**Introduction to Redis**

Redis is an open source (BSD licensed), in-memory **data structure store**, used as a database, cache and message broker. It supports data structures such as [strings](https://redis.io/topics/data-types-intro#strings), [hashes](https://redis.io/topics/data-types-intro#hashes), [lists](https://redis.io/topics/data-types-intro#lists), [sets](https://redis.io/topics/data-types-intro#sets), [sorted sets](https://redis.io/topics/data-types-intro#sorted-sets) with range queries, [bitmaps](https://redis.io/topics/data-types-intro#bitmaps), [hyperloglogs](https://redis.io/topics/data-types-intro" \l "hyperloglogs), [geospatial indexes](https://redis.io/commands/geoadd) with radius queries and [streams](https://redis.io/topics/streams-intro). Redis has built-in [replication](https://redis.io/topics/replication), [Lua scripting](https://redis.io/commands/eval), [LRU eviction](https://redis.io/topics/lru-cache), [transactions](https://redis.io/topics/transactions) and different levels of [on-disk persistence](https://redis.io/topics/persistence), and provides high availability via [Redis Sentinel](https://redis.io/topics/sentinel) and automatic partitioning with [Redis Cluster](https://redis.io/topics/cluster-tutorial).

You can run **atomic operations** on these types, like [appending to a string](https://redis.io/commands/append); [incrementing the value in a hash](https://redis.io/commands/hincrby); [pushing an element to a list](https://redis.io/commands/lpush); [computing set intersection](https://redis.io/commands/sinter), [union](https://redis.io/commands/sunion) and [difference](https://redis.io/commands/sdiff); or [getting the member with highest ranking in a sorted set](https://redis.io/commands/zrangebyscore).

In order to achieve its outstanding performance, Redis works with an **in-memory dataset**. Depending on your use case, you can persist it either by [dumping the dataset to disk](https://redis.io/topics/persistence#snapshotting) every once in a while, or by [appending each command to a log](https://redis.io/topics/persistence#append-only-file). Persistence can be optionally disabled, if you just need a feature-rich, networked, in-memory cache.