UniRitter
Laureate International Universities®

Banco de Dados NoSQL

(NS0203)

Prof. Giovane Barcelos
giovane_barcelos@uniritter.edu.br

Redis – Tipos de Dados
Conteúdo Programático

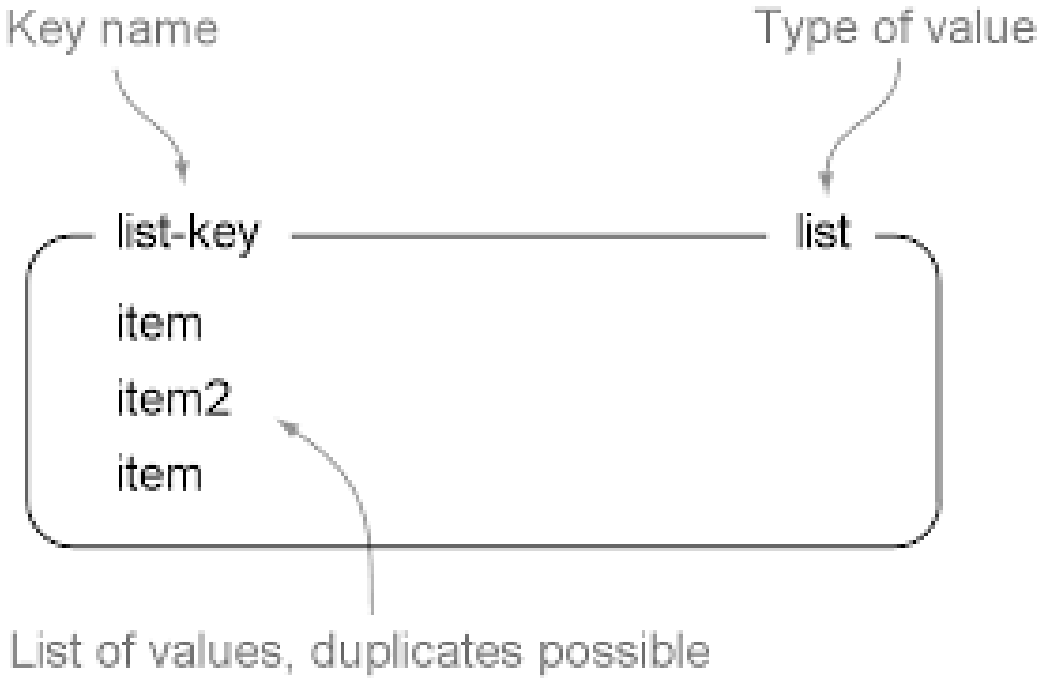
1. Introdução ao Redis
2. Primeiros Passos
3. Tipos de Dados
4. Persistência de Dados
5. Aplicação prática com Node.js

Redis – Tipos de Dados
Itens Abordados

1. Listas (***Lists***: *LPUSH, RPUSH, LRANGE, LREM*)
2. Conjuntos (Sets*: SADD, SREM*)
3. Conjuntos Ordenados *(Sorted Sets: ZADD, ZRANGE)*
4. *Hashes (HSET, HGET, HGETALL)*

3. Tipos de Dados - Listas
O que são Listas?

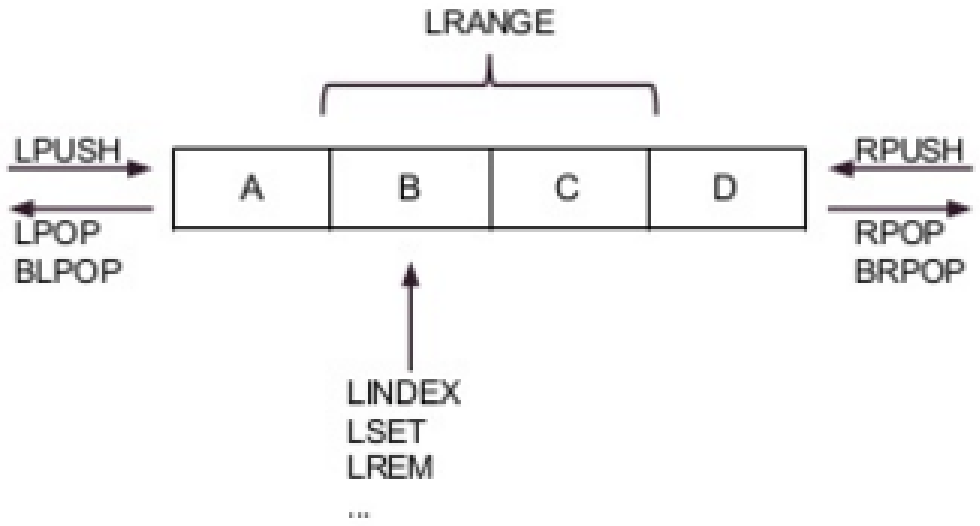
- Listas ou **grupos de *strings***
- Ordenado por **ordem de inserção**
- Os elementos podem ser inseridos no **início ou no fim da lista**
- Muitas vezes usado para pesquisas de **produtor / consumidor**



3. Tipos de Dados - Listas
Como inserir elementos na lista?

- ***LPUSH***: insere novo elemento no **início** (**esquerda**)
- ***RPUSH***: insere novo elemento no **fim** (**direita**)

- Uma nova lista é criada quando *LPUSH* ou *RPUSH* é executada sobre uma lista vazia
- A **chave é removida da lista** de chaves se uma operação sobre esta esvaziar a lista



3. Tipos de Dados - Listas
Exemplo de *LPUSH* e *RPUSH*

- ***LPUSH*** **mylist a**
  "a"
- ***LPUSH*** **mylist b**
  "b", "a"
- ***RPUSH*** **mylist "c"**
  "b", "a", "c"

3. Tipos de Dados - Listas
LRANGE

- **Retorna os elementos especificados na lista**
- **Os limites da lista são índices baseados em zero**
- **Os limites podem ser negativos indicando limites a partir partir do final da lista**

3. Tipos de Dados - Listas
Exemplo de *LRANGE*

["Penélope", "Medinho", "Vigarista"]

- ***LRANGE*** **amigos 0 -1**
  1. Penélope
  2. Medinho
  3. Vigarista

- ***LRANGE*** **amigos 1 2**
  1. Medinho
  2. Vigarista

3. Tipos de Dados - Listas
LLEN

- ***LLEN*** **retorna o comprimento da lista**

- ***LLEN*** **amigos**
  3

- **Se a lista estiver vazia, 0 é retornado**

3. Tipos de Dados - Listas
LPOP

- **Remove e retorna o primeiro elemento de uma lista**
- ***LPOP* amigos**

  Penélope

3. Tipos de Dados - Listas
RPOP

- **Remove e retorna o último elemento de uma lista**
- ***RPOP* amigos**

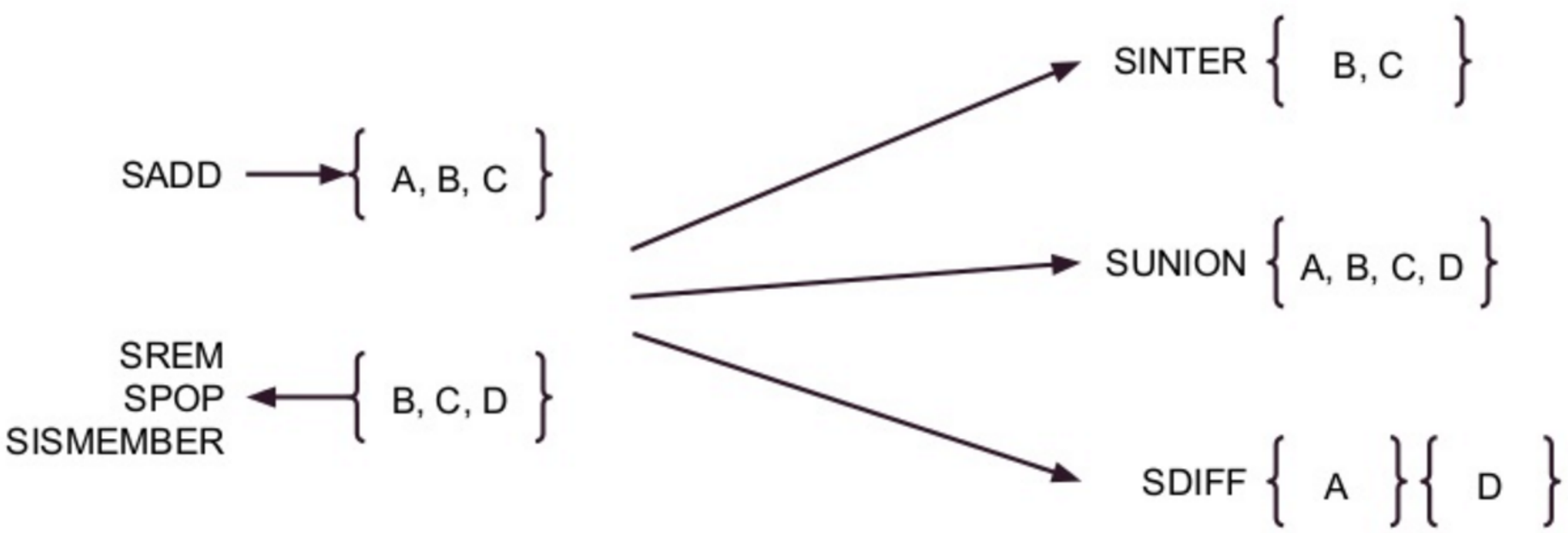
  Vigarista

3. Tipos de Dados - Listas
Vamos treinar com alguns comandos de Listas

```
LPUSH amigos "Medinho"
LPUSH amigos "Dick Vigarista"
RPUSH amigos "Muttley"
LRANGE amigos 0 -1
LRANGE amigos 1 2
LLEN amigos
LPOP amigos
RPOP amigos
LRANGE amigos 0 -1
LPUSH amigos "Penelope Charmosa"
LPUSH amigos "Bombarda"
LRANGE amigos 0 -1
LINSERT amigos BEFORE "Medinho" "Chorão"
LRANGE amigos 0 -1
```

3. Tipos de Dados - Conjuntos
O que são Conjuntos?

- **Coleções não ordenadas** de ***strings***
- **Podem adicionar, remover e testar**
- **NÃO permitem membros repetidos**
- **Suporta comandos do lado do servidor para calcular sobre conjuntos a partir de conjuntos existentes**



3. Tipos de Dados - Conjuntos
SADD && SREM

- ***SADD***
  - ✔ Adiciona valores a um conjunto
  - ✔ Os valores existentes são ignorados

- ***SREM***
  - ✔ Remove valores de um conjunto

- ***SADD*** fabricantes "Toyota" // Adiciona Toyota
- ***SREM*** fabricantes "Honda" // Remove Honda

3. Tipos de Dados - Conjuntos
SISMEMBER

- **Testa se o valor encontra-se no conjunto**
- **Retorna 1 se encontrar o valor e 0 se não encontrar**

SISMEMBER **fabricantes "Toyota"**

3. Tipos de Dados - Conjuntos
SMEMBERS

- **Retorna uma lista com todos os membros do conjunto**

***SMEMBERS* fabricantes**

3. Tipos de Dados - Conjuntos
SCARD

- **Retorna a contagem de membros / elementos em um conjunto**
- **Retorna 0 se a chave não existir**

***SCARD* fabricantes**

3. Tipos de Dados - Conjuntos
SMOVE

- **Move um membro de um conjunto para outro**

SMOVE **pessoas usuarios “Penélope Charmosa”**

3. Tipos de Dados - Conjuntos
SUNION

- **Combina dois ou mais conjuntos e retorna uma lista de membros**

  SUNION **fabricantes fabricantesCaminhao**

3. Tipos de Dados - Conjuntos
SDIFF

- **Retorna os membros do conjunto resultante da diferença entre o primeiro e todos os conjuntos sucessivos**
- **As chaves que não existem são considerados conjuntos vazios**

SDIFF **key1 key2**

3. Tipos de Dados - Conjuntos
SRANDMEMBER

- **Retorna um membro randômico do conjunto**
- **Parâmetro opcional para retornar uma contagem especificada**

SRANDMEMBER **fabricantes**
SRANDMEMBER **fabricantes 3**

3. Tipos de Dados - Conjuntos
SPOP

- **Remove e retorna um membro aleatório de um conjunto especificado**
- **Como *SRANDEMEMBER*, um segundo parâmetro é permitido para especificar uma contagem de membros**

***SPOP* fabricantes**

***SPOP* fabricantes 3**

3. Tipos de Dados - Conjuntos
Vamos treinar com alguns comandos de Conjuntos

```
SADD fabricantes "Toyota"
SADD fabricantes "Ford"
SADD fabricantes "Chevy"
SADD fabricantes "Honda"
SISMEMBER fabricantes "Honda"
SISMEMBER fabricantes "Hondas"
SMEMBERS fabricantes
SADD fabricantes "Honda"
SMEMBERS fabricantes
SCARD carmakes
SADD meusCarros "Acura"
SMOVE fabricantes meusCarros"Toyota"
SMEMBERS meusCarros
SMEMBERS fabricantes
SRANDMEMBER fabricantes
SPOP fabricantes
```

3. Tipos de Dados – Conjuntos Ordenados
O que são conjuntos ordenados?

- **Como os conjuntos não são ordenados, eles podem causar problemas para alguns projetos**
- **Os Conjuntos Ordenados foram criados para resolver esse problema**
- **Cada membro está associado a uma "pontuação"**
- **Pode acessar dados muito rapidamente**
- **Como conjuntos, os elementos só podem aparecer uma vez**

ZADD	ZRANK
ZCARD	ZREM
ZCOUNT	ZSCORE
ZRANGE	ZSCAN

3. Tipos de Dados – Conjuntos Ordenados
Propriedades das pontuações

- **Pontuação é requerida**
- **Deve ser um float / número**
- **Pontuação de 500 = 500,0**
- **A pontuação NÃO é única, os valores são**

3. Tipos de Dados – Conjuntos Ordenados
ZADD & ZREM

- **ZADD**: adiciona valores a um conjunto ordenado
- **ZREM**: remove valores de um conjunto ordenado

ZADD pessoas 1970 "Dick Vigarista"
ZREM pessoas "Dick Vigarista"

3. Tipos de Dados – Conjuntos Ordenados
ZRANGE

- ***ZRANGE*: funciona como *LRANGE* para listas. Obtém valores dentro de um intervalo especificado. Ordenado do menor ao maior em relação a pontuação**
- ***ZREVRANGE*: o mesmo que o ZRANGE, exceto que é ordenado do maior para o menor**

***ZRANGE* pessoas 0 -1**
***ZRANGE* pessoas 2 4**

3. Tipos de Dados – Conjuntos Ordenados
ZRANGEBYSCORE

- **Funciona como ZRANGE, mas usa uma variedade de valores de pontuação**

  ***ZRANGEBYSCORE* pessoas 1970 1990**

  Obtém as pessoas com uma pontuação entre 1970 e 1990

3. Tipos de Dados – Conjuntos Ordenados
ZRANK

- ***ZRANK*: retorna a classificação de um membro com pontuações ordenadas do maior para o menor. O Rank é baseado em 0**
- **ZREVRANK: obtém a classificação de um membro na ordem inversa**

***ZRANK* pessoas "Dick Vigarista"**
***ZREVRANK* pessoas "Dick Vigarista"**

3. Tipos de Dados – Conjuntos Ordenados
ZCARD & ZCOUNT

- ***ZCARD*: retorna o número de membros no conjunto ordenado**

  ***ZCARD* pessoas**

- ***ZCOUNT*: retorna o número de elementos no conjunto ordenado com uma pontuação entre mínimo e máximo**

  ***ZCOUNT* pessoas (1, 3)**

3. Tipos de Dados – Conjuntos Ordenados
ZINCRBY

- **Incrementa a pontuação do membro no conjunto ordenado**
- **Se o membro não existir, ele será adicionado com o valor do incremento como sua pontuação**
- **O valor da pontuação deve ser a representação de uma sequência de caracteres de um valor numérico e aceita números de ponto flutuante de precisão dupla. É possível fornecer um valor negativo para diminuir a pontuação.**

ZINCRBY **pessoas 1 "Dick Vigarista"**

3. Tipos de Dados – Conjuntos Ordenados
ZSCORE

- **Retorna a pontuação de um membro**
- **Se o membro não existir, nulo é retornado**

***ZSCORE* pessoas "Dick Vigarista"**

3. Tipos de Dados – Conjuntos Ordenados
Vamos treinar com alguns comandos de Conjuntos

ZADD **pessoas 1973 "Dick Vigarista"**
ZADD **pessoas 1985 "Penélope Charmosa"**
ZADD **pessoas 1990 "Medinho"**
ZRANGE **pessoas 0 -1**
ZRANGEBYSCORE **pessoas 1970**
ZRANK **pessoas "Penélope Charmosa"**
ZRANK **pessoas "Dick Vigarista"**
ZRANK **pessoas "Medinho"**
ZREVRANK **pessoas "Medinho"**
ZCARD **pessoas**
ZINCRBY **pessoas 1 "Dick Vigarista"**
ZSCORE **pessoas "Dick Vigarista"**

3. Tipos de Dados – Hash
O que é um Hash?

- **Mapas** entre campos de *string* e valores de sequencia de caracteres
- **Perfeito para representar objetos**
- Muito **compacto**

3. Tipos de Dados – Hash
HSET

- Define um campo no *hash*
- Se uma chave não existe, é criada uma nova chave que contém um *hash*
- Se o campo existe no *hash*, ele é substituído
- 1 é retornado se campo for um novo campo no *hash* e o valor foi definido
- 0 (zero) é retornado se o campo já existe no *hash* e o valor foi atualizado

*HSET usuario*1 nome "Dick Vigarista"

3. Tipos de Dados – Hash
HMSET

- **Define vários campos para seus respectivos valores**
- **Substitui todos os campos existentes no *hash***

HMSET **usuario2 nome "Medinho"**
e-mail "medinho@gmail.com"
idade "25"

3. Tipos de Dados – Hash
HGET

- **Obtém um valor associado a um campo em um *hash***
- **Retorna valor ou nulo se o campo não estiver presente**

***HGET* usuario2 nome**

3. Tipos de Dados – Hash
HMGET

- **Retorna valores associados a vários campos em um *hash***
- **Para cada campo que não existe, um valor nulo é retornado**

HMGET **usuario2 nome idade**

3. Tipos de Dados – Hash
HGETALL

- **Obtém todos os campos e valores em um *hash***
- **Retorna cada nome de campo seguido de seu valor**

***HGETALL* usuario2**

3. Tipos de Dados – Hash
HDEL

- **Remove os campos especificados do *hash***
- **Os campos especificados que não existem são ignorados**
- **Retorna o número de campos que foram removidos do *hash***

***HDEL* usuario2 idade**

3. Tipos de Dados – Hash
HEXISTS

- **Verifica um campo existente em um *hash***
- **Retorna 1 se o *hash* contém o campo e 0 se não**

HEXISTS **usuario2 nome**

3. Tipos de Dados – Hash
HINCRBY

- **Incrementa o número ordenado no *hash***
- **Se a chave não existir, é criada uma nova chave que contém um *hash***
- **Se o campo não existir, o valor é definido como 0 (zero) antes da operação ser executada**

HINCRBY **usuario3 idade 1**

3. Tipos de Dados – Hash
HKEYS

- **Retorna todos os nomes dos campos no *hash***

***HKEYS* usuario2**

3. Tipos de Dados – Hash
HLEN

- **Retorna o número de campos contidos no *hash***
- **Retorna 0 se a chave não existir**

***HLEN* usuario2**

3. Tipos de Dados – Hash
HVALS

- **Retorna todos os valores do *hash***

  ***HVALS* usuario2**

3. Tipos de Dados – Hash
HSTRLEN

- **Retorna o comprimento da sequencia do valor associado ao campo no *hash***
- **Se a chave ou o campo não existirem, 0 é retornado**

***HSTRLEN* usuario2 nome**

3. Tipos de Dados – Hash
Vamos treinar com alguns comandos *hash*

```
HSET usuario:dick nome "Dick Vigarista"
HGET usuario:dick nome
HMSET usuario:charmosa nome "Penélope Charmosa"
        email "charmosa@gmail.com" idade "30"
HGET usuario:charmosa idade
HGET usuario:charmosa email
HGET usuario:charmosa nome
HMGET usuario:charmosa nome idade
HGETALL usuario:charmosa
HKEYS usuario:charmosa
HVALS usuario:charmosa
HINCRBY usuario:charmosa idade
HVALS usuario:charmosa
HDEL usuario:charmosa idade
HGETALL usuario:charmosa
HLEN usuario:charmosa
```

Lembre-se

"Que a força esteja contigo."

Star Wars