数据库

-数据库的服务器：用来保存数据

-数据库的客户端：对数据进行增删改

启动服务器后，才用客户端连接数据库

可将数据库设置为系统服务（开机自启动）自己百度

数据库备份代码：mongodump -h 127.0.0.1 -d WebSql -o Desktop

数据库备份代码：（注意数据库版本保持一致）

mongoexport -h 54.223.100.13 -u admin -p Heywind2020 --authenticationDatabase=admin --authenticationMechanism=SCRAM-SHA-1 --port 37017 -d Unibone -c Product --type=csv -f \_id,Name,Color,SailfishColor,SellerSku,SailfishProductId,Alias,Size,ProductType,SiteCode,IsPublished,IsDeleted,Price,SeName,VisibleIndividually,CreatedOnUtc,UpdatedOnUtc -o Desktop/company.csv

mongo数据库讲解

数据库 放n个数据库（database）

集合 数据库，可放n个集合（collection）

文档 集合里可放多个文档（操作的为文档）(document)

在mongodb中，数据库和集合不需要手动创建

创建文档时，文档所在的集合和数据库不存在，如果不存在会自动创建数据库和集合

mongod 动mongod服务器 （关闭cmd窗口后服务器关闭，选中后服务停止）

mongo 启动客户端

--dbpath 路径 （设置数据库文件保存路径）

--port 数字 （设置端口号）4位以上 最大不能超过65535

显示所有数据库

show dbs (查看有多少数据库)

show databases

show collections (看数据库有几个集合)

进入某个数据库

use 数据库名 （进入到某个数据库中）

db (当前的数据库)

数据库的增删改

增添数据

db.<集合的名>.insert(doc) （向集合中插入一个或多个文档）

db.<stus>.insert({name:"zhangsan",age:18,gender:"male"});

db.<stus>.insert

([

{name:"张三",age:15},

{name:"张三",age:15},

{name:"张三",age:15}

])

插入一个文档对象

db.<>.insertOne()

插入多个文档对象

db.<>.insertMany()

查询数据

db.<集合的名>.find() （查询当前集合的所有文档）

db.<stus>.find()

查询数据

//查询最新的一条数据在数据库最下方

db.username.find().sort({\_id:-1}).limit(1);

删除数据文档

db.<集合的名>.remove()　 删除符合条件的所有数据，若第二个参数为ture则只删除一个

如果不传递任何参数，则为清空集合

db.<集合>.drop() 删除整个集合

db.<集合的名>.deleteOne() 删除一条数据

db.<集合的名>.deleteMany() 删除多条数据

db.dropDatabase() 删除整个数据库

db.stus.remove({\_id:"hello"}，ture) //删除id为hello

db.stus.drop()

数据库的修改  
db.WebAnimation.update({id:"Top2"},{"$set":{"name":"XXX"}})

数据库添加列

db.getCollection('WebAnimation').update({}, {**$set**: {content:14}}, {multi: 1})  
db.getCollection('WebAnimation').update({}, {**$set**: {beizhu:14}}, {beizh: 1})

数据库添加时间列

db.Webchat.insert  
([  
{username:"张三",chat:"sfdfs",time:ISODate("2020-02-20T15:42:33.303Z")},  
{username:"张三",chat:"sfdd",time:ISODate("2020-02-20T15:43:33.303Z")},  
{username:"张三",chat:"efsad",time:ISODate("2020-02-20T15:45:33.303Z")}  
])

数据库添加时自增

db.getCollection("A表").update(   
{ id: 1 },   
{ **$inc**: { pid: 1} }  
)

数据库整数类型查询

db.username.find({'$where':'this.number>1'})

数据库时间类型查询

db.Webchat.find({"time":{"$gt":ISODate("2020-02-20T15:42:33.303Z")}})

查询字符串

db.Double.find({'$where':'this.timeString==20200315105116'})

nodejs向mongodb数据库修改数据

Double.updateMany({Doublename:linshi},{$inc:{DegreeHeat:1}},function(err,results){

if (!err){

console.log(results.result.ok);

console.log("添加成功");

}else{

console.log("添加失败");

}

});

关联查询：db.product.aggregate([

{

$lookup:

{

from: "orders",

localField: "\_id",

foreignField: "pid",

as: "inventory\_docs"

}

}

])

$lookup:

db.product.insert({"\_id":1,"productname":"商品1","price":15})

db.product.insert({"\_id":2,"productname":"商品2","price":36})

db.orders.insert({"\_id":1,"pid":1,"ordername":"订单1"})

db.orders.insert({"\_id":2,"pid":2,"ordername":"订单2"})

db.orders.insert({"\_id":3,"pid":2,"ordername":"订单3"})

db.orders.insert({"\_id":4,"pid":1,"ordername":"订单4"})

db.product.find()

db.orders.find()

下面我简单介绍一些$lookup中的参数：

from：需要关联的表【orders】

localField: 【product】表需要关联的键。

foreignField：【orders】的matching key。

as：对应的外键集合的数据，【因为可能是一对多的，对吧】

db.WebAnimation.insert

([{id:1,name:"四月是你的谎言",type:"掠心",ViewingDegree:"该影评太长",evaluate:"感人到流泪",score:"5分",ViewingAddress:"https:/www.baidu.com"},{id:2,name:"罪恶王冠",type:"掠心",ViewingDegree:"该影评太长",evaluate:"感人到流泪",score:"5分",ViewingAddress:"https:/www.baidu.com"},{id:3,name:"斩赤红之瞳",type:"掠心",ViewingDegree:"该影评太长",evaluate:"感人到流泪",score:"5分",ViewingAddress:"https:/www.baidu.com"},{id:4,name:"叛逆的努努修",type:"烧脑",ViewingDegree:"该影评太长",evaluate:"感人到流泪",score:"5分",ViewingAddress:"https:/www.baidu.com"},{id:5,name:"从零开始的异界生活",type:"剧情",ViewingDegree:"该影评太长",evaluate:"感人到流泪",score:"5分",ViewingAddress:"https:/www.baidu.com"},{id:6,name:"夏目友人账",type:"治愈",ViewingDegree:"该影评太长",evaluate:"感人到流泪",score:"5分",ViewingAddress:"https:/www.baidu.com"},{id:7,name:"野良神",type:"打斗",ViewingDegree:"该影评太长",evaluate:"感人到流泪",score:"5分",ViewingAddress:"https:/www.baidu.com"},{id:8,name:"金之炼金术士",type:"炼药",ViewingDegree:"该影评太长",evaluate:"感人到流泪",score:"5分",ViewingAddress:"https:/www.baidu.com"},{id:9,name:"进击的巨人",type:"剧情",ViewingDegree:"很大的人",evaluate:"感人到流泪",score:"5分",ViewingAddress:"https:/www.baidu.com"},{id:10,name:"少年歌行",type:"剧情",ViewingDegree:"小和尚无心",evaluate:"感人到流泪",score:"5分",ViewingAddress:"https:/www.baidu.com"},{id:11,name:"白蛇:缘起",type:"爱情",ViewingDegree:"该影评太长",evaluate:"国产良心",score:"4分",ViewingAddress:"https:/www.baidu.com"},{id:12,name:"姜子牙",type:"剧情",ViewingDegree:"该影评太长",evaluate:"国产良心",score:"5分",ViewingAddress:"https:/www.baidu.com"},{id:13,name:"秦时明月1-5",type:"剧情",ViewingDegree:"国产动漫的扛把子",evaluate:"感人到流泪",score:"5分",ViewingAddress:"https:/www.baidu.com"},{id:14,name:"秦时明月1-5",type:"剧情",ViewingDegree:"国产动漫的扛把子",evaluate:"感人到流泪",score:"5分",ViewingAddress:"https:/www.baidu.com"}])

db.username.insert([{id:1,name:"admin",pwd:"123456",deta:"2019/06/01",QQ:"364931399",phone:"13297126970"},{id:2,name:"ttg",pwd:"ttgtest",deta:"2019/06/01",QQ:"658945535",phone:"15698756354"}])

MongoDB,表与表之间的关联

db.orders.aggregate([

{

**$lookup**:

{

from: "inventory",

localField: "item",

foreignField: "sku",

as: "so"

}

},

{**$unwind**:"$so"},

{

**$lookup**: {

from: "sanbiao",

localField: "so.sku",

foreignField: "key",

as:"list"

}

}

])

例子：

var a={ "\_id" : ObjectId("51e8636953dbe31d5f34a381"), "item" : "journal", "qty" : 10, "type" : "book" } var b={ "\_id" : ObjectId("51e8636953dbe31d5f34a382"), "item" : "journal", "qty" : 10, "type" : "book" } var c={ "\_id" : ObjectId("51e8636953dbe31d5f34a383"), "item" : "journal", "qty" : 10, "type" : "book" } var d={ "\_id" : ObjectId("51e8636953dbe31d5f34a384"), "item" : "journal", "qty" : 10, "type" : "book" } db.A.insert(a) db.A.insert(b) db.A.insert(c) db.A.insert(d) var Ba={Apid:[new DBRef('A',ObjectId("51e8636953dbe31d5f34a381"))],value:3} db.B.insert(Ba) var Ba={Apid:[new DBRef('A',ObjectId("51e8636953dbe31d5f34a382"))],value:4} db.B.insert(Ba) var Ba={Apid:[new DBRef('A',ObjectId("51e8636953dbe31d5f34a383"))],value:7} db.B.insert(Ba) var Ba={Apid:[new DBRef('A',ObjectId("51e8636953dbe31d5f34a384"))],value:8} db.B.insert(Ba) db.B.find() db.A.find() db.A.findOne({"\_id":db.B.findOne().Apid[0].$id})

关联外键查询

db.A.findOne({"\_id":db.B.findOne().Apid[0].$id})

db.B.find({"$db.test":{"Apid":DBRef("A", ObjectId("5f2149b8a26250f89002ecda"))} })db.B.find().Apid.forEach(function(ref){printjson(db[ref.$ref].findOne({"\_id":ref.$id}));})

网址：

https://blog.csdn.net/weixin\_30260399/article/details/95435489?utm\_medium=distribute.pc\_relevant.none-task-blog-BlogCommendFromMachineLearnPai2-1.channel\_param&depth\_1-utm\_source=distribute.pc\_relevant.none-task-blog-BlogCommendFromMachineLearnPai2-1.channel\_param

网址：https://stackoverflow.com/questions/50530363/aggregate-lookup-with-c-sharp

.Net 环境中通过外键查询mongodb多表