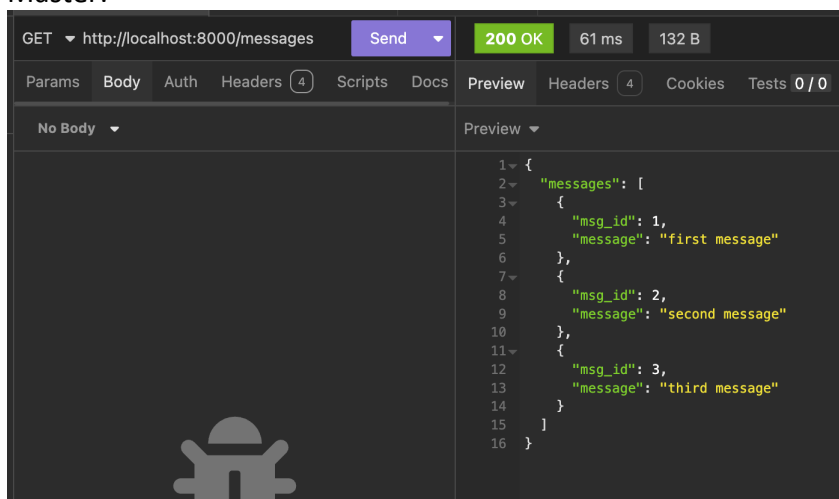
2025-06-05 15:53:33 master2 | 2025-06-05 12:53:33,089 - INFO - Master2: received 'second message' with msg_id=2, w=2
2025-06-05 15:53:33 secondary1_2 | 2025-06-05 12:53:33,178 - INFO - Secondary2 (sec1): received msg_id=2, sleeping 3s
2025-06-05 15:53:33 secondary2_2 | 2025-06-05 12:53:33,179 - INFO - Secondary2 (sec2): received msg_id=2, sleeping 3s
2025-06-05 15:53:36 secondary1_2 | 2025-06-05 12:53:36,188 - INFO - Secondary2 (sec1): stored msg_id=2
2025-06-05 15:53:36 secondary2_2 | 2025-06-05 12:53:36,188 - INFO - Secondary2 (sec2): stored msg_id=2
2025-06-05 15:53:36 secondary2_2 | INFO: 172.28.0.4:35236 - "POST /replicate HTTP/1.1" 200 OK
2025-06-05 15:53:36 secondary1_2 | INFO: 172.28.0.4:53986 - "POST /replicate HTTP/1.1" 200 OK
2025-06-05 15:53:36 master2 | 2025-06-05 12:53:36,217 - INFO - HTTP Request: POST http://secondary1:8001/replicate "HTTP/1.1 200 OK"
2025-06-05 15:53:36 master2 | 2025-06-05 12:53:36,221 - INFO - HTTP Request: POST http://secondary2:8001/replicate "HTTP/1.1 200 OK"
2025-06-05 15:53:36 master2 | 2025-06-05 12:53:36,222 - INFO - Master2: synchronous sent msg_id=2 to http://secondary1:8001 (ACK)
2025-06-05 15:53:36 master2 | 2025-06-05 12:53:36,223 - INFO - Master2: synchronous sent msg_id=2 to http://secondary2:8001 (ACK)
2025-06-05 15:53:36 master2 | 2025-06-05 12:53:36,224 - INFO - Master2: received 1 ACKs (required 1), replying to client
2025-06-05 15:53:36 master2 | 2025-06-05 12:53:36,224 - INFO - Master2: scheduling background task for msg_id=2 → http://secondary1:8001 (post-sync)
2025-06-05 15:53:36 master2 | 2025-06-05 12:53:36,225 - INFO - Master2: scheduling background task for msg_id=2 → http://secondary2:8001 (post-sync)
2025-06-05 15:53:36 master2 | INFO: 172.28.0.1:61248 - "POST /append HTTP/1.1" 200 OK
2025-06-05 15:53:36 secondary2_2 | 2025-06-05 12:53:36,324 - INFO - Secondary2 (sec2): received msg_id=2, sleeping 3s
2025-06-05 15:53:36 secondary1_2 | 2025-06-05 12:53:36,324 - INFO - Secondary2 (sec1): received msg_id=2, sleeping 3s
2025-06-05 15:53:39 secondary2_2 | 2025-06-05 12:53:39,332 - INFO - Secondary2 (sec2): msg_id=2 already exists, ignoring
2025-06-05 15:53:39 secondary1_2 | 2025-06-05 12:53:39,333 - INFO - Secondary2 (sec1): msg_id=2 already exists, ignoring
2025-06-05 15:53:39 secondary2_2 | INFO: 172.28.0.4:49416 - "POST /replicate HTTP/1.1" 200 OK
2025-06-05 15:53:39 secondary1_2 | INFO: 172.28.0.4:56924 - "POST /replicate HTTP/1.1" 200 OK
2025-06-05 15:53:39 master2 | 2025-06-05 12:53:39,358 - INFO - HTTP Request: POST http://secondary2:8001/replicate "HTTP/1.1 200 OK"
2025-06-05 15:53:39 master2 | 2025-06-05 12:53:39,361 - INFO - Master2: background sent msg_id=2 to http://secondary2:8001
2025-06-05 15:53:39 master2 | 2025-06-05 12:53:39,362 - INFO - HTTP Request: POST http://secondary1:8001/replicate "HTTP/1.1 200 OK"
2025-06-05 15:53:39 master2 | 2025-06-05 12:53:39,362 - INFO - Master2: background sent msg_id=2 to http://secondary1:8001
2025-06-05 15:53:40 secondary1_2 | INFO: 172.28.0.1:58514 - "GET /messages HTTP/1.1" 200 OK
2025-06-05 15:54:01 secondary2_2 | INFO: 172.28.0.1:59702 - "GET /messages HTTP/1.1" 200 OK

```

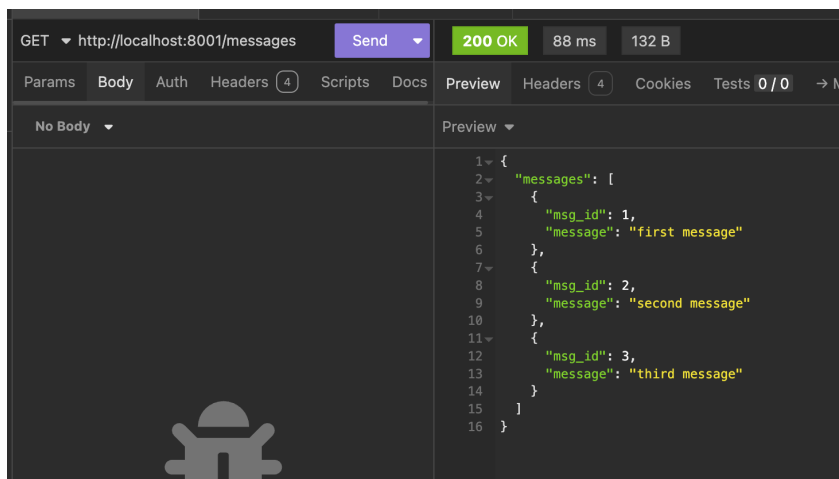
W=3



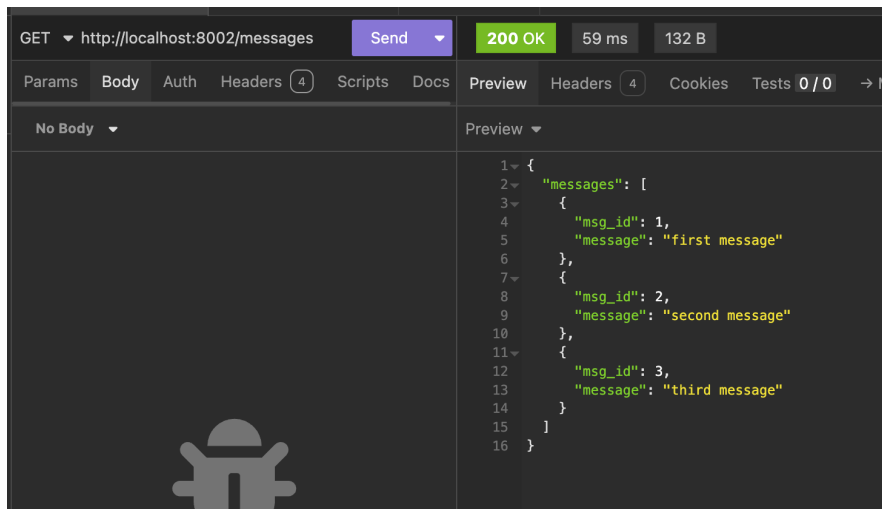
Master:



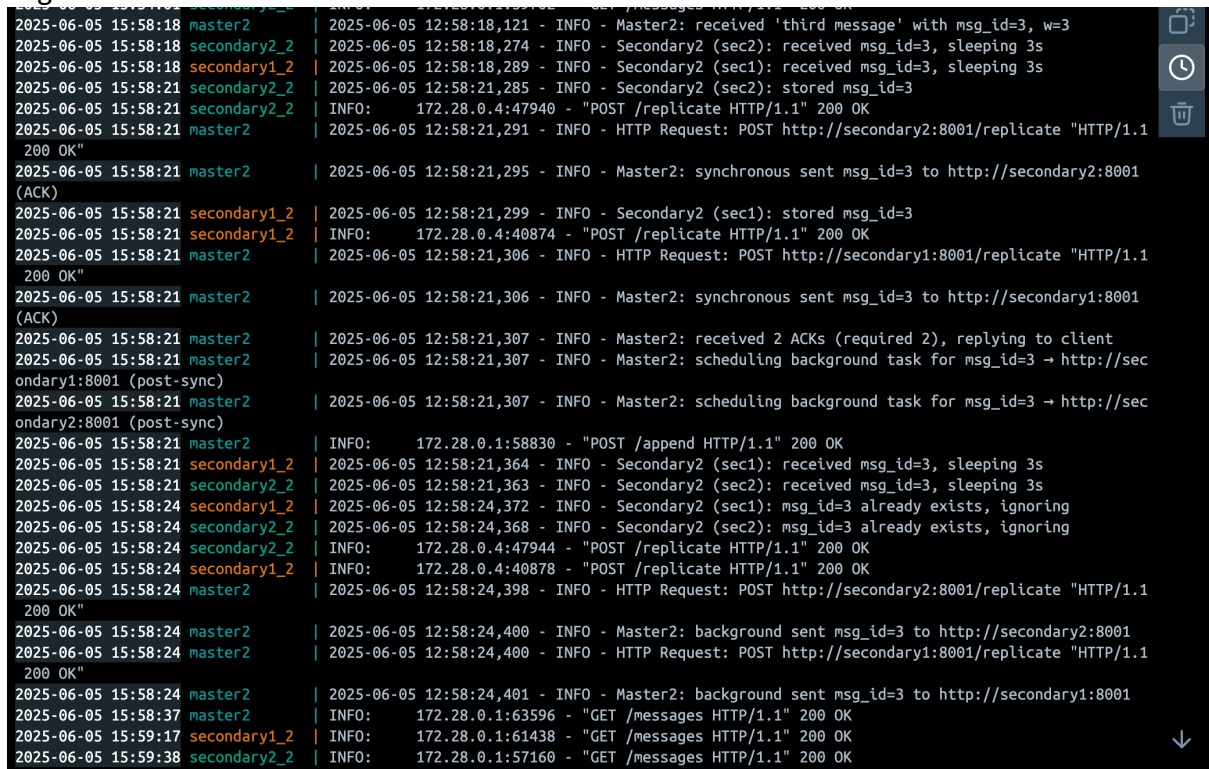
Secondary 1:



Secondary 2:



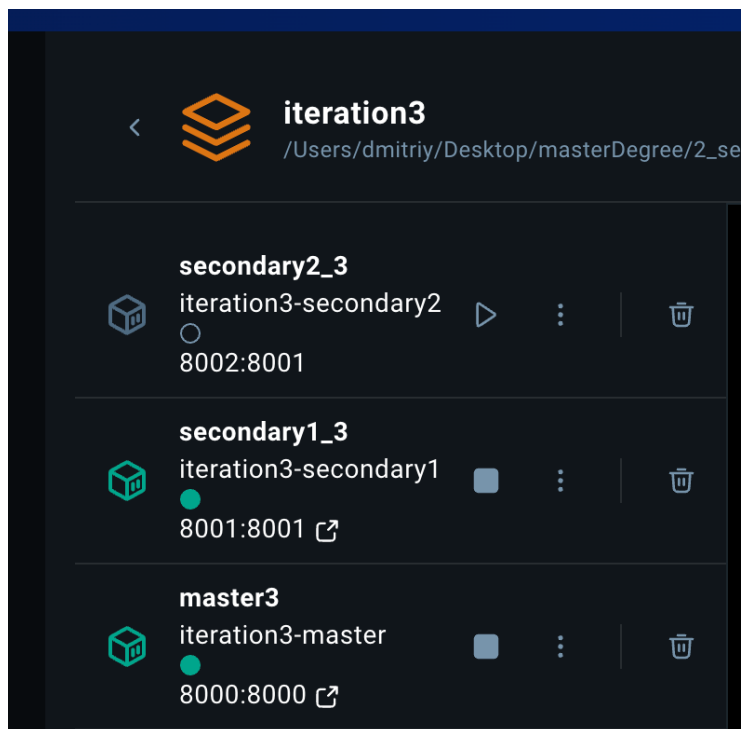
Logs:



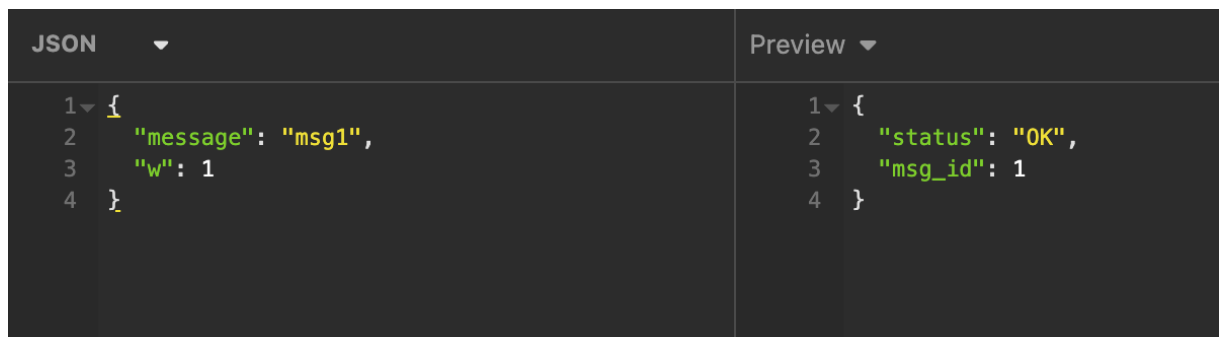
Iteration 3:

Self-check acceptance test:

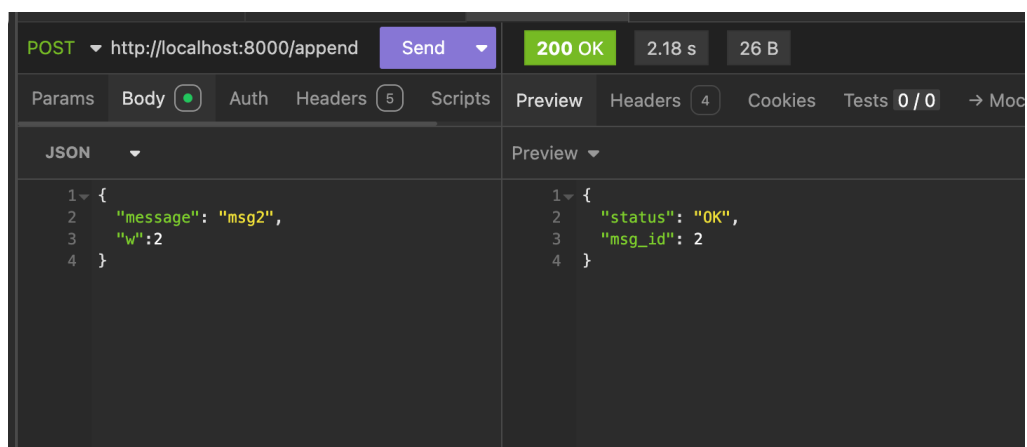
1. Start M + S1



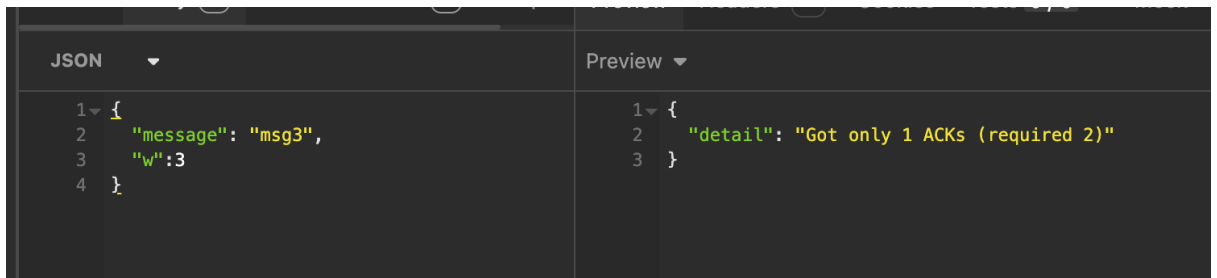
2. send (Msg1, W=1) – Ok



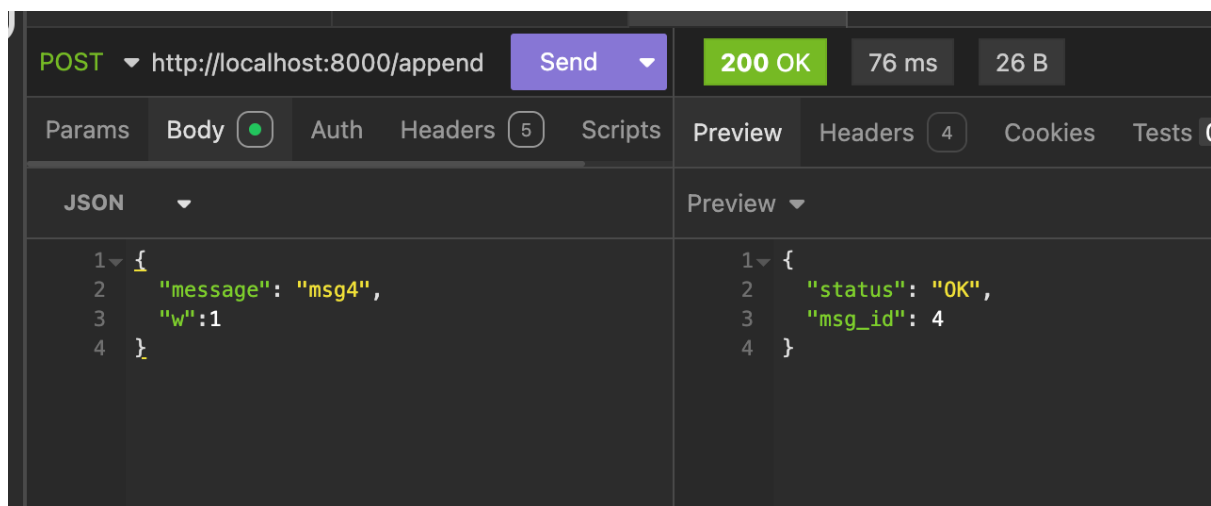
3. send (Msg2, W=2) – Ok



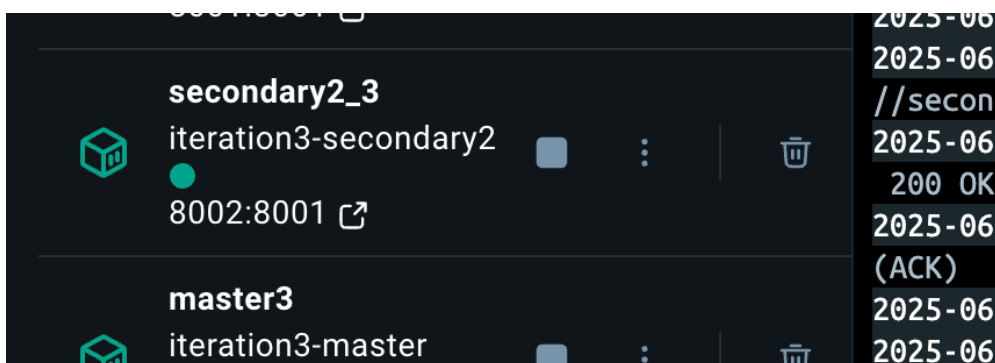
4. send (Msg3, W=3) – Wait



5. send (Msg4, W=1) – Ok



6. Start S2



7. Check messages on S2 - [Msg1, Msg2, Msg3, Msg4]

GET

▼

http://localhost:8000/mess

Send

▼

200 OK

79 ms

134 B

Params

Body

Auth

Headers

4

Script

Preview

Headers

4

Cookies


Tests

No Body

▼

Preview

▼



Enter a URL and send to get a

```
1 {
2   "messages": [
3     {
4       "msg_id": 1,
5       "message": "msg1"
6     },
7     {
8       "msg_id": 2,
9       "message": "msg2"
10    },
11    {
12      "msg_id": 3,
13      "message": "msg3"
14    },
15    {
16      "msg_id": 4,
17      "message": "msg4"
18    }
19  ]
20 }
```