Mongoose 中使用 aggregate 聚合管道

主讲教师：（大地）

合作网站：**www.itying.com**　　（IT 营）

我的专栏：https://www.itying.com/category-79-b0.html
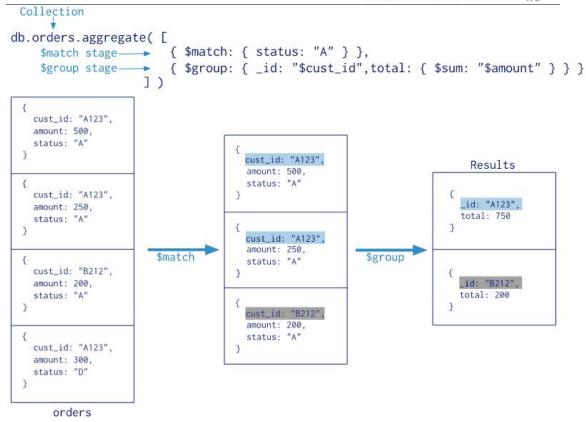
一、MongoDB 聚合管道（Aggregation Pipeline）

使用聚合管道可以对集合中的文档进行变换和组合。

实际项目：表关联查询、数据的统计。

MongoDB 中使用 db.COLLECTION_NAME.**aggregate([{<stage>},...])** 方法来构建和使用聚合管道。先看下官网给的实例，感受一下聚合管道的用法。

```
Collection
    ↓
db.orders.aggregate( [
    $match stage ——→    { $match: { status: "A" } },
    $group stage ——→    { $group: { _id: "$cust_id",total: { $sum: "$amount" } } }
                   ] )
```

二、MongoDB Aggregation 管道操作符与表达式

管道操作符	Description	
$project	增加、删除、重命名字段	
$match	条件匹配。只满足条件的文档才能进入下一阶段	
$limit	限制结果的数量	
$skip	跳过文档的数量	
$sort	条件排序。	
$group	条件组合结果 统计	
$lookup	$lookup 操作符 用以引入其它集合的数据 （表关联查询）	

SQL 和 NOSQL 对比:

WHERE	$match
GROUP BY	$group
HAVING	$match
SELECT	$project
ORDER BY	$sort
LIMIT	$limit
SUM()	$sum
COUNT()	$sum
join	$lookup

管道表达式:

管道操作符作为"键",所对应的"值"叫做管道表达式。

例如{$match:{status:"A"}}，$match 称为管道操作符，而 status:"A"称为管道表达式，是管道操作符的操作数(Operand)。

每个管道表达式是一个文档结构，它是由字段名、字段值、和一些表达式操作符组成的。

常用表达式操作符	Description
$addToSet	将文档指定字段的值去重
$max	文档指定字段的最大值
$min	文档指定字段的最小值
$sum	文档指定字段求和
$avg	文档指定字段求平均
$gt	大于给定值
$lt	小于给定值
$eq	等于给定值

三、 数据模拟

```
db.order.insert({"order_id":"1","uid":10,"trade_no":"111","all_price":100,"all_num":2})
db.order.insert({"order_id":"2","uid":7,"trade_no":"222","all_price":90,"all_num":2})
db.order.insert({"order_id":"3","uid":9,"trade_no":"333","all_price":20,"all_num":6})


db.order_item.insert({"order_id":"1","title":"商品鼠标 1","price":50,num:1})
db.order_item.insert({"order_id":"1","title":"商品键盘 2","price":50,num:1})
db.order_item.insert({"order_id":"1","title":"商品键盘 3","price":0,num:1})




db.order_item.insert({"order_id":"2","title":"牛奶","price":50,num:1})
db.order_item.insert({"order_id":"2","title":"酸奶","price":40,num:1})


db.order_item.insert({"order_id":"3","title":"矿泉水","price":2,num:5})
db.order_item.insert({"order_id":"3","title":"毛巾","price":10,num:1})
```


四、 $project

修改文档的结构，可以用来重命名、增加或删除文档中的字段。

要求查找 order 只返回文档中 trade_no 和 all_price 字段

```
db.order.aggregate([
  {
      $project:{ trade_no:1, all_price:1 }
  }
])
```

```
> db.order.aggregate([
... {
... $project:{ trade_no:1, all_price:1 }
... }
... ])
{ "_id" : ObjectId("5b743d8c2c327f8d1b360540"), "trade_no" : "111", "all_price" : 100 }
{ "_id" : ObjectId("5b743d902c327f8d1b360541"), "trade_no" : "222", "all_price" : 90 }
{ "_id" : ObjectId("5b743d962c327f8d1b360542"), "trade_no" : "333", "all_price" : 20 }
>
```

五、 $match

作用

用于过滤文档。用法类似于 find() 方法中的参数。

```
db.order.aggregate([
  {
      $project:{ trade_no:1, all_price:1 }
  },
  {
    $match:{"all_price":{$gte:90}}
  }
])
```

```
> db.order.aggregate([
... {
... $project:{ trade_no:1, all_price:1 }
... },
... {
... $match:{"all_price":{$gte:90}}
... }
... ])
{ "_id" : ObjectId("5b743d8c2c327f8d1b360540"), "trade_no" : "111", "all_price" : 100 }
{ "_id" : ObjectId("5b743d902c327f8d1b360541"), "trade_no" : "222", "all_price" : 90 }
>
```

六、 $group

将集合中的文档进行分组，可用于统计结果。

统计每个订单的订单数量，按照订单号分组

```
db.order_item.aggregate(
    [
        {
            $group: {_id: "$order_id", total: {$sum: "$num"}}
        }
    ]
)
```

```
> db.order_item.aggregate(
...     [
...         {
...             $group: {_id: "$order_id", total1: {$sum: 1}}
...         }
...     ]
... )
{ "_id" : "3", "total1" : 2 }
{ "_id" : "2", "total1" : 2 }
{ "_id" : "1", "total1" : 2 }
>
```

七、 $sort

将集合中的文档进行排序。

```
db.order.aggregate([
  {
      $project:{ trade_no:1, all_price:1 }
  },
  {
    $match:{"all_price":{$gte:90}}
  },
  {
    $sort:{"all_price":-1}
  }

])
```

```
> db.order.aggregate([
... {
... $project:{ trade_no:1, all_price:1 }
... },
... {
... $match:{"all_price":{$gte:90}}
... },
... {
... $sort:{ "all_price":1}
... }
... ])
{ "_id" : ObjectId("5b743d902c327f8d1b360541"), "trade_no" : "222", "all_price" : 90 }
{ "_id" : ObjectId("5b743d8c2c327f8d1b360540"), "trade_no" : "111", "all_price" : 100 }
>
```

```
> db.order.aggregate([
... {
... $project:{ trade_no:1, all_price:1 }
... },
... {
... $match:{"all_price":{$gte:90}}
... },
... {
... $sort:{ "all_price":-1}
... }
... ])
{ "_id" : ObjectId("5b743d8c2c327f8d1b360540"), "trade_no" : "111", "all_price" : 100 }
{ "_id" : ObjectId("5b743d902c327f8d1b360541"), "trade_no" : "222", "all_price" : 90 }
>
```

八、 $limit

```
db.order.aggregate([
  {
      $project:{ trade_no:1, all_price:1 }
  },
  {
    $match:{"all_price":{$gte:90}}
  },
  {
    $sort:{"all_price":-1}
  },
  {
    $limit:1
  }

])
```

```
> db.order.aggregate([
... {
... $project:{ trade_no:1, all_price:1 }
... },
... {
... $match:{"all_price":{$gte:90}}
... },
... {
... $sort:{"all_price":-1}
... },
... {
... $limit:1
... }
... ])
{ "_id" : ObjectId("5b743d8c2c327f8d1b360540"), "trade_no" : "111", "all_price" : 100 }
>
```

九、 $skip

```
db.order.aggregate([
  {
      $project:{ trade_no:1, all_price:1 }
  },
  {
    $match:{"all_price":{$gte:90}}
  },
  {
    $sort:{"all_price":-1}
  },
```

```
  {
    $skip:1
  }

])
```

```
> db.order.aggregate([
... {
... $project:{ trade_no:1, all_price:1 }
... },
... {
... $match:{"all_price":{$gte:90}}
... },
... {
... $sort:{"all_price":-1}
... },
... {
...   $skip:1
... }
...
... ])
{ "_id" : ObjectId("5b743d902c327f8d1b360541"), "trade_no" : "222", "all_price" : 90 }
>
```

十、 $lookup 表关联

```
db.order.aggregate([
    {
      $lookup:
        {
          from: "order_item",
          localField: "order_id",
          foreignField: "order_id",
          as: "items"
        }
    }
])
```

```
{
    "_id": ObjectId("5b743d8c2c327f8d1b360540"),
    "order_id": "1",
    "uid": 10,
    "trade_no": "111",
    "all_price": 100,
    "all_num": 2,
    "items": [{
        "_id": ObjectId("5b743d9c2c327f8d1b360543"),
        "order_id": "1",
        "title": "商品鼠标 1",
        "price": 50,
        "num": 1
    }, {
        "_id": ObjectId("5b743da12c327f8d1b360544"),
        "order_id": "1",
        "title": "商品键盘 2",
        "price": 50,
        "num": 1
    }, {
        "_id": ObjectId("5b74f457089f78dc8f0a4f3b"),
        "order_id": "1",
        "title": "商品键盘 3",
        "price": 0,
        "num": 1
    }]
} {
    "_id": ObjectId("5b743d902c327f8d1b360541"),
    "order_id": "2",
    "uid": 7,
    "trade_no": "222",
    "all_price": 90,
    "all_num": 2,
    "items": [{
        "_id": ObjectId("5b743da52c327f8d1b360545"),
        "order_id": "2",
        "title": "牛奶",
        "price": 50,
        "num": 1
    }, {
        "_id": ObjectId("5b743da92c327f8d1b360546"),
        "order_id": "2",
```

```
            "title": "酸奶",
            "price": 40,
            "num": 1
    }]
} {
    "_id": ObjectId("5b743d962c327f8d1b360542"),
    "order_id": "3",
    "uid": 9,
    "trade_no": "333",
    "all_price": 20,
    "all_num": 6,
    "items": [{
            "_id": ObjectId("5b743dad2c327f8d1b360547"),
            "order_id": "3",
            "title": "矿泉水",
            "price": 2,
            "num": 5
    }, {
            "_id": ObjectId("5b743dff2c327f8d1b360548"),
            "order_id": "3",
            "title": "毛巾",
            "price": 10,
            "num": 1
    }]
}
```

```
db.order.aggregate([
    {
        $lookup:
```

```
        {
            from: "order_item",
            localField: "order_id",
            foreignField: "order_id",
            as: "items"
        }
    },
    {
    $match:{"all_price":{$gte:90}}
    }

])



{
    "_id": ObjectId("5b743d8c2c327f8d1b360540"),
    "order_id": "1",
    "uid": 10,
    "trade_no": "111",
    "all_price": 100,
    "all_num": 2,
    "items": [{
        "_id": ObjectId("5b743d9c2c327f8d1b360543"),
        "order_id": "1",
        "title": "商品鼠标 1",
        "price": 50,
        "num": 1
    }, {
        "_id": ObjectId("5b743da12c327f8d1b360544"),
        "order_id": "1",
        "title": "商品键盘 2",
        "price": 50,
        "num": 1
    }, {
        "_id": ObjectId("5b74f457089f78dc8f0a4f3b"),
        "order_id": "1",
        "title": "商品键盘 3",
        "price": 0,
        "num": 1
    }]
```

```
}{
    "_id": ObjectId("5b743d902c327f8d1b360541"),
    "order_id": "2",
    "uid": 7,
    "trade_no": "222",
    "all_price": 90,
    "all_num": 2,
    "items": [{
        "_id": ObjectId("5b743da52c327f8d1b360545"),
        "order_id": "2",
        "title": "牛奶",
        "price": 50,
        "num": 1
    }, {
        "_id": ObjectId("5b743da92c327f8d1b360546"),
        "order_id": "2",
        "title": "酸奶",
        "price": 40,
        "num": 1
    }]
}
```

```
db.order.aggregate([
  {
      $lookup:
       {
         from: "order_item",
         localField: "order_id",
         foreignField: "order_id",
         as: "items"
       }
  },
  {
     $project:{ trade_no:1, all_price:1,items:1 }
  },
  {
    $match:{"all_price":{$gte:90}}
  },
  {
    $sort:{"all_price":-1}
```

```
  },
])


{
    "_id": ObjectId("5b743d8c2c327f8d1b360540"),
    "trade_no": "111",
    "all_price": 100,
    "items": [{
        "_id": ObjectId("5b743d9c2c327f8d1b360543"),
        "order_id": "1",
        "title": "商品鼠标 1",
        "price": 50,
        "num": 1
    }, {
        "_id": ObjectId("5b743da12c327f8d1b360544"),
        "order_id": "1",
        "title": "商品键盘 2",
        "price": 50,
        "num": 1
    }, {
        "_id": ObjectId("5b74f457089f78dc8f0a4f3b"),
        "order_id": "1",
        "title": "商 品键盘 3",
        "price": 0,
        "num": 1
    }]
} {
    "_id": ObjectId("5b743d902c327f8d1b360541"),
    "trade_no": "222",
    "all_price": 90,
    "items": [{
        "_id": ObjectId("5b743da52c327f8d1b360545"),
        "order_id": "2",
        "title": "牛奶",
        "price": 50,
        "num": 1
    }, {
        "_id": ObjectId("5b743da92c327f8d1b360546"),
        "order_id": "2",
        "title": "酸奶",
        "price": 40,
        "num": 1
```

```
    }]
}
```

```
> db.order.aggregate([
.. {
..        $lookup:
..        {
..          from: "order_item",
..          localField: "order_id",
..          foreignField: "order_id",
..          as: "items"
..        }
..     },
..     {
.. $project:{ trade_no:1, all_price:1,items:1 }
..     },
..     {
.. $match:{"all_price":{$gte:90}}
..     },
..     {
.. $sort:{"all_price":-1}
..     }
.. ])
{ "_id" : ObjectId("5b743d8c2c327f8d1b360540"), "trade_no" : "111", "all_price" : 100, "items" : [ { "_id" : ObjectId("5b743d9c2c327f8d1b
tle" : "商品鼠标1", "price" : 50, "num" : 1 }, { "_id" : ObjectId("5b743da12c327f8d1b360544"), "order_id" : "1", "title" : "商品键盘2",
{ "_id" : ObjectId("5b743d902c327f8d1b360541"), "trade_no" : "222", "all_price" : 90, "items" : [ { "_id" : ObjectId("5b743da52c327f8d1b3
e" : "牛奶", "price" : 50, "num" : 1 }, { "_id" : ObjectId("5b743da92c327f8d1b360546"), "order_id" : "2", "title" : "酸奶", "price" : 40
```

十一、 Mongoose aggregate 多表关联查询

```javascript
var mongoose=require('./db.js');
var OrderSchema=mongoose.Schema({
    order_id:String,
    uid:Number,
    trade_no:String,
    all_price:Number,
    all_num:Number
})
var OrderModel=mongoose.model('Order',OrderSchema,'order');
OrderModel.aggregate([
    {
        $lookup:{
            from:'order_item',
            localField:"order_id",
            foreignField:"order_id",
            as:"item"
        }
    }
],function(err,docs){
    console.log(docs)
});
```