cqlsh Cassandra query shell

CONCEPTS

Statements end with a semicolon; shell commands such as DESCRIBE or CONSISTENCY do not need one.

Cluster nodes, partitioner
- **Keyspace** replication settings
  - **Table** partition key + clustering
    - **Row** columns

CONNECT

`cqlsh <host> <port>`
Connect to a node (default port 9042)

`cqlsh -u <user> -p <password> <host>`
Authenticate on connect

`cqlsh -k <keyspace> <host>`
Start in a keyspace

`cqlsh --ssl <host>`
Connect over TLS

`cqlsh -e "<statement>" <host>`
Run one statement and quit

`cqlsh -f <file.cql> <host>`
Run a script file and quit

`cqlsh --request-timeout=<sec> <host>`
Raise the request timeout (default 10)

EXPLORE SCHEMA

`DESCRIBE KEYSPACES`
List all keyspaces

`DESCRIBE KEYSPACE <keyspace>`
DDL of one keyspace

`DESCRIBE TABLES`
Tables in current keyspace

`DESCRIBE TABLE <table>`
DDL of a table

SESSION CONTROL

`USE <keyspace>;`
Make a keyspace current

`CONSISTENCY <level>`
Set consistency level

SESSION CONTROL (cont.)

`TRACING ON`
Print a trace after each query

`EXPAND ON`
One column per line

`EXIT`
Quit the shell

SERVER AND FILES

`SHOW VERSION`
cqlsh, Cassandra and protocol versions

`SOURCE '<file.cql>'`
Run a CQL file from the prompt

BASIC CQL

```
CREATE KEYSPACE <ks> WITH
replication = {'class':
'SimpleStrategy',
'replication_factor': 1};
```
Create a keyspace

```
CREATE TABLE <table> (<id> uuid
PRIMARY KEY, <col> text);
```
Create a table

```
INSERT INTO <table> (<id>, <col>)
VALUES (<id>, '<val>');
```
Insert a row

`SELECT * FROM <table> WHERE <id> = <value>;`
Read by partition key

`SELECT * FROM <table> LIMIT <n>;`
Limit rows returned

`UPDATE <table> SET <col> = '<val>' WHERE <id> = <value>;`
Change a column

`DELETE FROM <table> WHERE <id> = <value>;`
Delete a row

COPY IMPORT AND EXPORT

`COPY <table> TO '<file.csv>' WITH HEADER = true;`
Export a table to CSV

`COPY <table> FROM '<file.csv>' WITH HEADER = true;`
Import CSV into a table