**Установка**

1. Установить с сайта <https://www.mongodb.com/try/download/community> программу MongoDB Community Server.

Для загрузки всех необходимых файлов выберем нужную операционную систему и подходящий тип пакета. MongoDB можно загрузить в ряде вариантов. Так, для Windows доступна загрузка установщика msi и также доступна загрузка zip-пакета. Для работы с сервером нам нужен клиент. Наиболее простым клиентом в данном случае является MongoDB Shell

1. Скачать с сайта <https://www.mongodb.com/try/download/tools> программу MongoDB Shell и распаковать в папку с программой MongoDB Community Server.

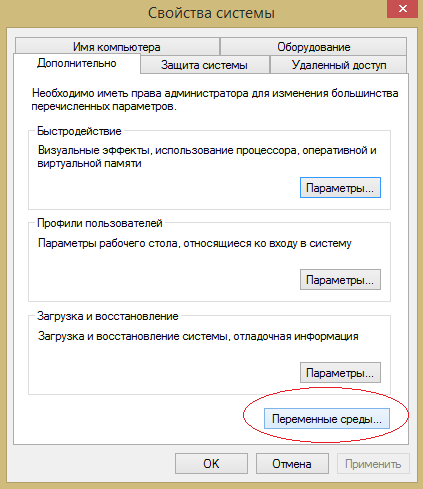
**C:\Program Files\MongoDB\**

Если до установки уже была установлена более ранняя версия MongoDB, то ее необходимо удалить.

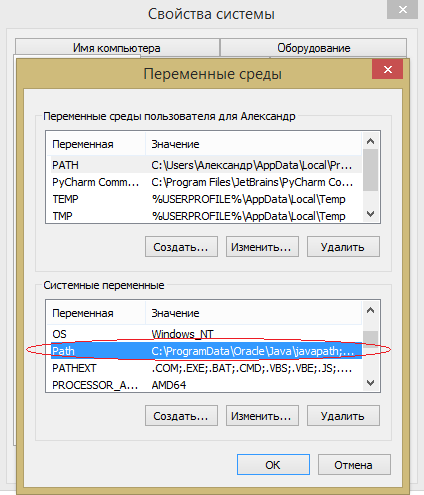
1. Скопировать путь к MongoDB Shell.

**C:\Program Files\MongoDB\mongosh-1.8.0-win32-x64\bin**

1. Открываем «Изменение системных переменных среды».



Открываем «Переменные среды». В «Системные переменные» выбираем **Path**. Вставляем (добавляем) скопированный путь.

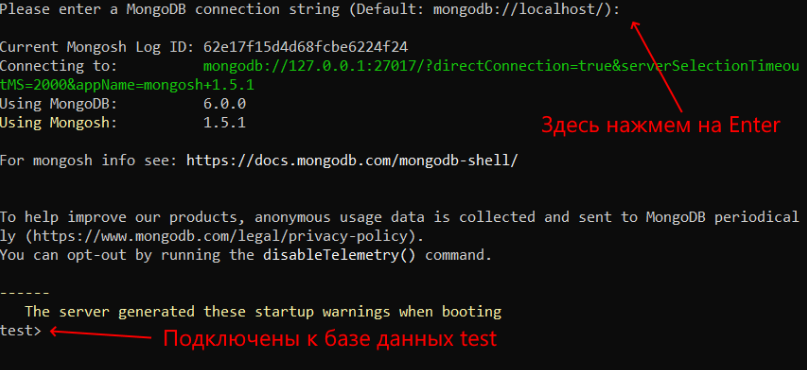


**Готово!**

Альтернативный вариант установки с использованием MongoDB Compass описан по ссылке <https://coderlessons.com/tutorials/bazy-dannykh/uchebnik-mongodb/3-ustanovka-i-nastroika-2>

**Подключение к серверу из клиента**

Используем выше установленное клиентское приложение mongosh для взаимодействия с сервером mongodb. (При работе с mongosh не стоит забывать, что у нас должен быть запущен сервер mongod). Итак, запустим файл mongosh, который располагается в выше рассмотренной папке установки. Открываем **командную строку** и вводим **mongosh.**



Все интересующие команды можно посмотреть в справочнике <https://www.mongodb.com/docs/mongodb-shell/>

Посмотреть все базы данных: **show dbs**

Создать новую базу данных **use** <название>: **use mydb**

С помощью функции **db.users.insertOne()** в коллекцию **users** базы данных **mydb** добавляется объект { name: "Tom" }. Идентификатор **db** представляет текущую базу данных. В нашем случае мы подключены к базе данных по умолчанию - то есть к базе данных **mydb**, соответственно **db** здесь представляет базу данных **mydb**. При этом не важно, есть или нет такая бд - если ее нет, то она создается.

После **db** идет **users** - это коллекция, в которую затем мы добавляем новый объект. Если в SQL нам надо создавать таблицы заранее, то коллекции MongoDB создает самостоятельно при их отсутствии.

Описание добавляемого объекта определяется в формате, с которым вы возможно знакомы, если имели дело с форматом JSON. То есть в данном случае у объекта определен один ключ "name", которому сопоставляется значение "Tom". То есть мы добавляем пользователя с именем Tom.

Если объект был успешно добавлен, то консоль выведет результат операции, в частности, идентификатор добавленного объекта.

Команда **db.users.find()** выводит на экран все объекты из бд **mydb**.

В выводе вы можете увидеть, что к начальным значениям объекта было добавлено какое-то непонятно поле **ObjectId**. MongoDB в качестве уникальных идентификаторов документа использует поле **\_id**. И в данном случае **ObjectId** как раз и представляет значение для идентификатора **\_id**.

**Задание.**

Разработайте и протестируйте запросы на выборку данных из созданных коллекций.

Создать коллекцию онлайн курсов со следующей информацией: название, продолжительность, дата начала, дата завершения, описание курса, лектор, стоимость.

Создать коллекцию пользователей онлайн курса со следующей информацией: идентификатор курса, ФИО, дата рождения, телефон, корпоративный клиент, страна, город, онлайн, пожелания.

Выполнить выборку данных:

1. Вывести средний возраст пользователей по каждому курсу.
2. Вывести курсы в которых физических клиентов, больше чем корпоративных клиент и их процентное соотношение.
3. Вывести курсы, которые уже завершились.
4. Вывести курсы, которые еще предстоят и количество записавшихся на них пользователей.
5. Вывести полученную прибыль от курсов, которые завершились и уже проходят.