

به نام خدا



دانشگاه صنعتی امیر کبیر
(پلی تکنیک تهران)

دستور کار آزمایشگاه پایگاه داده

جلسه هشتم

آشنایی با پایگاه داده‌های

NoSQL

استاد درس

دکتر شهریاری

مقدمه

پایگاه داده‌های NoSQL (مخفف عبارت Not only SQL یا Non-SQL) به اصطلاح یک پایگاه داده‌ی غیر رابطه‌ای (non relational) است که امکانی برای ذخیره و بازیابی اطلاعات به شکلی متفاوت از پایگاه داده‌های رابطه‌ای ایجاد می‌کند. در پایگاه داده‌های رابطه‌ای مرسوم از جبر رابطه‌ای استفاده می‌شود و در آن با جدول سر و کار داریم اما در پایگاه داده‌های غیر رابطه‌ای از داده‌هایی به شکل JSON و XML استفاده می‌کنیم که در بسیاری از مسائل متمرکز بر داده (data intensive) که سرعت و کارایی بالایی را نیاز دارد، عملکرد مناسبی دارد.

انواع NoSQL:

- پایگاه داده‌های NoSQL کلید-مقدار (Key-value NoSQL):

در اینگونه پایگاه داده‌ها از یک کلید (Key) به عنوان شناسه‌ی هر داده برای ذخیره و بازیابی مقدار داده (Value) استفاده می‌شود. این دسته به علت سادگی کارکرد مرسوم‌ترین نوع پایگاه داده‌های NoSQL می‌باشد.

- پایگاه داده‌های NoSQL مبتنی بر اسناد (Document NoSQL):

این نوع از پایگاه داده‌ها به منظور ذخیره‌سازی و کار با اسنادی با فرمت های ... , JSON , XML به کار می‌روند. از پایگاه داده‌های اسنادی NoSQL جهت ذخیره‌سازی داده های بدون ساختار مشخص با پراکندگی بالا استفاده می‌شود.

- پایگاه داده‌های NoSQL چند ستونه (Wide-column NoSQL):

پایگاه داده‌های چند ستونه در نگاه اول همانند پایگاه داده‌های SQL از جدول و ستون و سطر استفاده می‌کنند. اما عملکرد آن‌ها متفاوت است. برخلاف SQL هر ستون می‌تواند شامل داده‌هایی با فرمت و ساختار متفاوت باشد. به عبارتی دیگر نوع تعریف و فرمت یک ستون می‌تواند در هر سطر متفاوت باشد. این پایگاه داده‌ها انعطاف بسیار بالایی در ثبت و کار با داده‌های بسیار عظیم و متفاوت دارند.

- پایگاه داده‌های مبتنی بر گراف (Graph NoSQL):

پایگاه داده‌های گرافی به منظور ذخیره سازی حجم زیادی از داده‌های رابطه‌ای (Relational data) طراحی شده‌اند. به زبان ساده می‌توان اینگونه پایگاه داده‌ها را مانند گرافی شامل رئوسی به عنوان داده‌ها و یال‌هایی به

عنوان رابطه‌ها در یک گراف در نظر گرفت. از این دسته پایگاه داده‌ها در ذخیره‌سازی انواع معماری‌های داده‌های شبکه‌ای نیز استفاده می‌شود.

- پایگاه داده‌های NoSQL چند مدل (MultiModel NoSQL):

پایگاه داده‌های چند مدل می‌توانند تلفیقی از انواع دیگر پایگاه داده‌های NoSQL باشند. (مانند Key-value و مبتنی بر گراف)

معرفی برخی از پایگاه داده‌های NoSQL

- Aerospike:

یک پایگاه داده Key-value بسیار قدرتمند برای کلاسترینگ است که از لحاظ حجم پردازش داده و سرعت بالاترین رتبه را در Benchmark به دست آورده است. ذخیره‌سازی اطلاعات در این پایگاه داده به صورت ادغامی از Ram و SSD صورت می‌گیرد.

- Redis:

یک پایگاه داده Key-value که برای حجم کار کوچک‌تر به نسبت aerospike مناسب می‌باشد. تمامی اطلاعات این پایگاه داده در Ram ذخیره می‌شوند. این پایگاه داده قابلیت کلاستر شدن ندارد.

- MongoDB:

مونگو معروف‌ترین پایگاه داده NoSQL است و برای ذخیره‌سازی اسناد (Documents) استفاده می‌شود. اگر نیاز به ذخیره‌سازی حجم بالایی از داده‌های مختلف و پراکنده را دارید مونگو گزینه مناسبی برای شماست. کار با مونگو به نسبت ساده است و برای شروع کار و آشنایی با NoSQL ها گزینه‌ی مناسبی است.

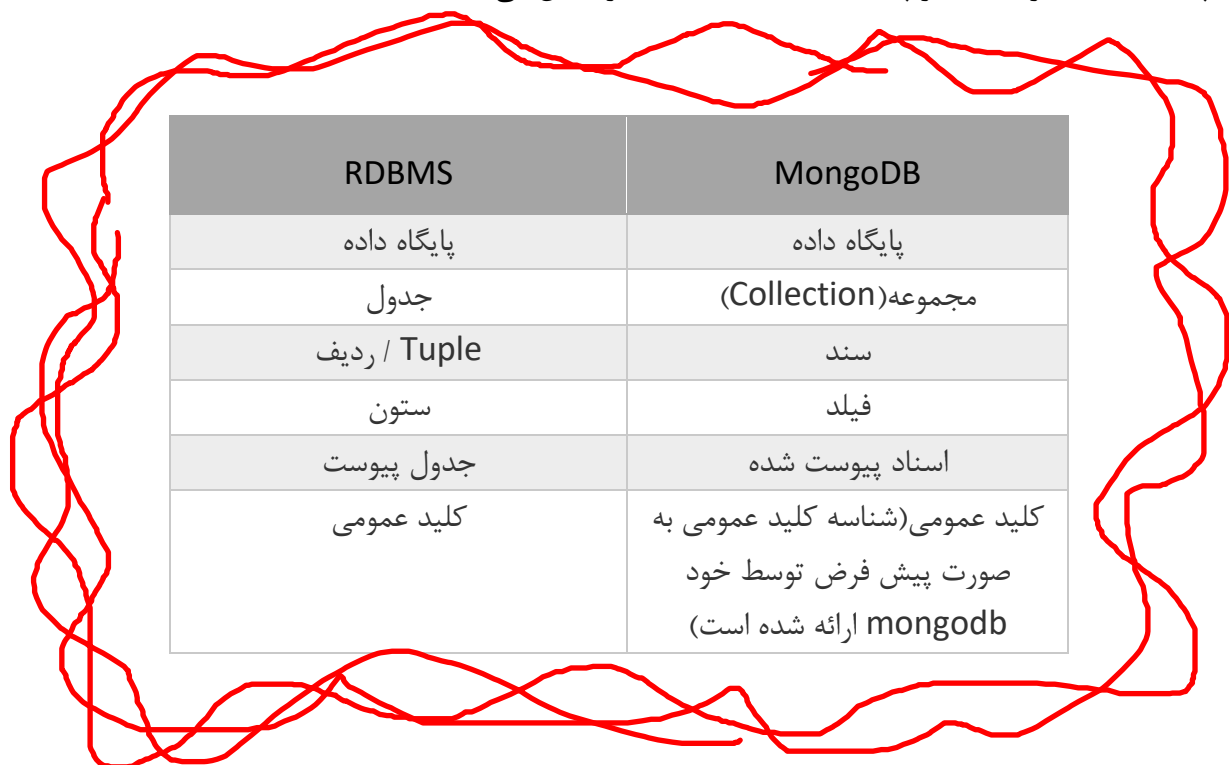
- Cassandra:

این پایگاه داده از قوی‌ترین پایگاه داده‌های NoSQL می‌باشد و توسط Apache ارائه شده است. پایگاه داده Cassandra در دسته Wide-column قرار دارد. اگر چه سرعت پردازش Aerospike از Cassandra بیشتر است اما قابلیت‌های فراوان و انعطاف بالای Cassandra به عقیده بسیاری آن را قدرتمندترین NoSQL کرده است.

• Neo4j:

پایگاه داده‌ی بر پایه گراف (Graph) که برای ذخیره سازی ساختمان داده‌های مختلف شبکه‌ای و داده‌های رابطه‌ای بسیار مناسب است.

در این کلاس ما به بررسی و استفاده از پایگاه داده‌ی MongoDB می‌پردازیم. جدول زیر جایگاه اصطلاحات پایگاه داده‌های رابطه‌ای در پایگاه داده‌ی MongoDB را نشان می‌دهد.



RDBMS	MongoDB
پایگاه داده	پایگاه داده
جدول	مجموعه (Collection)
Tuple / ردیف	سند
ستون	فیلد
جدول پیوست	اسناد پیوست شده
کلید عمومی	کلید عمومی (شناسه کلید عمومی به صورت پیش فرض توسط خود mongodb ارائه شده است)

نصب MongoDB

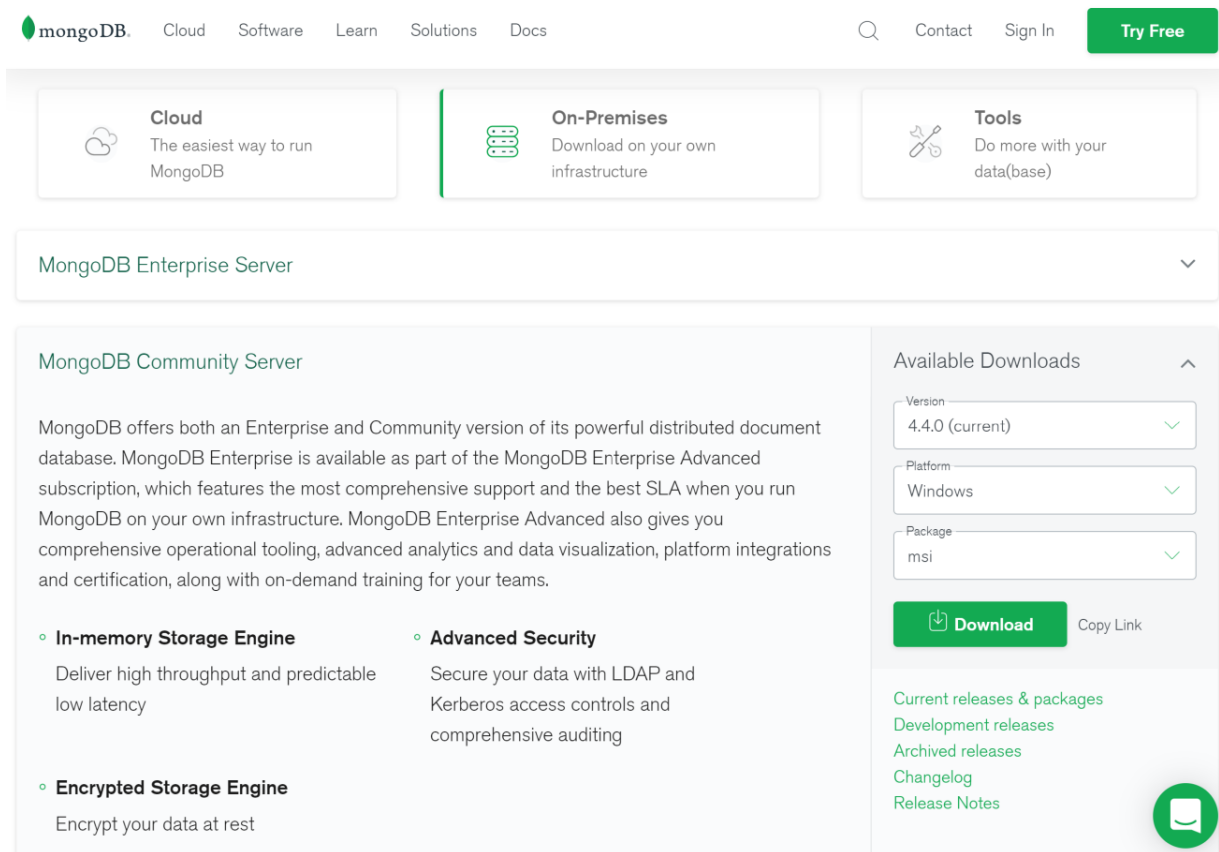
برای نصب MongoDB بر روی سیستم عامل‌های مختلف قسمتی در سایت رسمی آن وجود دارد که می‌توانید از آن استفاده کنید، اما در این نوشتار به آموزش نصب دو سیستم عامل ویندوز و اوبونتو به اختصار می‌پردازیم.

نصب روی ویندوز

از طریق لینک زیر وارد صفحه‌ی دانلود شوید:

<https://www.mongodb.com/try/download/community>

وارد صفحه‌ای مانند شکل ۱ می‌شوید. نسخه و سیستم‌عامل مورد نظر خود را انتخاب کرده و دانلود را آغاز کنید.



The screenshot shows the MongoDB download page. At the top, there are navigation links for Cloud, Software, Learn, Solutions, and Docs. A search bar and links for Contact, Sign In, and a Try Free button are also present. Below the navigation bar, there are three main sections: Cloud (The easiest way to run MongoDB), On-Premises (Download on your own infrastructure), and Tools (Do more with your data(base)). The On-Premises section is selected, and the MongoDB Enterprise Server is chosen. The MongoDB Community Server section is expanded, showing details about the database and its features. The Available Downloads section shows the current version (4.4.0) and the platform (Windows). The download button is highlighted.

شکل ۱

روند نصب را طی کنید (پیشنهاد می‌شود مطابق تنظیمات و روند پیش‌فرض نصب پیش بروید). محل نصب فایل‌های اجرایی نسخه‌ی ۴,۲ آن به صورت پیش‌فرض آدرس زیر است:

C:\Program Files\MongoDB\Server\4.2\bin\

برای اجرای سرویس MongoDB از طریق cmd، هربار نیاز به وارد کردن این آدرس داریم. به منظور تسهیل این مسئله ابتدا بر روی نشانگر Windows کلیک کرده و سپس عبارت Edit the system environment variables را تایپ می‌کنیم. بر روی عبارت یافت شده، کلیک می‌کنیم تا پنجره System Properties ظاهر

شود. سپس وارد تب Advanced شده و بر روی دکمه‌ی Environment Variables کلیک می‌کنیم تا پنجره مربوط به آن باز شود. در قسمت System variables بر روی ردیف Path دابل کلیک کرده تا پنجره Edit environment variable ظاهر شود. در پنجره‌ی ظاهر شده، بر روی دکمه New کلیک می‌کنیم تا یک سطر جدید در پایین لیست مسیرها ایجاد شود. در سطر خالی ایجاد شده، مسیر ریز را وارد کرده و سپس دکمه Enter را می‌زنیم.

```
C:\Program Files\MongoDB\Server\4.2\bin\
```

حال تمامی پنجره‌های باز را از طریق کلیک کردن بر روی دکمه OK آنها، می‌بندیم.

توجه: اگر در حین عملیات بالا پنجره‌ی PowerShell یا cmd باز باشد، نیاز است که آن را بسته و مجدد باز کنید.

با انجام مراحل گفته شده از این پس شما با اجرای دستور زیر امکان دسترسی به shell سرویس MongoDB را خواهید داشت:

```
mongo
```

نصب بر روی Ubuntu

برای نصب نسخه‌ی ۴,۴ در ابتدا در ترمینال دستورات زیر را به ترتیب اجرا کنید:

```
sudo apt-get install gnupg  
wget -qO - https://www.mongodb.org/static/pgp/server-4.4.asc | sudo apt-key  
add -
```

در ادامه لیست فایل `/etc/apt/sources.list.d/mongodb-org-4.4.list` را با توجه به نسخه‌ی اوبونتو خود با دستور زیر ایجاد کنید:

Ubuntu 20.04

```
echo "deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu focal/mongodb-org/4.4 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-4.4.list
```

Ubuntu 18.04

```
echo "deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu bionic/mongodb-org/4.4 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-4.4.list
```

Ubuntu 16.04

```
echo "deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu xenial/mongodb-org/4.4 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-4.4.list
```

سپس دستور زیر را برای به روزرسانی local package database اجرا کنید:

```
sudo apt-get update
```

حال برای نصب آخرین نسخه‌ی MongoDB دستور زیر را اجرا کنید:

```
sudo apt-get install -y mongodb-org
```

حال برای دسترسی به shell سرویس MongoDB می‌توانید از دستور زیر استفاده کنید:

```
mongo
```

نحوه‌ی مدل کردن داده در MongoDB

این پایگاه داده همان‌طور که پیش‌تر نیز مطرح شد، یک ساختار ذخیره‌سازی سندگرا است که از قالب BSON (مخفف JSON باینری) برای نمایش سندهای خود بهره می‌برد. قالب BSON از سوی دیگر، همان قالب JSON است که به‌صورت باینری کدگذاری شده و علاوه بر تمامی توانمندی‌ها و کارکردهای قالب JSON، ویژگی‌های منحصر به فرد دیگری مانند انواع داده‌ای بیش‌تر، از جمله Date و داده باینری (BinData) که جزء مشخصات JSON نبوده اما برای کارکرد MongoDB از اهمیت بالایی برخوردار هستند را در خود دارد. ذکر این نکته ضروری است که BinData در واقع معادل نوع داده BLOB در پایگاه‌های داده رابطه‌ای است که برای ذخیره‌سازی محتوای فایل‌ها درون پایگاه داده‌ها مورد استفاده قرار می‌گیرد.

هر سند در پایگاه داده MongoDB مجموعه‌ای از جفت‌های فیلد-مقدار است که مقدارها می‌تواند یکی از انواع زیر باشد:

- یک نوع BSON
- یک سند
- آرایه‌ای از هر نوع BSON
- آرایه‌ای از سند

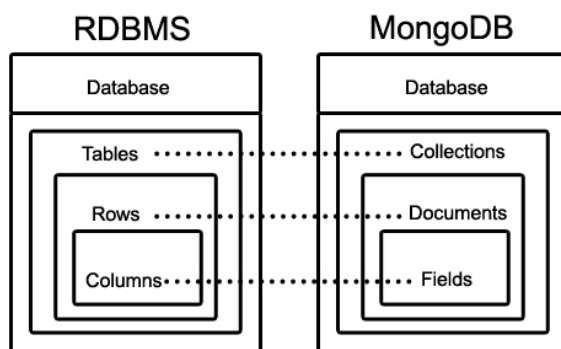
بنابراین، در این پایگاه داده این امکان وجود دارد که یک سند شامل سندهای دیگری باشد. این سندهای درونی بسیار قدرتمند بوده و در عین حال، انعطاف فراوانی را برای شیماهای آن با خود به‌همراه دارند، که البته باید با دقت فراوانی مورد استفاده قرار گیرند؛ چرا که حداکثر اندازه یک سند به‌طور معمول شانزده مگابایت است.

نکته‌ای که باید در اینجا به آن اشاره کرد، آن است که پایگاه داده MongoDB با مجموعه‌ها یا Collections کار می‌کند؛ که معادل جدول‌ها در پایگاه‌های داده رابطه‌ای بوده و سند، معادلی برای رکوردهای جدول‌ها در این رده از پایگاه است. به عبارت دیگر هر پایگاه داده MongoDB، متشکل از مجموعه‌هایی است که هر مجموعه خود شامل سندهایی — رکوردها — هستند که همان‌طور که در بخش قبل به آن اشاره شد، هر یک از این سندها می‌توانند خود مجموعه‌ای از سندها را در خود جای دهند.

از آن‌جا که هر سند متشکل از یک جفت فیلد اطلاعاتی و مقدار متناظر با آن است، نام مناسب برای شناسایی ویژگی هر سند به‌طور مشخص نام فیلد است که درواقع معادل نام ستون است، که در پایگاه‌های داده رابطه‌ای

مورد استفاده قرار می‌گیرد. همچنین هر سند، شامل یک کلید اصلی است که به صورت خودکار به فیلد id تخصیص داده می‌شود.

نکته‌ی حائز اهمیت در مورد مجموعه‌ها در پایگاه داده غیر رابطه‌ای MongoDB آن است که اگرچه مجموعه‌ها، به نوعی با مفهوم جدول‌ها هم ارز هستند که سندها را در خود جای می‌دهند، اما واقعیت آن است که این مجموعه‌ها هیچ ساختاری را بر سندها تحمیل نمی‌کنند و این بزرگ‌ترین تفاوت این رده از پایگاه‌های داده با انواع رابطه‌ای آن است. به عبارت دیگر، برخلاف جدول‌های پایگاه‌های رابطه‌ای که هر رکورد آن، ستون‌های یکسانی دارند، سندهای موجود در یک مجموعه می‌توانند ساختارهای – فیلدهای اطلاعاتی – کاملاً متفاوتی داشته باشند. اگرچه در کاربردهای معمول، سندهای موجود در یک مجموعه، معمولاً ساختار پایه‌ای مشترکی دارند.



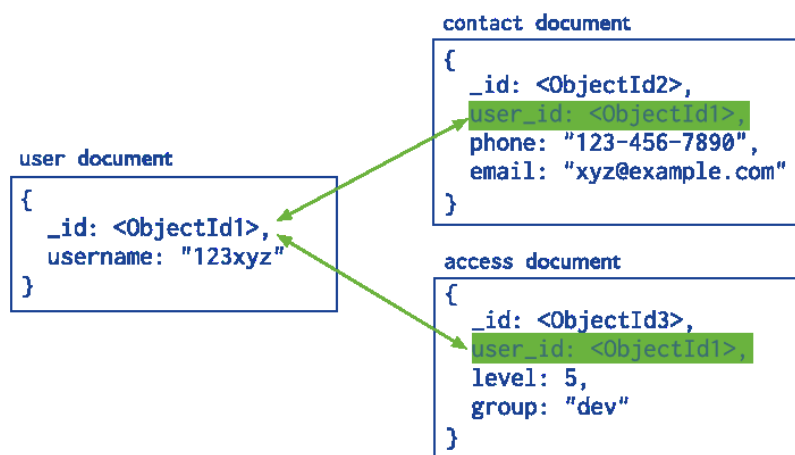
بنابراین، به طور خلاصه می‌توان ساختار پایگاه داده‌ای MongoDB را متشکل از مجموعه‌هایی دانست که هر مجموعه شامل سندهایی است و هر سند، معرف یک زوج فیلد اطلاعاتی و مقدار فیلد است که مقدار می‌تواند یکی از مقادیر مطرح شده در بخش‌های قبل را در خود جای داده و ساختار سندگرای تو در تو را فراهم آورد. هر سند با نام فیلد آن شناخته شده و کلید اصلی آن به صورت خودکار تخصیص داده می‌شود.

در ساختار MongoDB نحوه‌ی برقراری ارتباط میان دو سند با توجه به کاربرد مورد نظر به دو صورت قابل اجرا است:

- مدل داده‌ی نرمالسازی شده (Normalized Data Model)
- مدل داده‌ی جاسازی شده (Embedded Data Model)

مدل داده‌ی نرمالسازی شده (Normalized) یا ارجاعی

در این شیوه ارتباطات میان سندها از طریق ایجاد یک اتصال میان سندها برقرار می‌شود. در واقع با افزودن یک شناسه‌ی یک سند درون سند دیگر یک اشاره‌گر به سند بالادستی ایجاد می‌شود.



مدل داده‌ی جاسازی شده (Embedded)

در این شیوه سندها به صورت تو در تو ایجاد می‌شوند. در واقع سندی که قرار است به سند دیگری متصل شود، مستقیماً به عنوان یک جزء در سند دیگر قرار داده می‌شود.

