



链滴

Mongoose 聚合查询

作者: [JoeysCat](#)

原文链接: <https://ld246.com/article/1576166614334>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Author 模型

```
const mongoose = require('mongoose')
const moment = require('moment')

const Schema = mongoose.Schema

const AuthorSchema = new Schema(
  {
    first_name: { type: String, required: true, max: 100 },
    family_name: { type: String, required: true, max: 100 },
    date_of_birth: { type: Date },
    date_of_death: { type: Date }
  },
  {
    toJSON: { virtuals: true }
  }
)

module.exports = mongoose.model('Author', AuthorSchema)
```

数据集

```
{
  "RECORDS": [
    {
      "_id": "5deb0af03e15fb05116733f9",
      "first_name": "Isaac",
      "family_name": "Asimov"
    },
    {
      "_id": "5deb0af03e15fb05116733f7",
      "first_name": "迅",
      "family_name": "鲁"
    },
    {
      "_id": "5deb0af03e15fb05116733f8",
      "first_name": "言",
      "family_name": "莫"
    },
    {
      "_id": "5deb0af03e15fb05116733f6",
      "first_name": "笑笑生",
      "family_name": "兰陵"
    },
    {
      "_id": "5deb0af03e15fb05116733f5",
      "first_name": "迁",
      "family_name": "司马"
    },
    {
      "_id": "5df25414c9d219055b41f833",
```

```

    "first_name": "金",
    "family_name": "巴"
  },
  {
    "_id": "5df25439c9d219055b41f834",
    "first_name": "立波",
    "family_name": "周"
  },
  {
    "_id": "5df25449c9d219055b41f835",
    "first_name": "蒙",
    "family_name": "王"
  },
  {
    "_id": "5df2544fc9d219055b41f836",
    "first_name": "朔",
    "family_name": "王"
  },
  {
    "_id": "5df2546ac9d219055b41f837",
    "first_name": "士",
    "family_name": "王"
  }
]
}

```

需求

根据作者姓名进行查询，即 family_name+first_name

mongodb 聚合查询命令

```

db.authors.aggregate([
  {
    $project: {
      name: {
        $concat: ['$family_name', '$first_name']
      }
    }
  },
  {
    $match: {
      $or: [{
        name: {
          $regex: new RegExp('王', 'i')
        }
      }]
    }
  },
  {
    $skip: 0
  },
  {

```

```
    $limit: 10
  }
])
```

输出

```
[
  {
    "_id": ObjectId("5df25449c9d219055b41f835"),
    "name": "王蒙"
  },
  {
    "_id": ObjectId("5df2544fc9d219055b41f836"),
    "name": "王朔"
  },
  {
    "_id": ObjectId("5df2546ac9d219055b41f837"),
    "name": "王士"
  }
]
```

mongoose 查询方式

```
// query = { name: '王' }
exports.listName = query => {
  return new Promise((resolve, reject) => {
    let { name, page, limit } = query
    console.log(query)
    page = page ? page : 1
    limit = limit ? limit : 100
    let match = {}
    let like = []
    if (name) {
      like.push({ name: { $regex: new RegExp(name, 'i') } })
      match.$or = like
    }

    Author.aggregate([
      { $project: { name: { $concat: ['$family_name', '$first_name'] } } },
      { $match: match },
      { $skip: +(page - 1) * limit },
      { $limit: +limit }
    ]).exec((err, result) => {
      if (err) {
        return reject(err)
      }
      resolve({ items: result })
    })
  })
}
```

输出

```
{
  items: [
    { _id: 5df25449c9d219055b41f835, name: '王蒙' },
    { _id: 5df2544fc9d219055b41f836, name: '王朔' },
    { _id: 5df2546ac9d219055b41f837, name: '王士' }
  ]
}
```

参考

[选择记录匹配 MongoDB 中两个字段的 concat 值](#)

[Aggregation](#)