1. **В базе данных Redis подберите коллекцию для подсчета посещений с определенных IP-адресов.**
2. Создаем множество со списком IP адресов, для которых будем считать посещения:

SADD ip\_address 164.37.105.130 143.39.77.87 93.12.87.99 26.167.26.116 145.238.197.200 242.45.221.231 184.129.25.1 125.101.165.167 199.87.7.79 177.157.132.79

1. Создаем переменную - счетчик количества посещений:

SET visits\_count 0

1. Проверяем наличие IP адреса в множестве:

SISMEMBER ip\_address 93.12.87.99

Я не совсем понял, как данное задание можно сделать только средствами Redis. Я так понимаю, Redis не поддерживает условия. Поэтому необходимо использовать backend для проверки вхождения IP адреса в множество и принятия решения об увеличении счетчика.

1. Увеличиваем счетчик посещений

INCRBY visits 1

1. **При помощи базы данных Redis решите задачу поиска имени пользователя по электронному адресу и наоборот, поиск электронного адреса пользователя по его имени.**

В одну строну сделать все просто:

1. Создать пары значений:

SET dima dima@site.net

SET dima2 dima2@ site.net

…

1. Поиск электронного адреса по имени пользователя

GET dima

1. Поиска ключа по значению я не нашел, поэтому у меня только один вариант – создать противоположные значения:

SET dima@site.net dima

SET dima2@ site.net dima2

…

1. Поиск имени пользователя по электронному адресу

GET dima@site.net

1. **Организуйте хранение категорий и товарных позиций учебной базы данных shop в СУБД MongoDB.**

use shop  
db.shop.insert(  
{  
 "products": [  
 {  
 "id": 1,  
 "category": "Процессоры",  
 "name": "Intel Core i3-8100",  
 "description": "Процессор для настольных персональных компьютеров, основанных на платформе Intel.",  
 "price": 7890.00,  
 "created\_at": "2020-02-12 11:14:02",  
 "updated\_at": "2020-02-12 11:14:02"  
 },  
 {  
 "id": 5,  
 "category": "Материнские платы",  
 "name": "ASUS ROG MAXIMUS X HERO",  
 "description": "Материнская плата ASUS ROG MAXIMUS X HERO, Z370, Socket 1151-V2, DDR4, ATX",  
 "price": 19310.00,  
 "created\_at": "2020-02-12 11:14:02",  
 "updated\_at": "2020-02-12 11:14:02"  
 }  
]  
})