Моментальная аналитика с помощью Spring + Redis. Это возможно?

Артем Артемьев

ИТ-НЛМК

just_TimTlm

JPoint

NLMK

О чем речь?

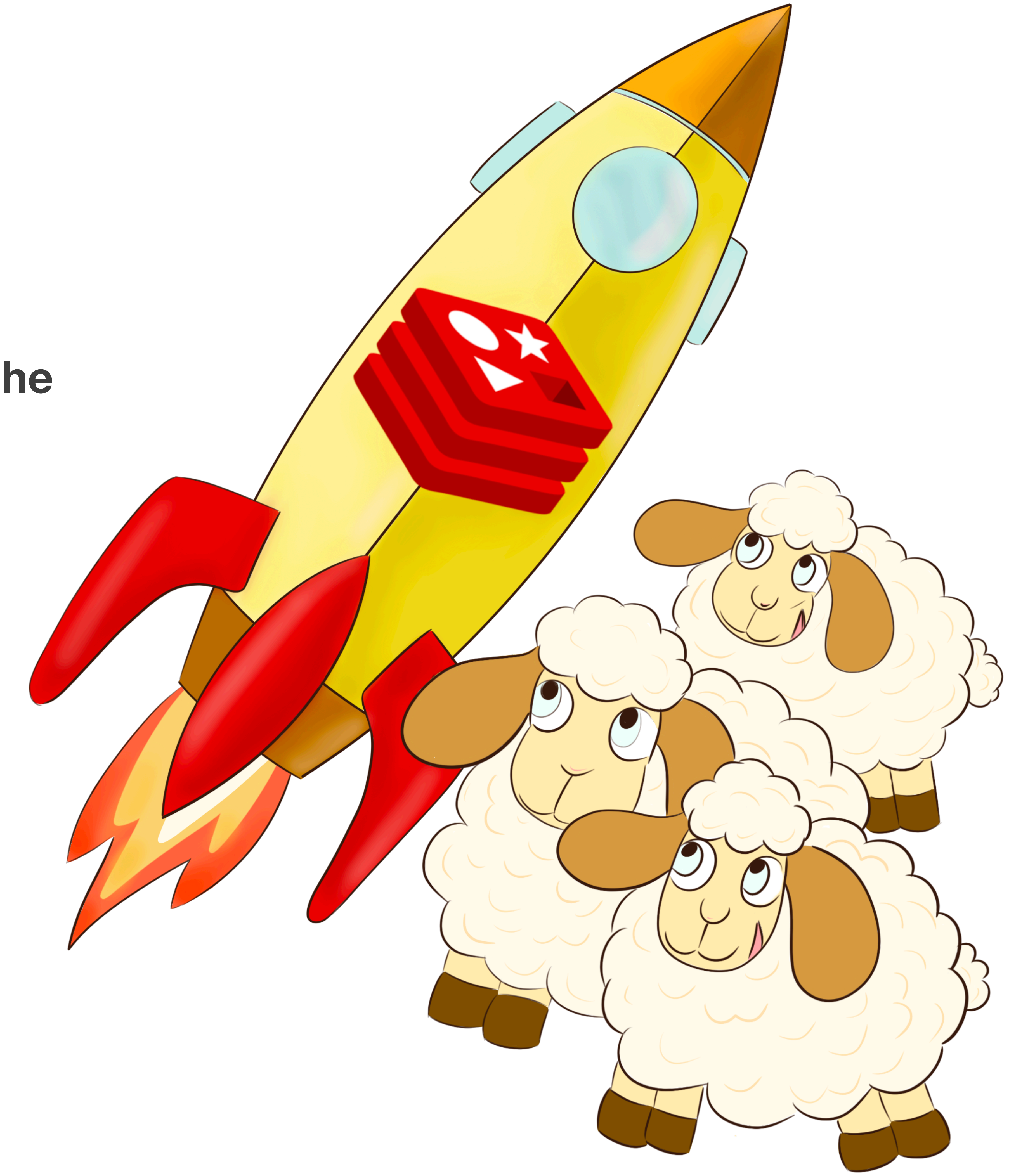
Несколько слов о нашем сервисе

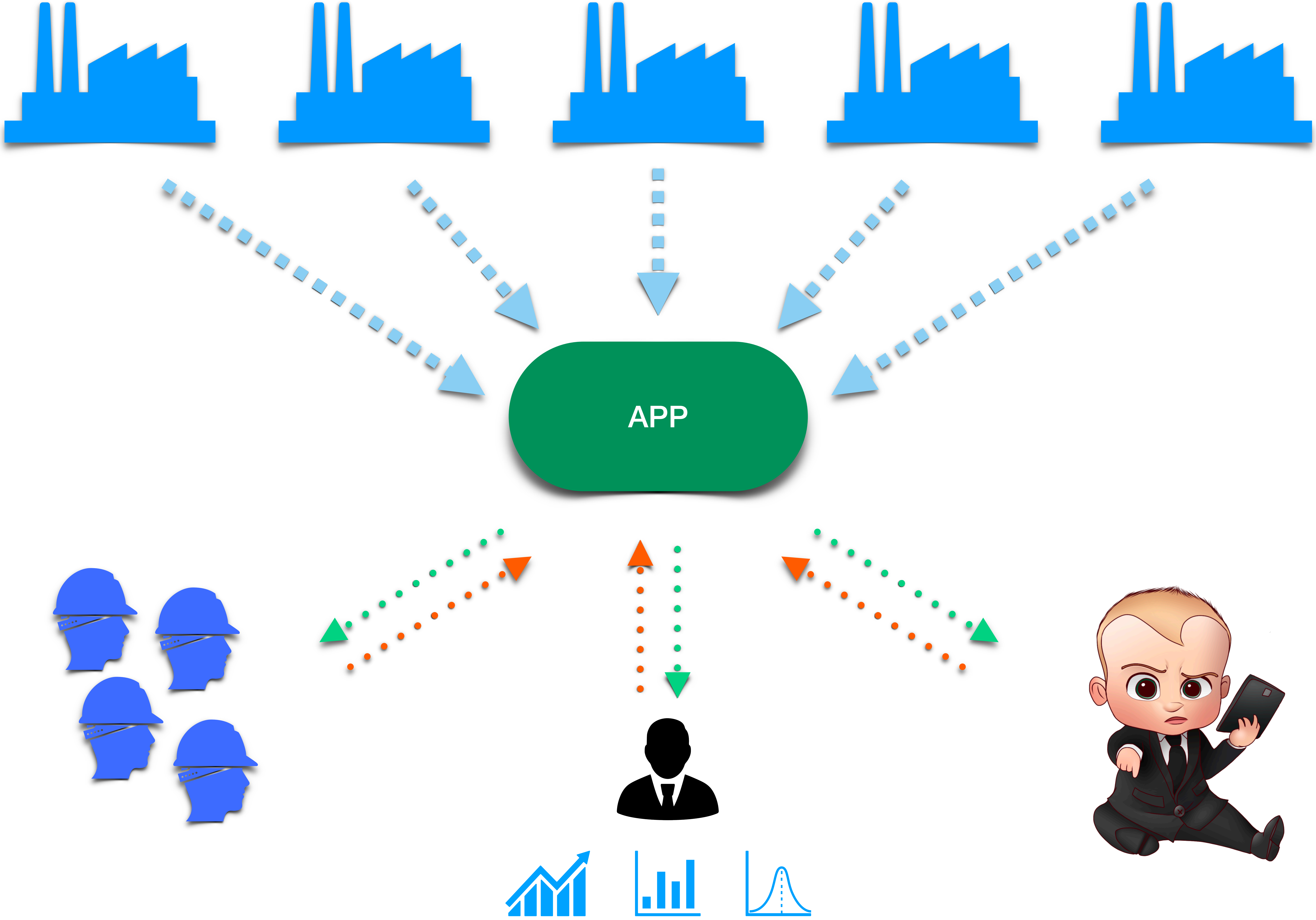
Наш кейс, который будем решать

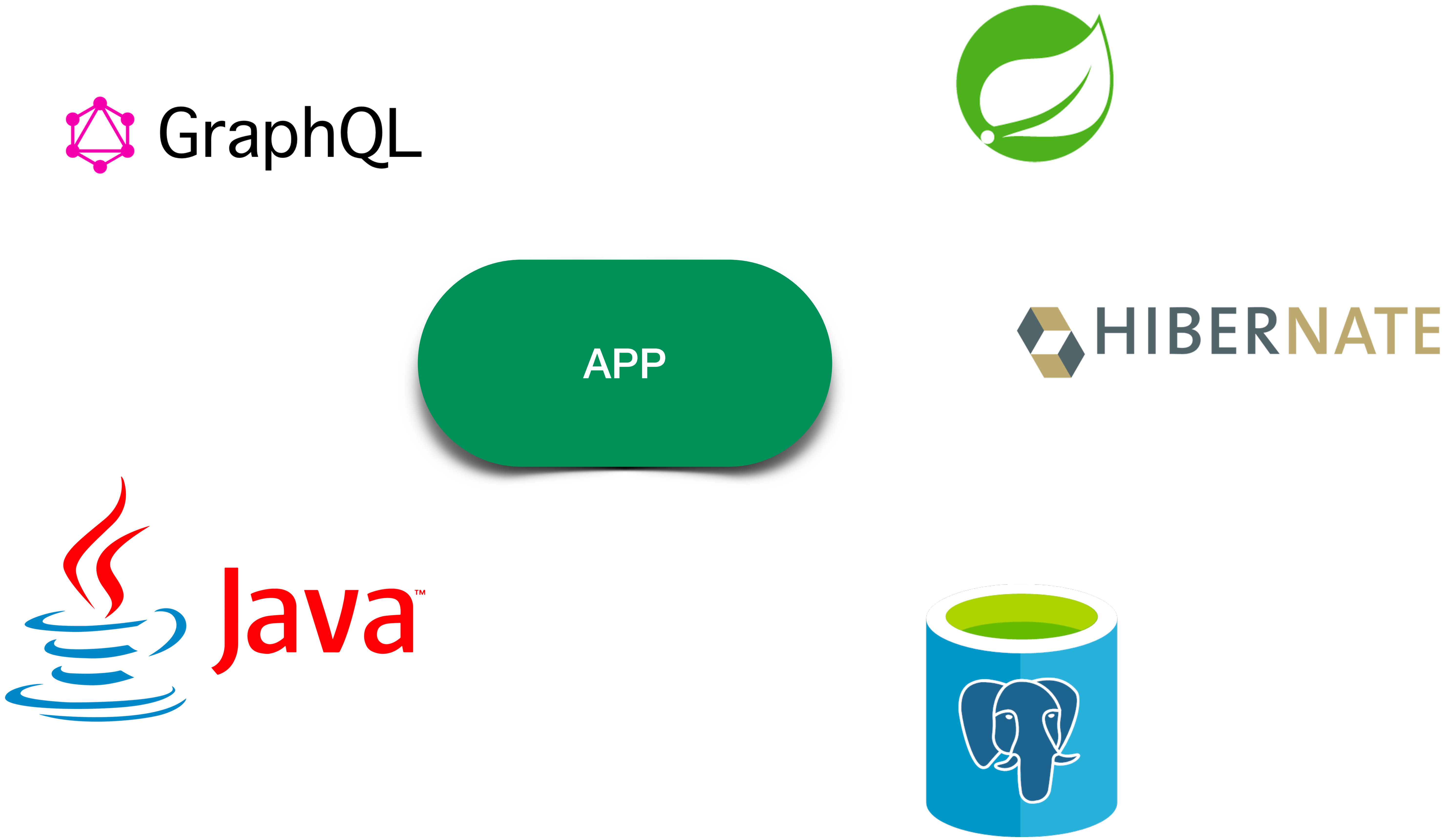
Некоторые детали нашего решения

Рассмотрим возможности Redis. Нет, не cache

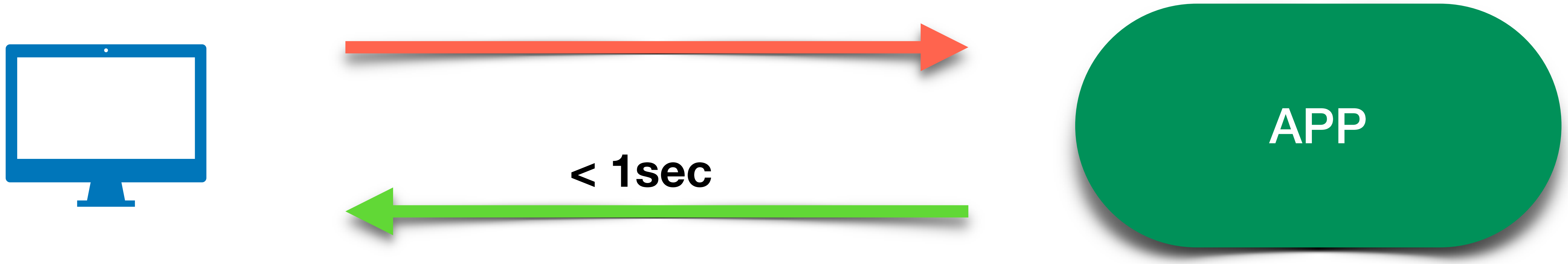
Посмотрим демо-проект

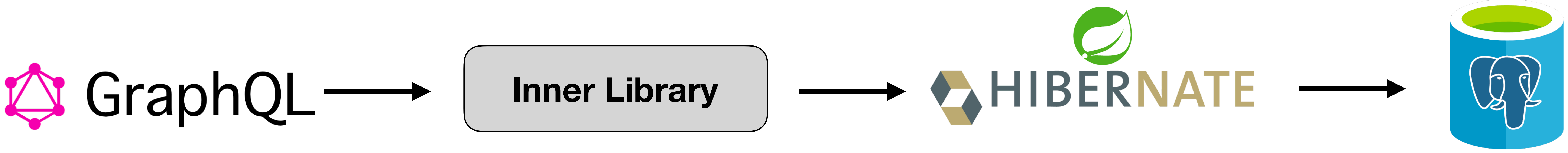



APP



APP


APP
< 1sec



GraphQL

Inner Library

HIBERNATE

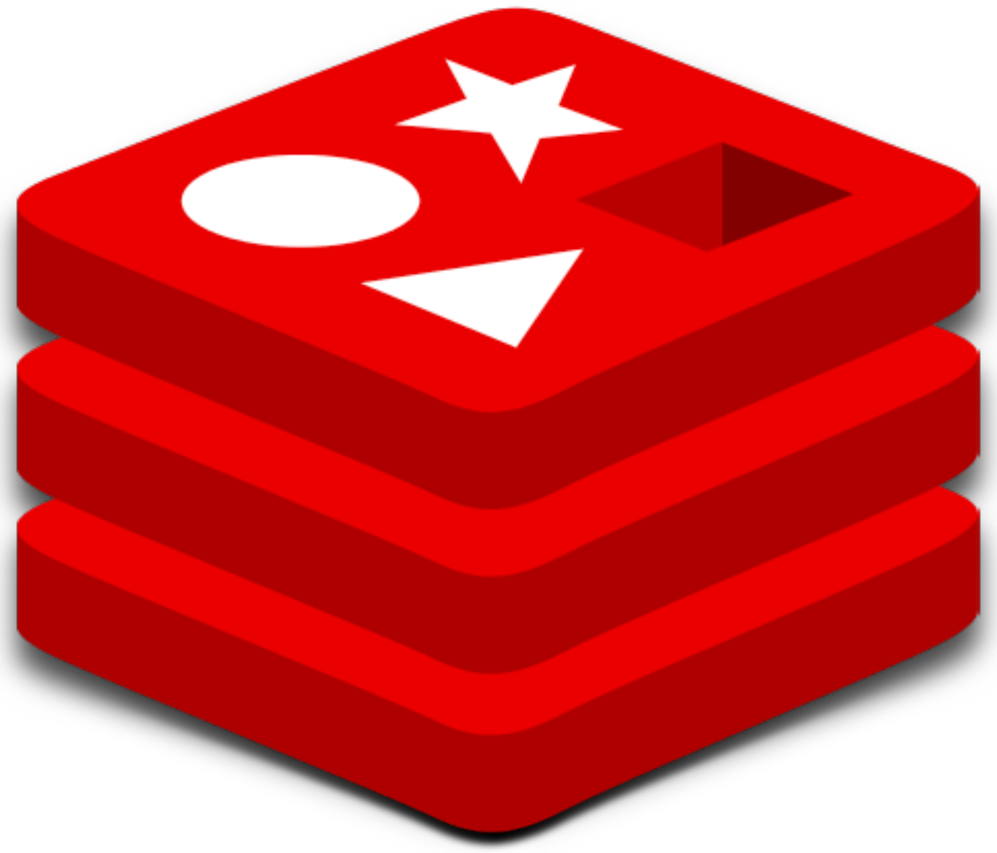
findAll(**Specification**<**T**> spec, **Pageable** pageable)

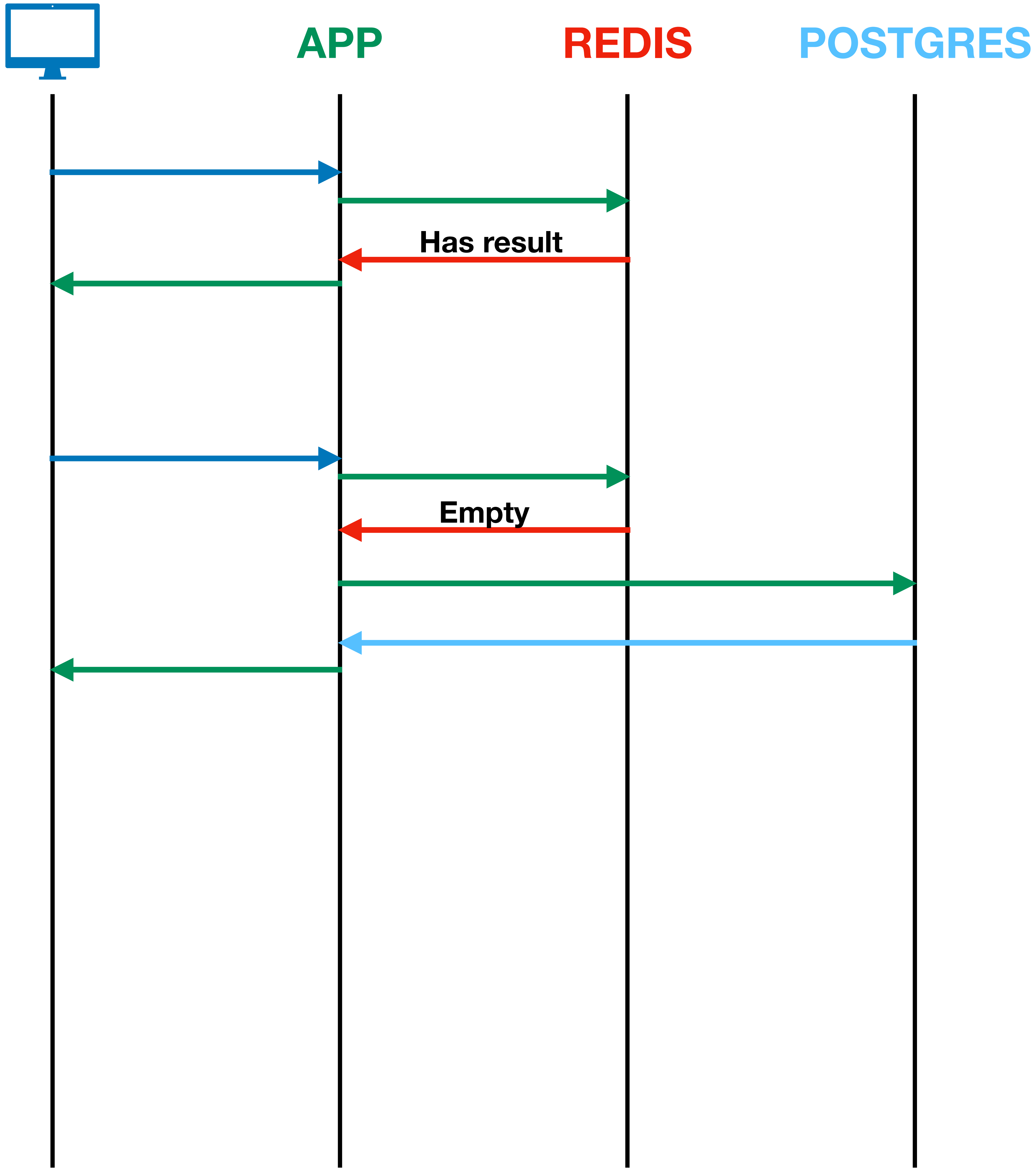
HHH000104: firstResult/maxResults specified with collection fetch; applying in memory!

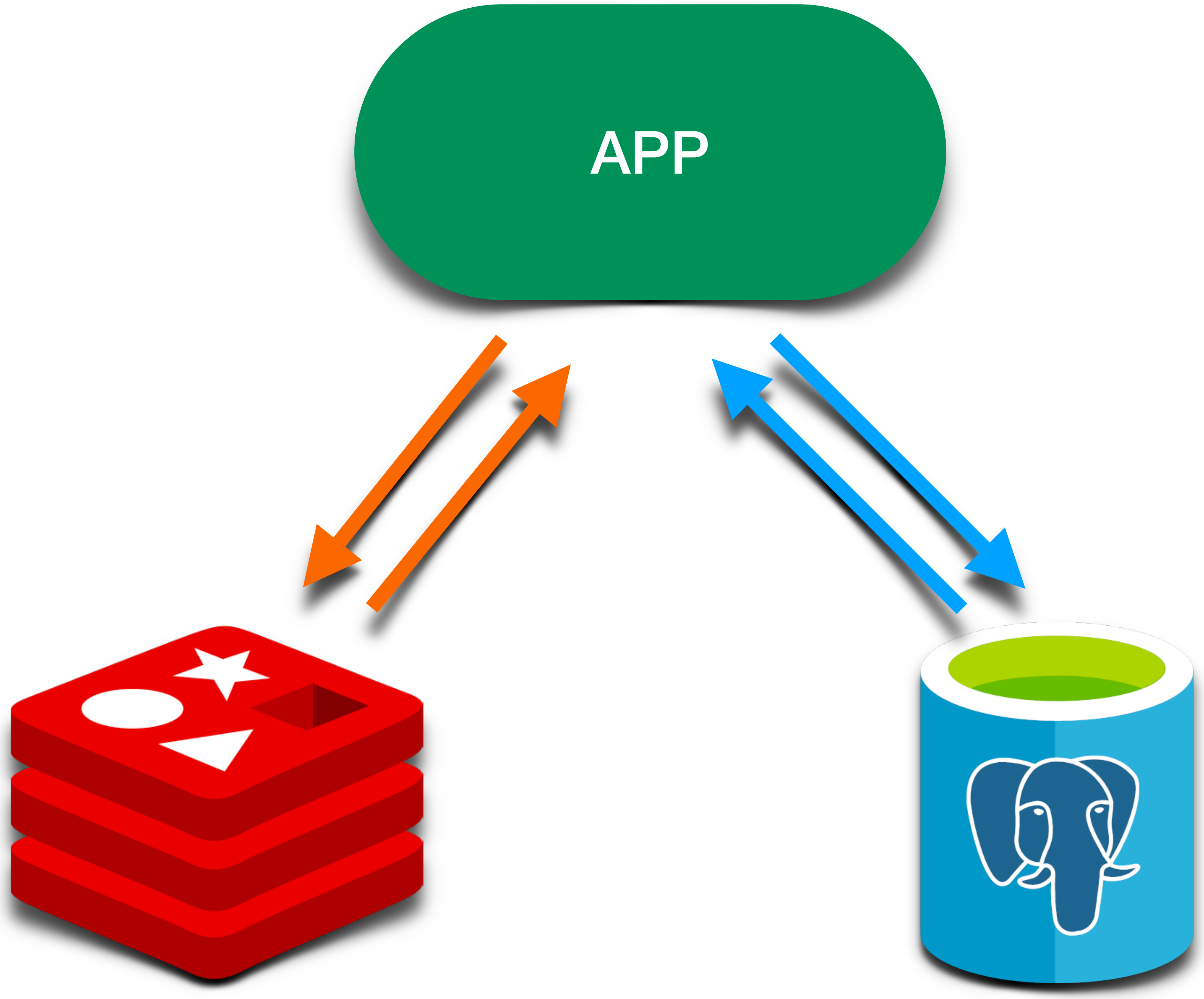
Разделяем на 2 запроса
1 - IDs
2 - данные по IDs

```
select table_.id as col_0_0_ from schema.table table_ where (table_.place_id in (? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? ,
? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? ,
? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? ,
? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? ,
? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? ,
? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? ,
? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? ,
? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? ,
? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? ,
? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? ,
? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? ,
? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ?)) and (table_.acc_date between ? and ? or table_.acc_date=? and (table_.shift_code in
(? , ? , ? , ?)) or table_.acc_date=? and (table_.shift_code in (? , ? , ?))) group by table_.id , table_.end_date order by table_.end_date asc, table_.id asc limit ?
```

```
select table_.id as id1_3_0_, responsibi1_.id as id1_48_1_, detailedca2_.id as id1_21_2_, cause3_.id as id1_12_3_, occurrence4_.id as id1_40_4_, rspbl5_.id as id1_57_5_, placep6_.id as id1_66_6_,
aggregateg8_.id as id1_7_7_, table_.create_date as create_d2_3_0_, table_.created_by as created_3_3_0_, table_.edit_date as edit_dat4_3_0_, table_.updated_by as updated_5_3_0_, table_.acc_date
as acc_date6_3_0_, table_.aggregate_id as aggregat7_3_0_, table_.blocking_date as blocking8_3_0_, table_.brigade_code as brigade_9_3_0_, table_.cause_id as cause_i28_3_0_,
table_.correct_description as correct10_3_0_, table_.corrective_action as correct11_3_0_, table_.detailed_cause_id as detaile12_3_0_, table_.document_unblocked as documen13_3_0_,
table_.downtime_group_id as downtim29_3_0_, table_.duration as duratio14_3_0_, table_.duration_time as duratio15_3_0_, table_.dwt_kind_id as dwt_kin30_3_0_, table_.dwt_type_id as
dwt_typ31_3_0_, table_.end_date as end_dat16_3_0_, table_.equipment_id as equipme32_3_0_, table_.is_description_added as is_desc17_3_0_, table_.is_description_required as is_desc18_3_0_,
table_.is_grouping_required as is_grou19_3_0_, table_.machine_id as machine33_3_0_, table_.mechanism_id as mechani34_3_0_, table_.note as note20_3_0_, table_.occurrence_factor_id as
occurre35_3_0_, table_.reason_note as reason_21_3_0_, table_.res_zone_id as respons36_3_0_, table_.rspble_user_id as respons22_3_0_, table_.rspble_work_place_id as respons37_3_0_,
table_.shift_code as shift_c23_3_0_, table_.source as source24_3_0_, table_.source_id as source_25_3_0_, table_.start_date as start_d26_3_0_, table_.status as status27_3_0_, table_.workshop_id as
worksho38_3_0_, responsibi1_.date_begin as date_beg2_48_1_, responsibi1_.date_end as date_end3_48_1_, responsibi1_.edited_date as edited_d4_48_1_, responsibi1_.id_mdm as
id_mdm5_48_1_, responsibi1_.is_active as is_activ6_48_1_, responsibi1_.code_mdm as code_mdm7_48_1_, responsibi1_.name as name8_48_1_, detailedca2_.date_begin as date_beg2_21_2_,
detailedca2_.date_end as date_end3_21_2_, detailedca2_.edited_date as edited_d4_21_2_, detailedca2_.id_mdm as id_mdm5_21_2_, detailedca2_.is_active as is_activ6_21_2_, detailedca2_.name as
name7_21_2_, cause3_.date_begin as date_beg2_12_3_, cause3_.date_end as date_end3_12_3_, cause3_.edited_date as edited_d4_12_3_, cause3_.id_mdm as id_mdm5_12_3_, cause3_.is_active as
is_activ6_12_3_, cause3_.category_reason_id as category9_12_3_, cause3_.code_mdm as code_mdm7_12_3_, cause3_.name as name8_12_3_, occurrence4_.date_begin as date_beg2_40_4_,
occurrence4_.date_end as date_end3_40_4_, occurrence4_.edited_date as edited_d4_40_4_, occurrence4_.id_mdm as id_mdm5_40_4_, occurrence4_.is_active as is_activ6_40_4_,
occurrence4_.name as name7_40_4_, rspbl5_.date_begin as date_beg2_57_5_, rspbl5_.date_end as date_end3_57_5_, rspbl5_.edited_date as edited_d4_57_5_, rspbl5_.id_mdm as id_mdm5_57_5_,
rspbl5_.is_active as is_activ6_57_5_, rspbl5_.code_mdm as code_mdm7_57_5_, rspbl5_.name as name8_57_5_, rspbl5_.number_rspble_work_place as number_r9_57_5_, rspbl5_.place_place_id as
technic10_57_5_, placep6_.date_begin as date_beg7_66_6_, placep6_.date_end as date_end8_66_6_, placep6_.edited_date as edited_d9_66_6_, placep6_.id_mdm as id_mdm2_66_6_,
placep6_.is_active as is_activ3_66_6_, placep6_.name as name4_66_6_, placep6_.parent_place_place_id as parent_t5_66_6_, placep6_.parent_place_place_id_mdm as parent_10_66_6_,
placep6_.sap_hcm_id as sap_hcm11_66_6_, placep6_.short_name as short_na6_66_6_, placep6_.place_level_id as technic13_66_6_, placep6_.unit_id_spep2 as unit_id12_66_6_,
aggregateg8_.date_begin as date_beg2_7_7_, aggregateg8_.date_end as date_end3_7_7_, aggregateg8_.edited_date as edited_d4_7_7_, aggregateg8_.id_mdm as id_mdm5_7_7_,
aggregateg8_.is_active as is_activ6_7_7_, aggregateg8_.name as name7_7_7_, aggregateg8_.unitgrp_id as unitgrp_8_7_7_, aggregateg7_.place_place_id as technica8_8_0__,
aggregateg7_.agg_gr_id as aggregat7_8_0__ from schema.table table_ left outer join dict.res_zone responsibi1_ on table_.res_zone_id=responsibi1_.id left outer join dict.detailed_cause detailedca2_
on table_.detailed_cause_id=detailedca2_.id left outer join dict.cause cause3_ on table_.cause_id=cause3_.id left outer join dict.occurrence_factor occurrence4_ on
table_.occurrence_factor_id=occurrence4_.id left outer join dict.rspble_work_place rspbl5_ on table_.rspble_work_place_id=rspbl5_.id left outer join dict.place_place placep6_ on
table_.aggregate_id=placep6_.id left outer join dict.agg_gr__aggregate aggregateg7_ on placep6_.id_mdm=aggregateg7_.place_place_id left outer join dict.agg_gr aggregateg8_ on
aggregateg7_.agg_gr_id=aggregateg8_.id_mdm where table_.id in (? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ? , ?)
```

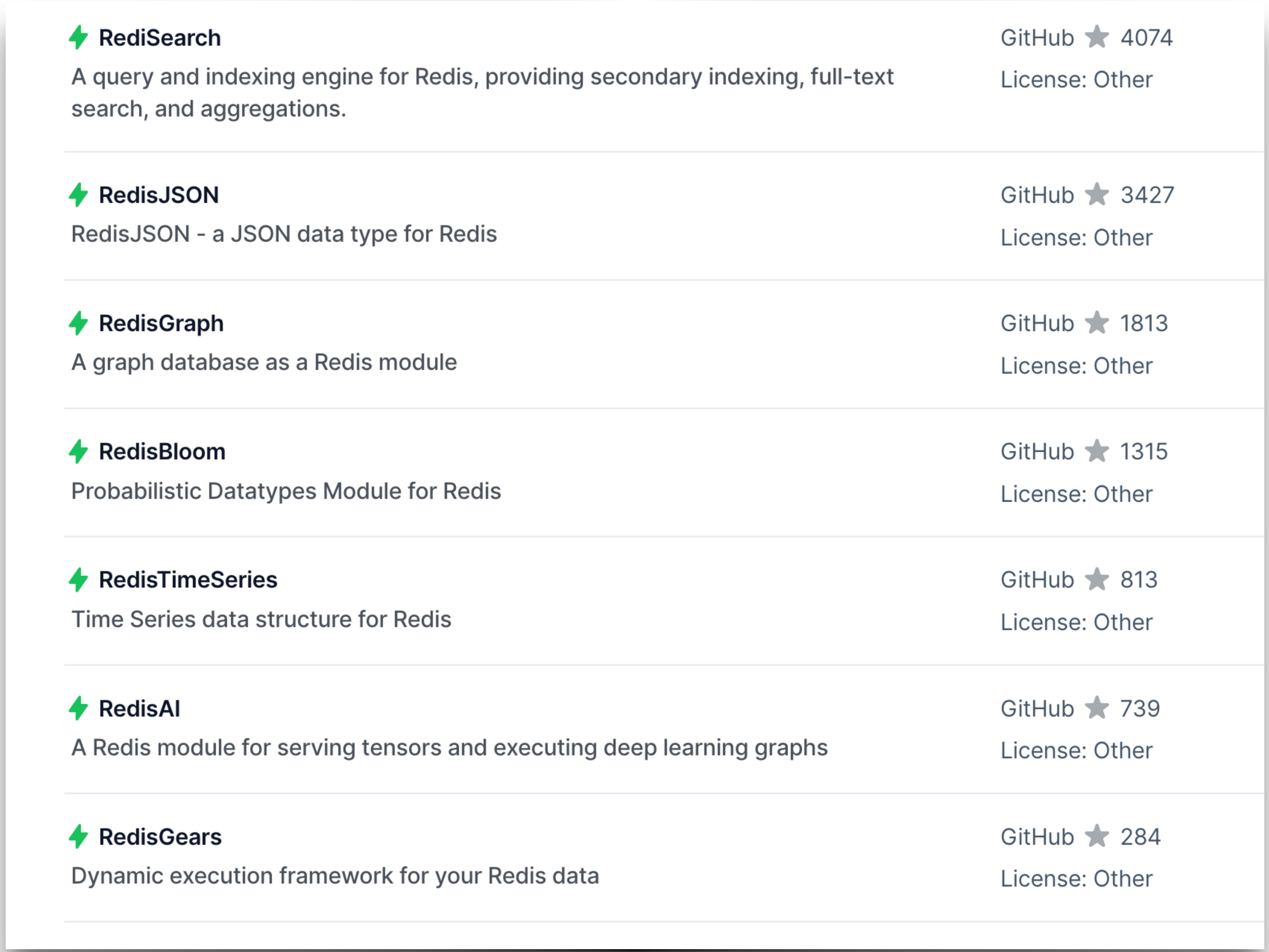



APP
REDIS
POSTGRES
Has result
Empty


APP

STRING
LIST
SET
ZSET
HASH

https://redis.io/resources/modules/



RediSearch GitHub ★ 4074

A query and indexing engine for Redis, providing secondary indexing, full-text search, and aggregations. License: Other

RedisJSON GitHub ★ 3427

RedisJSON - a JSON data type for Redis License: Other

RedisGraph GitHub ★ 1813

A graph database as a Redis module License: Other

RedisBloom GitHub ★ 1315

Probabilistic Datatypes Module for Redis License: Other

RedisTimeSeries GitHub ★ 813

Time Series data structure for Redis License: Other

RedisAI GitHub ★ 739

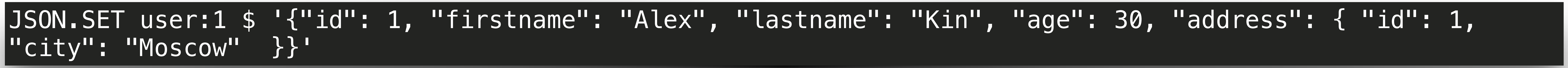
A Redis module for serving tensors and executing deep learning graphs License: Other

RedisGears GitHub ★ 284

Dynamic execution framework for your Redis data License: Other

RedisJSON

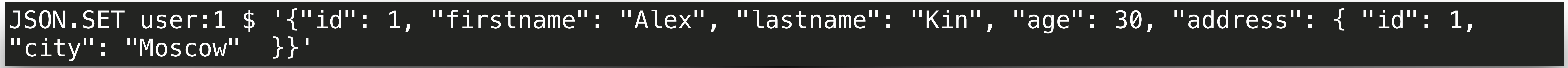
```
{
  "id": 1,
  "firstname": "Alex",
  "lastname": "Kin",
  "age": 30,
  "address": {
     "id": 1,
     «city": "Moscow"
  }
}
```



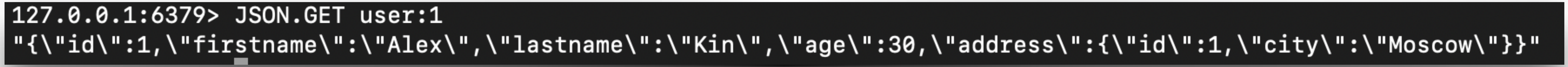
```
JSON.SET user:1 $ '{"id": 1, "firstname": "Alex", "lastname": "Kin", "age": 30, "address": { "id": 1,
"city": "Moscow"  }}'
```

RedisJSON

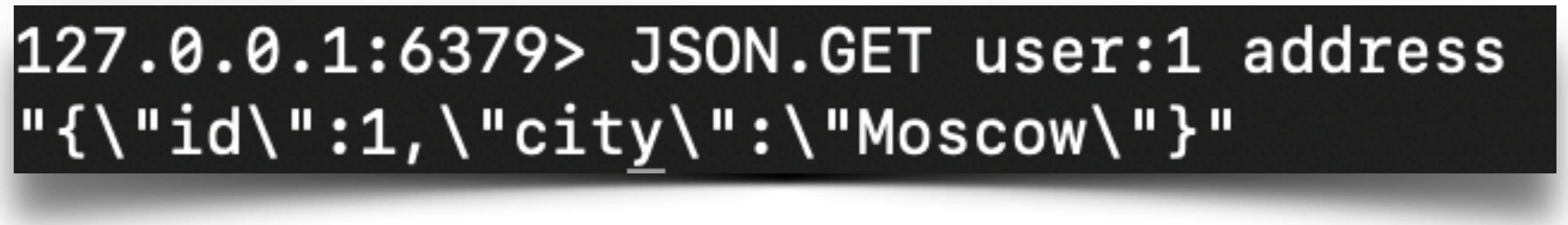
```
{
  "id": 1,
  "firstname": "Alex",
  "lastname": "Kin",
  "age": 30,
  "address": {
     "id": 1,
     «city": "Moscow"
  }
}
```



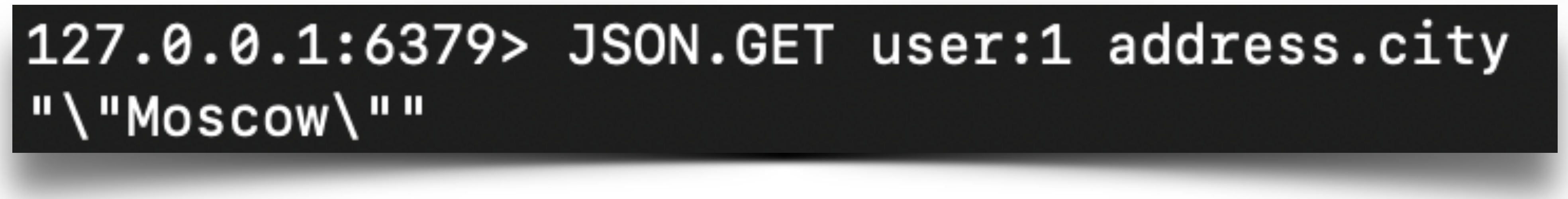
```
JSON.SET user:1 $ '{"id": 1, "firstname": "Alex", "lastname": "Kin", "age": 30, "address": { "id": 1,
"city": "Moscow"  }}'
```



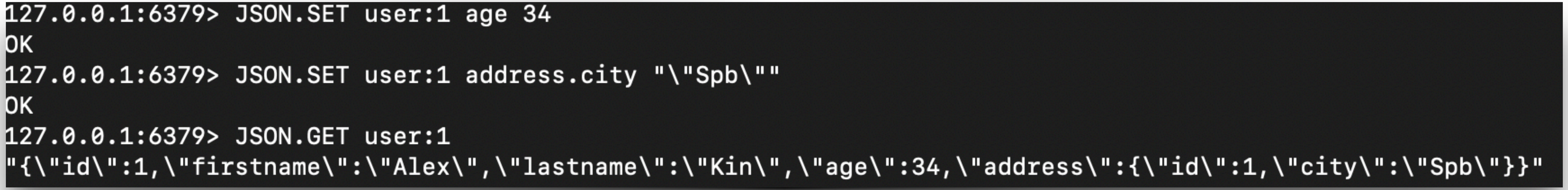
```
127.0.0.1:6379> JSON.GET user:1
"{\"id\":1,\"firstname\":\"Alex\",\"lastname\":\"Kin\",\"age\":30,\"address\":{\"id\":1,\"city\":\"Moscow\"}}"
```



```
127.0.0.1:6379> JSON.GET user:1 address
"{\"id\":1,\"city\":\"Moscow\"}"
```



```
127.0.0.1:6379> JSON.GET user:1 address.city
"\"Moscow\""
```

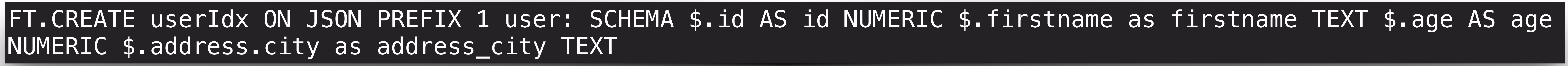


```
127.0.0.1:6379> JSON.SET user:1 age 34
OK
127.0.0.1:6379> JSON.SET user:1 address.city "\"Spb\""
OK
127.0.0.1:6379> JSON.GET user:1
"{\"id\":1,\"firstname\":\"Alex\",\"lastname\":\"Kin\",\"age\":34,\"address\":{\"id\":1,\"city\":\"Spb\"}}"
```

RediSearch

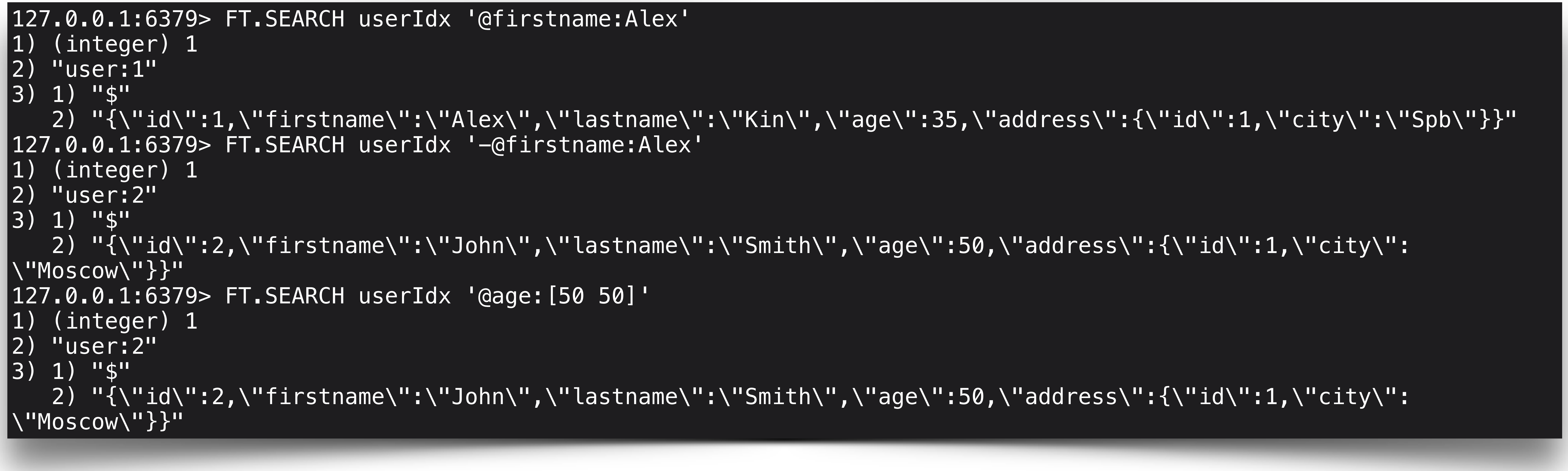
- Вторичное индексирование
- Запросы с логическими операторами
- Поиск по префиксу
- Полнотекстовый поиск
- Агрегации
- Группировки
- Операции редукции

RediSearch

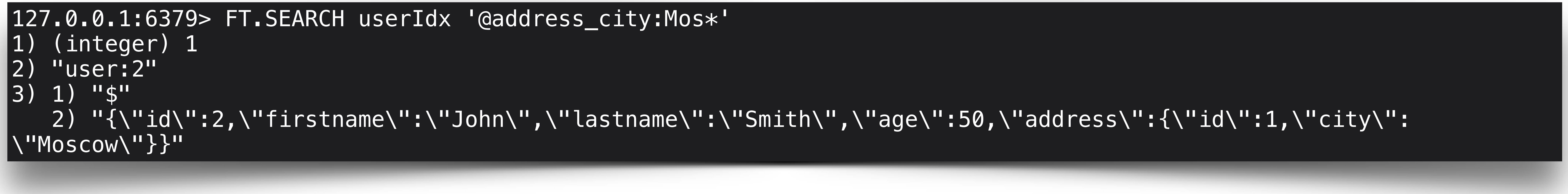


```
FT.CREATE userIdx ON JSON PREFIX 1 user: SCHEMA $.id AS id NUMERIC $.firstname as firstname TEXT $.age AS age
NUMERIC $.address.city as address_city TEXT
```

RediSearch

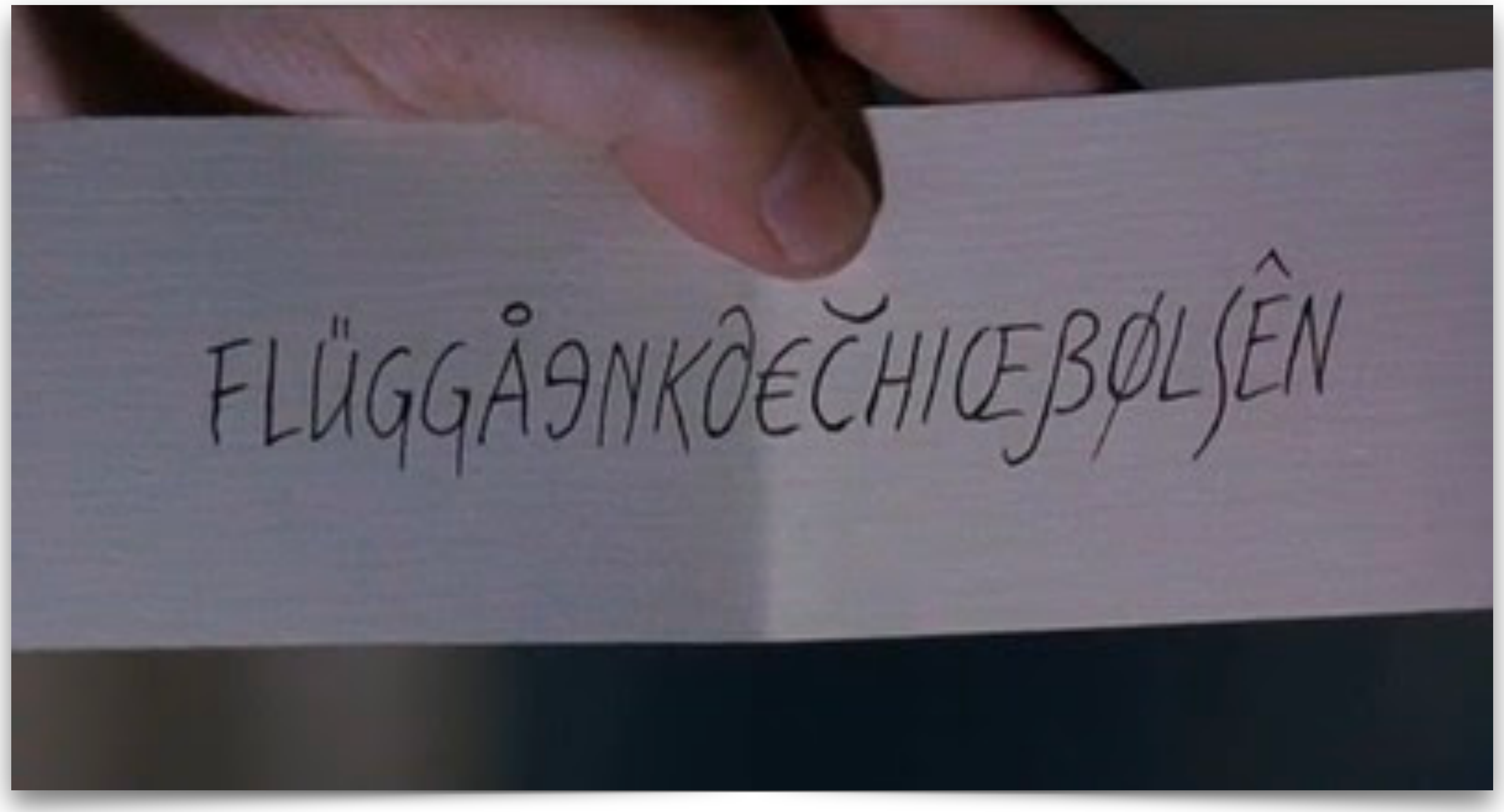


```
127.0.0.1:6379> FT.SEARCH userIdx '@firstname:Alex'
1) (integer) 1
2) "user:1"
3) 1) "$"
   2) "{\"id\":1,\"firstname\":\"Alex\",\"lastname\":\"Kin\",\"age\":35,\"address\":{\"id\":1,\"city\":\"Spb\"}}"
127.0.0.1:6379> FT.SEARCH userIdx '-@firstname:Alex'
1) (integer) 1
2) "user:2"
3) 1) "$"
   2) "{\"id\":2,\"firstname\":\"John\",\"lastname\":\"Smith\",\"age\":50,\"address\":{\"id\":1,\"city\":
\"Moscow\"}}"
127.0.0.1:6379> FT.SEARCH userIdx '@age:[50 50]'
1) (integer) 1
2) "user:2"
3) 1) "$"
   2) "{\"id\":2,\"firstname\":\"John\",\"lastname\":\"Smith\",\"age\":50,\"address\":{\"id\":1,\"city\":
\"Moscow\"}}"
```



```
127.0.0.1:6379> FT.SEARCH userIdx '@address_city:Mos*'
1) (integer) 1
2) "user:2"
3) 1) "$"
   2) "{\"id\":2,\"firstname\":\"John\",\"lastname\":\"Smith\",\"age\":50,\"address\":{\"id\":1,\"city\":
\"Moscow\"}}"
```

СТОП-СЛОВА



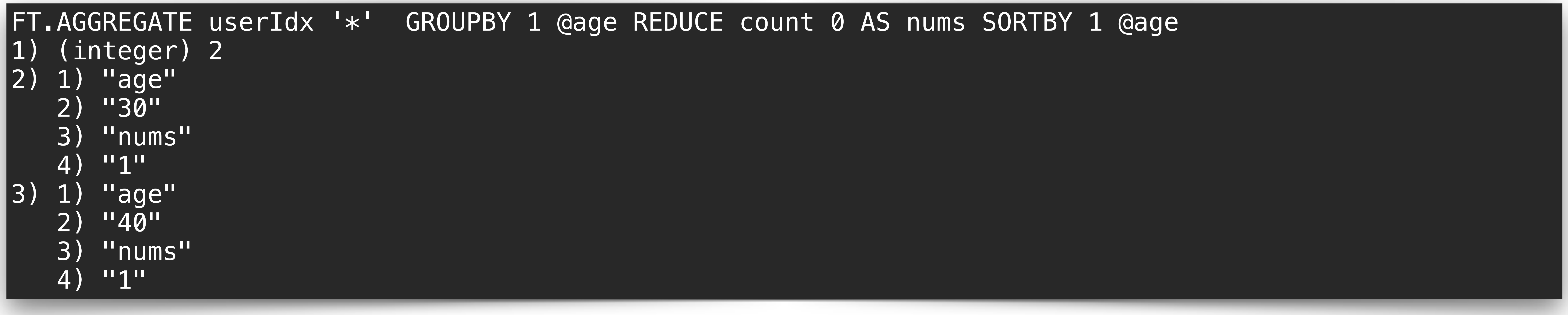
a, is, the, an, and, are, as, at, be, but, by, for, if, in, into, it, no, not, of, on, or, such, that, their, then, there, these, they, this, to, was, will, with

https://redis.io/docs/stack/search/reference/stopwords/

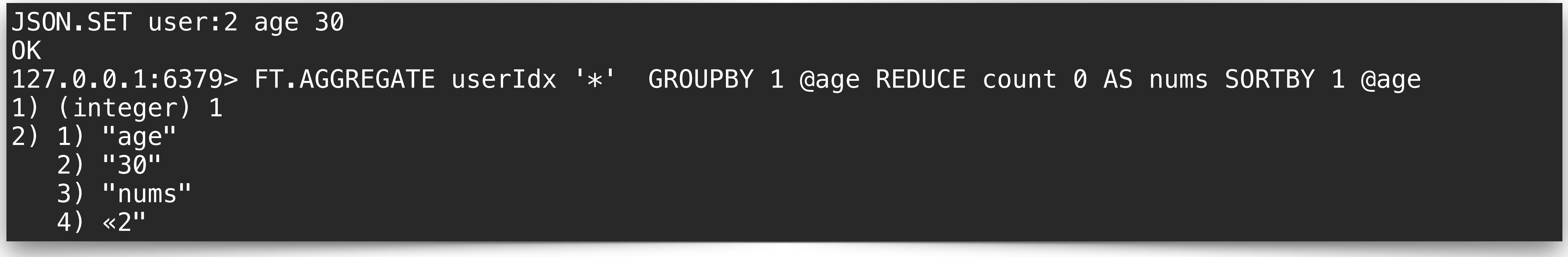
Агрегации

```
FT.AGGREGATE index query
  [VERBATIM]
  [ LOAD count field [field ...]]
  [TIMEOUT timeout]
  [LOAD *]
  [ GROUPBY nargs property [property ...] [ REDUCE function nargs arg [arg ...] [AS name] [ REDUCE function nargs arg [arg ...] [AS name] ...]]
  [ GROUPBY nargs property [property ...] [ REDUCE function nargs arg [arg ...] [AS name] [ REDUCE function nargs arg [arg ...] [AS name] ...]] ...]]
  [ SORTBY nargs [ property ASC I DESC [ property ASC I DESC ...]] [MAX num]]
  [ APPLY expression AS name [ APPLY expression AS name ...]]
  [ LIMIT offset num]
  [FILTER filter]
  [ WITHCURSOR [COUNT read_size] [MAXIDLE idle_time]]
  [ PARAMS nargs name value [ name value ...]]
  [DIALECT dialect]
```

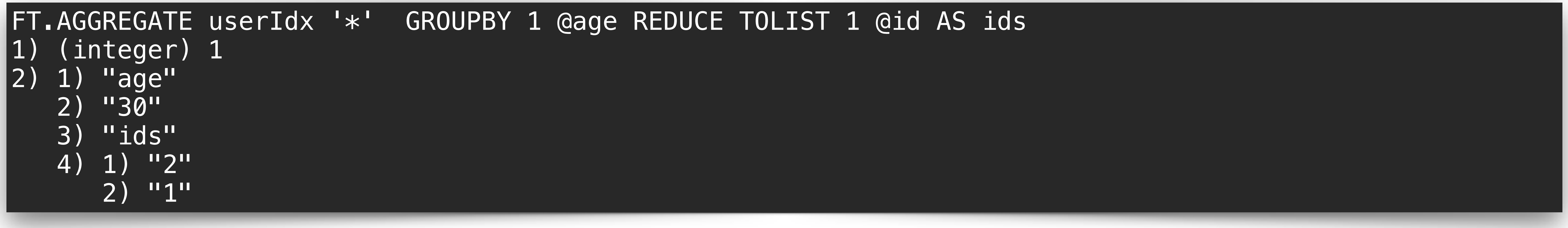
RediSearch



```
FT.AGGREGATE userIdx '*'  GROUPBY 1 @age REDUCE count 0 AS nums SORTBY 1 @age
1) (integer) 2
2) 1) "age"
   2) "30"
   3) "nums"
   4) "1"
3) 1) "age"
   2) "40"
   3) "nums"
   4) "1"
```

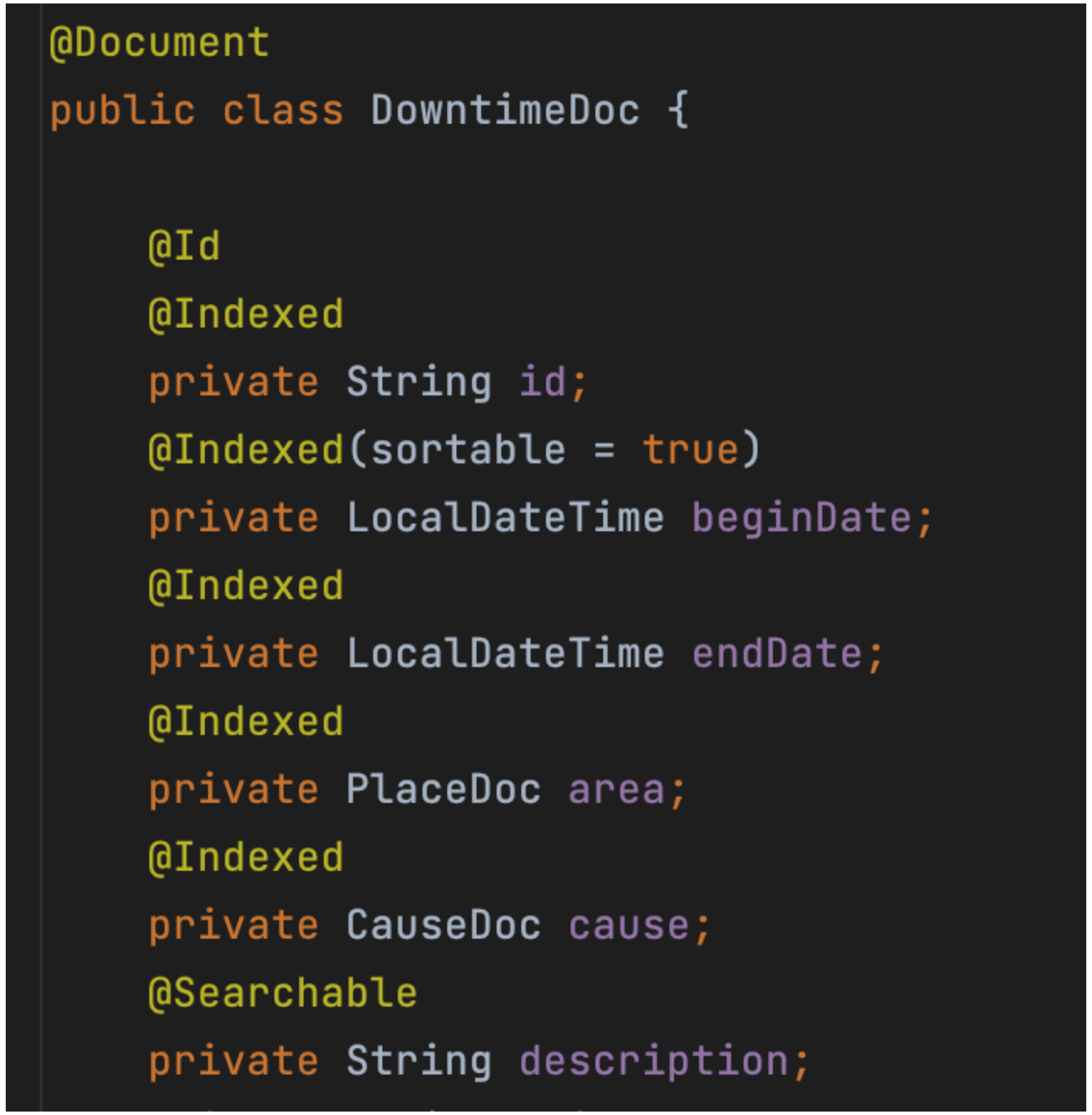


```
JSON.SET user:2 age 30
OK
127.0.0.1:6379> FT.AGGREGATE userIdx '*'  GROUPBY 1 @age REDUCE count 0 AS nums SORTBY 1 @age
1) (integer) 1
2) 1) "age"
   2) "30"
   3) "nums"
   4) «2"
```



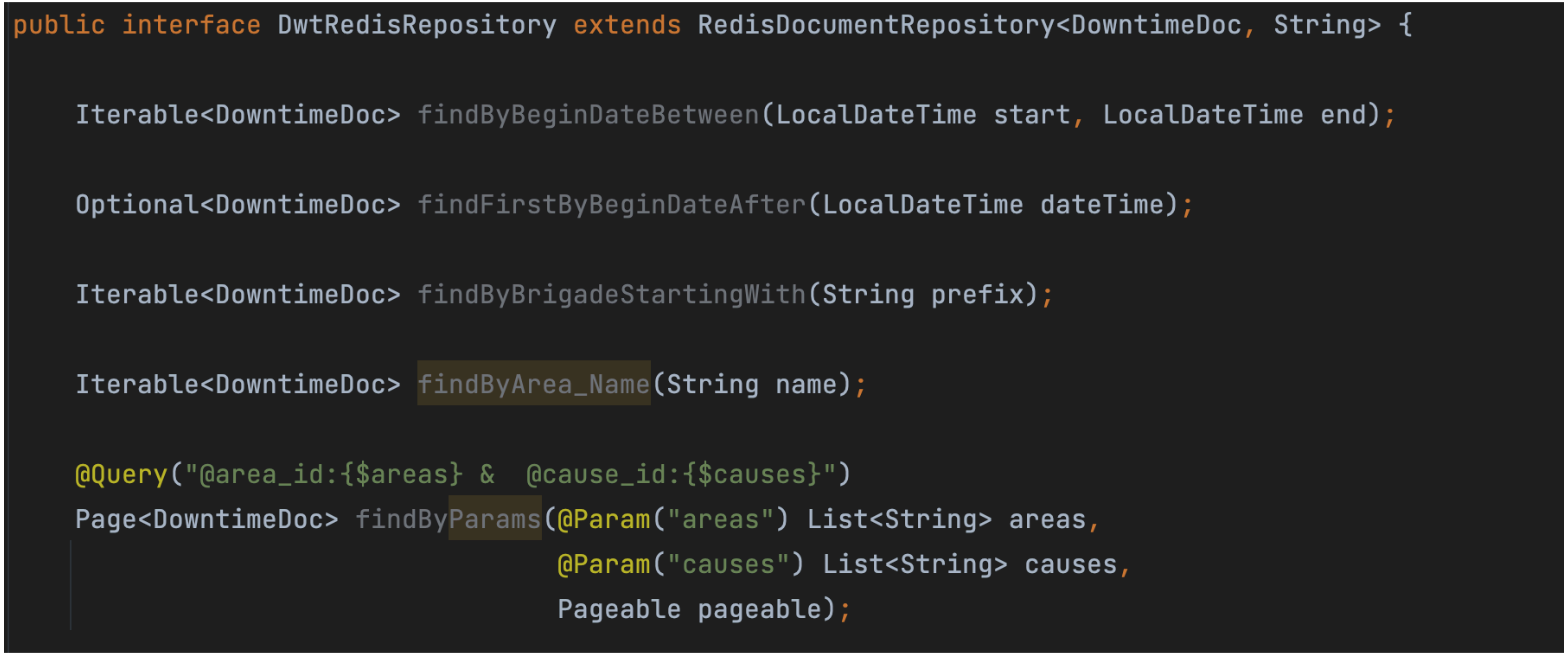
```
FT.AGGREGATE userIdx '*'  GROUPBY 1 @age REDUCE TOLIST 1 @id AS ids
1) (integer) 1
2) 1) "age"
   2) "30"
   3) "ids"
   4) 1) "2"
      2) "1"
```

Redis OM Spring



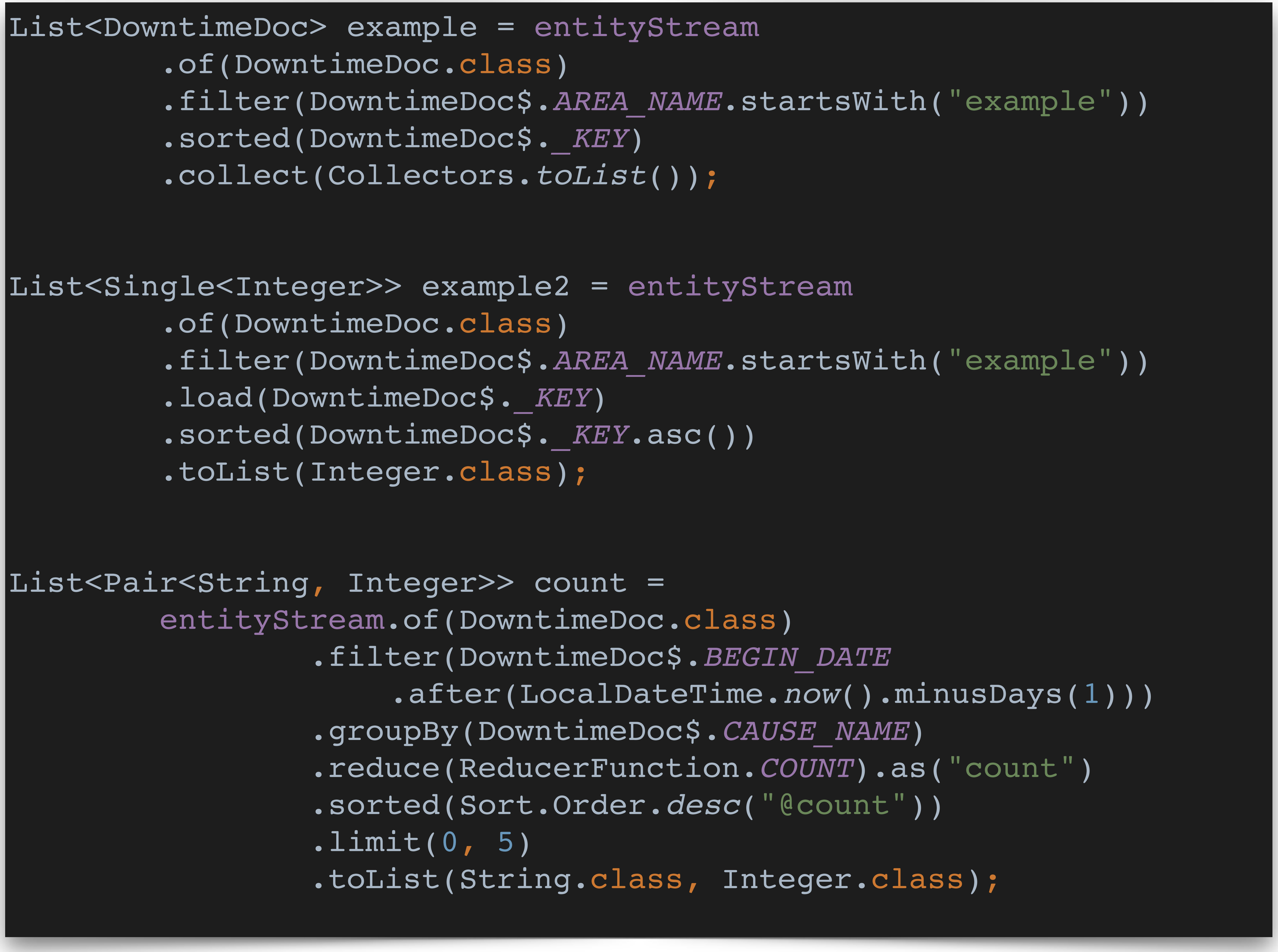
```
@Document
public class DowntimeDoc {

    @Id
    @Indexed
    private String id;
    @Indexed(sortable = true)
    private LocalDateTime beginDate;
    @Indexed
    private LocalDateTime endDate;
    @Indexed
    private PlaceDoc area;
    @Indexed
    private CauseDoc cause;
    @Searchable
    private String description;
```

Redis OM Spring

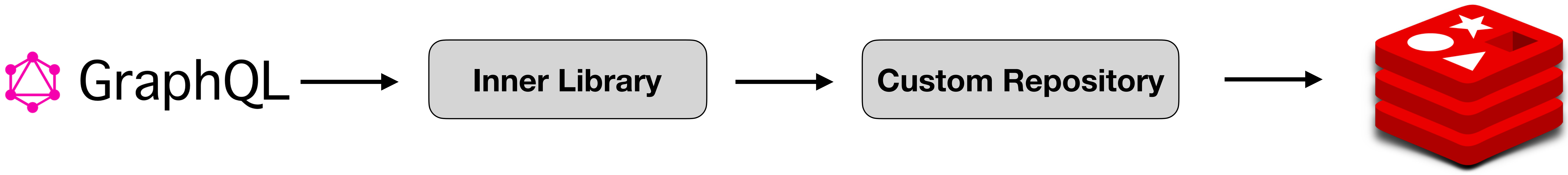


```java
public interface DwtRedisRepository extends RedisDocumentRepository<DowntimeDoc, String> {

    Iterable<DowntimeDoc> findByBeginDateBetween(LocalDateTime start, LocalDateTime end);

    Optional<DowntimeDoc> findFirstByBeginDateAfter(LocalDateTime dateTime);

    Iterable<DowntimeDoc> findByBrigadeStartingWith(String prefix);

    Iterable<DowntimeDoc> findByArea_Name(String name);

    @Query("@area_id:{$areas} &  @cause_id:{$causes}")
    Page<DowntimeDoc> findByParams(@Param("areas") List<String> areas,
                                   @Param("causes") List<String> causes,
                                   Pageable pageable);
```

Redis OM Spring



```
List<DowntimeDoc> example = entityStream
        .of(DowntimeDoc.class)
        .filter(DowntimeDoc$.AREA_NAME.startsWith("example"))
        .sorted(DowntimeDoc$._KEY)
        .collect(Collectors.toList());


List<Single<Integer>> example2 = entityStream
        .of(DowntimeDoc.class)
        .filter(DowntimeDoc$.AREA_NAME.startsWith("example"))
        .load(DowntimeDoc$._KEY)
        .sorted(DowntimeDoc$._KEY.asc())
        .toList(Integer.class);


List<Pair<String, Integer>> count =
        entityStream.of(DowntimeDoc.class)
                .filter(DowntimeDoc$.BEGIN_DATE
                    .after(LocalDateTime.now().minusDays(1)))
                .groupBy(DowntimeDoc$.CAUSE_NAME)
                .reduce(ReducerFunction.COUNT).as("count")
                .sorted(Sort.Order.desc("@count"))
                .limit(0, 5)
                .toList(String.class, Integer.class);
```

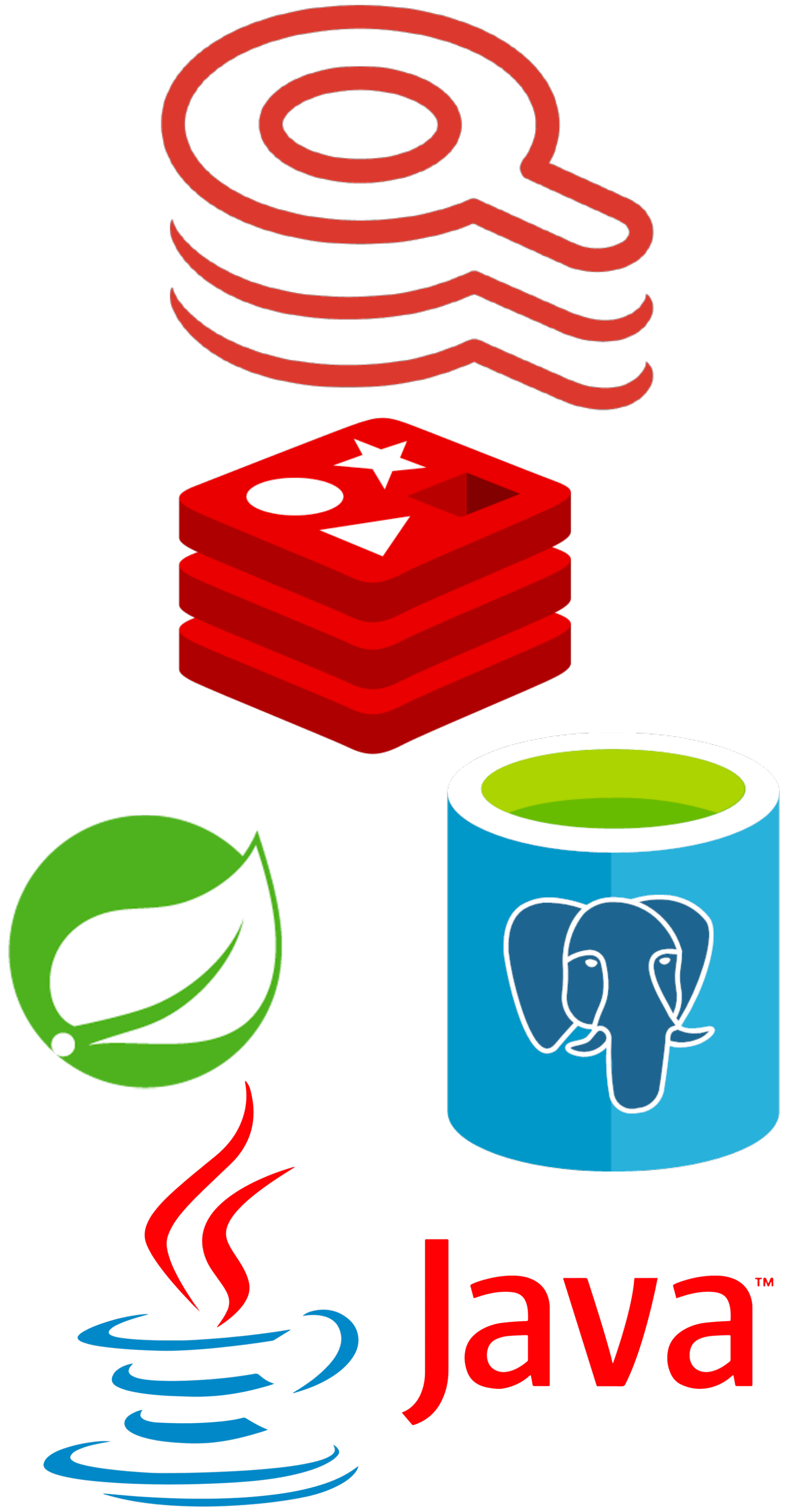


GraphQL

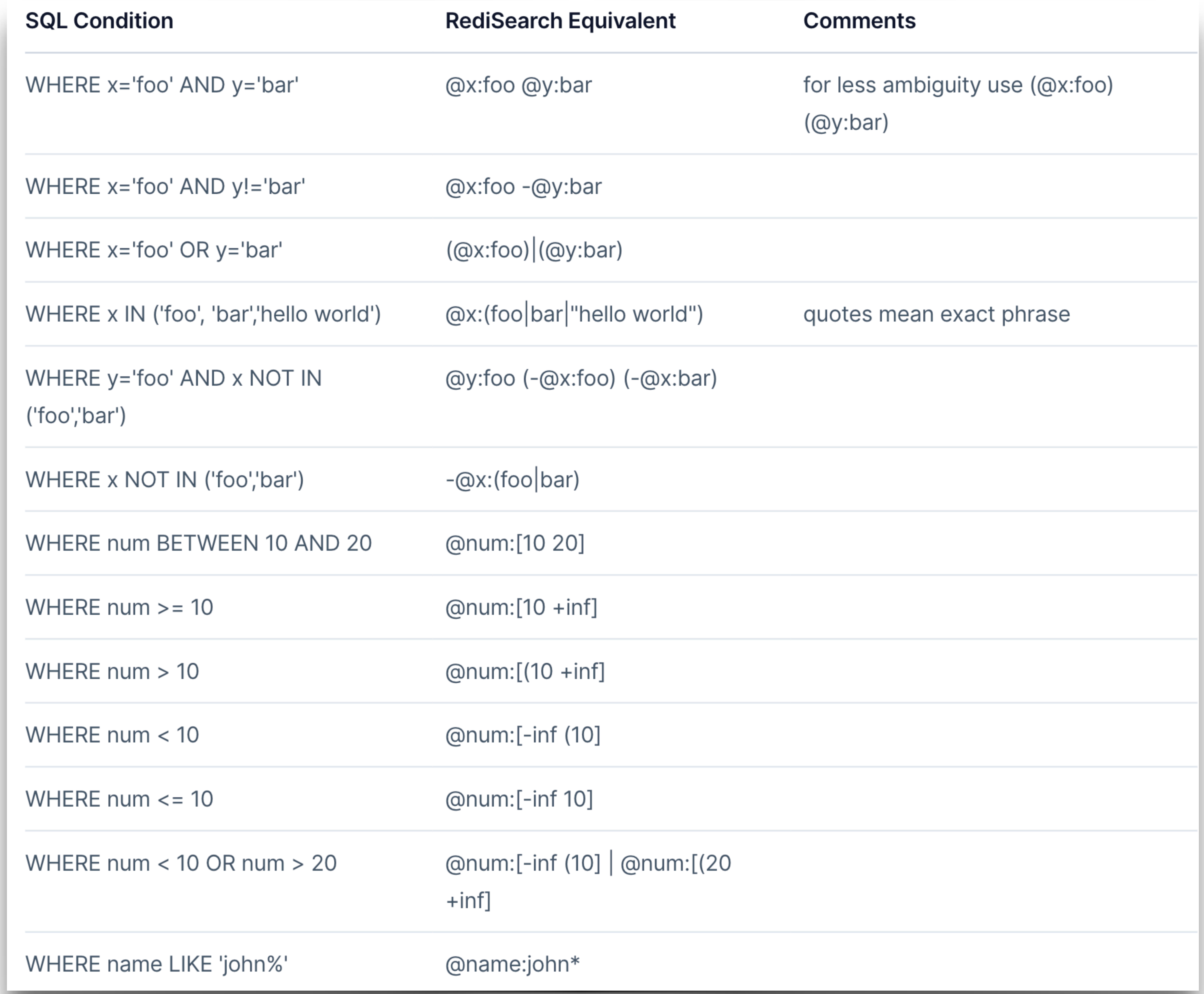
Inner Library

Custom Repository

DEMO PROJECT



Сопоставление общих предикатов SQL с RediSearch



SQL Condition	RediSearch Equivalent	Comments
WHERE x='foo' AND y='bar'	@x:foo @y:bar	for less ambiguity use (@x:foo) (@y:bar)
WHERE x='foo' AND y!='bar'	@x:foo -@y:bar	
WHERE x='foo' OR y='bar'	(@x:foo)|(@y:bar)	
WHERE x IN ('foo', 'bar','hello world')	@x:(foo|bar|"hello world")	quotes mean exact phrase
WHERE y='foo' AND x NOT IN ('foo','bar')	@y:foo (-@x:foo) (-@x:bar)	
WHERE x NOT IN ('foo','bar')	-@x:(foo|bar)	
WHERE num BETWEEN 10 AND 20	@num:[10 20]	
WHERE num >= 10	@num:[10 +inf]	
WHERE num > 10	@num:[(10 +inf]	
WHERE num < 10	@num:[-inf (10]	
WHERE num <= 10	@num:[-inf 10]	
WHERE num < 10 OR num > 20	@num:[-inf (10] | @num:[(20 +inf]	
WHERE name LIKE 'john%'	@name:john*	

Подводные камни Redis-OM-Spring

- Сложности с использованием UUID.

  ```
  @id:{2e5af72b\-02af\-443b\-8961\-248878aa381b}
  ```

  ```
  String regex = "(\\$" + key + «)(\\W+|\\*|\\+)(.*)»;
  ```

  https://redis.io/docs/stack/search/reference/escaping/

- Нужно передавать Pageable если ожидаете коллекцию

- В версиях 0.6.x и 0.8.x изменились конверторы, это влияет на результаты поиска

Подводные камни RediSearch

- При использовании пространства имен в БД вы должны использовать 0

Иначе вот такое сообщение «Cannot create index on db != 0»

- FT.SEARCH принимает только 1 параметр сортировки

Выводы

Помог ли нам Redis? - Да

Redis только кеш? - Нет

Redis подходит для легкой аналитики? - Да

Плюсы

- **Легко внедряется в проект**
- **Делает много шаблонной работы**
- **Быстро отрабатывает запросы**
- **Стабилен под нагрузкой**

Минусы

- **Не так много фич как в sql**
- **Мало документации**

Артем Артемьев

@just_TimTIm

https://www.linkedin.com/in/artem-artemev-b54043173/

Resources

https://redis.io/docs/stack/json/

https://redis.io/docs/stack/search/

https://redis.com/blog/introducing-redis-om-client-libraries

https://github.com/redis/redis-om-spring

GitHub Demo project https://github.com/justTimTim/demo-redisearch-and-sql