تحلیل ها و سیستم های داده های حجیم	دانشگاه خواجه نصیرالدین طوسی
فایل راهنمای پروژه اول: کار با MongoDB	مهر ماه1403

به نام خدا

mongo dbچیست؟

وقتی با بانک اطلاعاتی مانند دیتابیس mongo dB کار می کنیم، حجم اطلاعات بالا به معنای چیزی بیش از صدها هزار یا میلیون‌ها خط داده یا دیتا است. یا اگر در مقیاس حجم حافظه حساب کنیم، چندین گیگابایت تا چندین ترابایت از اطلاعات را شامل می‌شود. بدیهی است که وقتی حجم داده، بیش از ظرفیت حافظه رم سیستم است، بایستی به دنبال استراتژی دیگری نسبت به SQL برای ذخیره و پردازش داده بود.

از عمده دلایلی که استفاده از پایگاه داده غیر SQL را در اولویت قرار می‌دهد، حجم داده است. علاوه بر حجم، سرعت تغییرات ساختار داده نیز تاثیر بسیار مهمی دارد . ممکن است تغییر ساختار اطلاعاتی در برنامه‌ها، روزی چند مرتبه اتفاق بیافتد. در چنین شرایطی، ویرایش و مدیریت بانک‌های داده بر پایه SQL بسیار دشوار و زمان‌بر است. اما معماری و ابزاری چون پایگاه داده mongo dB این شرایط را بسیار سهل‌تر و سریع‌تر می‌کند.

پایگاه داده mongo dB این داده‌ها را با فرمت BSON ذخیره می‌کند. این فرمت، نوشتن اطلاعات به همان روش مرسوم و محبوب JSON است که به صورت BINARY کدگذاری و ذخیره شده است. فرمت JSON این برتری را دارد که برای تبدیل دوباره به ساختارهای برنامه بسیار منعطف است. داده‌ای که به صورت JSON ذخیره می‌شود، در آینده به راحتی ویرایش می‌شود. به این معنی که هم ساختار می‌تواند تغییر کند و هم حجم داده‌ها. مزیت بکارگیریBSON ، انعطاف پذیری بالای آن است.

در بانک اطلاعاتSQL ، تغییر ساختار یا حجم اطلاعات نیاز به بازنویسی کل جدول توسط برنامه مدیریت بانک اطلاعات دارد و دستور این کار یک عملیات جدا محسوب و این کار توسط ادمین یا برنامه نویس انجام می‌شود. به بیان دیگر، در بانک اطلاعات SQL این تغییر در طی روال ذخیره اطلاعات امکان پذیر نیست و در صورت وجود تفاوت در حجم یا ساختار داده، فرآیند ذخیره سازی با خطا مواجه می‌شود.

در پایگاه دادهmongo dB ، بر خلاف SQL که داده در جدول‌ها ذخیره می‌شود، می‌توان مدل‌های داده‌ای مانند key-value pairs و یا فرمت‌های گراف در آن ذخیره کرد. سرعت دسترسی به داده در مدل key-value بسیار بالا است.

در mongo dB مفهومی به نام collection وجود دارد که تمامdocument ها به آن نسبت داده می‌شود. همچنین، هر document می‌تواند از اندازه و ساختار کاملا متفاوت تبعیت کند. بخشی از یک document می‌تواند یک فایل تصویری باشد و بخش دیگری از آن، اطلاعات مربوط به تصویر. Collection معادل جدول و document معادل رکورد اطلاعاتی یعنی row در جدول است.

فهرست‌بندی (indexing) تاثیر بسیاری در افزایش سرعت پیدا کردن و خواندن اطلاعات بسیار مهم دارد. بنابراین، در پایگاه داده mongo dB می‌توان درخواست‌های پرتکرار را فهرست بندی کرد.

دیتابیس مونگو دی بی از قابلیت تجمیع (aggregation) برخوردار است. بدین معنی که می‌توان مجموعه‌ای اطلاعات را قالب یک پاسخ دریافت کرد. اطلاعات تجمیع شده چیزی مشابه یک گزارش یا آمار هستند. داده‌هایی که مقدار هر فیلد آن بر اساس پردازش خاصی تهیه شده است. این کار به طور معمول در منطق یک برنامه، توسط چرخش‌های متعدد در اطلاعات صورت می‌گیرید. زمانی که حجم این اطلاعات بالا باشد، مدیریت رم یا بهینه‌سازی این عملیات می‌تواند بسیار پیچیده و پر زحمت بشود.

از مزیت‌های بزرگ mongo dB این است که تجمیع داده را به طور بهینه به لحاظ تکرار، پردازش کرده و مطمئن به لحاظ مدیریت منابع، انجام می‌دهد.

زمانی که حجم یک collection بسیار بالا باشد، ثبت و جستجو در آن به مشکل می‌خورد. برای حل این شرایط مشکل‌ساز، mongo dB ابزار مفیدی به نام Sharding دارد که یکcollection را به قطعات کوچک‌تر تبدیل می‌کند. هر یک از این قطعه‌ها یک Shard گفته می‌شود. یک Shardمانند یک نمونه کامل و مستقل mongo dB عمل می‌کند و می‌تواند در سیستم‌های متعدد توزیع شود و Replicate شود. هر یک از shardها که در ماشین مجزا توسط mongo dB عمل می‌کنند که به آنها cluster گفته می‌شود.

پایگاه داده mongo dB برای چه کسانی مناسب است؟

- ✓ حجم اطلاعات بالا است.
- ✓ نیاز به سیستم بلادرنگ است. انجام پرس و جو با سرعت بالا.
- ✓ امکان تثبیت ساختار در پروژه وجود نداشته باشد (ساختار منعطف).
- ✓ نیاز به سیستم توزیع‌پذیر باشد.

حال که ما با مونگودی‌بی آشنا شدیم نیاز است که آن را با پایتون ادغام نماییم. برای چنین کاری نیاز است که سراغ یک پکیج استاندارد که توسط خود توسعه‌دهندگان مونگودی‌بی نوشته شده است برویم. این پکیج PyMongo نام دارد.

گام اول: نصب MongoDB

برای نصب و دانلود MongoDB می توانید به آدرس https://www.mongodb.com بروید.

گام دوم : پایتون برای دسترسی به پایگاه داده MongoDB نیاز به یک درایور MongoDB دارد. ما از درایور PyMongo استفاده خواهیم کرد.

برای نصب پای‌مونگو مانند هر پکیج دیگری از طریق pip وارد عمل شوید:

```
C:\Users\Your Name\AppData\Local\Programs\Python\Python36-32\Scripts>python -m pip install pymongo
```

برای فهمیدن اینکه آیا نصب با موفقیت انجام شده یا نه و یا اینکه آیا از قبل نصب شده است، دستور زیر را در یک فایل python اجرا کنید.

```
import pymongo
```

اگر کد بالا بدون خطا اجرا شود، به معنی این است که شما PyMongo را با موفقیت نصب کرده اید.

گام سوم: ساخت یک ارتباط

برای ساخت پایگاه داده یا دیتابیس در MongoDB، ابتدا یک شی MongoClient ایجاد می کنیم. سپس URL کانکشن و آدرس IP معتبر و نام پایگاه داده را مشخص می کنیم. MongoDB پایگاه داده را می سازد (اگر وجود نداشته باشد) و با آن ارتباط برقرار می کند. در مثال زیر یک پایگاه داده به نام mydatabase ایجاد کرده ایم:

```python
import pymongo

myclient = pymongo.MongoClient("mongodb://localhost:27017/")

mydb = myclient["mydatabase"]
```

در MongoDB تا زمانی که یک پایگاه داده محتوا نداشته باشد، ایجاد نشده است.

ایجاد collection و ایجاد document.

برای بررسی وجود داشتن یک دیتابیس می توانید آن را در بین تمام دیتابیس های موجود در سیستم جستجو کنید. مثال زیر لیست دیتابیس ها را نشان می دهد:

```python
print(myclient.list_database_names())
```

خروجی سیستم:

```
C:\Users\My Name>python demo_mongodb_check_db.py
['admin', 'local', 'mydatabase']
```

یک مجموعه یا Collection در MongoDB معادل جدول یا table در پایگاه های داده SQL است. برای ساخت کالکشن در MongoDB از شی پایگاه داده و نام کالکشن استفاده می کنیم. در مثال زیر یک کالکشن به نام customers ایجاد کرده ایم:

```python
import pymongo

myclient = pymongo.MongoClient("mongodb://localhost:27017/")
mydb = myclient["mydatabase"]

mycol = mydb["customers"]
```

توجه داشته باشید که تا زمانی که کالکشن Collection در MoongoDB محتوا نداشته باشد، ساخته نشده است. یک Document در MongoDB معادل یک رکورد در پایگاه های داده SQL است. برای درج یک Document در کالکشن، از تابع insert_one() استفاده می کنیم. اولین پارامتر این تابع یک دیکشنری حاوی مقادیر و کلید ها برای هر فیلد Document است. در مثال زیر یک document را در کالکشن customers درج کرده ایم:

```python
import pymongo

myclient = pymongo.MongoClient("mongodb://localhost:27017/")
mydb = myclient["mydatabase"]
mycol = mydb["customers"]

mydict = { "name": "John", "address": "Highway 37" }

x = mycol.insert_one(mydict)
```

تابع insert_one() یک شی InsertOneResult را بر می گرداند که یک خاصیت به نام inserted_id دارد. این خاصیت یا Property آی دی سند درج شده را در خود نگه می دارد.

```python
mydict = { "name": "Peter", "address": "Lowstreet 27" }

x = mycol.insert_one(mydict)

print(x.inserted_id)
```

خروجی

```
C:\Users\My Name>python demo_mongodb_insert_id.py
5b1910482ddb101b7042fcd7
```

اگر شما هیچ id فیلدی تعریف نکنید، MongoDB یک id منحصر به فرد برای هر فیلد اضافه می کند. در مثال بالا هیچ id تعریف نکردیم و MongoDB یک id منحصر به فرد برای document ایجاد کرده است.

مرجع: https://pvlearn.com/product/%D٪۸B٪۴D٪۸B٪۱D٪۸۸٪۹D٪۸B۹-%DA%A٪۹D٪۸A٪۷D٪۸B۱-%D٪۸A٪۸D٪۸A۷-mongodb/

دستورات aggregation در pymongo

https://www.analyticsvidhya.com/blog/۰۸/۲۰۲۰/how-to-create-aggregation-pipelines-in-a-mongodb-database-using-pymongo/

برخی دستورات در اینجا اورده شده.

http://jessezhuang.github.io/article/mongodb-aggregation-framework/

ساخت dump

https://stackoverflow.com/questions/۴۹۱۵۳۰۲۰/how-to-dump-a-collection-to-json-file-using-pymongo/۴۹۱۵۳۳۹۳

اندازه گیری زمان اجرا در پایتون

https://stackoverflow.com/questions/٧٣٧٠٨٠١/how-to-measure-elapsed-time-in-python