

MongoDB安装和核心语法

朱俊泽

2351114

1.MongoDB和MySQL区别

MongoDB是一个非关系型数据库，是一个集合型的数据库。

不需要事先创造好数据库和集合，也不需要预定好字段和长度，更加灵活。

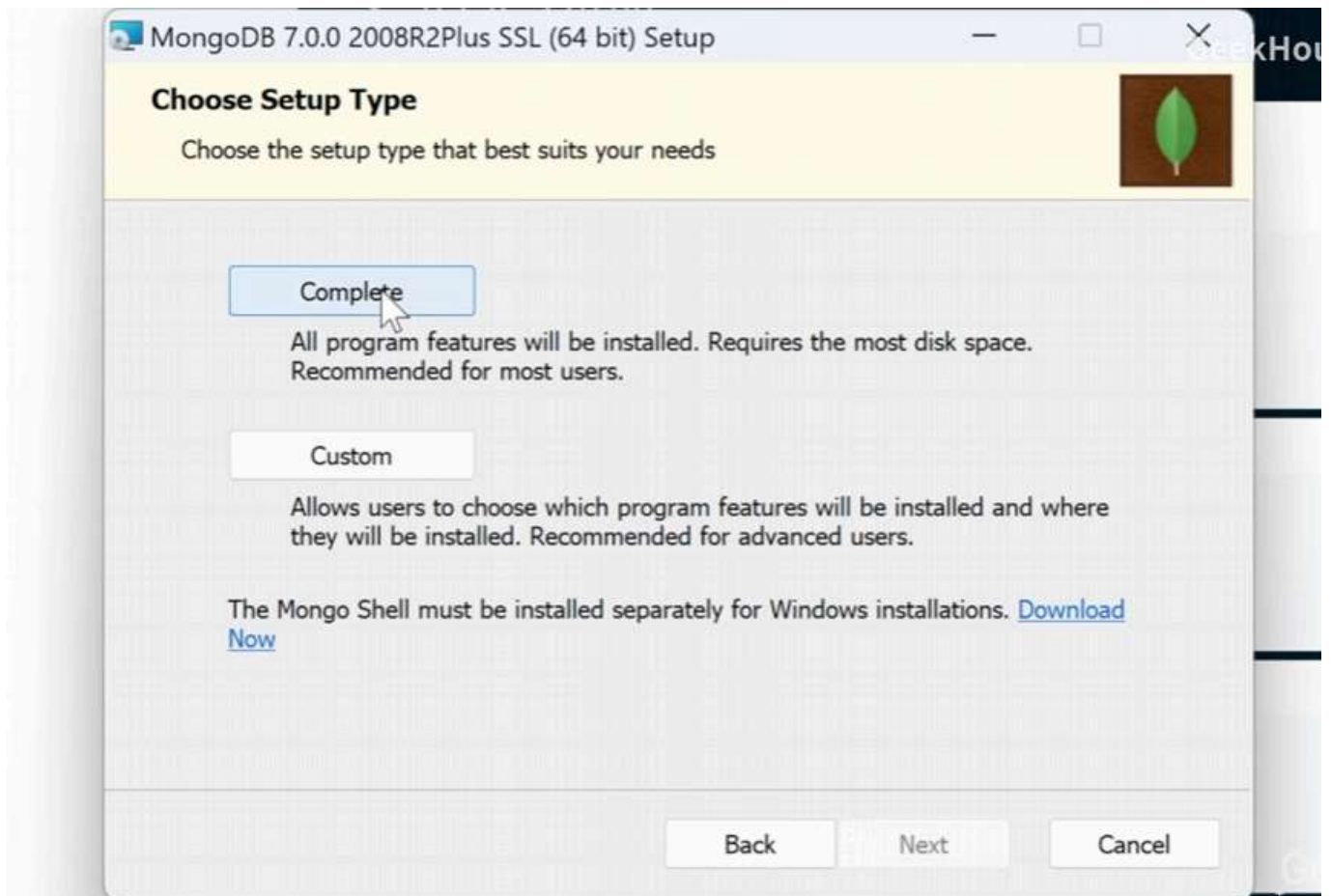
具体关系如下图



2.安装

从官网<https://www.mongodb.com/try> 下载好社区版本的MongoDB

complete完整安装就行



然后就可以安装了，注意他会安装一个图形化工具compass

以下分成两个部分的报告--图形化工具+普通版本工具

图形化不需要初始化环境变量，普通版本工具需要进行一系列的环境变量配置

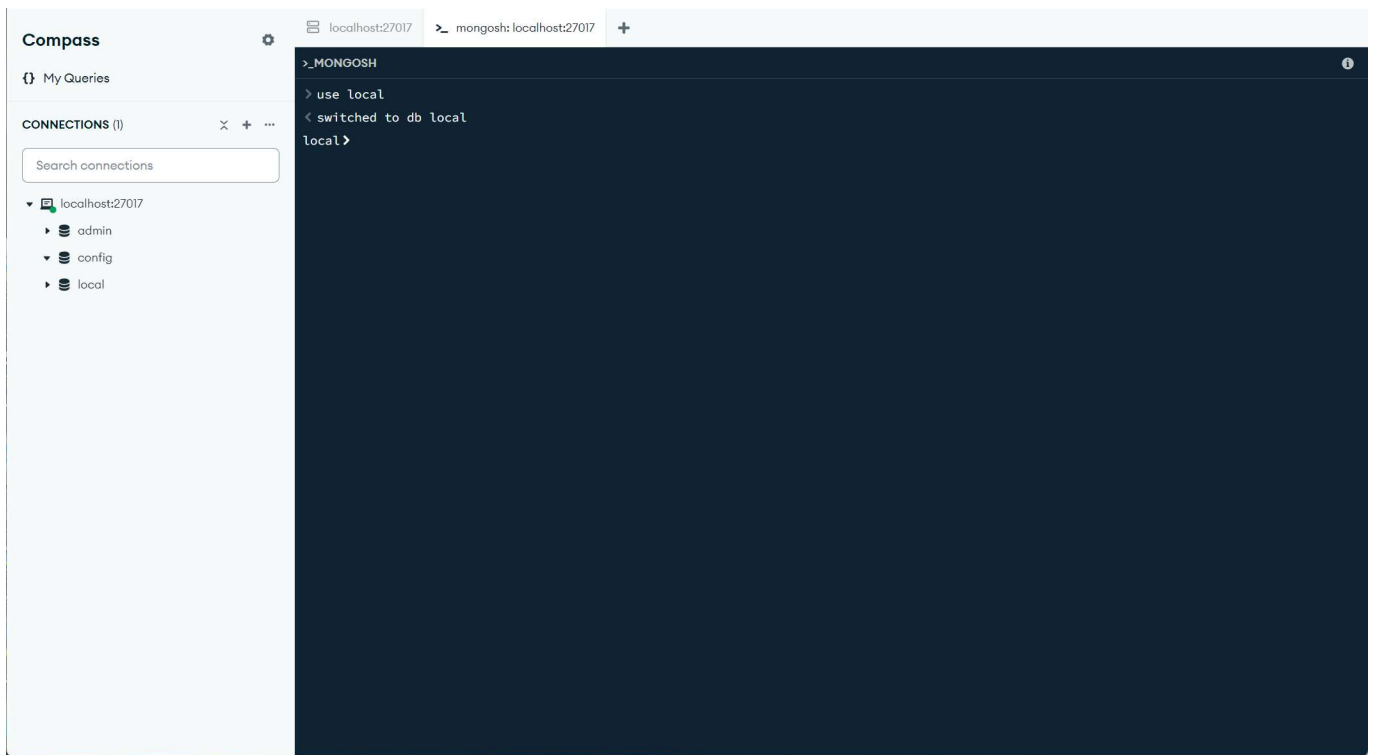
- 1.我们需要去把他的环境变量配置一下：右击此电脑=>属性=>高级系统设置=>环境变量=>找到Path双击进去，新建，内容是刚刚安装的MongoDB的Bin文件夹位置，我装的是E盘，路径为E:\MongoDB\bin
- 2.配置完环境变量，我们在C盘的根目录去创建一个data文件夹，在里面再建一个db文件夹
- 3.我们打开cmd（win+r打开运行输入cmd），输入mongod，他会跑出来一堆代码，这时候我们再去看刚刚新建的db文件夹，里面已经有东西了
- 4.我们再新建一个cmd，输入mongo（mongod用来启动服务器，mongo用来启动客户端），出现 >就差不多了

3.选择数据库

use 数据库名字

```
use local
```

和MySQL不一样的就是他可以use 不存在的库名来直接创建一个库



4.插入数据

4.1 单个插入

db.数据库名字.insertOne ({})

他会返回一个是否插入成功和插入数据的id

```
> db.game.insertOne({name:"laoyang"})
< {
  acknowledged: true,
  insertedId: ObjectId('66e6ab2f95ae50d90ce3a05d')
}
game >
```

4.2 多个插入

db.数据库名字.insertMany([{},{}])

```
> db.game.insertMany([{name:"laozhu"},{name:"man"}])
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('66e6ac3a95ae50d90ce3a05e'),
    '1': ObjectId('66e6ac3a95ae50d90ce3a05f')
  }
}
> db.game.find()
< {
  _id: ObjectId('66e6ab2f95ae50d90ce3a05d'),
  name: 'laoyang'
}
{
  _id: ObjectId('66e6ac3a95ae50d90ce3a05e'),
  name: 'laozhu'
}
{
  _id: ObjectId('66e6ac3a95ae50d90ce3a05f'),
  name: 'man'
}
game >
```

5.查找数据

5.1 普通限制

db.库名.find()

全局查找

```
> db.game.find()
< {
  _id: ObjectId('66e6ab2f95ae50d90ce3a05d'),
  name: 'laoyang'
}
game >
```

5.2 特殊限制查找

db.库名.find().sort({数据类型:1/-1,...}).limit().skip()

代表按照sort内部从左到右的顺序优先级限制输出顺序，1代表升序，-1代表降序。limit代表限制输出多少个数据。skip实现跳过功能

```
> db.game.find().sort({name:-1}).limit(2)
< {
  _id: ObjectId('66e6ac3a95ae50d90ce3a05f'),
  name: 'man'
}
{
  _id: ObjectId('66e6ac3a95ae50d90ce3a05e'),
  name: 'laozhu'
}
game >
```

5.3 自定义限制

db.库名.find({限制})

```
> db.game.find({name:"man"})
< {
  _id: ObjectId('66e6ac3a95ae50d90ce3a05f'),
  name: 'man'
}
game >
```

5.4 查询是否拥有关键字

db.库名.find({数据名:{\$exists:0/1}})

6.修改数据

遇到满足条件的第一条数据修改

db.task.update({"name":"zhangsan"},{\$set:{"name":"lisi"}})

修改所有满足条件的 db.task.updateMany({"name":"zhangsan"},{\$set:{"name":"lisi"}})

7.删除数据

7.1单数删除

db.task.remove({name:"zhangsan"})

```

> db.game.find()
< {
  _id: ObjectId('66e6ab2f95ae50d90ce3a05d'),
  name: 'laoyang'
}
{
  _id: ObjectId('66e6ac3a95ae50d90ce3a05e'),
  name: 'laozhu'
}
{
  _id: ObjectId('66e6ac3a95ae50d90ce3a05f'),
  name: 'man'
}
> db.game.remove({name:"man"})
< DeprecationWarning: Collection.remove() is deprecated. Use deleteOne, deleteMany, findOneAndDelete, or bulkWrite.
< {
  acknowledged: true,
  deletedCount: 1
}
> db.game.find()
< {
  _id: ObjectId('66e6ab2f95ae50d90ce3a05d'),
  name: 'laoyang'
}
{
  _id: ObjectId('66e6ac3a95ae50d90ce3a05e'),
  name: 'laozhu'
}
game >

```

7.2全部删除

```
db.task.remove({})
```

```
> db.game.find()
< {
  _id: ObjectId('66e6ab2f95ae50d90ce3a05d'),
  name: 'laoyang'
}
{
  _id: ObjectId('66e6ac3a95ae50d90ce3a05e'),
  name: 'laozhu'
}
> db.game.remove()
```

✖ ▶ **MongoshInvalidInputError:** [COMMON-10001] Missing required argument at

```
> db.game.remove({})
< {
  acknowledged: true,
  deletedCount: 2
}
> db.game.find()
<
game >
```

8.集合

8.1 创造集合

db.createCollection('集合名', [options])

8.2 查看集合

show collections; | show tables;

8.3 删除集合

db.集合名称.drop();