Task 7 - Реплікация у Cassandra
Автор: Потужний Богдан ФІ-03

1. Сконфігурувати кластер з 3-х нод:

- https://hub.docker.com/_/cassandra
- https://gokhanatil.com/2018/02/build-a-cassandra-cluster-on-docker.html
- https://www.jamescoyle.net/how-to/2448-create-a-simple-cassandra-cluster-with-3-nodes
- https://www.digitalocean.com/community/tutorials/how-to-run-a-multi-node-cluster-database-with-cassandra-on-ubuntu-14-04

```
PS C:\Users\bohda> docker run --name node1 -d cassandra
ff2e1010c8b951cd6dec5913e7c428c2bd758fbce3e44cba3e03a18961ec2596
PS C:\Users\bohda> docker run --name node2 -d --link node1:cassandra cassandra
32cb68b36ed40d5ee8632a46981bcc4002249bb949029624fdf775accafcb0cb
PS C:\Users\bohda> docker run --name node3 -d --link node1:cassandra cassandra
```

2. Перевірити правильність конфігурації за допомогою *nodetool status*

```
PS C:\Users\bohda> docker exec -it node1 nodetool status
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address     Load        Tokens  Owns (effective)  Host ID                               Rack
UN  172.17.0.4  104.25 KiB  16      76.0%             26c92459-fac5-4b86-9c02-19a167598094  rack1
UN  172.17.0.3  104.33 KiB  16      59.3%             9f65597d-db04-482e-bf0e-5b02a9facc10  rack1
UN  172.17.0.2  109.41 KiB  16      64.7%             ea281967-c280-4bc0-9d7f-481800ac16a6  rack1
```

3. Використовуючи *cqlsh*, створити три *Keyspace* з replication factor 1, 2, 3

https://www.tutorialspoint.com/cassandra/cassandra_create_keyspace.htm
https://docs.datastax.com/en/cql/3.1/cql/cql_reference/create_keyspace_r.html

```
PS C:\Users\bohda> docker exec -it node1 cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.3 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> CREATE KEYSPACE ks1 WITH replication = {'class':'SimpleStrategy', 'replication_factor' : 1};
cqlsh> CREATE KEYSPACE ks2 WITH replication = {'class':'SimpleStrategy', 'replication_factor' : 2};
cqlsh> CREATE KEYSPACE ks3 WITH replication = {'class':'SimpleStrategy', 'replication_factor' : 3};
cqlsh>
```

4. В кожному з кейспейсів створити таблиці

https://docs.datastax.com/en/cql/3.1/cql/cql_reference/create_table_r.html
https://www.tutorialspoint.com/cassandra/cassandra_create_table.htm

```
cqlsh> USE ks1;
cqlsh:ks1> CREATE TABLE table1 (id int PRIMARY KEY, name text);
cqlsh:ks1> USE ks2;
cqlsh:ks2> CREATE TABLE table2 (id int PRIMARY KEY, name text);
cqlsh:ks2> USE ks3;
cqlsh:ks3> CREATE TABLE table3 (id int PRIMARY KEY, name text);
cqlsh:ks3>
```

5. Спробуйте писати і читати на / та з різних нод.

```
cqlsh:ks1> INSERT INTO table1 (id, name) VALUES (1, 'Alice');
cqlsh:ks1> INSERT INTO table1 (id, name) VALUES (2, 'Bob');
cqlsh:ks1> exit
PS C:\Users\bohda> docker exec -it node2 cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.3 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> USE ks1;
cqlsh:ks1> SELECT * FROM table1;

 id | name
----+-------
  1 | Alice
  2 |   Bob

(2 rows)
cqlsh:ks1>
```

```
PS C:\Users\bohda> docker exec -it node1 cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.3 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> USE ks2;
cqlsh:ks2> INSERT INTO table2 (id, name) VALUES (1, 'Charlie');
cqlsh:ks2> INSERT INTO table2 (id, name) VALUES (2, 'Delta');
cqlsh:ks2>
cqlsh:ks2> USE ks3;
cqlsh:ks3> INSERT INTO table3 (id, name) VALUES (1, 'Echo');
cqlsh:ks3> INSERT INTO table3 (id, name) VALUES (2, 'Foxtrot');
cqlsh:ks3> exit
PS C:\Users\bohda> docker exec -it node2 cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.3 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> USE ks2;
cqlsh:ks2> SELECT * FROM table2;

 id | name
----+---------
  1 | Charlie
  2 |   Delta

(2 rows)
cqlsh:ks2>
cqlsh:ks2> USE ks3;
cqlsh:ks3> SELECT * FROM table3;

 id | name
```

```
 id | name
----+---------
  1 |    Echo
  2 | Foxtrot

(2 rows)
cqlsh:ks3>
```

6. Вставте дані в створені таблиці і подивіться на їх розподіл по вузлах кластера (для кожного з кейспесов - *nodetool status*)

https://docs.datastax.com/en/cql/3.1/cql/cql_reference/insert_r.html
https://docs.datastax.com/en/cql/3.1/cql/cql_reference/select_r.html
https://www.tutorialspoint.com/cassandra/cassandra_create_data.htm
https://www.tutorialspoint.com/cassandra/cassandra_read_data.htm

```
PS C:\Users\bohda> docker exec -it node1 nodetool status ks1
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address     Load        Tokens  Owns (effective)  Host ID                               Rack
UN  172.17.0.4  70.73 KiB   16      35.7%             26c92459-fac5-4b86-9c02-19a167598094  rack1
UN  172.17.0.3  104.94 KiB  16      31.6%             9f65597d-db04-482e-bf0e-5b02a9facc10  rack1
UN  172.17.0.2  110.01 KiB  16      32.7%             ea281967-c280-4bc0-9d7f-481800ac16a6  rack1

PS C:\Users\bohda> docker exec -it node1 nodetool status ks2
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address     Load        Tokens  Owns (effective)  Host ID                               Rack
UN  172.17.0.4  70.73 KiB   16      76.0%             26c92459-fac5-4b86-9c02-19a167598094  rack1
UN  172.17.0.3  104.94 KiB  16      59.3%             9f65597d-db04-482e-bf0e-5b02a9facc10  rack1
UN  172.17.0.2  110.01 KiB  16      64.7%             ea281967-c280-4bc0-9d7f-481800ac16a6  rack1

PS C:\Users\bohda> docker exec -it node1 nodetool status ks3
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address     Load        Tokens  Owns (effective)  Host ID                               Rack
UN  172.17.0.4  70.73 KiB   16      100.0%            26c92459-fac5-4b86-9c02-19a167598094  rack1
UN  172.17.0.3  104.94 KiB  16      100.0%            9f65597d-db04-482e-bf0e-5b02a9facc10  rack1
UN  172.17.0.2  110.01 KiB  16      100.0%            ea281967-c280-4bc0-9d7f-481800ac16a6  rack1
```

7. Для якогось запису з кожного з кейспейсу виведіть ноди на яких зберігаються дані

https://docs.datastax.com/en/dse/5.1/dse-admin/datastax_enterprise/tools/nodetool/toolsGetEndPoints.html

```
PS C:\Users\bohda> docker exec -it node1 nodetool getendpoints ks1 table1 1
172.17.0.2
```

8. Відключиіть одну з нод. Для кожного з кейспейсів визначить з якими рівнями *consistency* можемо читати та писати, і які з них забезпечують *strong consistency*

https://docs.datastax.com/en/cql/3.1/cql/cql_reference/consistency_r.html

```
PS C:\Users\bohda> docker pause node3
node3
PS C:\Users\bohda> docker exec -it node1 cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.3 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> CONSISTENCY ONE;
Consistency level set to ONE.
cqlsh> USE ks1;
cqlsh:ks1> INSERT INTO table1 (id, name) VALUES (5, 'Bob2');
NoHostAvailable: ('Unable to complete the operation against any hosts', {<Host: 127.0.0.1:9042 datacenter1>: Unavailable
('Error from server: code=1000 [Unavailable exception] message="Cannot achieve consistency level ONE" info={\'consistenc
y\': \'ONE\', \'required_replicas\': 1, \'alive_replicas\': 0}')})
cqlsh:ks1> SELECT * FROM table1;
NoHostAvailable: ('Unable to complete the operation against any hosts', {<Host: 127.0.0.1:9042 datacenter1>: Unavailable
('Error from server: code=1000 [Unavailable exception] message="Cannot achieve consistency level ONE" info={\'consistenc
y\': \'ONE\', \'required_replicas\': 1, \'alive_replicas\': 0}')})
cqlsh:ks1>
```

```
cqlsh:ks1> CONSISTENCY QUORUM;
Consistency level set to QUORUM.
cqlsh:ks1> SELECT * FROM table1;
NoHostAvailable: ('Unable to complete the operation against any hosts', {<Host: 127.0.0.1:9042 datacenter1>: Unavailable
('Error from server: code=1000 [Unavailable exception] message="Cannot achieve consistency level QUORUM" info={\'consist
ency\': \'QUORUM\', \'required_replicas\': 1, \'alive_replicas\': 0}')})
cqlsh:ks1> CONSISTENCY ALL;
Consistency level set to ALL.
cqlsh:ks1> SELECT * FROM table1;
NoHostAvailable: ('Unable to complete the operation against any hosts', {<Host: 127.0.0.1:9042 datacenter1>: Unavailable
('Error from server: code=1000 [Unavailable exception] message="Cannot achieve consistency level ALL" info={\'consistenc
y\': \'ALL\', \'required_replicas\': 1, \'alive_replicas\': 0}')})
cqlsh:ks1>
```

9. Зробить так щоб три ноди працювали, але не бачили одна одну по мережі (відключити зв'язок між ними)

```
PS C:\Users\bohda> docker network create --driver bridge network1
058576aaa447b824c1cc57dd06bd0e97e8e2669bf4091016121e42d177a4e448
PS C:\Users\bohda> docker network create --driver bridge network2
638e1c852043e74754c0de850011c23a1068dd9f9304813fccb061c599068a23
PS C:\Users\bohda> docker network create --driver bridge network3
da7a8bea24a25a2e6c046238cd51fd20388c717c9b5b30b0352f832cbaecd049
PS C:\Users\bohda> docker network disconnect bridge node1
PS C:\Users\bohda> docker network disconnect bridge node2
PS C:\Users\bohda> docker network disconnect bridge node3
PS C:\Users\bohda> docker network connect network1 node1
PS C:\Users\bohda> docker network connect network2 node2
PS C:\Users\bohda> docker network connect network3 node3
PS C:\Users\bohda> docker exec -it node1 nodetool status
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address     Load        Tokens  Owns  Host ID                               Rack
DN  172.17.0.4  70.73 KiB   16      ?     26c92459-fac5-4b86-9c02-19a167598094  rack1
DN  172.17.0.3  168.11 KiB  16      ?     9f65597d-db04-482e-bf0e-5b02a9facc10  rack1
UN  172.17.0.2  167.11 KiB  16      ?     ea281967-c280-4bc0-9d7f-481800ac16a6  rack1

Note: Non-system keyspaces don't have the same replication settings, effective ownersh
PS C:\Users\bohda> docker exec -it node2 nodetool status
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address     Load        Tokens  Owns  Host ID                               Rack
DN  172.17.0.4  70.73 KiB   16      ?     26c92459-fac5-4b86-9c02-19a167598094  rack1
```

10. Для кейспейсу з *replication factor* 3 задайте рівень consistency рівним 1. Виконайте запис одного й того самого значення, з однаковим primary key, але різними іншими значенням на кожну з нод (тобто створіть конфлікт)

```
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.3 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> CONSISTENCY ONE;
Consistency level set to ONE.
cqlsh> use ks3;
cqlsh:ks3> Select * from table3;

 id | name
----+---------
  1 |    Echo
  2 | Foxtrot

(2 rows)
cqlsh:ks3> INSERT INTO table3 (id, name) VALUES (3, 'Bob');
cqlsh:ks3> exit
PS C:\Users\bohda> docker exec -it node2 cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.3 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> use ks3;
cqlsh:ks3> insert into table3 (id, name) values (3, 'Alice');
cqlsh:ks3> exit
PS C:\Users\bohda> docker exec -it node3 cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.3 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> use ks3;
cqlsh:ks3>  insert into table3 (id, name) values (3, 'Bravo');
cqlsh:ks3> exit;
```

11. Об’єднайте ноди в кластер і визначте яке значення було прийнято кластером та за яким принципом

```
PS C:\Users\bohda> docker network disconnect network1 node1
PS C:\Users\bohda> docker network disconnect network2 node2
PS C:\Users\bohda> docker network disconnect network3 node3
PS C:\Users\bohda> docker network connect bridge node1
PS C:\Users\bohda> docker network connect bridge node2
PS C:\Users\bohda> docker network connect bridge node3
PS C:\Users\bohda> docker exec -it node1 nodetool status
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address     Load        Tokens  Owns  Host ID                               Rack
UN  172.17.0.4  133.95 KiB  16      ?     26c92459-fac5-4b86-9c02-19a167598094  rack1
UN  172.17.0.3  168.11 KiB  16      ?     9f65597d-db04-482e-bf0e-5b02a9facc10  rack1
UN  172.17.0.2  167.11 KiB  16      ?     ea281967-c280-4bc0-9d7f-481800ac16a6  rack1
```

```
PS C:\Users\bohda> docker exec -it node1 cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.3 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> use ks3;
cqlsh:ks3> select * from table3;

 id | name
----+---------
  1 |    Echo
  2 | Foxtrot
  3 |   Bravo

(3 rows)
cqlsh:ks3>
```

В даному випадку значення було обрано за принципом ‘last_write_wins’.

12. Перевірте поведінку *lightweight transactions* для попередніх пунктів у розділеному та не розділеному на три частини кластері https://docs.datastax.com/en/cql-oss/3.3/cql/cql_using/useInsertLWT.html

```
PS C:\Users\bohda> docker network connect network1 node1
PS C:\Users\bohda> docker network connect network2 node2
PS C:\Users\bohda> docker network connect network3 node3
PS C:\Users\bohda> docker exec -it node1 cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.3 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> INSERT INTO ks3.table3 (id, name) VALUES (1, 'Alice') IF NOT EXISTS;
NoHostAvailable: ('Unable to complete the operation against any hosts', {<Host: 127.0.0.1:9042 datacenter1>: Unavailable('Error from server: cod
e=1000 [Unavailable exception] message="Cannot achieve consistency level SERIAL" info={\'consistency\': \'SERIAL\', \'required_replicas\': 2, \'
alive_replicas\': 1}')})
cqlsh> UPDATE ks3.table3 SET name = 'Bob' WHERE id = 1 IF EXISTS;
NoHostAvailable: ('Unable to complete the operation against any hosts', {<Host: 127.0.0.1:9042 datacenter1>: Unavailable('Error from server: cod
e=1000 [Unavailable exception] message="Cannot achieve consistency level SERIAL" info={\'consistency\': \'SERIAL\', \'required_replicas\': 2, \'
alive_replicas\': 1}')})
cqlsh> DELETE FROM ks3.table3 WHERE id = 1 IF EXISTS;
NoHostAvailable: ('Unable to complete the operation against any hosts', {<Host: 127.0.0.1:9042 datacenter1>: Unavailable('Error from server: cod
e=1000 [Unavailable exception] message="Cannot achieve consistency level SERIAL" info={\'consistency\': \'SERIAL\', \'required_replicas\': 2, \'
alive_replicas\': 1}')})
```

```
PS C:\Users\bohda> docker network disconnect network1 node1
PS C:\Users\bohda> docker network disconnect network2 node2
PS C:\Users\bohda> docker network disconnect network3 node3
PS C:\Users\bohda> docker network connect bridge node1
PS C:\Users\bohda> docker network connect bridge node2
PS C:\Users\bohda> docker network connect bridge node3
PS C:\Users\bohda> docker exec -it node1 cqlsh
Connected to Test Cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.3 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> INSERT INTO ks3.table3 (id, name) VALUES (1, 'Alice') IF NOT EXISTS;

 [applied] | id | name
-----------+----+------
     False |  1 | Echo

cqlsh> UPDATE ks3.table3 SET name = 'Bob' WHERE id = 1 IF EXISTS;

 [applied]
-----------
      True

cqlsh> DELETE FROM ks3.table3 WHERE id = 1 IF EXISTS;

 [applied]
-----------
      True

cqlsh>
```

Вимогу до оформлення протоколу:

Завдання здається особисто без протоколу, або надсилається протокол який має містити:

- команди та результати їх виконання

Оформлено протокол.