

key通用操作

key 特征

- key是一个字符串，通过key获取redis中保存的数据



key应该设计哪些操作？

- 对于key自身状态的相关操作，例如：删除，判定存在，获取类型等
- 对于key有效性控制相关操作，例如：有效期设定，判定是否有效，有效状态的切换等
- 对于key快速查询操作，例如：按指定策略查询key
-

key通用操作

key 基本操作

- 删除指定key

```
del key
```

- 获取key是否存在

```
exists key
```

- 获取key的类型

```
type key
```

key 扩展操作 (时效性控制)

- 为指定key设置有效期

```
expire key seconds
```

```
pexpire key milliseconds
```

```
expireat key timestamp
```

```
pexpireat key milliseconds-timestamp
```

- 获取key的有效时间

```
ttl key
```

```
pttl key
```

- 切换key从时效性转换为永久性

```
persist key
```

key 其他操作

- 为key改名

```
rename key newkey
```

```
renamenx key newkey
```

- 对所有key排序

```
sort
```

- 其他key通用操作

```
help @generic
```

key通用操作

key 扩展操作（查询模式）

- 查询key

`keys pattern`

查询模式规则

* 匹配任意数量的任意符号 ? 配合一个任意符号 [] 匹配一个指定符号

<code>keys *</code>	查询所有
<code>keys it*</code>	查询所有以it开头
<code>keys *heima</code>	查询所有以heima结尾
<code>keys ??heima</code>	查询所有前面两个字符任意，后面以heima结尾
<code>keys user:?</code>	查询所有以user:开头，最后一个字符任意
<code>keys u[st]er:l</code>	查询所有以u开头，以er:l结尾，中间包含一个字母，s或t

■ 数据库通用操作

数据库



key 的重复问题

- key是由程序员定义的
- redis在使用过程中，伴随着操作数据量的增加，会出现大量的数据以及对应的key
- 数据不区分种类、类别混杂在一起，极易出现重复或冲突

解决方案

- redis为每个服务提供有16个数据库，编号从0到15
- 每个数据库之间的数据相互独立

db 基本操作

- 切换数据库

```
select index
```

- 其他操作

```
quit
```

```
ping
```

```
echo message
```

db 相关操作

- 数据移动

```
move key db
```

- 数据清除

```
dbsize
```

```
flushdb
```

```
flushall
```