Задание:

Используя СУБД MongoDB, разработайте базу данных, предназначенную  для хранения логов веб-сервера. Лог включает в себя следующие поля: адрес ресурса (URL), IP-адрес пользовательского компьютера, отметка времени начала просмотра ресурса, длительность просмотра ресурса.

Для создания БД нужно использовать команду:

use lab8, где lab8 будет названием нашей БД.

Для создания коллекции необходимо выбрав БД использовать команду:

db.createCollection("log"), где log название коллекции.

Для добавления записи можно воспользоваться командой:

db.log.insertOne({" \_id":1.0,

"URL":"[https://vk.com](https://vk.com/)",

"IP":"138.26.55.44",

"timeStamp":1.5870816e+09,

"timeSpent":20.0})

Таким образом мы добавим запись с полями \_id, URL, IP, timeStamp, timeSpent.

**Указания к заданию**

1. Разработайте консольную утилиту для преобразования лога веб-сервера в формате CSV (Comma Separated Values), в формат JSON. Лог должен содержать поля со следующими названиями: URL, IP, timeStamp, timeSpent.

Решение:

mongoimport --db lab8 --collection csvToLogs --file logs.csv --type csv —headerline

mongoexport --db lab8 --collection logs --out /home/user/logs.json

2. Разработайте запросы для загрузки полученных данных в формате JSON в СУБД MongoDB.

Решение:

mongoimport --db lab8 --collection jsonToLogs --file logs.json --type json

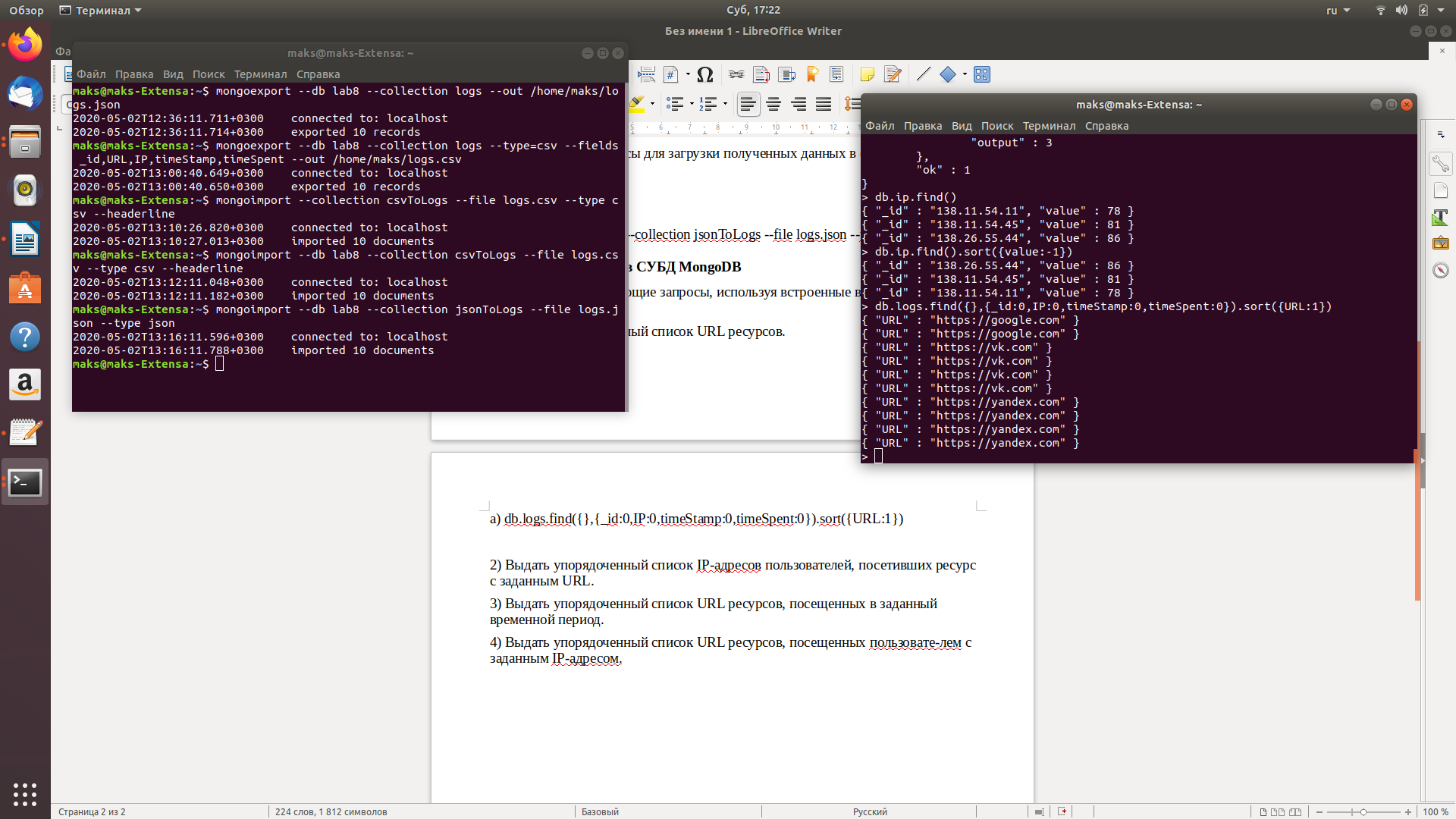
[**Разработка запросов в СУБД MongoDB**](../../media/maks/%D0%9B%D0%BE%D0%BA%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9%20%D0%B4%D0%B8%D1%81%D0%BA/%D1%83%D1%87%D0%B5%D0%B1%D0%B0/6%20%D1%81%D0%B5%D0%BC%D0%B5%D1%81%D1%82%D1%80/sitiod/lb8/lab8.htm#page6)

1. Разработайте следующие запросы, используя встроенные в СУБД MongoDB средствами выборки:

1) Выдать упорядоченный список URL ресурсов.

Решение:

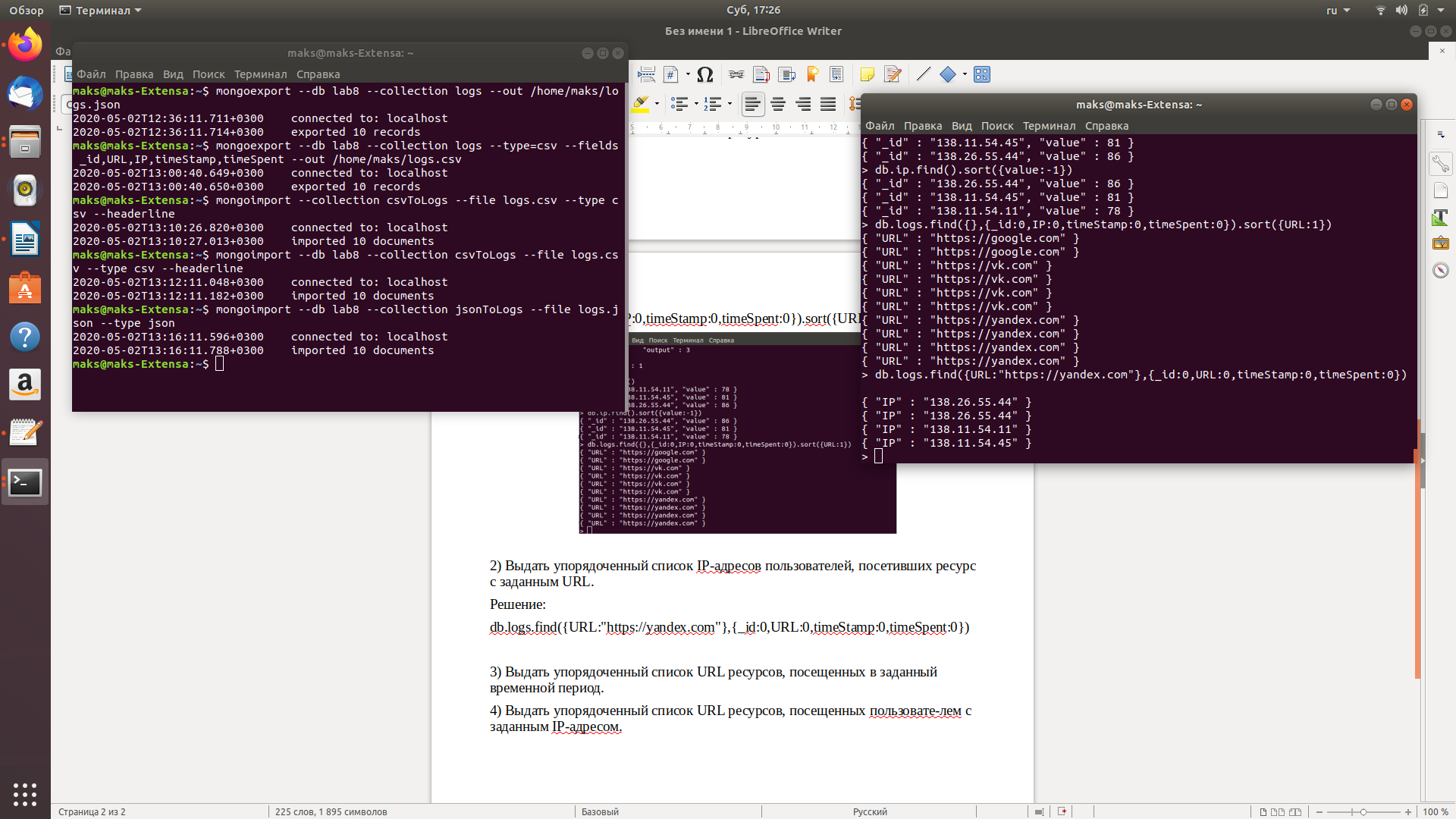
db.logs.find({},{\_id:0,IP:0,timeStamp:0,timeSpent:0}).sort({URL:1})



2) Выдать упорядоченный список IP-адресов пользователей, посетивших ресурс с заданным URL.

Решение:

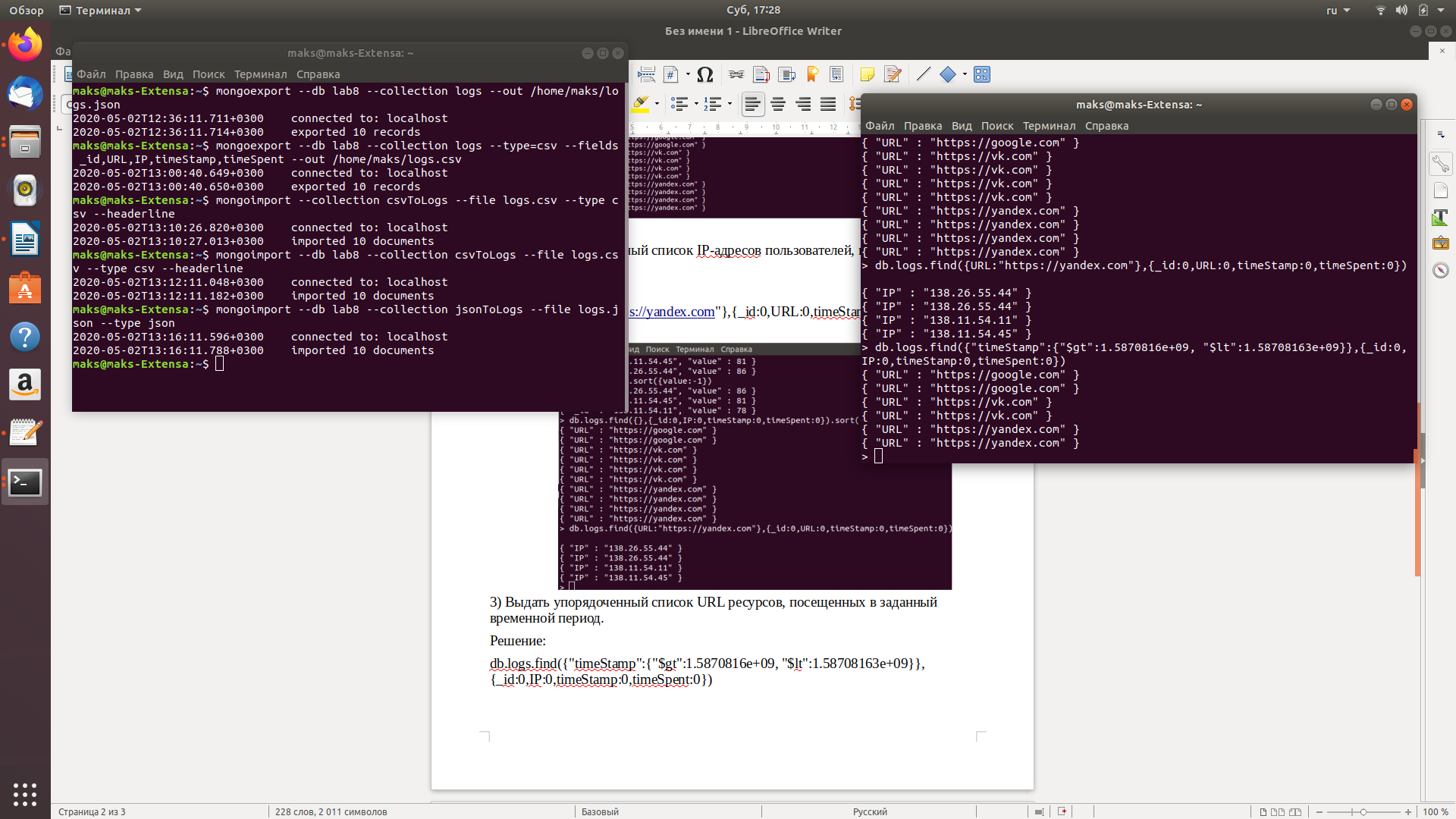
db.logs.find({URL:"[https://yandex.com](https://yandex.com/)"},{\_id:0,URL:0,timeStamp:0,timeSpent:0})



3) Выдать упорядоченный список URL ресурсов, посещенных в заданный временной период.

Решение:

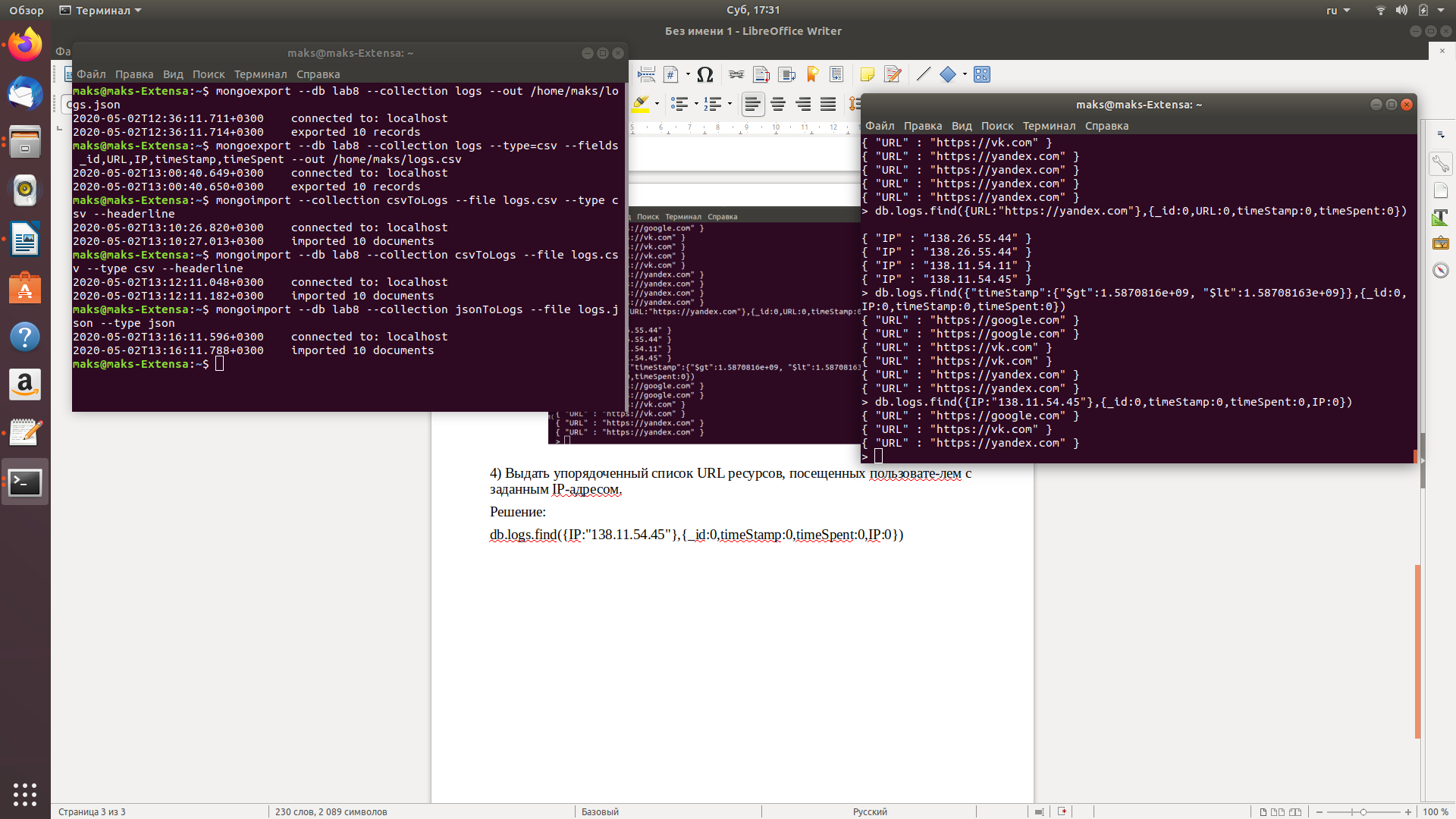
db.logs.find({"timeStamp":{"$gt":1.5870816e+09, "$lt":1.58708163e+09}},{\_id:0,IP:0,timeStamp:0,timeSpent:0})



4) Выдать упорядоченный список URL ресурсов, посещенных пользовате-лем с заданным IP-адресом.

Решение:

db.logs.find({IP:"138.11.54.45"},{\_id:0,timeStamp:0,timeSpent:0,IP:0})



2. Разработайте следующие запросы, используя встроенные в СУБД MongoDB средства программирования на основе парадигмы MapReduce:

1) Выдать список URL ресурсов с указанием суммарной длительности по-сещения каждого ресурса, упорядоченный по убыванию.

Решение:

function map() {

var urls = this.URL;

var length = this.timeSpent;

emit(urls, length);

}

function reduce(key, values) {

var totalTimeSpent = 0;

for (var i = 0; i < values.length; i++) {

totalTimeSpent += values[i];

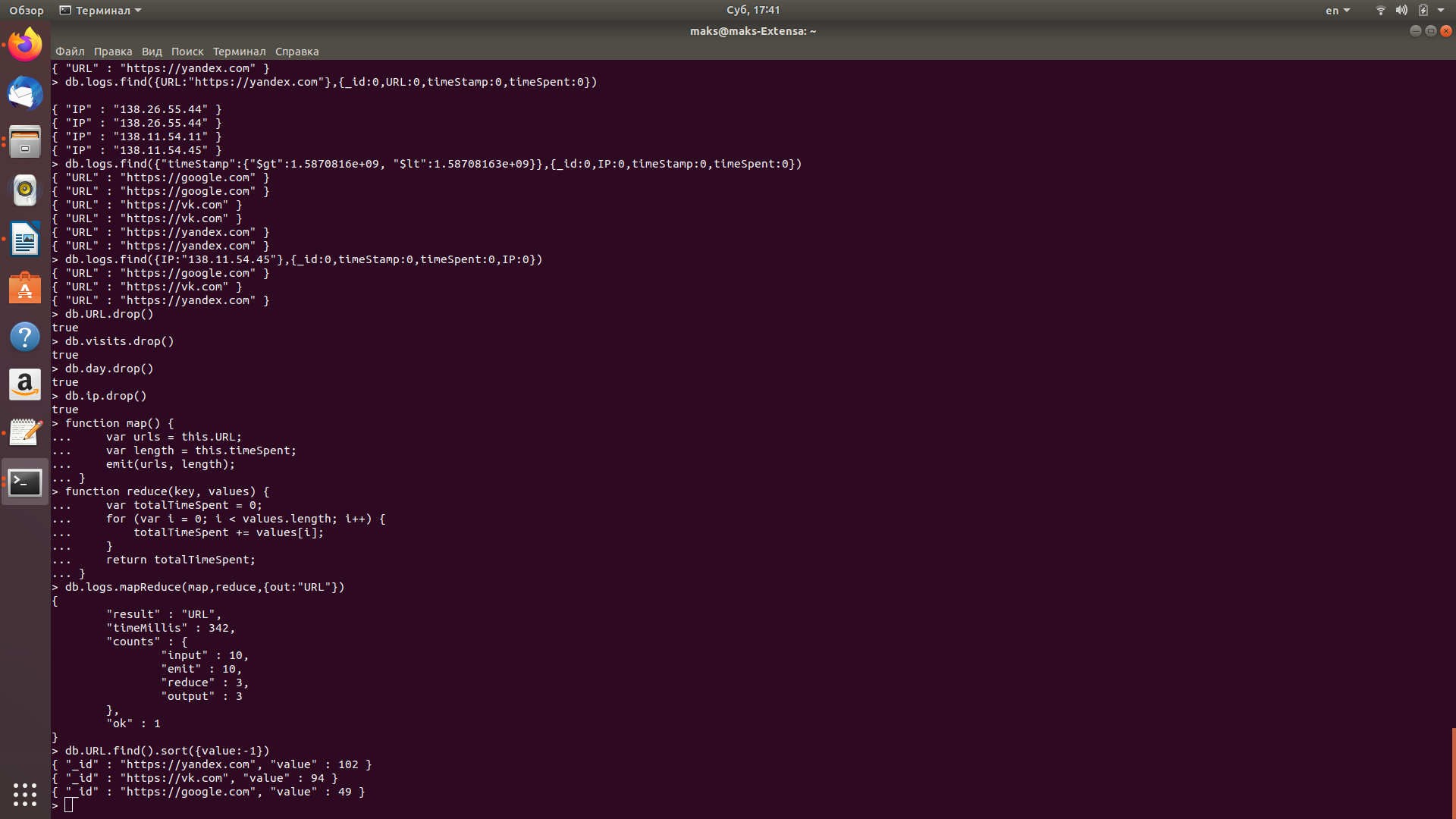
}

return totalTimeSpent;

}

db.logs.mapReduce(map,reduce,{out:"URL"})

db.URL.find().sort({value:-1})



2) Выдать список URL ресурсов с указанием суммарного количества посе-щений каждого ресурса, упорядоченный по убыванию.

Решение:

function map() {

var urls = this.URL;

emit(urls, 1);

}

function reduce(key, values) {

var count = 0;

for (var i = 0; i < values.length; i++) {

count += values[i];

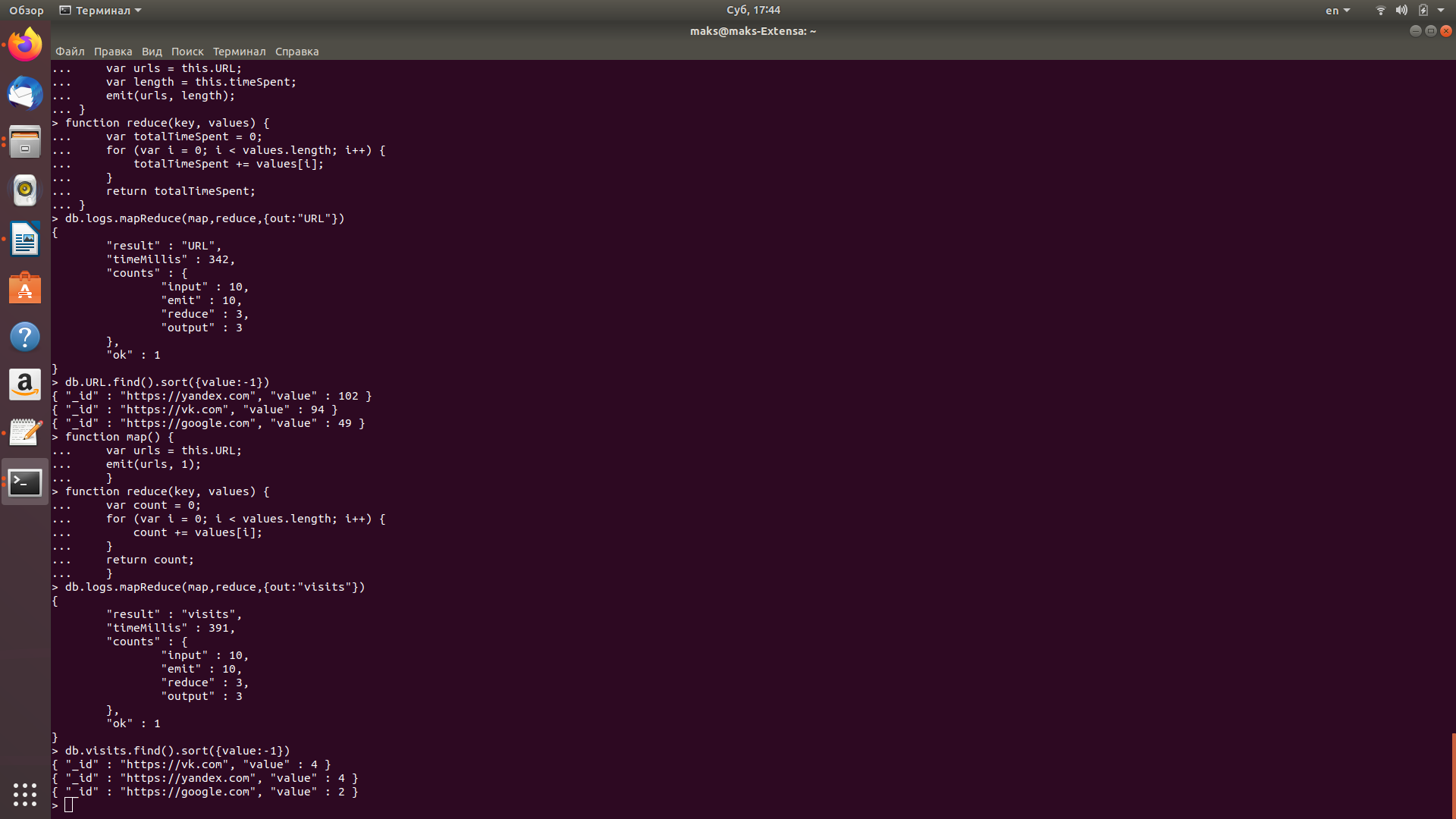
}

return count;

}

db.logs.mapReduce(map,reduce,{out:"visits"})

db.visits.find().sort({value:-1})



3) Выдать список URL ресурсов с указанием количества посещений каждого ресурса в день за заданный период, упорядоченный URL ресурса и убы-ванию количества посещений.

Решение:

function map() {

var url = this.URL;

var date = new Date(this.timeStamp \* 1000);

emit({url: url, date: date}, 1);

}

function reduce(key, values) {

var count = 0;

for (var i = 0; i < values.length; i++) {

count += values[i];

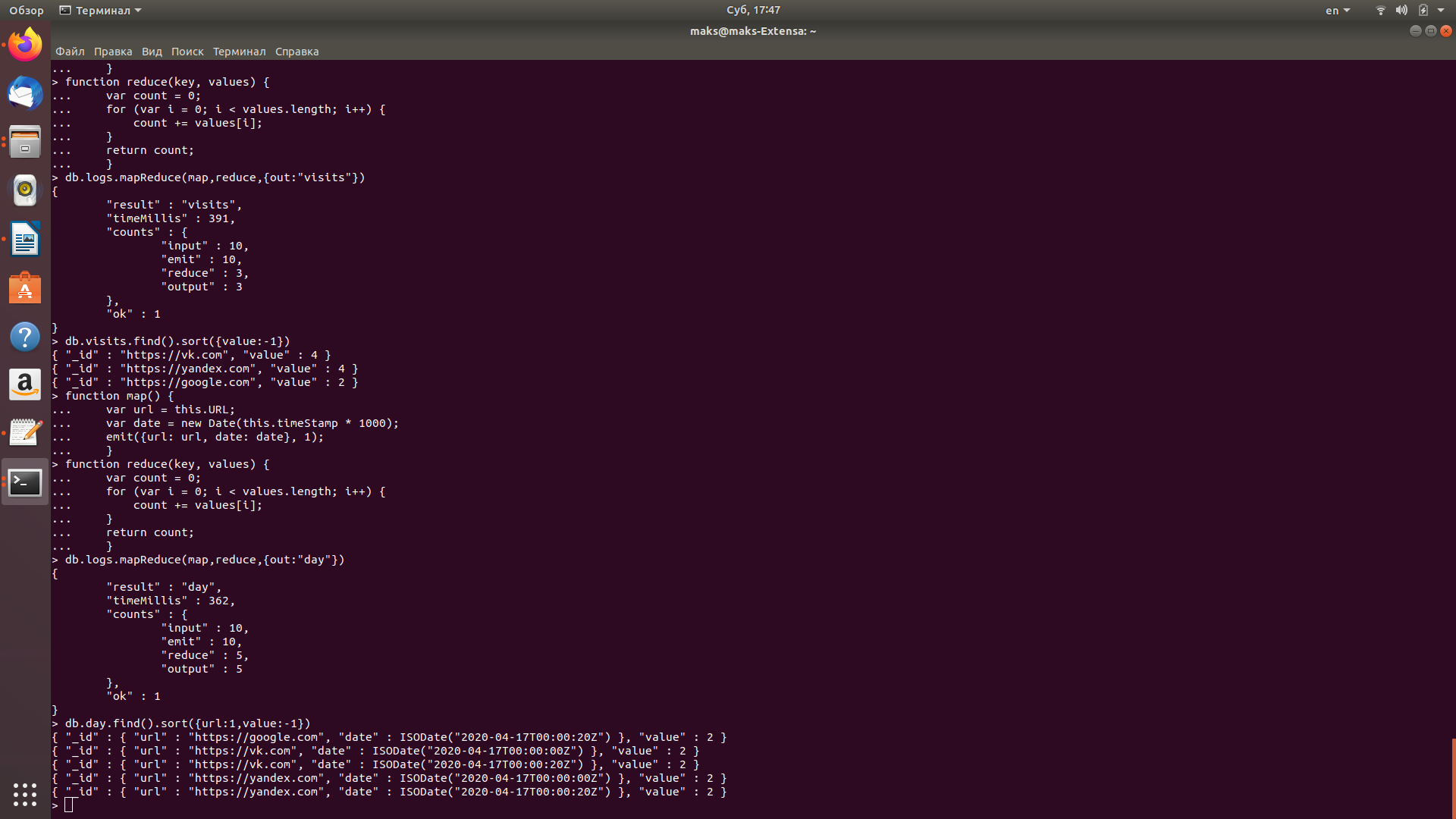
}

return count;

}

db.logs.mapReduce(map,reduce,{out:"day"})

db.day.find().sort({url:1,value:-1})



4) Выдать список IP-адресов c указанием суммарного количества и суммар-ной длительности посещений ресурсов, упорядоченный по адресу, убы-ванию количества и убыванию длительности.

Решение:

function map() {

var ip = this.IP;

emit(ip, 1);

}

function map() {

var ip = this.IP;

var timeSpent = this.timeSpent;

emit(ip, timeSpent);

}

function reduce(key, values) {

var count = 0;

for (var i = 0; i < values.length; i++) {

count += values[i];

}

return count;

}

db.logs.mapReduce(map,map,reduce,{out:"ip"})

db.ip.find().sort({value:-1})

