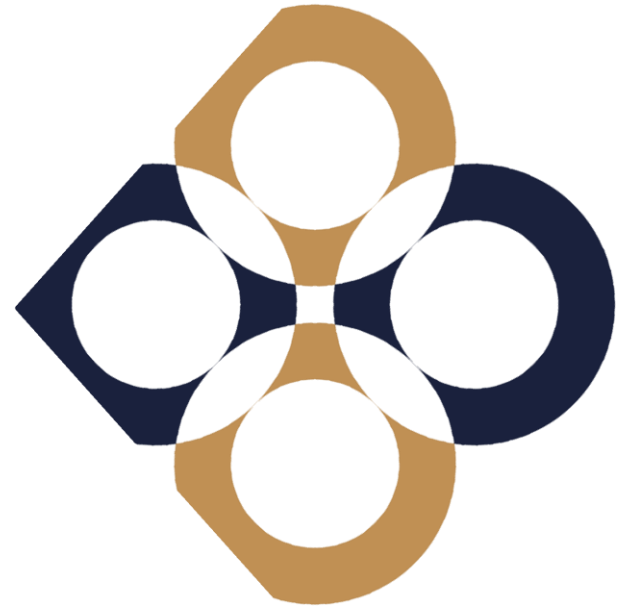


# Adatbázisok előadás 10.

Kulcs-érték adatbázisok



# Miről lesz szó?

- ☐ Kulcs-érték adatbázisok jellemzői
- ☐ A Redis adatbázis
  - ☐ Jellemzők
  - ☐ Adattípusok
  - ☐ Utasítások
  - ☐ Terminál
  - ☐ Elérés Python-ból
  - ☐ Feladatok megoldása

# Kulcs-érték adatbázisok

Olyan adatbázisok, amelyek az adatokat kulcs-érték párokban tárolják, és az adatok a kulcs alapján gyorsan elérhetők

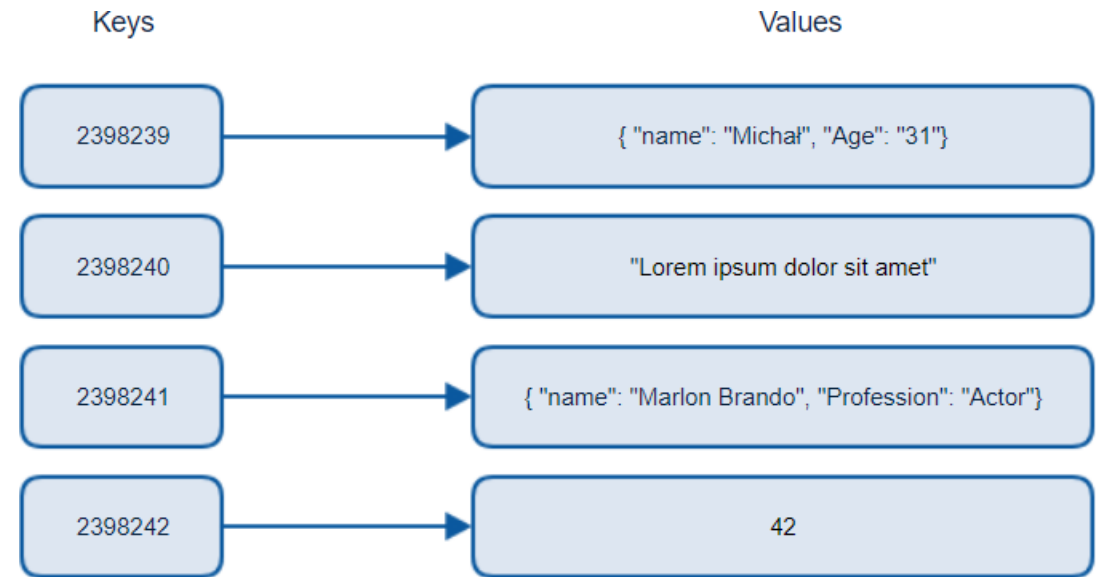
## ❑ Asszociatív tömbök (kulcs-érték párok)

### ❑ A kulcs **egyedi** azonosító

A kulcs és az érték lehet akár egyszerű, akár komplex adat

A kulcs-érték párnak lehet érvényességi ideje

Nagymértékben skálázhatók és partícionálhatók



# Kulcs-érték adatbázisok – előnyök és hátrányok

## Előnyök

- Egyszerűség
- Nagy hozzáférési sebesség
- Flexibilitás
- Skálázhatóság
- Kisebb tárolási méret
- Indexekre – alapesetben – nincs szükség

## Hátrányok

- Bonyolultabb adatmodellekhez nem javasolt
- Limitált lekérdezési lehetőségek
- Sok feladatot alkalmazás szintjén kell megoldani (pl. kényszerek)
- Adatok exportálása
- Adatok közötti kapcsolatok kezelése

# Kulcs-érték adatbázisok – hol használják őket?

Bevásárlókocsi

Cache

Felhasználói profil

Fórum

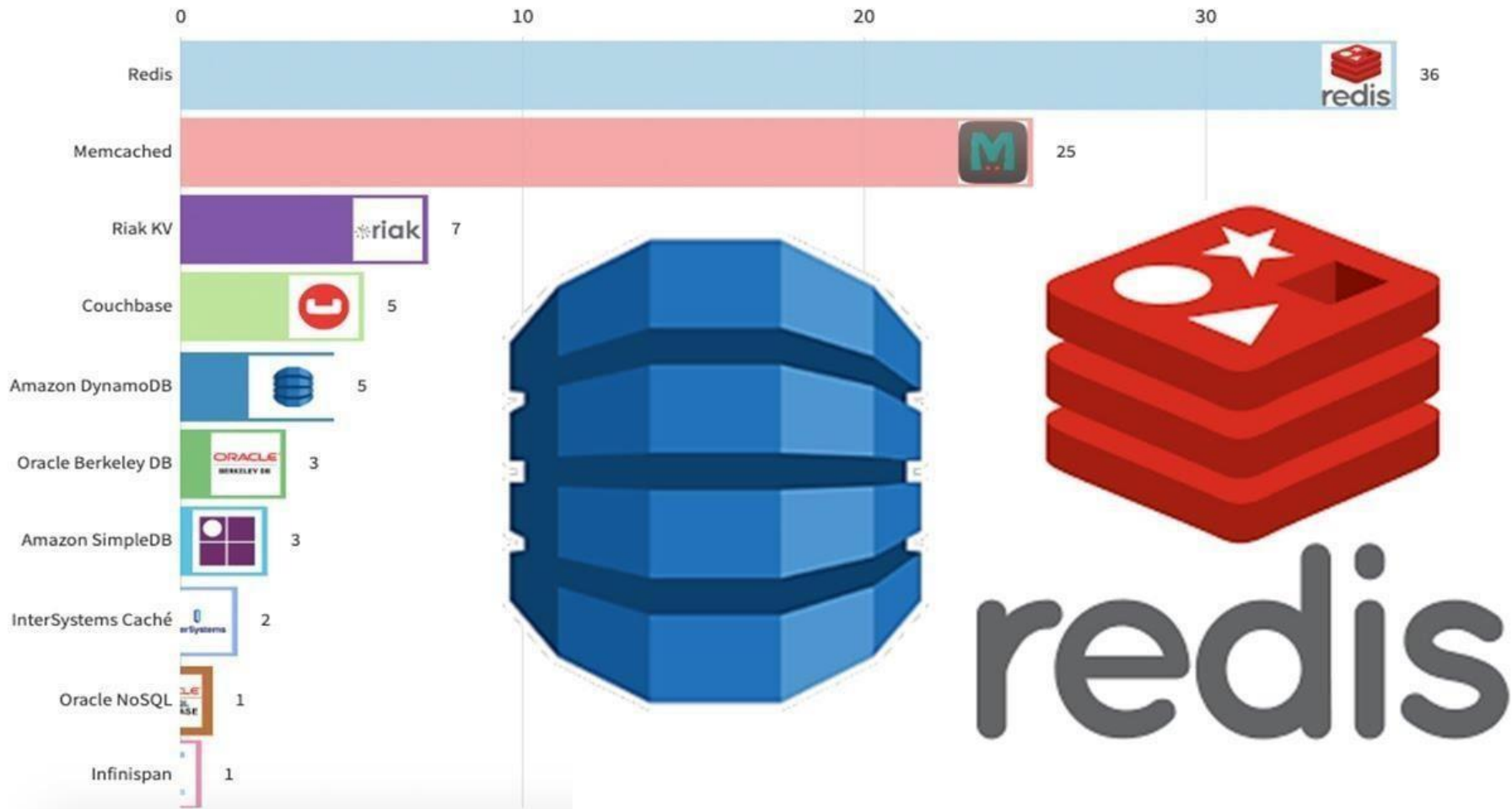
Valós idejű ajánló rendszerek

# Kulcs-érték adatbázisok vs. Relációs adatbázisok

Kulcs érték adatbázisok	Relációs adatbázisok
Adatbázis	(Logikai) Adatbázis
Sorted set	Tábla
Hash	Rekord
Flexibilis séma	Fix séma
A kapcsolatok megfelelő kulcs elnevezésekkel valósíthatók meg	A kapcsolatok idegen kulcsokkal valósulnak meg
Nincs beépített lekérdezési lehetőség	Lekérdezés SQL nyelven

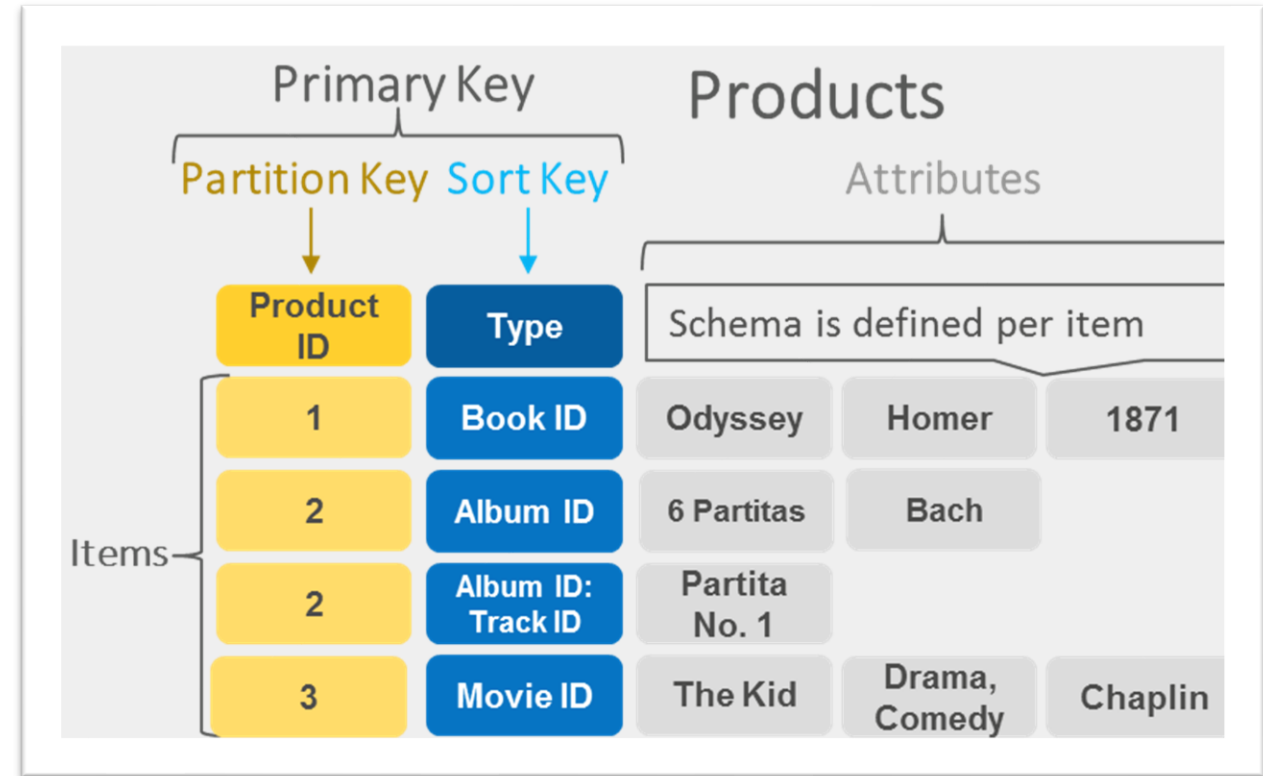
# Kulcs-érték adatbázisok - Példák

## Most Popular Database Engine of Key-value Stores (2012 - 2019)



# Kulcs-érték adatbázisok - DynamoDB

- ❑ A kulcs-érték adatbázisok úttörője
- ❑ Állítható konzisztencia szint
- ❑ Költséghatékony
- ❑ Hibatűrő
- ❑ Integrálható más szolgáltatásokkal





# Kulcs-érték adatbázis – Redis

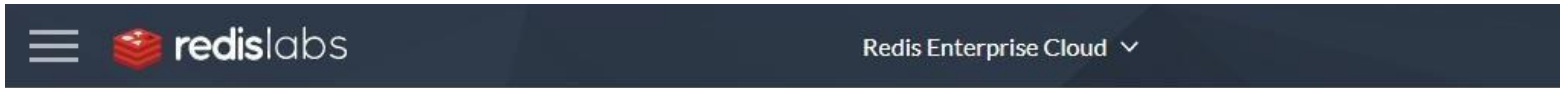
- ❑ Nyílt forráskódú adatbázis
- ❑ Kulcs-érték adatmodell, in-memory adatbázis
- ❑ Futtatási/telepítési lehetőségek
  - ❑ Cloud próba ([Try Redis](#), illetve [Try Redis Enterprise | Redis Labs](#))
  - ❑ On-premise

[Redis](#) - Linux

[How To Install Redis on Windows 10 - DEV Community](#) – Windows

[Install and config Redis on Mac OS X via Homebrew | by Pete Houston | Medium - Mac](#)

# Redis – Cloud



View Database    

Database Name

Metrics Slowlog **Configuration**

Subscription	#1406889 Redis Cloud/Fixed Plan/AWS/eu-central-1/Standard/30MB
Protocol	Redis
Used Memory	2.03 MB
Replication 	Disabled
Activated On	03/26/2021 20:56:01
Last Changed	03/26/2021 20:56:38

A csatlakozáshoz szükséges adatok:

- ☐ Port
- ☐ Endpoint
- ☐ Default user password

Csatlakozási lehetőségek

- ☐ redis-cli (parancssoros interfész)
- ☐ Programkód (pl. Python)

# Redis – Cli

```
C:\Windows\System32\cmd.exe - redis-cli -h redis-13613.c250.eu-central-1-1.ec2.cloud.redislabs.com -p 13613 -a iH3vsvrHTZog4DXLtmvQUMn240NFq0iO
Microsoft Windows [Version 10.0.18363.1440]
(c) 2019 Microsoft Corporation. Minden jog fenntartva.

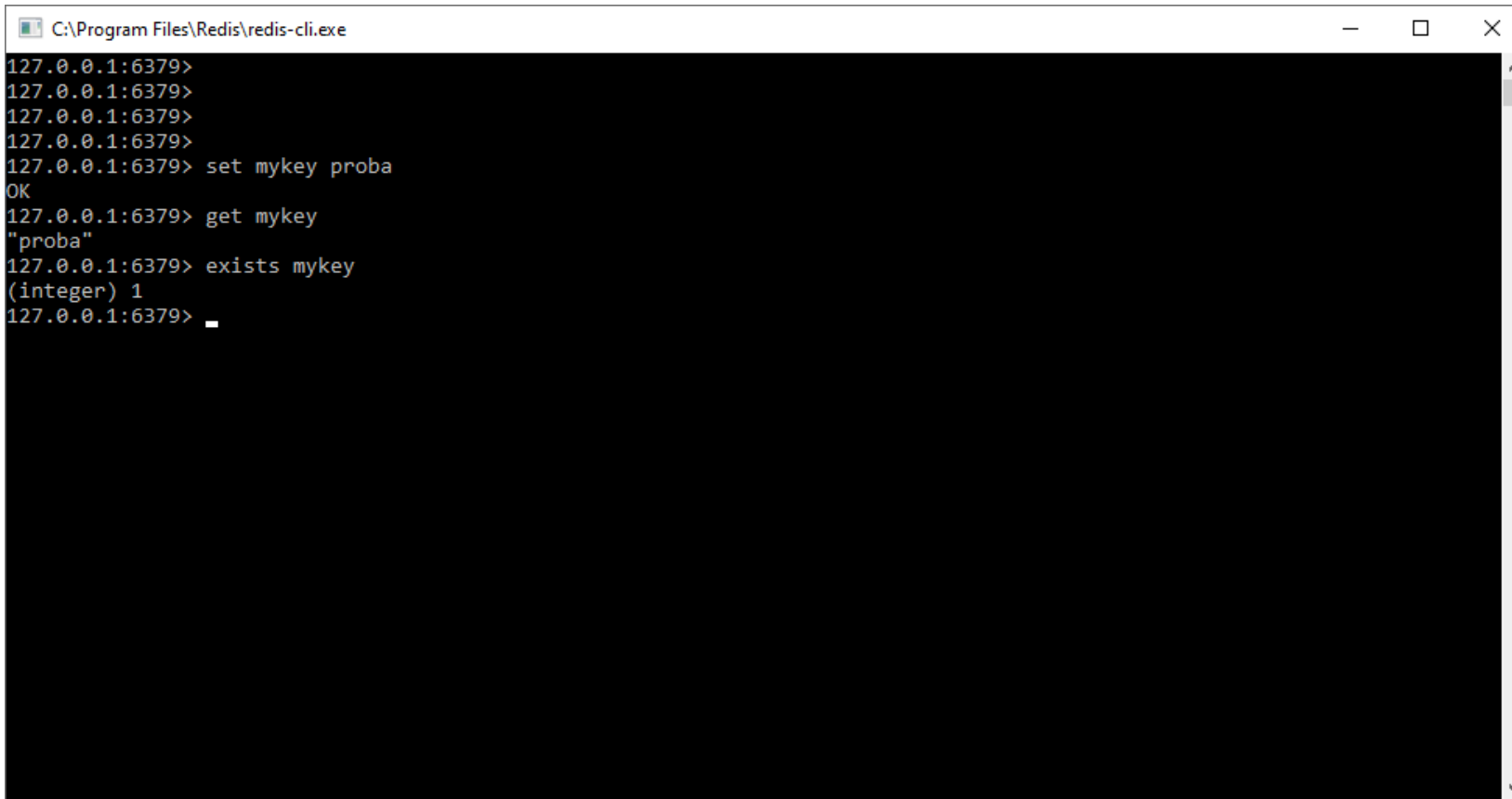
C:\Program Files\Redis>redis-cli -h redis-13613.c250.eu-central-1-1.ec2.cloud.redislabs.com -p 13613 -a iH3vsvrHTZog4DXLtmvQUMn240NFq0iO
redis-13613.c250.eu-central-1-1.ec2.cloud.redislabs.com:13613> ping
PONG
redis-13613.c250.eu-central-1-1.ec2.cloud.redislabs.com:13613> set hello world
OK
redis-13613.c250.eu-central-1-1.ec2.cloud.redislabs.com:13613> get hello
"world"
redis-13613.c250.eu-central-1-1.ec2.cloud.redislabs.com:13613>
```

Aktiválja a Windowst  
Aktiválja a Windows rendszert a Gépházban.

Csatlakozás a Redis Cloud adatbázishoz:

```
redis-cli -h <endpoint> -p <port> -a <password>
```

# Redis – On-premise

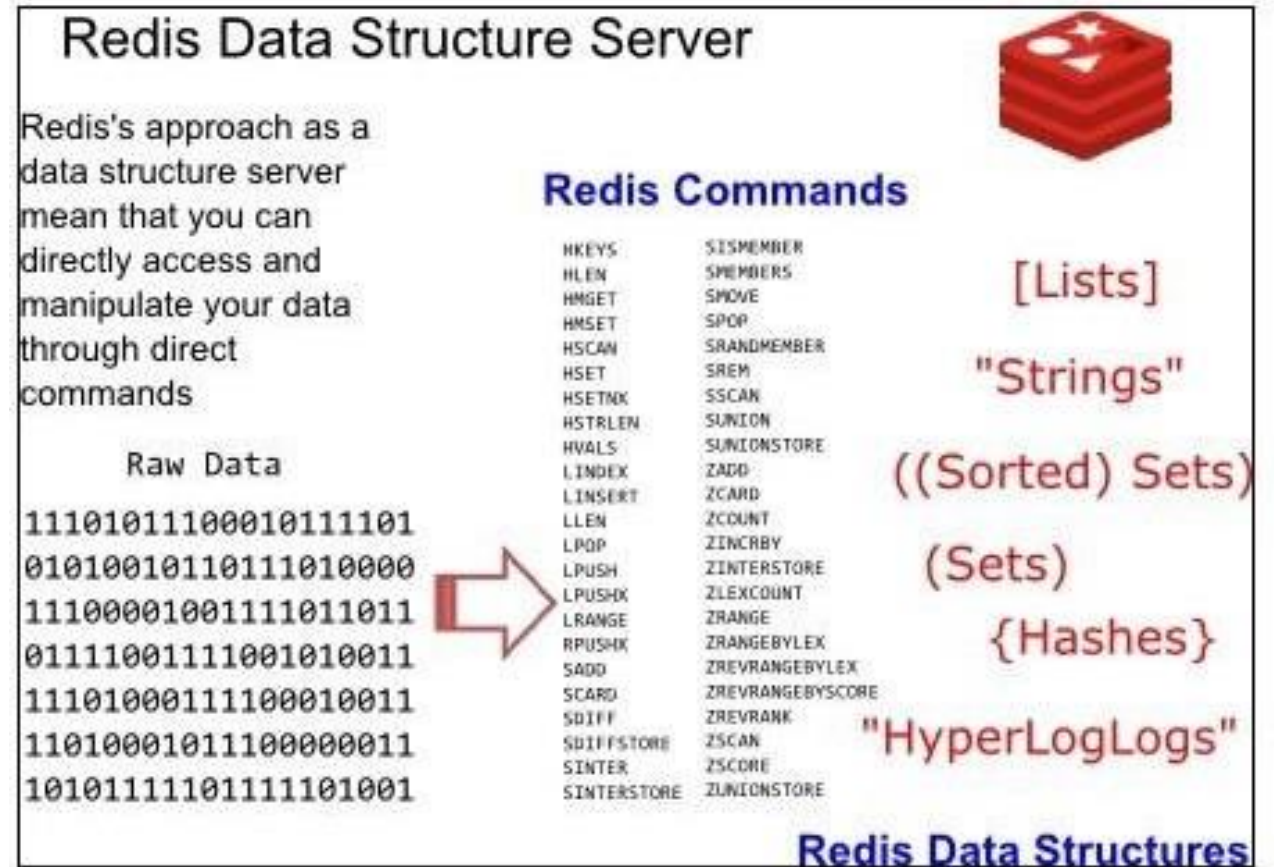
A screenshot of a Windows command prompt window titled "C:\Program Files\Redis\redis-cli.exe". The terminal shows a series of commands and responses from the Redis server running on 127.0.0.1:6379. The commands include multiple empty prompts, a 'set mykey proba' command returning 'OK', a 'get mykey' command returning '"proba"', and an 'exists mykey' command returning '(integer) 1'. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

```
C:\Program Files\Redis\redis-cli.exe
127.0.0.1:6379>
127.0.0.1:6379>
127.0.0.1:6379>
127.0.0.1:6379>
127.0.0.1:6379> set mykey proba
OK
127.0.0.1:6379> get mykey
"proba"
127.0.0.1:6379> exists mykey
(integer) 1
127.0.0.1:6379> _
```

Csatlakozás az on-premis rendszerhez: redis-cli

# Redis adattípusok

- ☐ Key – tulajdonság neve\*
- ☐ String – szöveg
- ☐ List – szöveges adatok rendezett listája
- ☐ Hashes – objektumhoz hasonló adattípus, amely objektum kulcsból és kulcs-érték párokból áll
- ☐ Set – egyedi szöveges adatok rendezetlen halmaza
- ☐ Sorted set – egyedi szöveges adatok rendezett halmaza



# Utasítások\* I. – Key, string

## SET kulcs érték [opciók]

Új kulcs-érték pár létrehozása, vagy a meglévő felülírása  
Fontosabb opciók: EX seconds – lejárati idő, NX – csak akkor jön létre a kulcs, ha még nem létezik, GET – a régi érték megjelenítése, amennyiben létezik

## GET kulcs

A kulcs értékének lekérdezése – ha nem létezik, akkor a visszaadott érték üres (nil)

## EXISTS kulcs(ok)

Ellenőrzi, hogy az adott kulcs létezik-e. Egymás után szóközzel elválasztva több kulcs is megadható

## DEL kulcs(ok)

Törli a megadott kulcsot vagy kulcsokat

\* A Redis verziótól függően a felsorolt utasításoknak csak egy része támogatott

# Utasítások I. – Key, string (folytatás)

COPY kulcs1 kulcs2 [logikai adatbázis]  
[REPLACE]

A kulcs1 értékével létrehozza a még nem létező kulcs2-t. A kulcs2 lehet egy másik logikai adatbázisban is. A REPLACE segítségével törölhető a meglévő kulcs2.

MOVE kulcs logikai\_adatbázis

A kulcsot átmozgatja egy másik logikai adatbázisba, amennyiben ott az nem létezik

RENAME kulcsnév újkulcsnév

Új nevet ad a kulcsnak

EXPIRE kulcs seconds

Beállítja a kulcs élettartamát (másodpercben)

TYPE kulcs

Megadja a kulcshoz tartozó érték típusát

# Utasítások I. - Példák

```
C:\Windows\System32\cmd.exe - redis-cli
127.0.0.1:6379>
127.0.0.1:6379>
127.0.0.1:6379>
127.0.0.1:6379> set mainap hetfo
OK
127.0.0.1:6379> get mainap
"hetfo"
127.0.0.1:6379> set mainap hetfo ex 5
OK
127.0.0.1:6379> get mainap
(nil)
127.0.0.1:6379> exists mainap
(integer) 0
127.0.0.1:6379> set mainap kedd
OK
127.0.0.1:6379> exists mainap
(integer) 1
127.0.0.1:6379> del mainap
(integer) 1
127.0.0.1:6379> exists mainap
(integer) 0
```

- ☐ Létrehozunk egy új kulcsot mainap néven, amelynek értéke hetfo
- ☐ Lekérdezzük az értékét, majd beállítjuk a lejáratí idejét 5 másodperce
- ☐ Utána a kulcs törlődik (nil), azaz már nem létezik
- ☐ Ismét létrehozzuk a mainap kulcsot, most értéke kedd
- ☐ Ellenőrizzük, hogy létezik-e, majd töröljük. Utána ismét megnézzük, hogy létezik-e



# Utasítások II. – Hashes

HSET | HMSET kulcs mező érték [mező érték]

Beállítja a mező(k) értékét

HGET kulcs mező

Lekérdezi a kulcs adott mezőjének értékét

HGETALL kulcs

Lekérdezi a kulcs összes mezőjének értékét

HEXISTS kulcs mező

Megadja, hogy létezik-e a kulcs adott mezője

HDEL kulcs mező [kulcs mező]

Törli az adott mező(ke)t.

HKEYS kulcs

Megadja, hogy milyen mezői vannak a kulcsnak

# Utasítások II. - Példák

```
C:\Windows\System32\cmd.exe - redis-cli
127.0.0.1:6379> hmset tanulo nev "Kiss Bela"
szulido 2000.01.01 neptun abcdef
OK
127.0.0.1:6379> hgetall tanulo
1) "nev"
2) "Kiss Bela"
3) "szulido"
4) "2000.01.01"
5) "neptun"
6) "abcdef"
127.0.0.1:6379> type tanulo
hash
127.0.0.1:6379> hkeys tanulo
1) "nev"
2) "szulido"
3) "neptun"
127.0.0.1:6379> hdel tanulo neptun
(integer) 1
127.0.0.1:6379>
```

- ☐ Létrehozzuk a tanulo kulcsot több mezővel (nev, szulido, neptun)
- ☐ Lekérdezzük a tanulo kulcsot
- ☐ Lekérdezzük a kulcs típusát
- ☐ Lekérdezzük a kulcs mezőit
- ☐ Töröljük a tanulo kulcs neptun mezőjét

# Utasítások III. – List

LSET kulcs index érték

Beállítja a lista adott indexű elemének értékét. Az index 0-ról indul. Speciálisan a -1 az utolsó elemre, -2 az utolsó előttiire utal stb.

LPUSH kulcs elem [elem]

Elem(ek)et szúr be a lista elejére

RPUSH kulcs elem [elem]

Elem(ek)et szúr be a lista végére

LPOP kulcs [n]

Törli és visszaadja a lista első (n) elemét

RPOP kulcs [n]

Törli és visszaadja a lista utolsó (n) elemét

LINSERT kulcs BEFORE | AFTER elem újelem

Új elemet szúr be a lista adott eleme elé vagy után

LRANGE kulcs start stop

Visszaadja a lista elemeit start indextől a stop-ig

# Utasítások III. - Példák

```
C:\Windows\System32\cmd.exe - redis-cli
127.0.0.1:6379>
127.0.0.1:6379> rpush napok hetfo kedd
(integer) 2
127.0.0.1:6379> linsert napok after kedd szerda
(integer) 3
127.0.0.1:6379> lrange napok 0 -1
1) "hetfo"
2) "kedd"
3) "szerda"
127.0.0.1:6379> lpop napok
"hetfo"
127.0.0.1:6379> lrange napok 0 -1
1) "kedd"
2) "szerda"
127.0.0.1:6379>
```

- ☐ Létrehozunk egy napok listát, tartalma: hetfo, kedd
- ☐ Beszúrjuk a szerdát a végére
- ☐ Megjelenítjük a lista elemeit
- ☐ Töröljük az első napot
- ☐ Megjelenítjük az így kapott lista minden elemét

# Utasítások IV. – Set

SADD kulcs elem [elem]

Elem(ek)et ad hozzá a halmazhoz. Ha a halmaz még nem létezik, akkor létrehozza. Ha az elem már létezik, akkor nem kerül be még egyszer a halmazba

SCARD kulcs

Visszaadja a halmaz elemszámát

SPOP kulcs [n]

Töröl és visszaad (n) elemet a halmazból.

SISMEMBER kulcs érték

Visszaadja, hogy az adott érték benne van-e a halmazban

SMEMBERS kulcs

Listázza a halmaz összes elemét

SDIFF | SUNION | SINTER kulcs1 kulcs2

Listázza a halmazok különbségét, unióját, metszetét

SORT kulcs [BY minta] [LIMIT interval]  
[ASC | DESC] [ALPHA]

Rendezi a list, set és ordered set kulcs elemeit

# Utasítások IV. - Példák

```
C:\Windows\System32\cmd.exe - redis-cli
127.0.0.1:6379> sadd evszakok tavasz nyar osz tel
(integer) 4
127.0.0.1:6379> type evszakok
set
127.0.0.1:6379> smembers evszakok
1) "tel"
2) "osz"
3) "tavasz"
4) "nyar"
127.0.0.1:6379> scard evszakok
(integer) 4
127.0.0.1:6379> sadd kedvencevszak nyar
(integer) 1
127.0.0.1:6379> sdiff evszakok kedvencevszak
1) "tavasz"
2) "osz"
3) "tel"
127.0.0.1:6379> sort evszakok alpha
1) "nyar"
2) "osz"
3) "tavasz"
4) "tel"
```

- ☐ Létrehozunk egy halmazt evszakok neven, majd lekérdezzük a típusát
- ☐ Listázzuk a halmaz tartalmát
- ☐ Megjelenítjük a halmaz elemszámát
- ☐ Létrehozunk egy új halmazt kedvencevszak neven
- ☐ Képezzük a két halmaz különbségét
- ☐ Rendezve jelenítjük meg az evszakok halmaz elemeit

# Utasítások V. – Sorted set

ZADD kulcs [opciók] pontszám elem  
[pontszám elem]

Új elem(ek)et ad a sorted set-hez. Ha a set nem létezik, akkor létrehozza. A létező elemek felülíródnak. Opciók: XX – nem ad hozzá új elemet, csak update, LT/GT – csak akkor módosítja a meglévőt, ha az új elem kisebb/nagyobb. A rendezés az adott pontszámok (score) alapján történik

ZCARD kulcs

Visszaadja a sorted set elemszámát

ZCOUNT kulcs min max

Visszaadja a min és max közötti elemek számát

ZINCRBY kulcs n elem

A sorted set adott elemének értékét n-nel növeli

ZINTER | ZUNION | ZDIFF n kulcsok

Visszadja az adott számú halmaz különbségét, unióját, metszetét

ZRANGE kulcs min max [opciók]

Visszadja a sorted set adott indextartományba eső elemeit

# Utasítások V. - Példák

```
C:\Windows\System32\cmd.exe - redis-cli
127.0.0.1:6379>
127.0.0.1:6379>
127.0.0.1:6379> zadd nevsor 1 laci 2 nora 3 zita
(integer) 3
127.0.0.1:6379> type nevsor
zset
127.0.0.1:6379> zcard nevsor
(integer) 3
127.0.0.1:6379> zcount nevsor 2 3
(integer) 2
127.0.0.1:6379> zrange nevsor 0 -1 withscores
1) "laci"
2) "1"
3) "nora"
4) "2"
5) "zita"
6) "3"
127.0.0.1:6379> zincrby nevsor 5 nora
"7"
127.0.0.1:6379> zrange nevsor 0 -1
1) "laci"
2) "zita"
3) "nora"
127.0.0.1:6379>
```

- ☐ Létrehozunk egy sorted set-et nevsor néven (laci, nora, zita)
- ☐ Lekérdezzük a nevsor típusát, elemszámát
- ☐ Lekérdezzük, hogy hány elem van, amelynek pontszáma 2 és 3 közötti
- ☐ Lekérdezzük a nevsor minden pontszámát és elemét
- ☐ Növeljük zita pontszámát 5-tel, utána ismét listázzuk a nevsor elemeit  
➔ a sorrend megváltozik



# Lekérdezések

SELECT \* FROM tábla – minden táblára

SELECT \* FROM tábla WHERE ... LIKE ...

SELECT \* FROM tábla WHERE id=érték

SELECT SUM(), AVG(), MIN(), MAX()  
FROM tábla

SELECT FROM ... GROUP BY

SCAN 0 – Az adatbázisban lévő összes kulcsot megadja

SCAN | SSCAN | HSCAN | ZSCAN iterátor  
[MATCH minta] [COUNT n] [TYPE típus]

HGETALL kulcs

Nincs beépített lehetőség – csak program segítségével (ciklusokkal) oldható meg

Nincs beépített lehetőség – csak program segítségével (ciklusokkal) oldható meg

# Lekérdezések- Példák

```
C:\Windows\System32\cmd.exe - redis-cli
127.0.0.1:6379>
127.0.0.1:6379> scan 0
1) "7"
2) 1) "mainap"
   2) "napok"
   3) "evszakok"
   4) "mykey"
   5) "a"
   6) "hello"
   7) "tegnap"
   8) "kedvenevszak"
   9) "nevsor"
  10) "nev"
127.0.0.1:6379> zscan nevsor 0 match *a*
1) "0"
2) 1) "laci"
   2) "1"
   3) "zita"
   4) "3"
   5) "nora"
   6) "7"
127.0.0.1:6379>
```

- ☐ Lekérdezzük az aktuális adatbázisban tárolt összes kulcsot
- ☐ Listázzuk a nevsor sorted set azon elemeit és pontszámait, ahol az elem nevében van a betű

# Táblák megfelelője Redis-ben

Table  
Products

ID	Name	Description	Price
10001	AAAA	Description of AAAA	100
10002	BBBB	Description of BBBB	200
10003	CCCC	Description of CCCC	200
.	.	.	.
.	.	.	.
10999	XXYZ	Description of XXYZ	500



Sorted Set		Sorted Set	
product_list		product_price	
10001	product:10001	100	product:10001
10002	product:10002	200	product:10002
10003	product:10003	200	product:10003
.	.	.	.
.	.	.	.
10999	product:10999	500	product:10999

Aktiválja a W

[How to get SQL-like Experience with Redis? | Redis Labs](#)

- ❑ A tábla oszlopainak a sorted set adattípus feleltethető meg, a tábla sorainak pedig a hash típus.
- ❑ Az elsődleges kulcsot a kulcs megfelelő elnevezésével lehet megvalósítani, pl. táblanév:elsődleges\_kulcs formában

# Tábla létrehozása - Példa

```
C:\Windows\System32\cmd.exe - redis-cli
127.0.0.1:6379> hmset product:10001 name AAAA description "Description of AAAA"
price 100
OK
127.0.0.1:6379> hmset product:10002 name BBBB description "Description of BBBB "
price 200
OK
127.0.0.1:6379> scan 0 match product*
1) "2"
2) 1) "product:10002"
   2) "product:10002:name"
   3) "product:10001"
   4) "product:10001:description"
127.0.0.1:6379> set product:10001:name AAAA
OK
127.0.0.1:6379> set product:10001:description "Description of AAAA"
127.0.0.1:6379> set product:10001:price 100
OK
127.0.0.1:6379> set product:10002:name BBBB
OK
127.0.0.1:6379> set product:10002:description "Description of BBBB"
OK
127.0.0.1:6379> set product:10002:price 200
OK
127.0.0.1:6379> zadd product_price 100 product:10001 200 product:10002
(integer) 2
```

- ☐ Létrehozzuk a products tábla első két sorának megfelelő adatszerkezetet (soronként) name, description és price mezőkkel
- ☐ Lekérdezzük a product karaktersorozattal kezdődő kulcsokat
- ☐ Létrehozzuk a tábla első két sorának megfelelő adatszerkezetet (mezőnként)
- ☐ Létrehozzuk a tábla price oszlopának megfelelő sorted set-et

# Kapcsolat létrehozása Redis-ben

employee_id	first_name	last_name	address
1	John	Doe	New York
2	Benjamin	Button	Chicago
3	Mycroft	Holmes	London

payment_id	employee_id	amount	date
1	1	50,000	01/12/2017
2	1	20,000	01/13/2017
3	2	75,000	01/14/2017
4	3	40,000	01/15/2017
5	3	20,000	01/17/2017
6	3	25,000	01/18/2017

[From RDBMS to Key-Value Store: Data Modeling Techniques | by Wishmitha S. Mendis | Medium](#)






- ❑ A kapcsolatot a kulcs megfelelő elnevezésével lehet létrehozni
- ❑ Pl: payment:2:1, ahol 2 – elsődleges kulcs, 1 – idegen kulcs

# Kapcsolat létrehozása- Példa (Előző dia alapján)

```
C:\Windows\System32\cmd.exe - redis-cli
127.0.0.1:6379>
127.0.0.1:6379> set employee:1:first_name John
OK
127.0.0.1:6379> set employee:1:last_name Doe
OK
127.0.0.1:6379> set employee:1:address "New York"
OK
127.0.0.1:6379> set payment:1:1:amount 50000
OK
127.0.0.1:6379> set payment:1:1:date "01/12/2017"
OK
127.0.0.1:6379> set payment:2:1:amount 20000
OK
127.0.0.1:6379> set payment:2:1:date "01/13/2017"
OK
127.0.0.1:6379>
```

- ☐ Létrehozzuk (mezőnként) az employee tábla első két sorának megfelelő kulcs-érték párokat
- ☐ Létrehozzuk (mezőnként) a payment tábla első két sorának megfelelő kulcs-érték párokat

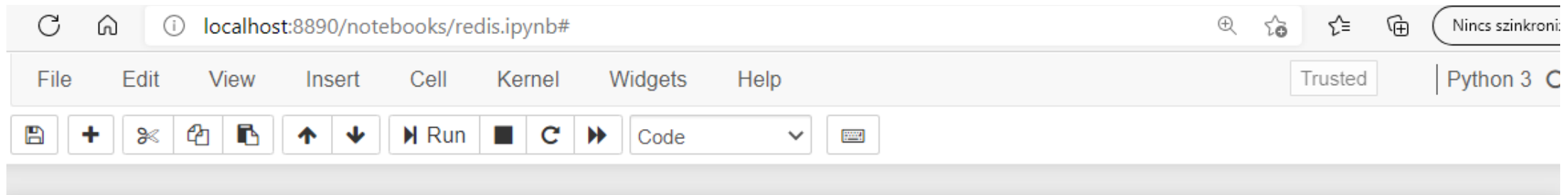
# Redis – kiegészítő komponensek - példák

Redisearch		Full-Text search over Redis	<a href="#">dvirsky</a> <a href="#">RedisLabs</a>	Redis Source Available License
neural-redis		Online trainable neural networks as Redis data types.	<a href="#">antirez</a>	BSD
RedisGraph		A graph database with a Cypher-based querying language using sparse adjacency matrices	<a href="#">swilly22</a> <a href="#">RedisLabs</a>	Redis Source Available License
redisSQL		A redis module that provides a full SQL capabilities embedding SQLite	<a href="#">siscia</a> <a href="#">RedBeardLab</a>	AGPL-3.0
RedisJSON		A JSON data type for Redis	<a href="#">itamarhaber</a> <a href="#">RedisLabs</a>	Redis Source Available License

→ Keresési, aggregálási lehetőségek

→ SQL lekérdezési lehetőség

# Redis elérés Python-ból



```
In [ ]: !pip install redis
```

```
In [ ]: import redis
```

```
In [ ]: #a host felhős rendszerénél az endpoint értéke, on-premise esetben localhost  
#a port értéke on-premise rendszerénél 6379  
#a jelszó on-premise rendszerénél - ha nem állítottuk be - nem kell  
  
r = redis.Redis(host='', port= , password='')  
r.set('hello', 'world')  
print(r.get('hello'))
```



# Feladat I.

Csatlakozzon a Redis Cloud-hoz ([Redis Labs | The Home of Redis](https://redislabs.com/))!

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

# Feladat II.

Csatlakozzon a Redis Cloud adatbázisához a következő (parancssorba beírt) utasítás segítségével:

*redis-cli -h endpoint -p port -a default\_password*, ahol a megfelelő helyekre az előző feladatban kiolvasott endpoint, port és default password értékeit kell behelyettesíteni

- a. A csatlakozás után adja ki az `info server` parancsot
- b. A parancs futtatásának eredményeként kapott sorokat másolja be a válaszhoz

# Feladat III.

A redis-cli-ben hozzon létre három új kulcs-érték párt: ev 2021, hónap 4, nap 26

a. A kulcsok lejáratí ideje 10 másodperc legyen

b. A kulcsok létrehozásához szükséges utasításokat adja meg válaszként!

# Feladat IV.

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

# Feladat V.

A redis-cli-ben hozzunk létre új sorted set-et személyek néven, ahol az egyes pontszámok az életkorok legyenek: Andras 15 éves, Peter 20 éves, Juli 18 éves

- a. Kérdezzük le a 16-18 pontszámok közötti személyek számát!
- b. A sorted set-et létrehozó és a lekérdező utasításokat (két utasítás) adjuk meg válaszként!

# Feladat VI.

A redis-cli-ben a korábban létrehozott személyek sorted set-ben növeljük Juli életkorát 10 évvel, majd ismét listázzuk a személyek sorted set elemeit!

a. A szükséges utasításokat (két utasítás) adjuk meg válaszként!

# Feladat VII.

Hozza létre mezőnként és soronként külön kulcsok segítségével a dolgozo tábla következő rekordjait:

nev	munkakor	kod
Nagy Eva	titkarno	1
Kiss Ilona	konyvelo	2

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

# Feladat VIII.

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!



# Feladat IX.

A redis-cli-ben hozzunk létre új két új halmaz típusú kulcsot:

numbers1: 10, 20, 30, 40, 50, 60 és numbers2: 15, 30, 45, 60

a. Képezzük a halmazok metszetét!

b. A halmazokat létrehozó, és a metszetüket lekérdező utasításokat (3 utasítás) adja meg a válaszhoz!

# Feladat X.

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

- a. Ezután adjuk meg a létező kulcsok számát a dbsize utasítás segítségével!
- b. A parancsokat és eredményüket mutató képernyőrészletet adjuk meg a válaszhoz kép formájában!



**Köszönöm  
a figyelmet!**