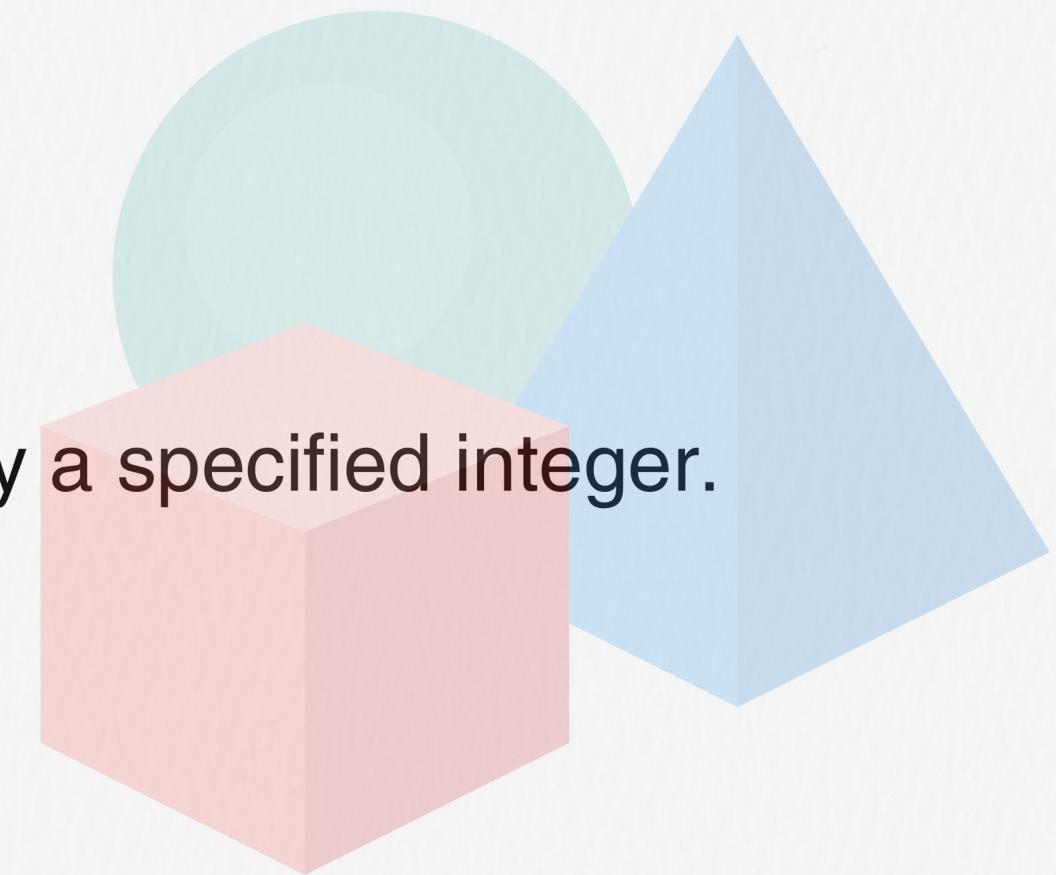


- In Redis, a hash is a data structure that maps fields to values. It is similar to a **dictionary** or a **map** in other programming languages.
- Hashes are useful for **representing objects with multiple attributes**, as they allow you to store and retrieve several related fields with a single key
- **Commands**
  - **HSET**: Set the value of a field in a hash.
  - **HGET**: Get the value of a field in a hash.
  - **HDEL**: Delete one or more fields from a hash.
  - **HGETALL**: Get all the fields and values in a hash.
  - **HINCRBY**: Increment the value of a field in a hash by a specified integer.
  - **HKEYS**: Get all the fields in a hash.
  - **HVALS**: Get all the values in a hash.
  - **HLEN**: Get the number of fields in a hash.

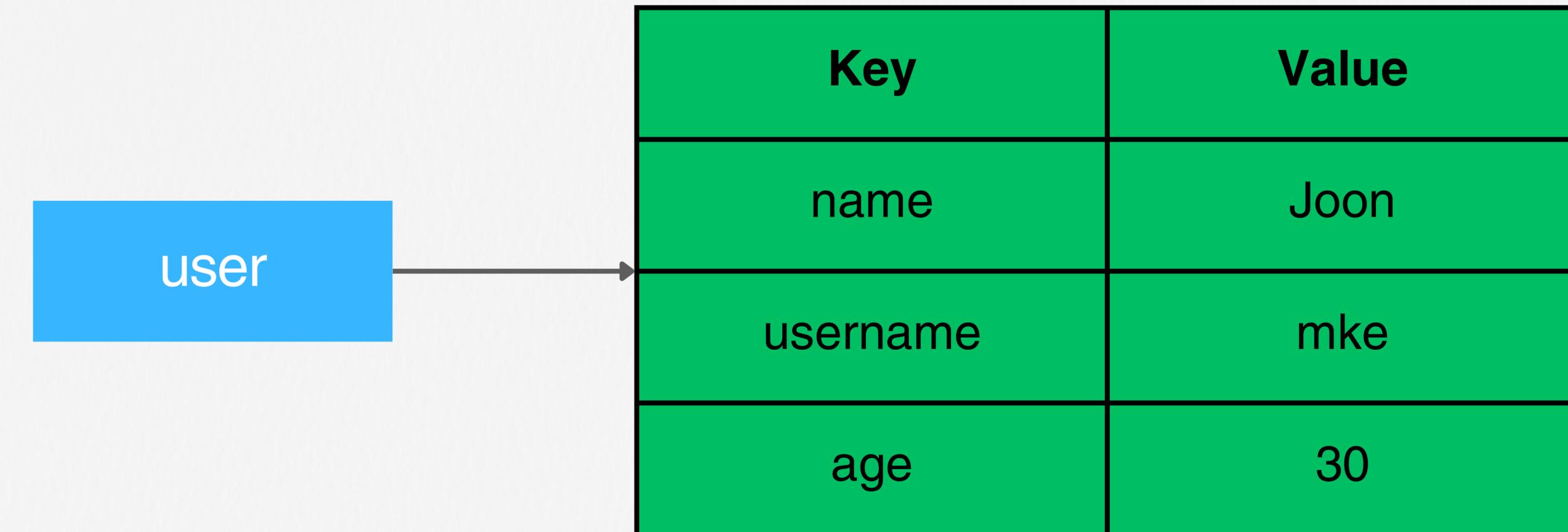


# HSET

SILICON VALLEY ENGINEER



```
1 127.0.0.1:6379> HSET user name Joon username mke age 30
2 (integer) 3
```

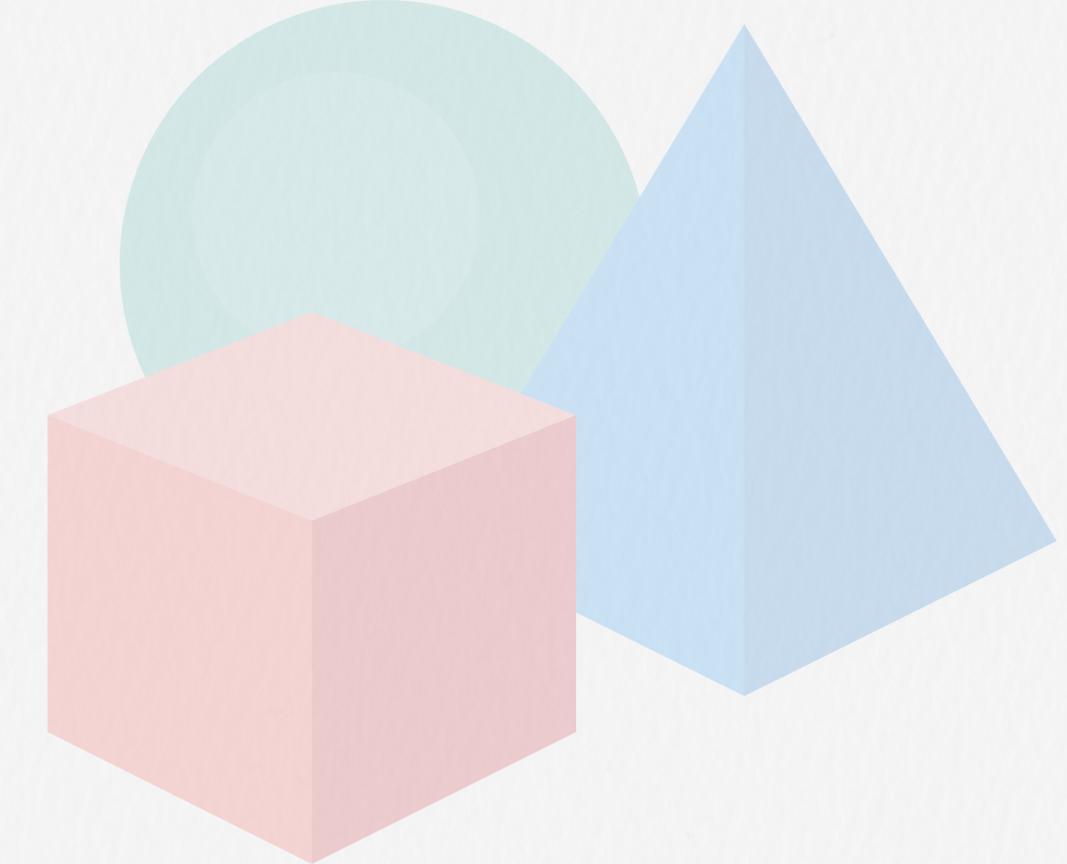


# Limitation

SILICON VALLEY ENGINEER

- No nested hashes

```
1  {
2      "name": "Joon",
3      "username": "mke",
4      "age": 30
5  }
6
7  {
8      "name": "Joon",
9      "username": "mke",
10     "age": 30,
11     "loc": {
12         "country": "usa",
13         "state": "california"
14     }
15 }
```



# But... kind of nested hash(not official)

SILICON VALLEY ENGINEER

- the **colon (:) is used as a delimiter or separator to structure the key** in Redis.
- This is a **common convention** used in Redis and many other key-value stores to organize keys in a hierarchical or namespace-like manner.

```
● ● ●  
1 # HSET  
2 127.0.0.1:6379> HSET user name Joon username mke age 30  
3 (integer) 3  
4  
5 # HSET nested hash(not official)  
6 127.0.0.1:6379> HSET user:address street "123 Main St" city "Anytown" zip "12345"  
7 (integer) 3
```