Министерство образования Республики Беларусь

Учреждение образования

Белорусский государственный университет информатики и радиоэлектроники

Инженерно-экономический факультет

Кафедра экономической информатики

Дисциплина: Системы интеллектуальной обработки данных

ОТЧЕТ

по лабораторной работе №8

на тему

**NoSQL базы данных**

|  |  |
| --- | --- |
| Выполнил: | Студент группы 772302  П. Е. Буря |
| Проверил: | Ассистент кафедры ЭИ  А. А. Кунцевич |

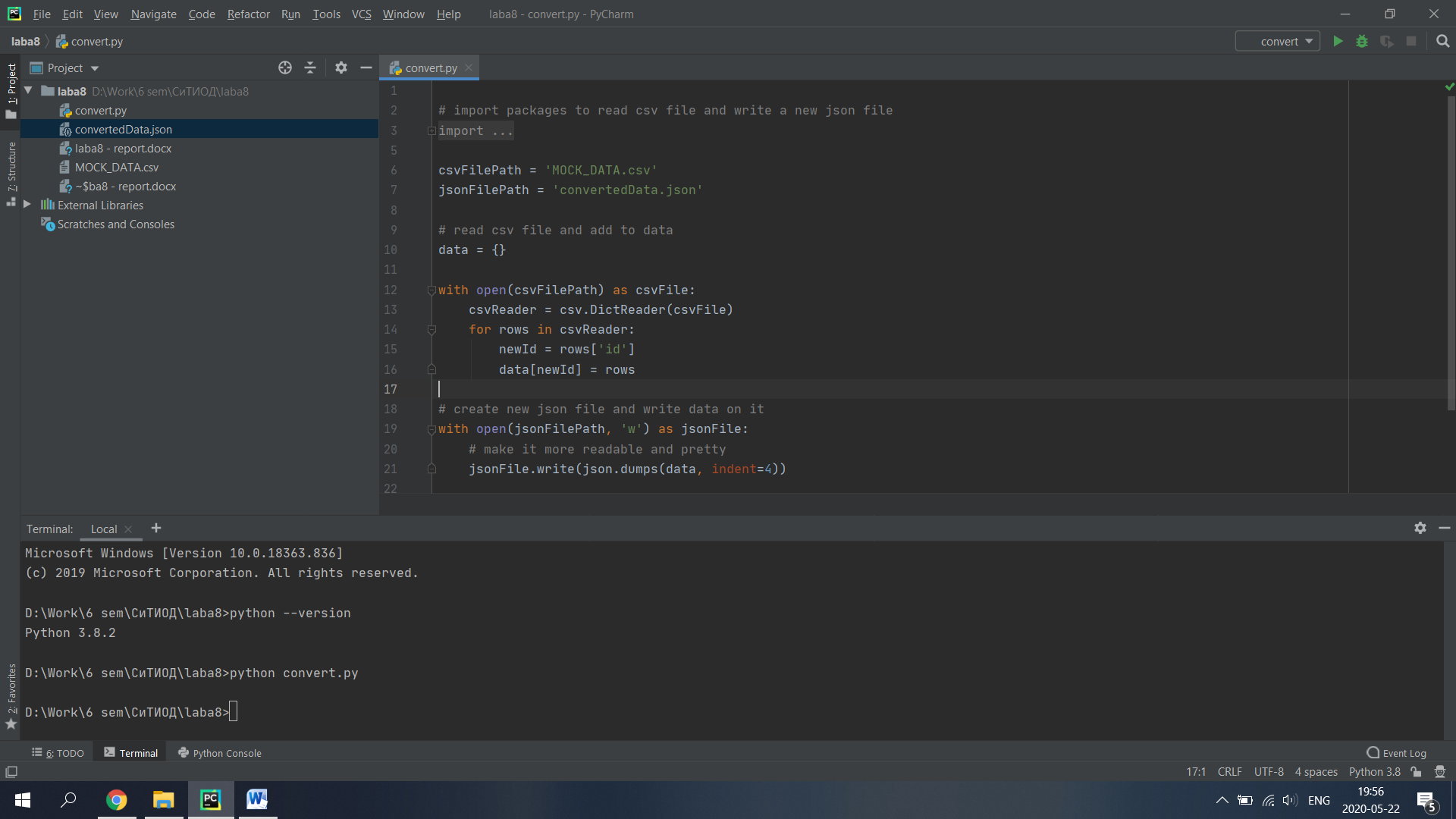
Минск 2020

**Задание**

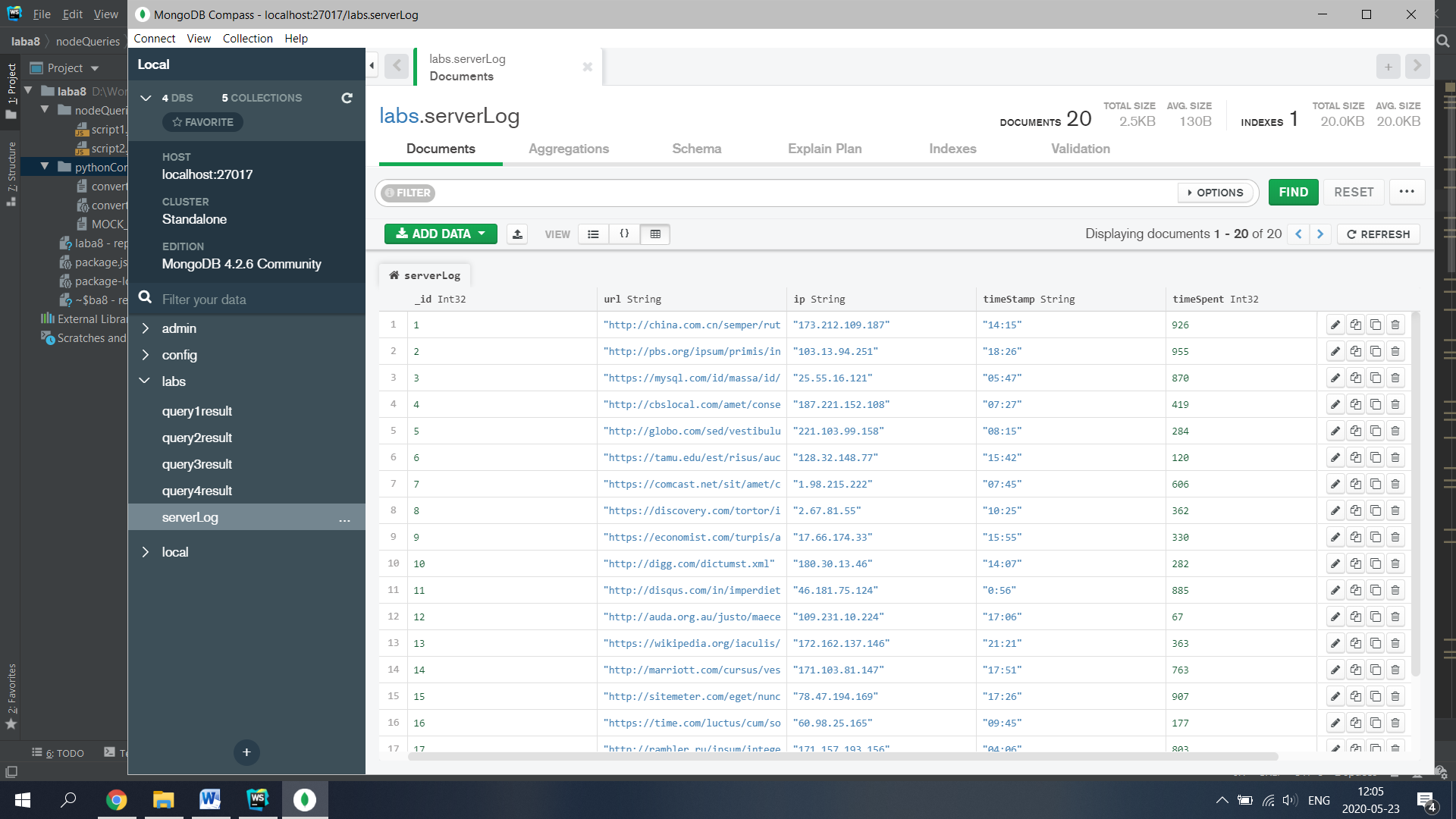
1. Разработайте консольную утилиту для преобразования лога веб-сервера в формате CSV (Comma Separated Values), в формат JSON. Лог должен содержать поля со следующими названиями: URL, IP, timeStamp, timeSpent.
2. Разработайте запросы для загрузки полученных данных в формате JSON в СУБД MongoDB.

**Ход работы**

Код конвертера из csv в json:



Структура базы данных в монго:



Код запросов на монго скрипте с использованием FIND:

let ***conn*** = new Mongo();  
let ***db*** = ***conn***.getDB("labs");  
let ***serverLog*** = ***db***.serverLog;  
let ***result***;  
  
  
// -- FIND --  
  
// get sorted url's  
  
***result*** = ***serverLog*** .find({}, { \_id: 0, url: 1 })  
 .sort({ url: 1 });  
  
while (***result***.hasNext()) {  
 printjson(***result***.next());  
}  
print();  
  
  
// get sorted ip's with the specified url  
  
***result*** = ***serverLog*** .find(  
 { url: "https://discovery.com/tortor/id/nulla/ultrices.js" },  
 { \_id: 0, ip: 1 }  
 )  
 .sort({ ip: 1 });  
  
while (***result***.hasNext()) {  
 printjson(***result***.next());  
}  
print();  
  
  
// get sorted url's within the specified time period  
  
***result*** = ***serverLog*** .find(  
 { timeStamp: { $gt: '12:00', $lt: '23:59' } },  
 { \_id: 0, url: 1 }  
 )  
 .sort({ url: 1 });  
  
while (***result***.hasNext()) {  
 printjson(***result***.next());  
}  
print()  
  
// get sorted url's with the specified ip  
  
***result*** = ***serverLog*** .find(  
 { ip: "1.98.215.222" },  
 { \_id: 0, url: 1 }  
 )  
 .sort({ url: 1 });  
  
while (***result***.hasNext()) {  
 printjson(***result***.next());  
}  
print();

let ***conn*** = new Mongo();  
let ***db*** = ***conn***.getDB("labs");  
let ***serverLog*** = ***db***.serverLog;  
let ***result***;

Код запросов на монго скрипте с использованием MAP REDUCE:

// -- MAP REDUCE --  
  
// get url and it's sum of timeSpent (sorted desc)  
  
***serverLog***.mapReduce(  
 function () {  
 emit(this.url, this.timeSpent);  
 },  
 function (key, value) {  
 return ***Array***.sum(value);  
 },  
 { out: "query1result" }  
);  
  
***result*** = ***db***.query1result.find().sort( { value: -1 } )  
  
while (***result***.hasNext()) {  
 printjson(***result***.next());  
}  
print();  
  
  
// get url and it's sum of ip's (sorted desc)  
  
***serverLog***.mapReduce(  
 function () {  
 emit(this.url, 1);  
 },  
 function (key, value) {  
 return ***Array***.sum(value);  
 },  
 { out: "query2result" }  
);  
  
***result*** = ***db***.query2result.find().sort( { value: -1 } )  
  
while (***result***.hasNext()) {  
 printjson(***result***.next());  
}  
print();  
  
  
// get url and it's sum of visits within the specified time period (sorted by url, and by visits number (desc))  
  
***serverLog***.mapReduce(  
 function () {  
 if (this.timeStamp.localeCompare('23:00') === -1) emit(this.url, 1);  
 else emit (this.url, 0);  
 },  
 function (key, value) {  
 return ***Array***.sum(value);  
 },  
 { out: "query3result" }  
);  
  
***result*** = ***db***.query3result.find().sort( { \_id: 1, value: -1 } )  
  
while (***result***.hasNext()) {  
 printjson(***result***.next());  
}  
print();  
  
  
// get url and it's sum of visits and sum of timeSpent (sorted by url, visits number (desc), timeSpent(desc))  
  
***serverLog***.mapReduce(  
 function () {  
 emit(this.url, {visits: 1, timeSpent: this.timeSpent});  
 },  
 function (key, value) {  
 let returnValue = {  
 visits: 0,  
 timeSpent: 0,  
 };  
 for (let i = 0; i < value.left; i++) {  
 returnValue.visits += value.visits;  
 returnValue.timeSpent += value.timeSpent;  
 }  
 return returnValue;  
 },  
 { out: "query4result" }  
);  
  
***result*** = ***db***.query4result.find().sort( { \_id: 1, 'value.visits': -1, 'value.timeSpent': -1 } )  
  
while (***result***.hasNext()) {  
 printjson(***result***.next());  
}  
print();