

المواد ذات الصلة

البرنامج التعليمي الأساسي Ogre اثنين

MongoDB أساسيات البرنامج التعليمي - الجزء 2 عمليات MongoDB الأساسية (1)

عملية بيونجو البرنامج التعليمي الأساسي MongoDB

سلسلة البرنامج التعليمي الأساسية MongoDB - خط الأنايب التجميعي
البرنامج التعليمي الأساسي MongoDB

"البرنامج التعليمي البسيط MongoDB" الفصل 2 أوامر MongoDB الأساسية (Shell)

البرنامج التعليمي الأساسي لـ HTML (اثنان)

برنامج تعليمي kail (اثنان) - الإعدادات الأساسية

برنامج سحابي صغير تطوير البرنامج التعليمي الأساسي القتالي اثنين

تنصيب البرنامج التعليمي MongoDB

تنصيب البرنامج التعليمي mongodb

البرنامج التعليمي MongoDB الأساسي (اثنان)

يتضمن: mongodb

واحد ، عامل التشغيل

< gt — greater than\$

=< gte — gt equal\$

> lt — less than\$

=> lte — lt equal\$

=! ne — not equal\$

= eq — equal\$

على سبيل المثال:

```
1 db.col.find({likes : {$lte : 150}})
```

ابحث عن مستندات أقل من أو تساوي 150 في العمود
بيان sql مماثل:

```
1 Select * from col where likes <= 150;
```

أحصل على البيانات التي تحتوي على عدد "الإعجابات" أكبر من 100 ولكن أقل من 200 في مجموعة "العمود".

```
1 db.col.find({likes : {$lt : 200, $gt : 100}})
```

مماثل لـ SQL:

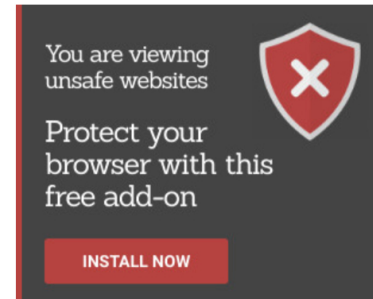
```
1 Select * from col where likes>100 AND likes<200;
```

الثاني ، نوع المشغل \$

يعتمد عامل التشغيل \$ type على نوع BSON لاسترداد نوع البيانات المطابقة في المجموعة وإرجاع النتيجة.

الأنواع التي يمكن استخدامها في MongoDB موضحة في الجدول التالي:

التي	التي	التي
Double	1	
String	2	
Object	3	
Array	4	
Binary data	5	
Undefined	6	已废弃。
Object id	7	
Boolean	8	
Date	9	
Null	10	
Regular Expression	11	
JavaScript	13	



Symbol	14	
JavaScript (with scope)	15	
32-bit integer	16	
Timestamp	17	
64-bit integer	18	
Min key	255	Query with -1.
Max key	127	

على سبيل المثال ، للحصول على البيانات بعنوان String في مجموعة "col":

```
1 db.col.find({"title" : {$type : 2}})
```

3. طريقة الحد والتخطي

1. تقبل طريقة limit () معلمة رقمية تحدد عدد السجلات المقروءة من MongoDB.

```
1 db.COLLECTION_NAME.find().limit(NUMBER)
```

إذا لم يتم تحديد المعلمة في طريقة limit () ، فسيتم عرض جميع البيانات في المجموعة.

2. تستخدم طريقة skip () لتخطي كمية محددة من البيانات ، كما تقبل طريقة التخطي معلمة رقمية باعتبارها عدد السجلات المطلوب تخطيها.

```
1 db.COLLECTION_NAME.find().limit(NUMBER).skip(NUMBER)
```

المعلمة الافتراضية للطريقة skip () هي 0

قراءة 100 سجل من 10 سجلات ، وهو ما يعادل الحد الأقصى (10100) في sql.

```
db.COLLECTION_NAME.find().skip(10).limit(100)
```

أساليب التخطي والحد مناسبان فقط للترحيل بكمية صغيرة من البيانات. إذا كانت الملايين من المستويات ، ستكون الكفاءة منخفضة للغاية ، لأن أسلوب التخطي هو عدد أجزاء البيانات في الماضي ، فمن المستحسن استخدام where_limit. بمعنى آخر ، إذا كان بإمكانك إرفاق مثل هذه الشروط ، فسيتم تحسين كفاءة الاستعلام

```
db.test.find({amount:{$gt:2399927}}).sort({"amount":1}).limit(10) //53ms
```

عند الاستعلام ، استخدم الفرز والتخطي والحد في نفس الوقت. بغض النظر عن المركز ، يتم تنفيذ الأمر أولاً ، ثم الفرز ، والتخطي ، ثم الحد.

الرابع ، نوع MongoDB

تقوم طريقة الفرز () بفرز البيانات ، ويمكن لطريقة الفرز () تحديد حقل الفرز حسب المعلمات ، واستخدام 1 و -1 لتحديد طريقة الفرز ، حيث 1 للترتيب التصاعدي ، و -1 للترتيب التنازلي.

```
db.COLLECTION_NAME.find().sort({KEY:1})
```

الدعامة الأولى: حيث الشرط ؛

القوس الثاني 1: not display ، 0: display ؛

فرز بترتيب تنازلي ؛

```
> db.col.find(0, {"title":1,_id:0}).sort({"likes":-1})
{"title": "PHP 教程"}
{"title": "PHP 教程"}
{"title": "Java 教程"}
{"title": "Java 教程"}
{"title": "MongoDB 教程"}
{"title": "MongoDB 教程"}
>
```

عندما يتم تنفيذ skip () و limit () و sort () معاً ، يكون ترتيب التنفيذ هو sort () ، ثم تخطي () ، وأخيراً الحد المعروض ().

خمسة ، فهرس

يمكن للفهارس بشكل عام تحسين كفاءة الاستعلام بشكل كبير. إذا لم يكن هناك فهرس ، يجب على MongoDB مسح كل ملف في المجموعة وتحديد تلك السجلات التي تلبي شروط الاستعلام عند قراءة البيانات.

هذا النوع من كفاءة الاستعلام لمسح المجموعة الكاملة منخفض جدًا ، خاصة عند معالجة كمية كبيرة من البيانات ، يمكن أن يستغرق الاستعلام عشرات الثواني أو حتى دقائق ، وهو أمر قاتل جدًا لأداء موقع الويب.

الفهرس عبارة عن بنية بيانات خاصة ، ويتم تخزين الفهرس في مجموعة بيانات يسهل اجتيازها وقراءتها. والفهرس عبارة عن هيكل يفرز قيم عمود واحد أو أكثر في جدول قاعدة البيانات.

```
db.COLLECTION_NAME.ensureIndex({KEY:1})
في طريقة ()
```

عن طريق إضافة الخلفية: الخيار الصحيح عند إنشاء الفهرس ، يتم تنفيذ عمل الإنشاء في الخلفية

```
1 db.values.ensureIndex({open: 1, close: 1}, {background: true})
```

ستة ، التجميع

يستخدم التجميع بشكل أساسي لمعالجة البيانات (مثل المتوسط الإحصائي والتجميع وما إلى ذلك) وإرجاع نتيجة البيانات المحسوبة. يشبه إلى حد ما count (*) في عبارة SQL.

```
db.COLLECTION_NAME.aggregate(AGGREGATE_OPERATION)
```

احسب عدد المقالات التي كتبها كل مؤلف ، واستخدم التجميع () لحساب النتيجة على النحو التالي:

```
1 db.mycol.aggregate([{$group : {_id : "$by_user",num_tutorial : {$sum : 1}}}]])
```

تعبير إجمالي:

表达式	描述	实例
\$sum	计算总和。	db.mycol.aggregate([{\$group : {_id : "\$by_user", num_tutorial : {\$sum : "\$likes"}}}])
\$avg	计算平均值	db.mycol.aggregate([{\$group : {_id : "\$by_user", num_tutorial : {\$avg : "\$likes"}}}])
\$min	获取集合中所有文档对应值得最小值。	db.mycol.aggregate([{\$group : {_id : "\$by_user", num_tutorial : {\$min : "\$likes"}}}])
\$max	获取集合中所有文档对应值得最大值。	db.mycol.aggregate([{\$group : {_id : "\$by_user", num_tutorial : {\$max : "\$likes"}}}])
\$push	在结果文档中插入值到一个数组中。	db.mycol.aggregate([{\$group : {_id : "\$by_user", url : {\$push : "\$url"}}}])
\$addToSet	在结果文档中插入值到一个数组中，但不创建副本。	db.mycol.aggregate([{\$group : {_id : "\$by_user", url : {\$addToSet : "\$url"}}}])
\$first	根据资源文档的排序获取第一个文档数据。	db.mycol.aggregate([{\$group : {_id : "\$by_user", first_url : {\$first : "\$url"}}}])
\$last	根据资源文档的排序获取最后一个文档数据	db.mycol.aggregate([{\$group : {_id : "\$by_user", last_url : {\$last : "\$url"}}}])

خط أنابيب

ينقل خط تجميع MongoDB مستندات MongoDB إلى خط الأنابيب التالي للمعالجة بعد المعالجة في خط أنابيب واحد. يمكن تكرار عمليات خطوط الأنابيب.

التعبير: معالجة مستند الإدخال والإخراج. التعبيرات عديمة الحالة ويمكن استخدامها فقط لحساب المستندات في خط أنابيب التجميع الحالي ، ولا يمكن معالجة المستندات الأخرى.

نقدم هنا العديد من العمليات شائعة الاستخدام في إطار التجميع:

- مشروع \$: تعنيل هيكل مستند الإدخال. يمكن استخدامها لإعادة تسمية الحقول أو إضافتها أو حذفها ، كما يمكن استخدامها لإنشاء نتائج الحساب والمستندات المتداخلة.

match : استعمال البثريمنفور قبالنسبة الى , فقطتخسر خارجرمز اجمعمقالة - سلعاقطعنصالملفات | يستخد match عمليات الاستعلام المعيارية لـ MongoDB.

- حد \$: يُستخدم للحد من عدد المستندات التي يتم إرجاعها بواسطة خط تجميع MongoDB.
- تخطي \$: تخطي العدد المحدد من المستندات في مسار التجميع وأعد المستندات المتبقية.
- فك \$: قسّم حقل نوع مصفوفة في المستند إلى أجزاء متعددة ، كل منها يحتوي على قيمة واحدة في المصفوفة.
- مجموعة \$: تجميع المستندات في المجموعة ، والتي يمكن استخدامها للنتائج الإحصائية.
- فرز \$: قم بفرز مستندات الإدخال وإخراجها.
- geoNear \$: إخراج المستندات المطلوبة بالقرب من موقع جغرافي معين.

مثيل المشروع \$

```
1 db.article.aggregate(  
2   { $project : {  
3     title : 1 ,  
4     author : 1 ,  
5   } }  
6 );
```



Surfing questionable
sites? Protect
yourself



Play now and get
your warship for
!free



World of Warships:
!Play For Free



World of Warships:
Naval Warfare