МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«МИРЭА – РОССИЙСКИЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»
РТУ МИРЭА

Институт: Институт комплексной безопасности и цифровых технологий (ИКБ)

Специальность (направление): Информационные системы и технологии

Кафедра: КБ-14 «Цифровые технологии обработки данных»

Дисциплина: Программные средства манипулирования данными

Отчёт по практической работе №6

Подключение и выгрузка сторонних неструктурированных данных

Студент: _____ 9.11.2022 Иванов С.К., Денега А.П.
подпись *дата* *инициалы и фамилия*

Группа: БСБО-06-20

Преподаватель: _____ _____ Котилевец И.Д.
подпись *дата* *инициалы и фамилия*

Москва 2022 г.

Задания на практическую работу:

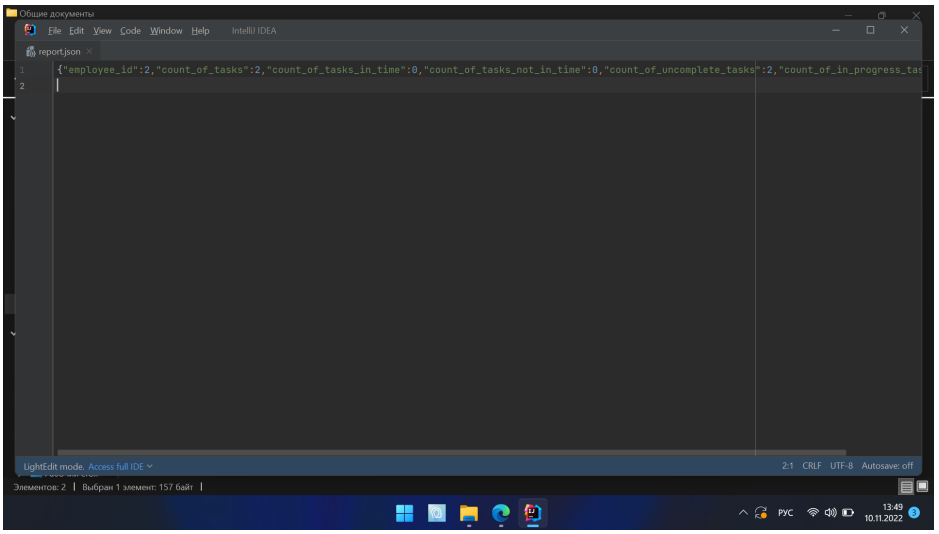
1. На основе практических работ №2-5 разработать механизм отчёта по заданиям, который выгружается в формате .json и может храниться в нереляционных базах данных.
2. Выбрать одну из нереляционных баз данных (MongoDB, CouchDB, Cassandra, CockroachDB) и выполнить пять запросов к отчёту через неё.

1. Разработать механизм отчёта в формат .json

В практических работах №2-5 мы уже создавали выгрузку отчёта в формате .csv. Выгрузка в файла .json почти ничем не отличается, но для начала требуется выполнить конвертацию строки в json объект. Для этого нужно использовать функцию «row_to_json()», которая как раз занимается переводом строки в json объект. В скобках надо указать, что именно мы хотим конвертировать.

```
--Выгрузка отчёта в json формат

-- 1. Нужно преобразовать строки в json объекты с помощью команды row_to_json()

   COPY (SELECT row_to_json(report) FROM get_report(1,
'2022-10-24'::timestamp without time zone,
'2022-11-30'::timestamp without time zone) as report)

TO 'C:\Users\Public\Documents\report.json';

-- WITH (FORMAT text, HEADER false) применяется дефолтно
```

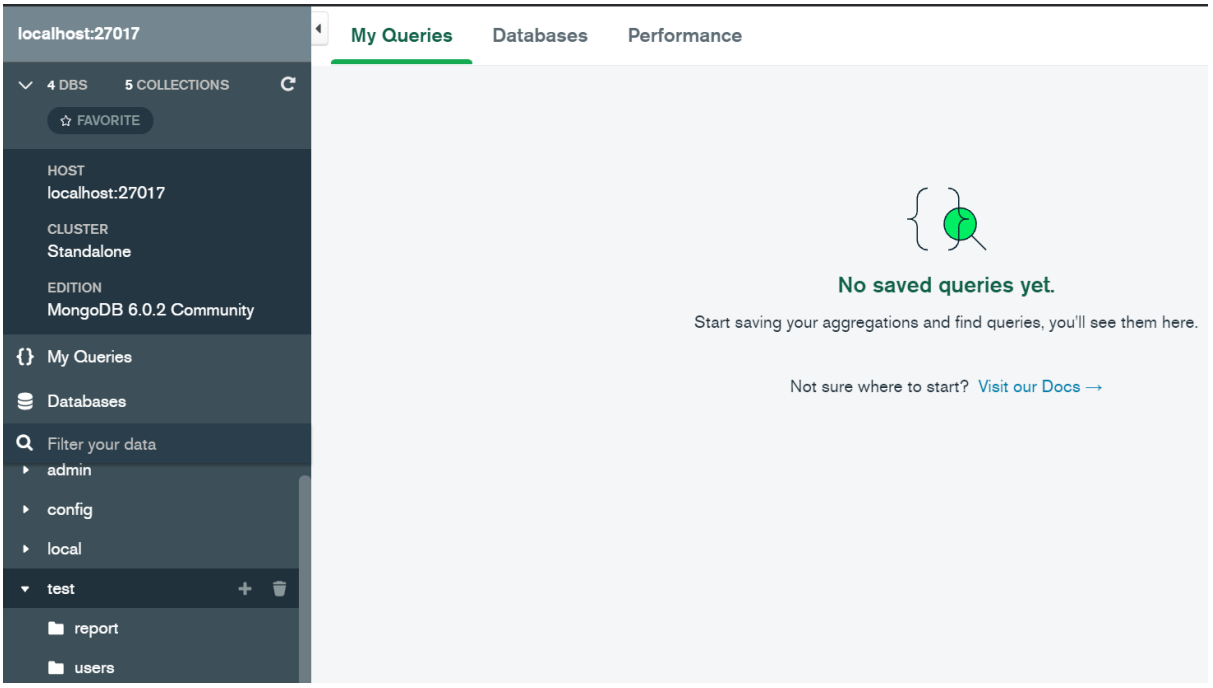
После выполнения этой команды мы получаем файл .json в нашей директории:



2. Выполнить пять запросов к отчёту через выбранную базу данных

Для выполнения этой части практической работы мы выбрали MongoDB и установили MongoDB server с официального сайта. Там есть несколько версий под разные нужды, но мы скачали бесплатную версию «Community», так как её функционала нам достаточно. После установки сервера установщик также поставил графический редактор для этой нереляционной базы данных – «MongoDB Compass».

Далее мы подключаемся к нашему серверу и можем создать свою БД и коллекцию в ней (test, report):



Добавляем в коллекцию данные из нашего .json файла с отчётом и выполняем к нему несколько запросов.

```
> db.report.find()
< { _id: ObjectId("636bb7d33ac0eb3762b21729"),
    employee_id: 1,
    count_of_tasks: 3,
    count_of_tasks_in_time: 1,
    count_of_tasks_not_in_time: 0,
    count_of_uncomplete_tasks: 0,
    count_of_in_progress_tasks: 2 }
```

```
> db.report.find({employee_id: 1})
< { _id: ObjectId("636bb7d33ac0eb3762b21729"),
    employee_id: 1,
    count_of_tasks: 3,
    count_of_tasks_in_time: 1,
    count_of_tasks_not_in_time: 0,
    count_of_uncomplete_tasks: 0,
    count_of_in_progress_tasks: 2 }
```

```
> db.report.insertOne({"employee_id":2,"count_of_tasks":2,"count_of_tasks_in_time":0,"count_of_tasks_not_in_time":0,"count_of_uncomplete_tasks":2,"count_of_in_pr
< { acknowledged: true,
  insertedId: ObjectId("636bba53ff503ace9763a05f") }
```

```
> db.report.find()
< { _id: ObjectId("636bb7d33ac0eb3762b21729"),
    employee_id: 1,
    count_of_tasks: 3,
    count_of_tasks_in_time: 1,
    count_of_tasks_not_in_time: 0,
    count_of_uncomplete_tasks: 0,
    count_of_in_progress_tasks: 2 }
  { _id: ObjectId("636bba53ff503ace9763a05f"),
    employee_id: 2,
    count_of_tasks: 2,
    count_of_tasks_in_time: 0,
    count_of_tasks_not_in_time: 0,
    count_of_uncomplete_tasks: 2,
    count_of_in_progress_tasks: 0 }
```

```
> db.report.updateOne({employee_id: 2}, {$set:{count_of_tasks: 3, count_of_tasks_in_time: 1}})
< { acknowledged: true,
    insertedId: null,
    matchedCount: 1,
    modifiedCount: 1,
    upsertedCount: 0 }
> db.report.find()
```

```
> db.report.deleteOne({employee_id: 2})
< { acknowledged: true, deletedCount: 1 }
```