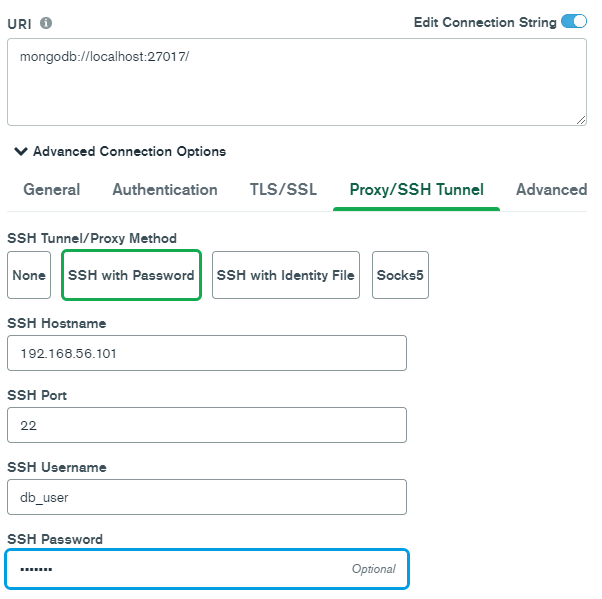
1. Подключиться к серверу Linux на виртуальной машине

db\_user dbStudy

2. Откройте оболочку Compass и подключитесь к базе данных mongo через SSH соединение (предварительно убедившись в том, что сервис mongodb запущен на виртуальной машине).



3. С помощью инструмента mongosh (в дальнейшем, писать команды будем тут).

4. Переключитесь на базу данных provider (если ее у вас нет, см. описание лабораторной работы 2).

5. С помощью функции группировки $group и функции $count сосчитайте, используя коллекцию internet, сколько сессий интернет по дням открывали клиенты.

db.internet.aggregate ([{$group: {\_id: "$session\_date", countNumberOfDocumentsForState: {$count: {}}}}])

6. С помощью функции группировки $group выведите все уникальные имена пользователей из коллекции users, отсортировав массив в возрастающем порядке.

db.users.aggregate ([{$group: {\_id: “$first\_name”}}]).sort ({\_id: 1})

7. В базу данных agg добавьте коллекцию sales со следующими документами:

db.sales.insertMany([

{ "\_id" : 1, "item" : "abc", "price" : NumberDecimal("10"), "quantity" : NumberInt("2"), "date" : ISODate("2014-03-01T08:00:00Z") },

{ "\_id" : 2, "item" : "jkl", "price" : NumberDecimal("20"), "quantity" : NumberInt("1"), "date" : ISODate("2014-03-01T09:00:00Z") },

{ "\_id" : 3, "item" : "xyz", "price" : NumberDecimal("5"), "quantity" : NumberInt( "10"), "date" : ISODate("2014-03-15T09:00:00Z") },

{ "\_id" : 4, "item" : "xyz", "price" : NumberDecimal("5"), "quantity" : NumberInt("20") , "date" : ISODate("2014-04-04T11:21:39.736Z") },

{ "\_id" : 5, "item" : "abc", "price" : NumberDecimal("10"), "quantity" : NumberInt("10") , "date" : ISODate("2014-04-04T21:23:13.331Z") },

{ "\_id" : 6, "item" : "def", "price" : NumberDecimal("7.5"), "quantity": NumberInt("5" ) , "date" : ISODate("2015-06-04T05:08:13Z") },

{ "\_id" : 7, "item" : "def", "price" : NumberDecimal("7.5"), "quantity": NumberInt("10") , "date" : ISODate("2015-09-10T08:43:00Z") },

{ "\_id" : 8, "item" : "abc", "price" : NumberDecimal("10"), "quantity" : NumberInt("5" ) , "date" : ISODate("2016-02-06T20:20:13Z") }

])

8. Сгруппируйте продаваемые продукты по названию, сосчитайте сумму продаж по каждой группе и выведите эту сумму продаж для тех продуктов, где эта сумма превышает значение 100.

db.sales.aggregate ([{$group: {\_id: "$item", totalSaleAmount: { $sum: { $multiply: [ "$price", "$quantity" ]}}}}, {$match: {"totalSaleAmount": {$gte: 100}}} ])

9. Ознакомьтесь с идентификаторами типов данных BSON, приведенными по ссылке https://www.mongodb.com/docs/manual/reference/bson-types/.

10. Вернитесь к базе данных provider и попробуйте вывести пользователей из коллекции users, у которых first\_name имеет тип данных double (числовой).

db.users.find ({first\_name: {$type : 1}})

11. Создайте коллекцию education со следующими документами:

db.education.insert ([{\_id: "МГУ", name: "Московский Государственный Университет"}, {\_id: "ВШЭ", name: "Высшая Школа Экономики"}])

12. В коллекции personnel, созданной ранее, в документе с именем Иванов добавим атрибут study со значением МГУ.

db.personnel.update ({name: "Петров"}, {$set: {study: "МГУ"}}, {upsert: true})

13. Соедините в запросе find данные Петрова из коллекции personnel и коллекции study. Узнайте в каком университете учится Петров.

var stud = db.personnel.findOne ({name: "Петров"});

db.education.find ({\_id: stud.study})

14. Закройте Compass и закончите сессию в виртуальной машине.

exit