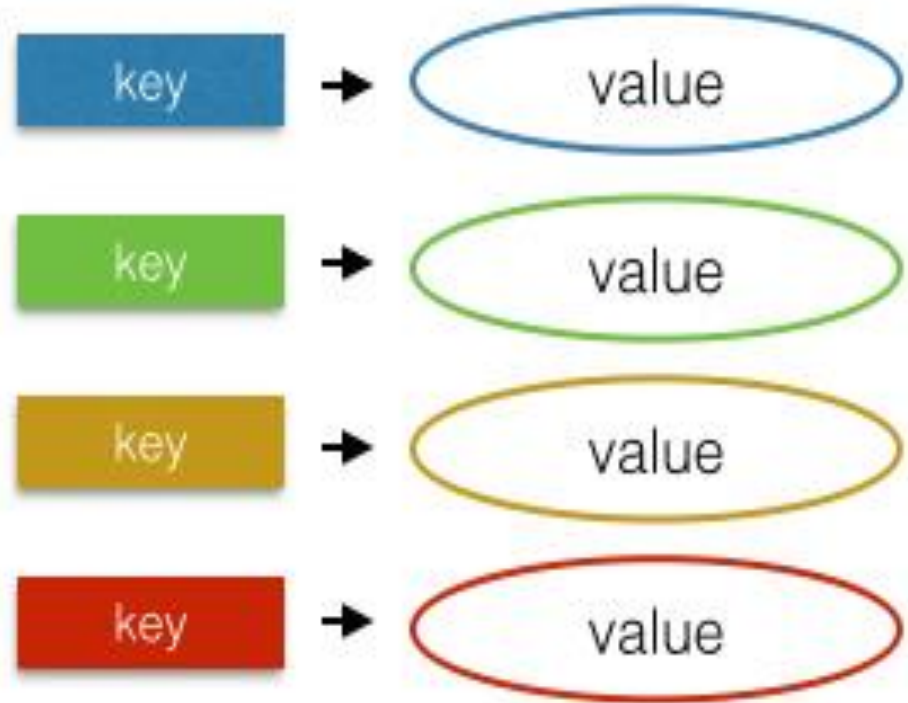


# NOSQL

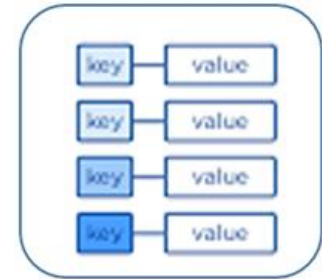
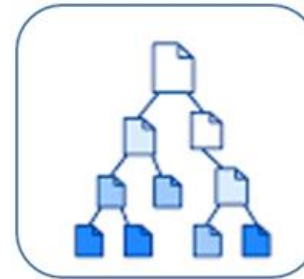
## CHAVE-VALOR

## Key-value



# Modelos de Dados

- Modelos de Dados mais comuns:
  - **Chave-valor**
  - Orientado a Documentos
  - Orientado a Colunas
  - de Grafo



# Chave-valor

- É o tipo de BD NoSQL mais simples de usar a partir da perspectiva de uma API.
- O usuário pode:
  - Obter o valor de uma determinada chave
  - Inserir um valor para uma dada chave
  - Remover uma chave do BD
- Valor é um blob ou texto ou JSON ou XML que o BD guarda sem se preocupar ou saber o que há dentro dele
  - É responsabilidade da aplicação entender o que está armazenado
- Já que o acesso é sempre feito pela chave primária:
  - Um bom desempenho pode ser alcançado
  - Podem ser escaláveis facilmente



# Chave-valor

## Prós

- Alta escalabilidade
- Baixo custo de armazenamento
- Velocidade de consulta
  - Alguns rodam 100% em memória RAM
- Curva de aprendizado rápida

## Contras

- Consultas apenas pela chave
  - Precisa-se conhecer todas as chaves e seus tipos
- Relacionamentos entre dados quase inexistente
- Operações com múltiplas chaves precisa ser feita na aplicação



# Prática

## NoSQL Chave-valor

Utilizando o Redis



# Redis

- Acrônimo para **RE**mote **D**ictionary **S**erver
  - Desenvolvido a partir de 2009
  - Criado para tratar problemas de logging
- Armazena os dados em memória
  - Não precisa de acesso a disco para leitura ou escrita
  - Mas tem a opção de persistir os dados em disco (snapshots individuais ou agendados)



# Redis

- Ferramenta *open source* desenvolvida em Ansi C
- Download disponível em <https://redis.io/>
- Executa em Linux, BSD e OS X
  - Não há suporte para Windows, mas tem versões que executam disponíveis [AQUI](#)



# Redis

- Permite representação no modelo Chave-valor
  - Disponibiliza representação de dados simples como números e textos até listas e conjuntos
  - Também permite representação de bitmaps e dados geoespaciais





# Redis

## Quem usa?

- Twitter
- GitHub
- Pinterest
- Snapchat
- Stackoverflow
- ...

## Quando usar?

- Cache
- Memory Broker
- Replicação de Dados
- Classificação de Jogadores
- Armazenamento de imagem
- IoT
- ...

## Concorrentes

- Riak
- DynamoDB
- Memcache
- Berkeley DB
- ...



# Chave-valor do Redis

- **CHAVE:** é uma string
  - Case-sensitive (nome é diferente de Nome)
  - Pode conter espaços e caracteres especiais
  - Evitar nomes longos
- **VALOR:** é um array de bytes que pode representar algum destes tipos:
  - Números (representados internamente como strings), String, Hash, Lista, Conjuntos, Mapa de bits, Dados geoespaciais, ...

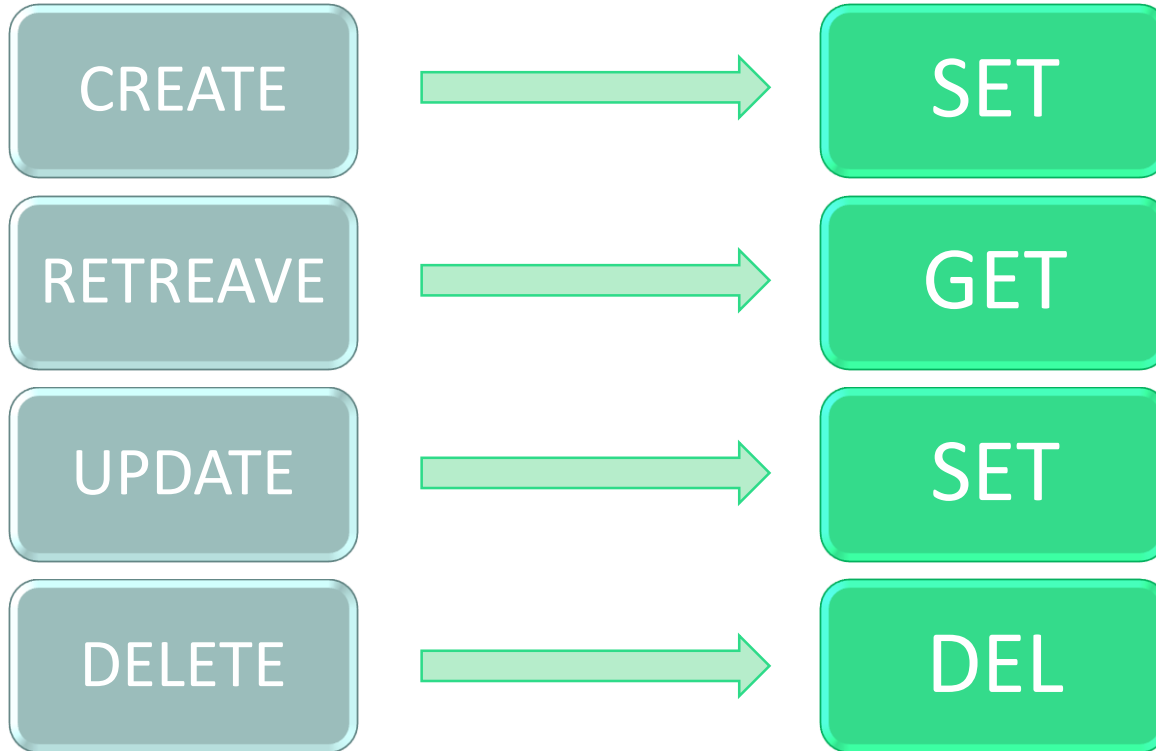


# Operações Básicas

CRUD, Lists, Hash, Set



# Operações CRUD



# Comando SET

## SET chave valor

- Guarda o dado “valor” na “chave” dada.
- Dois comportamentos possíveis:
  1. Criar a chave caso ela não tenha sido criada anteriormente
  2. Atualiza o valor dela caso ela já exista.



# Comando SET

- **Exemplos**

1. Criar chaves contendo o nome, a idade e a formação superior da professora

- > set nomeProfa "Carol Torres"

- > set idadeProfa 34

- > set formacaoProfa "Ciências da Computação"

2. Carol casou e adotou o sobrenome do marido, atualizar o nome dela.

- > set nomeProfa "Carol Ribeiro"



# Comando GET

## GET chave

- Retorna o valor associado à “chave” dada.
  - Retorno será nulo (nil) caso a chave não exista

- **Exemplos**

1. Qual a idade e a formação da professora?

```
> get idadeProfa           // 34
```

```
> get formacaoProfa        // "Ciências da Computação"
```

```
> Get formaçãoProfa        // (nil)
```

# Comando DEL

## DEL chave

- Faz a deleção da “chave” determinada
- **Exemplos**
  1. Apagar a formação e a idade da professora
    - > del formacaoProfa // (integer) 1
    - > del IdadeProfA // (integer) 0





# Comandos Auxiliares

## EXISTS chave

- Retorna a existência ou não de uma chave
  - 0 (FALSE) ou 1 (TRUE)

## TYPE chave

- Retorna o tipo de dado guardado na chave dada
  - *none* será o retorno se a chave não existir

## INCR chave

- Trata o valor na string como inteiro e faz o incremento do valor
  - Mostra o novo valor
  - Erro se tentar com um valor não inteiro

LINK: <https://redis.io/commands/?group=string>

# Comando Auxiliares

- **Exemplos**

1. A chave NomeProfa existe?

```
> EXISTS NomeProfa // 0
```

2. Carol fez aniversário, a idade dela precisa ser atualizada

```
> INCR idadeProfa // 35
```

3. Qual o tipo da chave idadeProfa?

```
> TYPE idadeProfa // "string"
```



# Redis Lists

- As listas no Redis são implementadas como listas ligadas
  - Assegurando maior velocidade de escrita no começo ou no fim da lista
  - Comprometendo o tempo de leitura de uma posição específica



# Redis Lists

**RPUSH chave  
v1 v2 v3 ...**

- adiciona os valores dados ao final da lista associada à chave dada

**LPUSH chave  
v1 v2 v3 ...**

- funciona como o RPUSH, mas adiciona os valores no começo da lista

**LRANGE chave  
começo fim**

- retorna uma sublista a partir das posições fornecidas

**LINDEX chave  
posição**

- retorna o valor da posição dada

**LPOP chave**

- retorna o valor do começo da lista e o remove

**RPOP chave**

- retorna o valor do fim da lista e o remove.



# Redis Lists

- **Exemplos:**

- Criar uma lista contendo a ordem de apresentação do projetos finais da disciplina de BD2 (EqA, EqD, EqC)

- > RPush ordem EqA EqD EqC

- A equipe B chegou atrasada e será a primeira a se apresentar

- > LPush ordem EqB

- Quais as duas próximas apresentações?

- > LRange ordem 0 1



# Redis Lists

- **Exemplos:**

- Qual a última equipe a se apresentar?  
> LRANGE ordem 3 100
- A próxima equipe se apresentará agora.  
> LPOP ordem
- A última equipe desistiu de se apresentar pois ficaria tarde demais o horário.  
> RPOP ordem

# Redis Hash

- Estrutura que normalmente é mais utilizada para guardar objetos no Redis
  - É possível parear campo a campo do seu objeto com o objeto guardado no Redis
- Os objetos não seguem um esquema de dados fixo predeterminado
  - Não necessariamente possui os mesmo campos que os objetos que estão nas outras chaves

# Redis Hash

## **HSET chave campo valor**

- salva o valor dado no campo dado no hash associado à chave dada

## **HMSET chave c1 v1 c2 v2 ...**

- funciona como vários HSETs seguidos e é útil para salvar um objeto de uma só vez no Redis.

## **HDEL chave campo**

- deleta o campo que está salvo dentro do objeto hash associado à chave dada.

## **HGETALL chave**

- retorna os campos e os valores associados à chave dada

## **HGET chave campo**

- retorna o valor associado ao campo da chave dada





# Redis Hash

Matrícula: 3857  
Nome: Carolina  
Idade: 34  
status: ativo

- **Exemplos:**

- Adicionar os dados da professora em um objeto hash.
  - > HMSET prof-3857 matricula 3857 nome Carol idade 34 status ativo
- Carol foi desativada do sistema.
  - > HSET prof-3857 status desativado
- Qual o status e a idade de Carol?
  - HGET prof-3857 idade
  - HGET prof-3857 status
- Apagar o campo matrícula do objeto.
  - > HDEL prof-3857 matricula
- Quais as informações salvas no objeto?
  - > HGETALL prof-3857



# Redis Set

- Representam conjuntos de valores que **não se repetem**.
  - Ao fazer uma adição de um elemento já existe em um set, o comando set é ignorado.
- Oferecem operação entre conjuntos (união, interseção e diferença)

# Redis Set

## **SADD chave valor**

- tenta adicionar o valor dado à chave dada
- 1 (conseguiu adicionar), 0 (não conseguiu)

## **SMEMBERS chave**

- retorna todo o set

## **SISMEMBER chave valor**

- retorna se o valor dado está ou não no set.

## **SREM chave valor**

- tenta remover o valor dado do set.



# Redis Set

**SUNION chave1  
chave2**

- retorna a união dos sets das chaves dadas

**SINTER chave1  
chave2**

- retorna a interseção dos sets das chaves dadas

**SDIFF chave1  
chave2**

- retorna a diferença entre os sets das chaves dadas



# Redis Set

- **Exemplos:**

- Criar um conjunto contendo os nomes das disciplinas que a professora Carol ministra (BD, BD1, BD2)
  - > SADD disciplinasCarol BD BD1 BD2
- Carol começará a ministrar “Testes de Software”
  - > SADD disciplinasCarol "Testes de Software"
- Quais as disciplinas ministradas pela professora Carol?
  - > SMEMBERS disciplinasCarol
- Josino é outro professor e ele ensina BD1, Web 2 e PPP3
  - > SADD disciplinasJosino BD1 "Web 2" PPP3



# Redis Set

- **Exemplos:**

- Carol ministra BD3?

> SISMEMBER disciplinasCarol BD3 // 0

- Josino não ensina mais Web 2

> SREM disciplinasJosino "Web 2"

- Quais disciplinas que Carol e Josino ensinam ao mesmo tempo?

> SINTER disciplinasCarol disciplinasJosino

- Quais as disciplinas que Carol e Josino ensinam ao todo?

> SUNION disciplinasCarol disciplinasJosino



# Links interessantes

- Comandos Redis: <https://redis.io/commands>
  - LIST: <https://redis.io/commands#list>
  - HASH: <https://redis.io/commands#hash>
  - SET: <https://redis.io/commands#set>
- Testar os comandos sem necessidade de instalar, inclusive oferecendo tutorial: <https://try.redis.io/>



# NOSQL

## CHAVE-VALOR

## Key-value

