

11. gyakorlat

#1 ✓ ? Csatlakozzon a Redis Cloud-hoz!

- Kattintson a korábban létrehozott adatbázis nevére, majd válassza ki a Configuration fület!
- Adja meg válaszként a következőket: Endpoint, Port, Default password

Public endpoint	redis-13916.c100.us-east-1-4.ec2.redns.redis-cloud.com:13916
-----------------	--

Default user password

hXwFfZU8HChY4HluCYdsseiesdMAcRai

```
>_ CLI

Try Workbench, our advanced CLI. Check out our Quick Guides to learn more about Redis capabilities.

Connecting...

Pinging Redis server on redis-13916.c100.us-east-1-4.ec2.redns.redis-cloud.com:13916
Connected.
Ready to execute commands.

> SET életkor 25
"OK"

> GET életkor
"25"

> EXPIRE életkor 100
(integer) 1

> SET életkor 26
"OK"
```

lista típusú elem: rpush

hash alapú elem: hset

```
> RPush nevek juci laci maci
(integer) 3

> LRange nevek 0 2
1) "juci"
2) "laci"
3) "maci"

> HSet tanulo08 nev "Nagy Ivo" neptunkod "xxx888" hianyzas 3
(integer) 3

> HGetall tanulo08
1) "nev"
2) "Nagy Ivo"
3) "neptunkod"
4) "xxx888"
5) "hianyzas"
6) "3"
```

halmaz típusú elem: sadd

```
> SAdd tantargyak Adatbazisok Programozas Matematika
(integer) 3

> SAdd vizsgatargy Programozas Matematika
(integer) 2

> SUnion tantargyak vizsgatargy
1) "Adatbazisok"
2) "Programozas"
3) "Matematika"

> SDiff tantargyak vizsgatargy
1) "Adatbazisok"

> SInter tantargyak vizsgatargy
1) "Programozas"
2) "Matematika"
```

sorted set típusú elem: zadd

```
> ZADD eredmények 50 laci 70 juci 10 maci
(integer) 3

> ZRANGE eredmények 0 -1
1) "maci"
2) "laci"
3) "juci"

> ZRANGE eredmények 0 -1 withscores
1) "maci"
2) "10"
3) "laci"
4) "50"
5) "juci"
6) "70"
```

Megoldás

#4 ✓ ? A redis-cli-ben hozzon létre egy új hash kulcsot auto néven, amelynek értéke a következő rekordnak megfelelő legyen:

rendszám típus evjarat ertek xxx-111 Skoda Fabia 2012 1500000

```
C:\Windows\System32\cmd.exe - redis-cli -h redis-13542.c293.eu-central-1-1.ec2.cloud.redislabs.com -p 13542 -a 4eTzDUpAht196Vmsx4d1F0ucX4Lrp3R6
redis-13542.c293.eu-central-1-1.ec2.cloud.redislabs.com:13542> hset auto rendszam xxx-111
típus "Skoda Fabia" evjarat 2012 ertek 1500000
(integer) 4
```

#5 ✓ ? A redis-cli-ben hozzon létre új lista típusú kulcsot tantargyak néven, amelyek értékei a következők legyenek: programozas alapjai, matematika, adatbaziskezeles

- a. A lista végére szűrjön be még egy tantárgyat: operacios rendszerek
- b. A listát létrehozó és a lista végére beszűrő parancsokat (két parancs) adja meg válaszként!

Megoldás

```
C:\Windows\System32\cmd.exe - redis-cli -h redis-13542.c293.eu-central-1-1.ec2.cloud.redislabs.com -p 13542 -a 4eTzDUpAht196Vmsx4d1F0ucX4Lrp3R6
redis-13542.c293.eu-central-1-1.ec2.cloud.redislabs.com:13542> rpush tantargyak "programo
zas alapjai" matematika adatbaziskezeles
(integer) 3
```

```
redis-13542.c293.eu-central-1-1.ec2.cloud.redislabs.com:13542> rpush tantargyak "operacio
s rendszerek"
```

```
redis-13542.c293.eu-central-1-1.ec2.cloud.redislabs.com:13542> lrange tantargyak 0 -1
1) "programozas alapjai"
2) "matematika"
3) "adatbaziskezeles"
4) "operacios rendszerek"
```

#8 ✓ ? A redis-cli-ben kérdezzük le az adatbázisban lévő azon kulcsokat, amelyek nevében van a betű!

a. A parancsot és eredményét mutató képernyőrészletet adjuk meg a válaszhoz kép formájában!

➤ [Megoldás](#)

```
> scan 0 match *a*
1) "17"
2) 1) "munka:2:3"
   2) "munkahelyek"
   3) "tant\xc3\xa1rgyak"
   4) "diak:3"
   5) "vizsgatargy"
   6) "diak:1"

> keys *a*
1) "diak:1"
```

#12 ✓ ? Hozza létre mezőnként és soronként külön kulcsok segítségével a dolgozo tábla következő rekordjait:

nev munkakor kod Nagy Eva titkarno 1 Kiss Ilona
konyvelo 2

sorokat hoz létre

```
> HSET dolgozo:1 nev "Nagy Eva" munkakor "titkarno" kod 1 H
(integer) 4

> HSET dolgozo:2 nev "Kiss Ilona" munkakor "konyvelo" kod 2
(integer) 3

>
```

#13 ✓ ? A redis-cli-ben hozzuk létre a projekt táblának megfelelő adatszerkezetet soronként és mezőnként külön kulcsokkal! A kulcsok megadásánál ügyeljünk a projekt és a dolgozo tábla közötti kapcsolatra!

projektkod projektnev dolgozokod 1 EURO-33 1 3 TRANS-22 2

a. A szükséges utasításokat tartalmazó képernyőrészt adjuk meg válaszként kép formájában!

```
> HSET project:1:1 projektnev "EURO-33"
(integer) 1

> HSET project:3:2 projektnev "TRANS-22"
(integer) 1
```

#14 ✓ ? A redis-cli-ben hozzuk létre a dolgozo tábla (ld. 12-es feladat) nev oszlopának megfelelő adatszerkezetet dolgozo_nev néven!

oszlopokat hoz létre

```
> ZADD dolgozo:nev 1 "Nagy Eva" 2 "Kiss Ilona"  
(integer) 2
```

elemeket hoz létre

```
> SET dolgozo:1:nev "Nagy Eva"  
"OK"  
  
> SET dolgozo:1:munkakor "titkarno"  
"OK"
```

CASSANDRA

#20 ✓ ? A DataStax CQL konzolon kérdezzük le, hogy mi azoknak a filmeknek a címe, amelyek típusa Movie, és 2018-ban jelentek meg (release_year)!

a. A szükséges csak az első 3 találatot jelenítse meg

```
token@cqlsh:movies> SELECT title FROM movies_and_tv WHERE type = 'Movie' AND release_year=2018 LIMIT 3;
```

title
#Roxy
14 Cameras
1st Summoning

(3 rows)

#21 ✓ ? A DataStax Astra CQL konzolon kérdezzük le, hogy a movies_and_tv táblában típusonként, azon belül évenként hány rekord van!

a. A lekérdezést adjuk meg válaszként!

```
token@cqlsh:movies> SELECT type, release_year, COUNT(*) FROM movies_and_tv GROUP BY type, release_year;
```

type	release_year	count
Movie	1958	2
Movie	1965	2
TV Show	2018	19
Movie	1995	14
Movie	2013	153
Movie	1976	6
Movie	2020	5
Movie	2018	517
Movie	1942	1
Movie	1966	1

#23 ✓ ? A Cassandra CQL Shell-ben tegye aktuálissá az előzőleg létrehozott kps keyspace-t!

a. Hozzon létre egy új táblát Szemely néven, amelynek mezői: nev - szöveg, szulev - egész, foglalkozas - szöveg

b. A partition key legyen: (nev, szulev), a clustered key pedig foglalkozas

c. A táblát létrehozó parancsot adjuk meg válaszként!

```
token@cqlsh:Movies> use kps;
token@cqlsh:kps> create table Szemely
... (
... nev text,
... szulev int,
... foglalkozas text,
... PRIMARY KEY ((nev, szulev), foglalkozas);
```

```
token@cqlsh:movies> INSERT INTO személy(nev, szulev,foglalkozas)
... VALUES ('Kiss Béla', 2000, 'lakatos');
token@cqlsh:movies> INSERT INTO személy(nev, szulev,foglalkozas) VALUES ('Tóth Ottó', 2000, 'pincér');
token@cqlsh:movies> SELECT * FROM személy
... ;
```

nev	szulev	foglalkozas
Tóth Ottó	2000	pincér
Kiss Béla	2000	lakatos

(2 rows)

```
token@cqlsh:movies>
```

#25 ✓ ? A Cassandra CQL Shell-ben készítsen indexet a Szemely táblához a foglalkozas oszlop alapján!

- Az index neve legyen i_foglalkozas
- A szükséges parancsot adja meg válaszként!

```
token@cqlsh:movies> create index i_foglalkozas on személy(foglalkozas);
token@cqlsh:movies> desc table személy;
```

```
CREATE TABLE movies.szemely (
  nev text,
  szulev int,
  foglalkozas text,
  autok map<text, text>,
  nyelvtudas set<text>,
  vegzettseg list<text>,
  PRIMARY KEY ((nev, szulev), foglalkozas)
) WITH CLUSTERING ORDER BY (foglalkozas ASC)
AND additional_write_policy = '99PERCENTILE'
AND bloom_filter_fp_chance = 0.01
AND caching = {'keys': 'ALL', 'rows_per_partition': 'NONE'}
AND comment = ''
AND compaction = {'class': 'org.apache.cassandra.db.compaction.UnifiedCompactionStrategy'}
AND compression = {'chunk_length_in_kb': '64', 'class': 'org.apache.cassandra.io.compress.LZ4Compressor'}
AND crc_check_chance = 1.0
AND default_time_to_live = 0
AND gc_grace_seconds = 864000
AND max_index_interval = 2048
AND memtable_flush_period_in_ms = 0
AND min_index_interval = 128
AND read_repair = 'BLOCKING'
AND speculative_retry = '99PERCENTILE';
CREATE CUSTOM INDEX i_foglalkozas ON movies.szemely (foglalkozas) USING 'org.apache.cassandra.index.sai.StorageAttachedIndex';
```

#26 ✓ ? A Cassandra CQL Shell-ben kérdezze le azokat a személyeket, akiknek foglalkozása pincer vagy lakatos!

- Csak a személyek neve jelenjen meg!
- A szükséges lekérdezést adja meg válaszként! (szükség esetén a lekérdezés végén használja az ALLOW FILTERING záradékot)
- A lekérdezést adja meg válaszként!

```
token@cqlsh:movies> select nev from személy where foglalkozas in('pincer','lakatos')allow filtering;
```

nev
Tóth Ottó
Kiss Béla
Kiss Béla

#27 ✓ ? A Cassandra CQL Shell-ben bővítse a Személy táblát egy új mezővel

- A mező neve legyen Vegzettseg, típusa LIST
- Kiss Bela végzettsegei legyenek 'gepesz' és 'muszeresz' (az UPDATE utasítás WHERE feltételében minden kulcsmezőt meg kell adni!)
- Az új mezőt létrehozó, és a végzettségeket megadó parancsokat (2 db) adja meg válaszként!

```
token@cqlsh:movies> ALTER TABLE személy
... ADD vegzettseg LIST<TEXT>;
token@cqlsh:movies> UPDATE személy
... SET vegzettseg= ['gepesz','műszerész']
... WHERE nev= 'Kiss Béla' AND szulev=200 AND foglalkozas='lakatos';
token@cqlsh:movies> SELECT * FROM személy
... ;
```

nev	szulev	foglalkozas	vegzettseg
Kiss Béla	200	lakatos	['gepesz', 'műszerész']
Tóth Ottó	2000	pincér	null
Kiss Béla	2000	lakatos	null

#28 ✓ ? A Cassandra CQL Shell-ben bővítsük a Szemely táblát egy új mezővel!

- A mező neve legyen Nyelvtudas, típusa pedig SET
- Nagy Ivo nyelvtudása legyen: 'angol' és 'francia'
- A szükséges parancsokat (2 db) adjuk meg válaszként!

```
token@cqlsh:movies> ALTER TABLE személy ADD Nyelvtudas SET<TEXT>;
```



```
token@cqlsh:movies> UPDATE személy SET Nyelvtudas={ 'angol','francia'} WHERE nev= 'Tóth Ottó'  
AND szulev=200 AND foglalkozas='pincér' ;  
token@cqlsh:movies> SELECT * FROM személy ;
```

nev	szulev	foglalkozas	nyelvtudas	vegzettseg
Tóth Ottó	200	pincér	{'angol', 'francia'}	null
Kiss Béla	200	lakatos	null	['gepesz', 'műszerész']
Tóth Ottó	2000	pincér	null	null
Kiss Béla	2000	lakatos	null	null

(4 rows)

#29 ✓ ? A Cassandra CQL Shell-ben adjon hozzá még egy új oszlopot a Szemely táblához!

- Az oszlop neve legyen Autok, típusa MAP<TEXT, TEXT>
- Rögzítsük, hogy Kiss Bela esetén az Autok mező értéke: {'Skoda Fabia': 'abc-111', 'Audi A4': 'xyz-222' }
- A szükséges utasításokat (2 db) adjuk meg válaszként!

```
token@cqlsh:movies> ALTER TABLE személy ADD Autok MAP<TEXT,TEXT>;
token@cqlsh:movies> UPDATE személy SET Autok={ 'Skoda Fabia': 'abc-111','Audi A4': 'xyz-222'} W
HERE nev= 'Tóth Ottó' AND szulev=200 AND foglalkozas='pincér' ;
token@cqlsh:movies> SELECT * FROM személy ;
```

nev	szulev	foglalkozas	autok	nyelvtu das	vegzettseg
Tóth Ottó	200	pincér	{'Audi A4': 'xyz-222', 'Skoda Fabia': 'abc-111'}	{'angol', 'francia'}	['barmixer']
Kiss Béla	200	lakatos			['gepesz', 'műszerész']
Tóth Ottó	2000	pincér			

#30 ✓ ? A Cassandra CQL Shell-ben módosítsuk a Személy tábla következő adatait:

- Nagy Ivó megtanult németül is, vegyük fel ezt is a nyelvtudásai közé
- Toth Otto elvégezte a bármixer szakot, vegyük fel ezt a végzettségei közé
- A szükséges 2 utasítást adjuk meg válaszként!

```
token@cqlsh:movies> update személy
... set vegzettseg=vegzettseg+['barmixer']
... where nev='Tóth Ottó' and szulev=200 and foglalkozas='pincer';
token@cqlsh:movies> select * from személy;
```

nev	szulev	foglalkozas	autok	nyelvtudas	vegzettseg
Tóth Ottó	200	pincer			['barmixer']
Tóth Ottó	200	pincér	{'Audi A4': 'xyz-222', 'Skoda Fabia': 'abc-111'}	{'angol', 'francia'}	['gepesz', 'műszerész']
Kiss Béla	200	lakatos			
Tóth Ottó	2000	pincér			
Kiss Béla	2000	lakatos			

(5 rows)