







**LỜI CẢM ƠN**

Em xin gửi lời cảm ơn chân thành và sâu sắc đến cô Phan Thị Phương Nam, người đã tận tình hướng dẫn và hỗ trợ em trong suốt quá trình thực hiện đề tài "Thiết kế và cài đặt hệ thống thi trắc nghiệm trực tuyến tiếng Anh kỹ năng Reading".

Nhờ vào sự chỉ bảo tận tình, những đóng góp quý báu và sự kiên nhẫn của cô, em đã có cơ hội học hỏi, rèn luyện và hoàn thiện các kỹ năng chuyên môn liên quan đến việc thiết kế và cài đặt hệ thống cơ sở dữ liệu. Cô không chỉ giúp em định hướng rõ ràng về nội dung nghiên cứu mà còn truyền cảm hứng để tôi hoàn thành đề tài một cách tốt nhất.

Em biết ơn những lời khuyên và sự hỗ trợ kịp thời của cô trong quá trình thực hiện đồ án cơ sở ngành. Đó là động lực to lớn giúp em vượt qua thử thách và đạt được những kết quả tốt trong đề tài này.

Trong quá trình nghiên cứu đề tài, do kiến thức còn hạn chế nên em có nhiều thiếu sót trong quá trình tìm hiểu, đánh giá và trình bày đề tài. Em rất mong nhận được những đóng góp ý kiến của thầy/cô để khắc phục và phát triển từng bước hoàn thiện hơn.

Em xin chân thành cảm ơn.**MỤC LỤC**

[DANH MỤC HÌNH ẢNH 5](#_Toc4276)

[DANH MỤC BẢNG BIỂU 6](#_Toc31390)

[TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH 7](#_Toc31020)

[MỞ ĐẦU 8](#_Toc12816)

[CHƯƠNG 1: TỔNG QUAN 10](#_Toc28011)

[1.1. Tổng quan về vấn đề 10](#_Toc15921)

[1.2. Mục tiêu nghiên cứu 10](#_Toc25739)

[1.3. Nội dung nghiên cứu 10](#_Toc1000)

[1.4. Phạm vi nghiên cứu 10](#_Toc6464)

[CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT 11](#_Toc16092)

[2.1. Tổng quan về cơ sở dữ liệu NoSQL 11](#_Toc7506)

[2.1.1. Cơ sở dữ liệu NoSQL là gì 11](#_Toc21380)

[2.1.2. Đặc tính cơ sở dữ liệu NoSQL 11](#_Toc15839)

[2.1.3. Các dạng cơ sở dữ liệu NoSQL 11](#_Toc4839)

[2.1.4. Một số thuật ngữ tương đương trong Relational Database và NoSQL database 13](#_Toc26289)

[2.2. Tổng quan về MongoDB 13](#_Toc1311)

[2.2.1. Các khái niệm cơ bản trong MongoDB 14](#_Toc25892)

[2.2.2. Các kiểu dữ liệu trong MongoDB 15](#_Toc14612)

[2.2.3. Các thao tác dữ liệu trên MongoDB 16](#_Toc12153)

[2.2.4. Làm việc với Documents 20](#_Toc19223)

[CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU 26](#_Toc15243)

[3.1. Thiết kế cơ sở dữ liệu 26](#_Toc8507)

[3.1.1. Tìm hiểu 26](#_Toc2098)

[3.1.2. Thiết kế cấu trúc lược đồ cơ sở dữ liệu 26](#_Toc5032)

[3.1.3. Lập bảng phân cấp thuộc tính 30](#_Toc22248)

[3.2. Cài đặt môi trường làm việc 35](#_Toc10290)

[CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU 38](#_Toc16232)

[4.1. Chọn dữ liệu mẫu thử 38](#_Toc1018)

[4.2. Nhập dữ liệu mẫu thử 40](#_Toc21848)

[4.3. Đặt và chạy các câu truy vấn 43](#_Toc2799)

[CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 55](#_Toc6213)

[5.1 Kết luận 55](#_Toc9310)

[5.1.1 Kết quả đạt được 55](#_Toc26743)

[5.1.2 Hạn chế 55](#_Toc5361)

[5.2 Hướng phát triển 55](#_Toc20325)

[DANH MỤC TÀI LIỆU THAM KHẢO 56](#_Toc8660)

[PHỤ LỤC 57](#_Toc8403)

**DANH MỤC HÌNH ẢNH**

*[Hình](#_Toc8405)* [1](#_Toc8405) *[. Cài đặt MongoDB Community Server](#_Toc8405)* [36](#_Toc8405)

*[Hình](#_Toc24522)* [2](#_Toc24522) *[. Cài đặt MongoDB Shell](#_Toc24522)* [37](#_Toc24522)

*[Hình](#_Toc29340)* [3](#_Toc29340) *[. Setup cài đặt MongoDB](#_Toc29340)* [37](#_Toc29340)

*[Hình](#_Toc30308)* [4](#_Toc30308) *[. Giao diện MongoDB Compass](#_Toc30308)* [38](#_Toc30308)

*[Hình](#_Toc18035)* [5](#_Toc18035) *[. Minh họa đề 1](#_Toc18035)* [39](#_Toc18035)

*[Hình](#_Toc14771)* [6](#_Toc14771) *[. Minh họa đề 2](#_Toc14771)* [40](#_Toc14771)

*[Hình](#_Toc10270)* [7](#_Toc10270) *[. Minh họa đề 3](#_Toc10270)* [41](#_Toc10270)

**DANH MỤC BẢNG BIỂU**

*[Bảng](#_Toc25061)* [1](#_Toc25061) *[. Thuật ngữ Relational Database và NoSQL databse](#_Toc25061)* [13](#_Toc25061)

*[Bảng](#_Toc3489)* [2](#_Toc3489) *[. Mối quan hệ giữa các thuật ngữ của RDBMS và MongoDB](#_Toc3489)* [15](#_Toc3489)

[Bảng 3 . Miêu tả tham số 17](#_Toc19133)

[Bảng 4 . Danh sách các options 18](#_Toc1199)

[Bảng 5 . Các phép truy vấn 23](#_Toc10832)

*[Bảng](#_Toc9882)* [6](#_Toc9882) *[. Phân cấp thuộc tính](#_Toc9882)* [35](#_Toc9882)

**TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH**

**Vấn đề nghiên cứu:**

* Cấu trúc cơ bản của một hệ thống thi tiếng Anh Reading trực tuyến.
* Nội dung và thể thức của một kỳ thi chuẩn VSTEP.
* Khó khăn trong việc quản lý dữ liệu các thí sinh.

**Hướng tiếp cận:**

* Sử dụng cơ sở dữ liệu NoSQL (MongoDB) để thiết kế và cài đặt dữ liệu cho hệ thống thi Reading tiếng Anh trực tuyến.

**Cách giải quyết vấn đề:**

* Phân tích dữ liệu thực tế từ trang web thi VSTEP và các tài liệu liên quan.
* Sử dụng MongoDB với cấu trúc dữ liệu linh hoạt để lưu trữ thông tin thí sinh, đề thi và kết quả thi.

**Một số kết quả đạt được:**

* Xây dựng được hệ thống cơ sở dữ liệu có khả năng mở rộng, tối ưu cho việc quản lý thông tin thí sinh.
* Hiểu rõ hơn về đề tài nghiên cứu.

**MỞ ĐẦU**

**Lý do chọn đề tài**

Hiện nay, nhiều trường đại học và tổ chức giáo dục đã triển khai các kỳ thi VSTEP trực tuyến và cung cấp nền tảng luyện thi đáng tin cậy. Trường đại học Trà Vinh là một trong nhiều cơ sở để tổ chức thi trực tuyến VSTEP nhằm hỗ trợ thí sinh luyện thi hiệu quả với các bộ đề tiêu chuẩn. Việc phát triển hệ thống cơ sở dữ liệu thi trắc nghiệm tiếng Anh kỹ năng Reading trực tuyến giúp:

**Quản lý hiệu quả:**

* Tối ưu hóa nguồn lực: Các cơ sở có thể quản lý thông tin dễ dàng, tối ưu hóa việc vận hành và giảm tải lên các địa điểm đăng ký trực tiếp.
* Nâng cao chất lượng dịch vụ: Hệ thống giúp trang web truy vấn nhanh thông tin và kết quả của thí sinh, đề thi hay cải thiện chất lượng của đề thi qua từng kỳ.

**Dữ liệu chính xác đầy đủ:**

* Hỗ trợ công tác phân tích và phát triển: Các cơ sở có thể sử dụng dữ liệu từ hệ thống để phân tích năng lực và xu hướng lựa chọn đáp án của các thí sinh trong từng kỳ để đánh giá năng lực theo vùng hoặc cải thiện chất lượng đào tạo.

**Đáp ứng nhu cầu của xã hội:**

* Phù hợp với xu hướng số hóa: Hệ thống thi trắc nghiệm tiếng Anh Reading trực tuyến góp một phần trong quá trình chuyển đổi số của ngành giáo dục.
* Đáp ứng nhu cầu của người dùng: Ngày càng nhiều người có nhu cầu đăng ký và thi bằng chứng chỉ trực tuyến.

**Mục tiêu nghiên cứu:**

Vận dụng quy trình thu thập thông tin và lý thuyết về hình thức thi trực tuyến kỹ năng tiếng Anh Reading qua văn bản, trang web và các hoạt động tham gia để xây dựng cơ sở dữ liệu thực tế.

**Đối tượng nghiên cứu:**

Xây dựng cơ sở dữ liệu về hình thức thi trực tuyến kỹ năng Reading bằng NoSQL.

**CHƯƠNG 1: TỔNG QUAN**

* 1. **Tổng quan về vấn đề**

Hình thức thi trực tuyến ngày càng trở nên phổ biến, đặc biệt là với sự hỗ trợ của công nghệ và giáo dục, thì việc học và thi online trở nên phổ biến. Trong đó, hình thức thi ngoại ngữ được sử dụng phổ biến và rộng rãi hơn bởi tính thuận tiện của nó. Trong thi ngoại ngữ thì kỹ năng Reading là một kỹ năng quan trọng trong việc đánh giá khả năng ngôn ngữ của thí sinh. Do đó việc xây dựng một cơ sở dữ liệu tạo tiền đề cho việc xây dựng một hệ thống quản lý việc tham gia và cách thức thi trực tuyến tiếng Anh là cần thiết.

* 1. **Mục tiêu nghiên cứu**
* Vận dụng quy trình thu thập các thông tin về hình thức thi trực tuyến kỹ năng Reading qua các văn bản hướng dẫn, trang web, thể lệ tham gia, và các hoạt động tham gia để xây dựng cơ sở dữ liệu.
* Vận dụng lý thuyết xây dụng cơ sở dữ liệu vào việc xây dựng các cơ sở dữ liệu thực tế.
  1. **Nội dung nghiên cứu**
* Xây dựng cơ sở dữ liệu về hình thức thi trực tuyến kỹ năng Reading theo hướng cơ sở dữ liệu NoSQL.
* Cài đặt cơ sở dữ liệu, nhập mẫu thủ và truy vấn dữ liệu theo hướng được lưu trữ trong hệ quản trị cơ sở dữ liệu MongoDB.
* Viết câu truy vấn minh họa các giai đoạn thi trắc nghiệm tiếng Anh kỹ năng Reading.
  1. **Phạm vi nghiên cứu**
* Nội dung: nghiên cứu các hình thức thi trực tuyến kỹ năng Reading của sinh viên tại Trường Đại học Trà Vinh.
* Không gian: Trường Đại học Trà Vinh
* Thời gian: Học kỳ I năm học 2024 - 2025

# CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

**2.1. Tổng quan về cơ sở dữ liệu NoSQL**

**2.1.1. Cơ sở dữ liệu NoSQL là gì**

Thuật ngữ NoSQL được sử dụng lần đầu tiên vào năm 1998 cho một cơ sở dữ liệu không có giao diện SQL. NoSQL phát triển mạnh suốt những năm 2000 trong sự mở rộng nhanh chống của Internet. Cơ sở dữ liệu NoSQL là một hệ thống quản lý dữ liệu phi quan hệ (Non-relational Data Management System) có lược đồ (schema) linh hoạt, dễ mở rộng. Mục đích chính việc sử dụng cơ sở dữ liệu NoSQL là cho các kho dữ liệu lớn phân tán với nhu cầu lưu trữ dữ liệu lớn.

**2.1.2. Đặc tính cơ sở dữ liệu NoSQL**

Cơ sở dữ liệu NoSQL có các đặc tính sau:

* Mô hình cơ sở dữ liệu phi quan hệ.
* Tập trung cho khả năng phân tán và mở rộng thuộc tính theo chiều rộng.
* Quy tắc cho lược đồ thì yếu hoặc không có.
* Sao chép dữ liệu dễ dàng.
* Dễ dàng truy cập được cung cấp thông qua một API.
* Mô hình nhất quán không phải là ACID.

Cơ sở dữ liệu quan hệ RDBMS (Relational Database Management System) sử dụng cú pháp SQL để lưu trữ và truy xuất dữ liệu. Đối với hệ thống cơ sở dữ liệu NoSQL, bao gồm một loạt các công nghệ cơ sở dữ liệu có thể lưu trữ dữ liệu có cấu trúc, bán cấu trúc, phi cấu trúc và đa hình.

**2.1.3. Các dạng cơ sở dữ liệu NoSQL**

Có nhiều hệ thống cơ sở dữ liệu NoSQL như: MongoDB, RavenDB, Redis, Neo4j,… Xong có thể chia NoSQL thành 4 loại:

* Key-Value stores;
* Document stores;
* Column-Family Databases;
* Graph Databases.

**1. Key-Value stores**

Cơ sở dữ liệu Key-Value Stores khi có các đặc tính như sau:

* Có một tập các đối tượng dữ liệu nhận dạng, các khóa;
* Đối với mỗi khóa, có chính xác một đối tượng dữ liệu mô tả có liên quan, giá trị cho khóa đó;
* Chỉnh định một khóa cho phép truy vấn giá trị liên quan trong cơ sở dữ liệu.

Để truy vấn dữ liệu trong CSDL, dựa vào key để lấy dữ liệu ra. Các CSDL dạng này có tốc độ truy vấn rất nhanh.

**Database tiêu biểu**: Riak, Redis, MemCache, Project Voldemort, CouchBase.

**Ứng dụng**: Do tốc độ truy xuất nhanh, key-value stores thường được dùng để làm cache cho ứng dụng (Tiêu biểu là Redis và MemCache). Ngoài ra,còn được dùng để lưu thông tin trong sessions, profiles/preferences của user…

**2. Document stores**

Mỗi đối tượng sẽ được lưu trữ trong CSDL dưới dạng một document. Dữ liệu sẽ được lưu trữ dưới dạng BSON/JSON/XML dưới database. Dữ liệu không có lược đồ cứng như SQL. Do đó có thể thêm/sửa các thuộc tính, thay đổi table, … rất nhanh và đơn giản. Cơ sở dữ liệu dạng này có tốc độ truy vấn nhanh, có thể thực hiện các câu truy vấn phức tạp, dễ mở rộng (scalability). Mỗi cơ sở dữ liệu có một kiểu truy vấn riêng.

**Database tiêu biểu**: MongoDB, RavenDB, CouchDB, TerraStone, OrientDB.

**Ứng dụng**: Do nhanh và linh động, document database thường đóng vài trò làm database cho các ứng dụng prototype, big data, e-commerce, CMS. Ngoài ra, ta còn dùng để lưu log hoặc history.

**3. Column-Family Database**

Giới thiệu: Dữ liệu được lưu trong CSDL dưới dạng các cột, thay vì các hàng như SQL. Mỗi hàng sẽ có một key/id riêng. Điểm đặt biệt là các hàng trong một bảng sẽ có số lượng cột khác nhau. Câu lệnh truy vấn của nó khá giống SQL.

**Database tiêu biểu**: Cassandra (Phát triển bởi Facebook), HyperTable, Apache Hbase.

**Ứng dụng**: Column-Family Database được sử dụng khi cần ghi một số lượng lớn dữ liệu, big data. Còn được ứng dụng trong một số CMS và ứng dụng e-commerce.

**4. Graph Database**

Dữ liệu trong graph database được lưu dưới dạng các node. Mỗi node sẽ có một label, một số properties như một row trong SQL. Các node này được kết nối với nhau bằng các relationship. Graph database tập trung nhiều vào relationship giữa các node, áp dụng nhiều thuật toán duyệt node để tăng tốc độ.

**Database tiêu biểu**: Neo4j, InfiniteGraph, OrientDB, HYPERGRAPHDB.

**Ứng dụng**: Khi cần truy vấn các mối quan hệ, graph database truy vấn nhanh và dễ hơn nhiều so với database. Được dùng trong các hệ thống: mạng nơ ron, chuyển tiền bạc, mạng xã hội (tìm bạn bè), giới thiệu sản phẩm (dựa theo sở thích/lịch sử mua sắm của người dùng)…

**2.1.4. Một số thuật ngữ tương đương trong Relational Database và NoSQL database**

*Bảng 1**. Thuật ngữ Relational Database và NoSQL databse*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **RELATIONAL** | **KEY-VALUE (RIAK)** | **DOCUMENT**  **(MONGODB)** | **COLUMN-FAMILY**  **(CASSANDRA)** | **GRAPH (NEO4J)** |
| instance | cluster | mongod | cluster | instance |
| table | bucket | collection | column-family | label |
| row | key-value | document | row | node |
| row-id | key | \_id |  |  |
| schema |  | database |  | schema |

Bảng này chỉ có tính tương đối. Các node trong Graph DB không có key, mà có properties của nó sẽ đóng vai trò thay thế key.

**2.2. Tổng quan về MongoDB**

MongoDB là một cơ sở dữ liệu đa nền tảng, hoạt động trên các khái niệm collection và document, nó cung cấp hiệu suất cao, tính khả dụng cao và khả năng mở rộng dễ dàng.

Các lợi thế của MongoDB so với RDBMS:

* Ít Schema hơn: MongoDB là một cơ sở dữ liệu dựa trên document, trong đó một collection giữ các document khác nhau. Số trường, nội dung và kích cỡ của document này có thể khác với document khác.
* Cấu trúc của một đối tượng là rõ ràng.
* Không có các Join phức tạp.
* Khả năng truy vấn sâu hơn. MongoDB hỗ trợ các truy vấn động trên các document bởi sử dụng một ngôn ngữ truy vấn dựa trên document mà mạnh mẽ như SQL.
* MongoDB là dễ dàng để mở rộng.
* Việc chuyển đổi/ánh xạ của các đối tượng ứng dụng đến các đối tượng cơ sở dữ liệu là không cần thiết.
* Sử dụng bộ nhớ nội tại để lưu giữ phần công việc, giúp truy cập dữ liệu nhanh hơn.

**2.2.1. Các khái niệm cơ bản trong MongoDB**

**Database**

Database là một nơi chứa vật lý cho các collection. Mỗi database lấy tập hợp các file riêng của nó trên hệ thống file. Mỗi MongoDB SERVER có thể có nhiều cơ sở dữ liệu.

**Collection**

Collection là một nhóm các document trong MongoDB. Collection tương đương như một table trong RDBMS – Ralational Database Managament System. Do đó, một collection tồn tại bên trong một cơ sở dữ liệu duy nhất. Các collection không có ràng buộc (Relationship) như các hệ quản trị cơ sở dữ liệu quan hệ nên việc truy xuất rất nhanh. Do đó mỗi collection có thể chứa nhiều thể loại khác nhau, không giống như table trong hệ quản trị cơ sở dữ liệu quan hệ có các field cố định. Các document bên trong một collection có thể có nhiều trường khác nhau. Đặc biệt, tất cả các document trong một collection là tương tự nhau hoặc với cùng mục đích liên quan.

**Document**

Một document trong MongoDB, có cấu trúc tương tự như kiểu dữ liệu JSON, là một tập hợp các cặp key-value. Các document có lược đồ động, nghĩa là document trong cùng một collection không cần thiết phải có cùng một tập hợp các 27 trường hoặc cấu trúc giống nhau, và các trường chung trong document của một collection có thể giữ các kiểu dữ liệu khác nhau.

Mối quan hệ thuật ngữ giữa các thuật ngữ của hệ quản trị cơ sở dữ liệu quan hệ và MongoDB:

*Bảng 2**. Mối quan hệ giữa các thuật ngữ của RDBMS và MongoDB*

|  |  |
| --- | --- |
| **RDBMS** | **MongoDB** |
| Database | Database |
| Table | Collection |
| Tuple/Row | Document |
| Column | Field |
| Table Join | Embedded Documents |
| Primary Key | Primary Key (Giá trị mặc định \_id được cung cấp bởi chính MongoDB) |

\_id là một số thập lục phân để đảm bảo tính duy nhất của mỗi Document. Người dùng có thể cung cấp \_id trong khi chèn vào Document. Nếu người dùng không cung cấp \_id thì MongoDB sẽ cung cấp một id duy nhất cho mỗi Document.

**2.2.2. Các kiểu dữ liệu trong MongoDB**

* **Chuỗi**: Đây là kiểu dữ liệu được sử dụng phổ biến nhất để lưu giữ dữ liệu. Chuỗi trong MongoDB phải là UTF-8 hợp lệ.
* **Số nguyên**: Kiểu dữ liệu này được sử dụng để lưu một giá trị số. Số nguyên có thể là 32 bit hoặc 64 bit phụ thuộc vào SERVER của bạn.
* **Boolean**: Kiểu dữ liệu này được sử dụng để lưu giữ một giá trị Boolean (true/false). − Double: Kiểu dữ liệu này được sử dụng để lưu các giá trị số thực dấu chấm động.
* **Min/ Max keys**: Kiểu dữ liệu này được sử dụng để so sánh một giá trị với các phần tử BSON thấp nhất và cao nhất.
* **Mảng**: Kiểu dữ liệu này được sử dụng để lưu giữ các mảng hoặc danh sách hoặc nhiều giá trị vào trong một key.
* **Timestamp**: Giúp thuận tiện cho việc ghi chép hoặc đánh dấu thời điểm một document được sửa đổi hoặc được thêm vào.
* **Object:** Kiểu dữ liệu này được sử dụng cho các document được nhúng vào.
* **Null:** Kiểu dữ liệu này được sử dụng để lưu một giá trị Null. 29
* **Symbol:** Kiểu dữ liệu này được sử dụng giống như một chuỗi, tuy nhiên, nói chung nó được dành riêng cho các ngôn ngữ mà sử dụng kiểu symbol cụ thể.
* **Date**: Kiểu dữ liệu này được sử dụng để lưu giữ date và time hiện tại trong định dạng UNIX time. Có thể xác định date time riêng bằng việc tạo đối tượng Date và truyền ngày, tháng, năm vào trong đó.
* **Object ID:** Kiểu dữ liệu này được sử dụng để lưu giữ ID của document.
* **Binary data:** Kiểu dữ liệu này được sử dụng để lưu giữ dữ liệu nhị phân.
* Code: Kiểu dữ liệu này được sử dụng để lưu giữ JavaScrip code vào trong document.
* **Regular expression**: Kiểu dữ liệu này được sử dụng để lưu giữ Regular Expresion

**2.2.3. Các thao tác dữ liệu trên MongoDB**

1. **Tạo cơ sở dữ liệu**

Sử dụng lệnh use để tạo cơ sở dữ liệu, các cơ sở dữ liệu được xác định bởi tên của nó. Lệnh use sẽ tạo một cơ sở dữ liệu mới nếu CSDL chưa tồn tại, nếu CSDL đã tồn tại thì kết quả sẽ trả về cơ sở dữ liệu đang tồn tại.

Cú pháp cơ bản: **use 'DATABASE\_NAME'**

Trong đó quy tắt đặt tên cho < DATABASE\_NAME > như sau:

* Gồm chữ và số;
* Phân biệt chữ hoa hay chữ thường;
* Không chứa chuỗi rỗng (“”);
* Không chứa các ký tự sau: /, \, ., ", \*, , :, |, ?, $, (khoảng trắng), or \0 (the null character)
* Tên cơ sở dữ liệu giới hạn tối đa 64 bytes.

1. **Hiển thị tất cả các databases**

Sử dụng lệnh: **show dbs**

Cơ sở dữ liệu mặc định trong MongoDB là test, nếu người dùng không tạo thêm bất kỳ cơ sở dữ liệu nào thì các collection sẽ được lưu trữ trong test.

Khi một cơ sở dữ liệu được tạo mà chưa có bất kỳ collection nào thì tên cơ sở dữ liệu đó sẽ không hiện lên trong danh sách các cơ sở dữ liệu khi sử dụng lệnh show dbs.

1. **Tạo một collection**

Sử dụng lệnh: **db.createCollection(name, options)**

Trong lệnh trên, name là tên của Collection. Options là một Document và được sử dụng để xác định cấu hình cho Collection.

Bảng 3. Miêu tả tham số

|  |  |  |
| --- | --- | --- |
| **Tham số** | **Kiểu** | **Miêu tả** |
| Name | Chuỗi | Tên của Collection |
| Options | Document | (Tùy ý) Xác định các tùy chọn về kích cỡ bộ nhớ và việc lập chỉ mục |

Tham số options là tùy ý, vì thế bạn chỉ cần xác định tên của Collection. Dưới đây là danh sách các tùy chọn bạn có thể sử dụng:

Bảng 4. Danh sách các options

| **Trường** | **Kiểu** | **Miêu tả** |
| --- | --- | --- |
| capped | Boolean | (Tùy ý) Nếu true, kích hoạt một Capped Collection. Đây là một Collection có kích cỡ cố định mà tự động ghi đè các entry cũ nhất khi nó tiếp cận kích cỡ tối đa. **Nếu bạn xác định là true, thì bạn cũng cần xác định tham số size** |
| autoIndexID | Boolean | (Tùy ý) Nếu true, tự động tạo chỉ mục trên các trường \_id. Giá trị mặc định là false |
| size | Số | (Tùy ý) Xác định kích cỡ tối đa (giá trị byte) cho một Capped Collection. **Nếu tham số capped là true, thì bạn cũng cần xác định trường này** |
| max | Số | (Tùy ý) Xác định số Document tối đa được cho phép trong một Capped Colleciton |

Trong khi thực hiện việc chèn dữ liệu vào Document, đầu tiên MongoDB kiểm tra trường size của Capped Collection, sau đó nó kiểm tra trường max.

**Ví dụ:**

Cú pháp cơ bản của phương thức createCollection() mà không có các options là như sau:

>use test

switched to db test>db.createCollection("mycollection"){ "ok" : 1 }>

Bạn có thể kiểm tra Collection đã tạo bởi sử dụng lệnh show collections như sau:

>show collections

mycollection

system.indexes

1. **Xóa một collection**

Cú pháp cơ bản: **db.COLLECTION\_NAME.drop()**

**Ví dụ:**

Đầu tiên, bạn kiểm tra các Collection có sẵn bên trong cơ sở dữ liệu mydb:

>use mydb

switched to db mydb>show collections

mycol

mycollection

system.indexes

tutorialspoint>

Bây giờ, bạn xóa Collection với tên mycollection như sau:

>db.mycollection.drop()true>

Phương thức drop() sẽ trả về true, nếu Collection đã chọn bị xóa thành công, nếu không, nó sẽ trả về false.

>show collections

mycol

system.indexes

tutorialspoint>

**2.2.4. Làm việc với Documents**

1. **Chèn**

Cú pháp cơ bản: >db.COLLECTION\_NAME.insert(document)

**Ví dụ:**

Trong Document được chèn, nếu chúng ta không xác định tham số \_id, thì MongoDB gán một ObjectId duy nhất cho Document này.

>db.mycol.insert({

\_id: ObjectId(7df78ad8902c),

title: 'MongoDB Overview',

description: 'MongoDB is no sql database',

by: 'tutorials point',

url: 'http://www.tutorialspoint.com',

tags: ['mongodb', 'database', 'NoSQL'],

likes: 100})

Ở đây, mycol là tên của Collection, đã được tạo trong chương trước. Nếu Collection này chưa tồn tại trong cơ sở dữ liệu, thì MongoDB sẽ tạo Collection này và sau đó chèn Document vào trong nó.

\_id là một số thập lục phân duy nhất, dài 12 byte cho mỗi Document trong một Collection. 12 byte được phân chia như sau (đã được mô tả trong các chương trước):

\_id: ObjectId(4 bytes timestamp, 3 bytes machine id, 2 bytes process id, 3 bytes incrementer)

Để chèn nhiều Document trong một truy vấn đơn, bạn có thể truyền một mảng các Document trong lệnh insert() như sau:

>db.post.insert([{

title: 'MongoDB Overview',

description: 'MongoDB is no sql database',

by: 'tutorials point',

url: 'http://www.tutorialspoint.com',

tags: ['mongodb', 'database', 'NoSQL'],

likes: 100},{

title: 'NoSQL Database',

description: 'NoSQL database doesn't have tables',

by: 'tutorials point',

url: 'http://www.tutorialspoint.com',

tags: ['mongodb', 'database', 'NoSQL'],

likes: 20,

comments: [

{

user:'user1',

message: 'My first comment',

dateCreated: new Date(2013,11,10,2,35),

like: 0

}

]}])

Để chèn dữ liệu vào trong Document, bạn cũng có thể sử dụng db.post.save(document). Nếu bạn không xác định \_id trong Document, thì phương thức save() sẽ làm việc giống như phương thức insert(). Nếu bạn xác định \_id, thì nó sẽ thay thế toàn bộ dữ liệu của Document chứa \_id khi được xác định trong phương thức save().

1. **Truy vấn**

Cú pháp cơ bản:>db.COLLECTION\_NAME.find()

Phương thức find() sẽ hiển thị tất cả Document ở dạng không có cấu trúc (hiển thị không theo cấu trúc nào).

Ngoài ra, để hiển thị các kết quả theo một cách đã được định dạng, bạn có thể sử dụng phương thức pretty() như sau:

>db.mycol.find().pretty(){

"\_id": ObjectId(7df78ad8902c),

"title": "MongoDB Overview",

"description": "MongoDB is no sql database",

"by": "tutorials point",

"url": "http://www.tutorialspoint.com",

"tags": ["mongodb", "database", "NoSQL"],

"likes": "100"}>

Ngoài phương thức find(), trong MongoDB còn có phương thức findOne() sẽ chỉ trả về một Document.

Để truy vấn Document dựa trên một số điều kiện nào đó, bạn có thể sử dụng các phép toán sau:

Bảng 5. Các phép truy vấn

| **Phép toán** | **Cú pháp** | **Ví dụ** | **Mệnh đề WHERE tương đương** |
| --- | --- | --- | --- |
| Equality | {<key>:<value>} | db.mycol.find({"by":"tutorials point"}).pretty() | where by = 'tutorials point' |
| Less Than | {<key>:{$lt:<value>}} | db.mycol.find({"likes":{$lt:50}}).pretty() | where likes < 50 |
| Less Than Equals | {<key>:{$lte:<value>}} | db.mycol.find({"likes":{$lte:50}}).pretty() | where likes <= 50 |
| Greater Than | {<key>:{$gt:<value>}} | db.mycol.find({"likes":{$gt:50}}).pretty() | where likes > 50 |
| Greater Than Equals | {<key>:{$gte:<value>}} | db.mycol.find({"likes":{$gte:50}}).pretty() | where likes >= 50 |
| Not Equals | {<key>:{$ne:<value>}} | db.mycol.find({"likes":{$ne:50}}).pretty() | where likes != 50 |

1. **Cập nhật**

Phương thức update() hoặc save() trong MongoDB được sử dụng để cập nhật Document vào trong một Collection. Phương thức update() cập nhật các giá trị trong Document đang tồn tại trong khi phương thức save() thay thế Document đang tồn tại với Document đã truyền trong phương thức save() đó.

Cú pháp cơ bản:

>db.COLLECTION\_NAME.update(SELECTIOIN\_CRITERIA, UPDATED\_DATA)

Ví dụ:

Bạn theo dõi Collection có tên mycol có dữ liệu sau:

{ "\_id" : ObjectId(5983548781331adf45ec5), "title":"MongoDB Overview"}

{ "\_id" : ObjectId(5983548781331adf45ec6), "title":"NoSQL Overview"}

{ "\_id" : ObjectId(5983548781331adf45ec7), "title":"Tutorials Point Overview"}

Ví dụ sau sẽ thiết lập tiêu đề mới 'New MongoDB Tutorial' của Document có tiêu đề là 'MongoDB Overview':

>db.mycol.update({'title':'MongoDB Overview'},{$set:{'title':'New MongoDB Tutorial'}})

>db.mycol.find()

{ "\_id" : ObjectId(5983548781331adf45ec5), "title":"New MongoDB Tutorial"}

{ "\_id" : ObjectId(5983548781331adf45ec6), "title":"NoSQL Overview"}

{ "\_id" : ObjectId(5983548781331adf45ec7), "title":"Tutorials Point Overview"}>

Theo mặc định, MongoDB sẽ chỉ cập nhật một Document đơn, để cập nhật nhiều Document, bạn thiết lập tham số 'multi' thành true.

>db.mycol.update({'title':'MongoDB Overview'},{$set:{'title':'New MongoDB Tutorial'}},{multi:true})

1. **Lưu**

Cú pháp cơ bản:

>db.COLLECTION\_NAME.save({\_id:ObjectId(),NEW\_DATA})

**Ví dụ:**

Ví dụ sau sẽ thay thế Document với \_id là '5983548781331adf45ec7'.

>db.mycol.save(

{

"\_id" : ObjectId(5983548781331adf45ec7), "title":"Tutorials Point New Topic", "by":"Tutorials Point"

})

>db.mycol.find()

{ "\_id" : ObjectId(5983548781331adf45ec5), "title":"Tutorials Point New Topic", "by":"Tutorials Point"}

{ "\_id" : ObjectId(5983548781331adf45ec6), "title":"NoSQL Overview"}

{ "\_id" : ObjectId(5983548781331adf45ec7), "title":"Tutorials Point Overview"}>

**CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU**

1. **Thiết kế cơ sở dữ liệu**
2. **Tìm hiểu**

* Cấu trúc đề thi tiếng Anh Reading chuẩn VSTEP gồm có 4 đoạn, mỗi đoạn có 10 câu. Tổng cộng 40 câu trắc nghiệm.
* Một kỳ thi thường được tổ chức khoảng 1 đến 2 ngày tùy theo số lượng thí sinh đăng ký thi.

1. **Thiết kế cấu trúc lược đồ cơ sở dữ liệu**

**Collection**: Thi\_sinh

{

* id: mã thí sinh
* ngay\_thi: ngày thi
* ten: tên người dùng
* email: email đăng ký
* mat\_khau: mật khẩu của thí sinh (đã mã hóa)
* ngay\_tao: ngày tạo tài khoản
* diem\_so: điểm số của thí sinh sau khi thi
* de\_thi: mỗi thí sinh chỉ thi 1 đề mỗi lần

{

* ma\_de: mã đề thi
* noi\_dung: nội dung của đề thi

{

* doan\_mot: đoạn 1 của đề thi

{

* noi\_dung: nội dung đề của đoạn 1
* cau\_mot: câu hỏi thứ nhất trong đoạn 1
* noi\_dung: nội dung của câu 1
* A: lựa chọn 1 của câu 1
* B: lựa chọn 2 của câu 1
* C: lựa chọn 3 của câu 1
* D: lựa chọn 4 của câu 1
* dap\_an\_dung: đáp án đúng của câu 1
* dap\_an\_thi\_sinh: đáp án của thí sinh
* cau\_hai: câu hỏi thứ hai trong đoạn 1
* noi\_dung: nội dung của câu 2
* A: lựa chọn 1 của câu 2
* B: lựa chọn 2 của câu 2
* C: lựa chọn 3 của câu 2
* D: lựa chọn 4 của câu 2
* dap\_an\_dung: đáp án đúng của câu 2
* dap\_an\_thi\_sinh: đáp án của thí sinh
* cau\_ba: câu hỏi thứ ba trong đoạn 1
* noi\_dung: nội dung của câu 3
* A: lựa chọn 1 của câu 3
* B: lựa chọn 2 của câu 3
* C: lựa chọn 3 của câu 3
* D: lựa chọn 4 của câu 3
* dap\_an\_dung: đáp án đúng của câu 3
* dap\_an\_thi\_sinh: đáp án của thí sinh
* cau\_bon: câu hỏi thứ bốn trong đoạn 1
* noi\_dung: nội dung của câu 4
* A: lựa chọn 1 của câu 4
* B: lựa chọn 2 của câu 4
* C: lựa chọn 3 của câu 4
* D: lựa chọn 4 của câu 4
* dap\_an\_dung: đáp án đúng của câu 4
* dap\_an\_thi\_sinh: đáp án của thí sinh
* cau\_nam: câu hỏi thứ năm trong đoạn 1
* noi\_dung: nội dung của câu 5
* A: lựa chọn 1 của câu 5
* B: lựa chọn 2 của câu 5
* C: lựa chọn 3 của câu 5
* D: lựa chọn 4 của câu 5
* dap\_an\_dung: đáp án đúng của câu 5
* dap\_an\_thi\_sinh: đáp án của thí sinh
* cau\_sau: câu hỏi thứ sáu trong đoạn 1
* noi\_dung: nội dung của câu 6
* A: lựa chọn 1 của câu 6
* B: lựa chọn 2 của câu 6
* C: lựa chọn 3 của câu 6
* D: lựa chọn 4 của câu 6
* dap\_an\_dung: đáp án đúng của câu 6
* dap\_an\_thi\_sinh: đáp án của thí sinh
* cau\_bay: câu hỏi thứ bảy trong đoạn 1
* noi\_dung: nội dung của câu 7
* A: lựa chọn 1 của câu 7
* B: lựa chọn 2 của câu 7
* C: lựa chọn 3 của câu 7
* D: lựa chọn 4 của câu 7
* dap\_an\_dung: đáp án đúng của câu 7
* dap\_an\_thi\_sinh: đáp án của thí sinh
* cau\_tam: câu hỏi thứ tám trong đoạn 1
* noi\_dung: nội dung của câu 8
* A: lựa chọn 1 của câu 8
* B: lựa chọn 2 của câu 8
* C: lựa chọn 3 của câu 8
* D: lựa chọn 4 của câu 8
* dap\_an\_dung: đáp án đúng của câu 8
* dap\_an\_thi\_sinh: đáp án của thí sinh
* cau\_chin: câu hỏi thứ chín trong đoạn 1
* noi\_dung: nội dung của câu 9
* A: lựa chọn 1 của câu 9
* B: lựa chọn 2 của câu 9
* C: lựa chọn 3 của câu 9
* D: lựa chọn 4 của câu 9
* dap\_an\_dung: đáp án đúng của câu 9
* dap\_an\_thi\_sinh: đáp án của thí sinh

**…**

* cau\_muoi: câu hỏi thứ mười trong đoạn 4
* noi\_dung: nội dung của câu 10
* A: lựa chọn 1 của câu 10
* B: lựa chọn 2 của câu 10
* C: lựa chọn 3 của câu 10
* D: lựa chọn 4 của câu 10
* dap\_an\_dung: đáp án đúng của câu 10
* dap\_an\_thi\_sinh: đáp án của thí sinh

}}}}

Mô tả cấu trúc: Lược đồ cơ sở dữ liệu của một Documents thể hiện thông tin của một thí sinh và nội dung đề thi thí sinh đã làm, cùng với đáp án sau khi đã hoàn thành bài thi và được lưu trữ kết quả vào hệ thống.

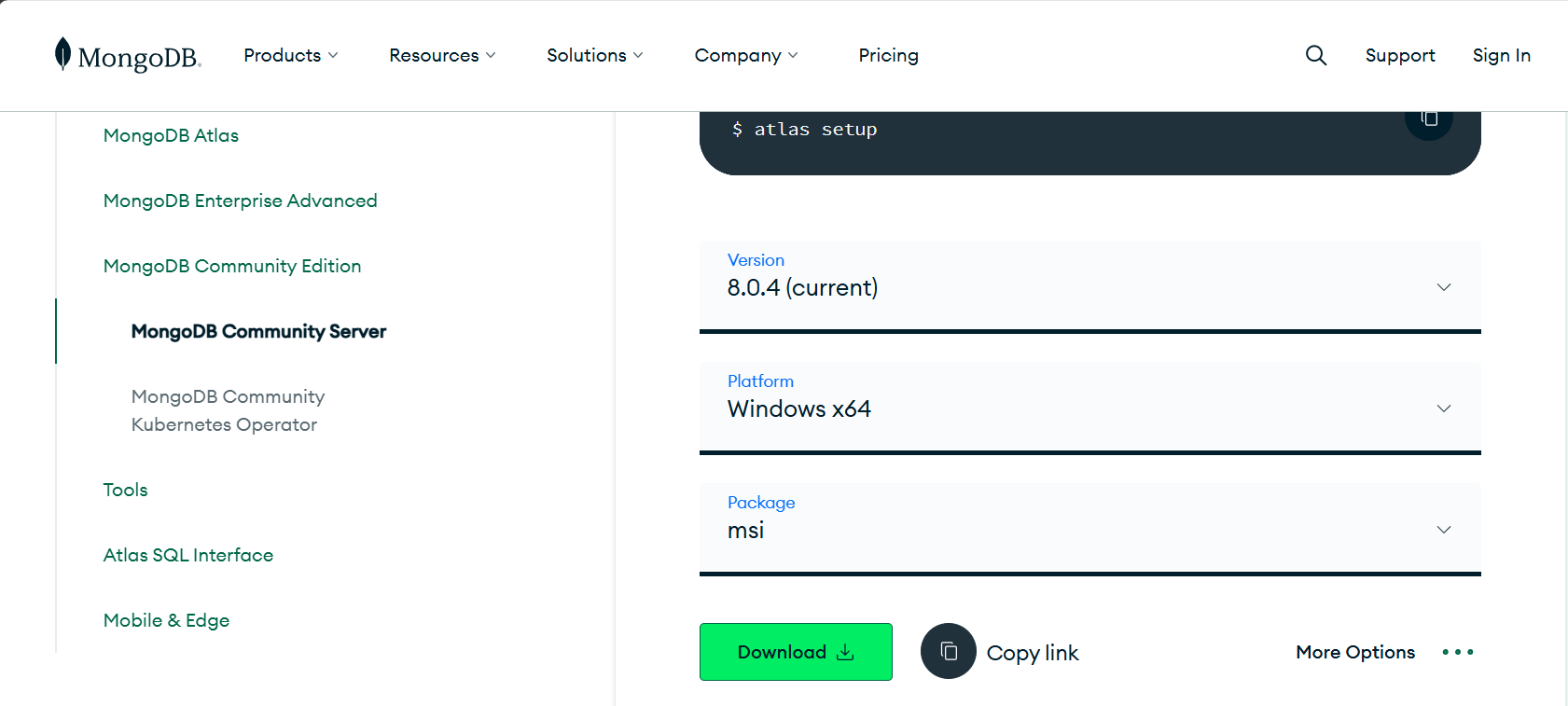
1. **Lập bảng phân cấp thuộc tính**

*Bảng 6**. Phân cấp thuộc tính*

| **STT** | **Thuộc tính cấp 1** | **Thuộc tính cấp 2** | **Thuộc tính cấp 3** | **Thuộc tính cấp 4** | **Thuộc tính cấp 5** |
| --- | --- | --- | --- | --- | --- |
| 1 | ngay\_thi |  |  |  |  |
| 2 | thi\_sinh |  |  |  |  |
| 3 | ten |  |  |  |  |
| 4 | id |  |  |  |  |
| 5 | email |  |  |  |  |
| 6 | mat\_khau |  |  |  |  |
| 7 | ngay\_tao |  |  |  |  |
| 8 | diem\_so |  |  |  |  |
| 9 | de\_thi |  |  |  |  |
| 10 |  | ma\_de |  |  |  |
| 11 |  | noi\_dung |  |  |  |
| 12 |  |  | doan\_mot |  |  |
| 13 |  |  |  | noi\_dung  cau\_mot  cau\_hai  cau\_ba  cau\_bon  cau\_nam  cau\_sau  cau\_bay  cau\_tam  cau\_chin  cau\_muoi |  |
| 14 |  |  |  |  | A  B  C  D  dap\_an\_dung  dap\_an\_thi\_sinh |
| 15 |  |  | doan\_hai |  |  |
| 16 |  |  |  | noi\_dung  cau\_mot  cau\_hai  cau\_ba  cau\_bon  cau\_nam  cau\_sau  cau\_bay  cau\_tam  cau\_chin  cau\_muoi |  |
| 17 |  |  |  |  | A  B  C  D  dap\_an\_dung  dap\_an\_thi\_sinh |
| 18 |  |  | doan\_ba |  |  |
| 19 |  |  |  | noi\_dung  cau\_mot  cau\_hai  cau\_ba  cau\_bon  cau\_nam  cau\_sau  cau\_bay  cau\_tam  cau\_chin  cau\_muoi |  |
| 20 |  |  |  |  | A  B  C  D  dap\_an\_dung  dap\_an\_thi\_sinh |
| 21 |  |  | doan\_bon |  |  |
| 22 |  |  |  | noi\_dung  cau\_mot  cau\_hai  cau\_ba  cau\_bon  cau\_nam  cau\_sau  cau\_bay  cau\_tam  cau\_chin  cau\_muoi |  |
| 21 |  |  |  |  | A  B  C  D  dap\_an\_dung  dap\_an\_thi\_sinh |

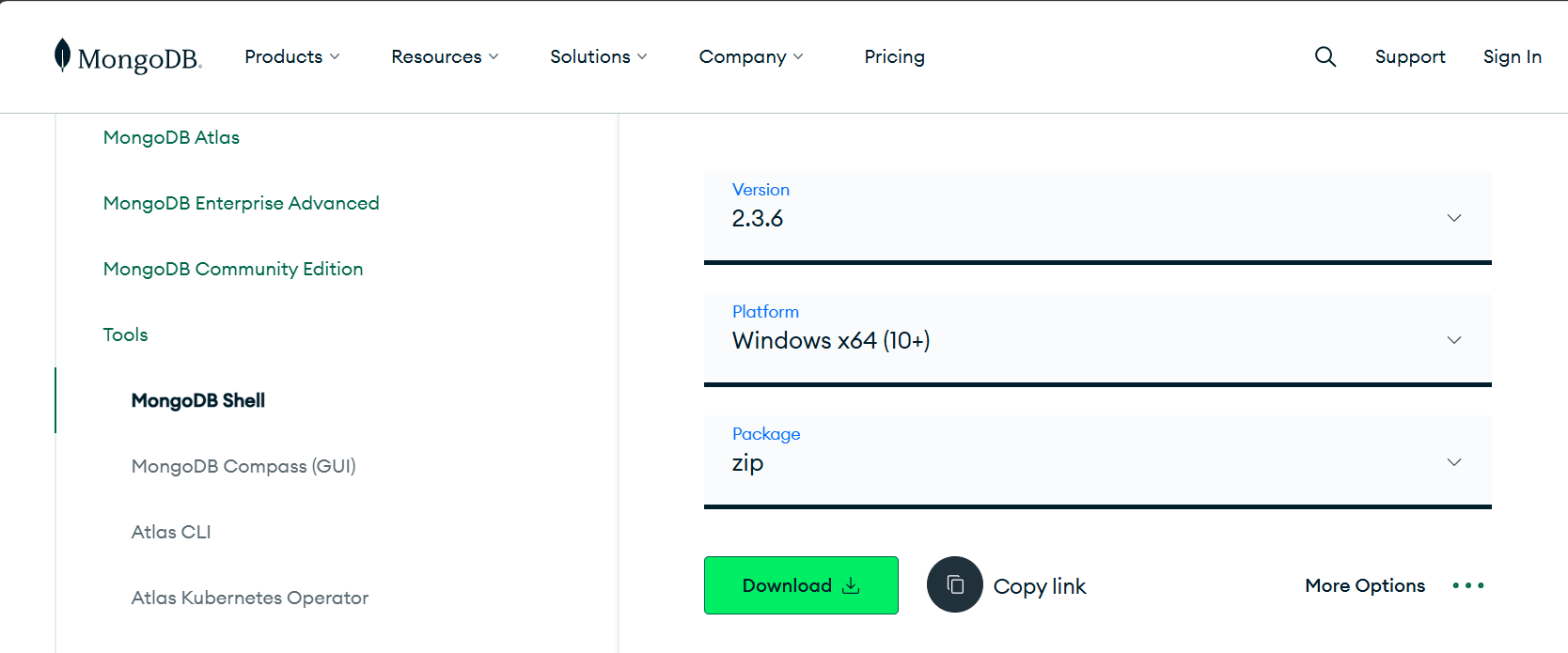
1. **Cài đặt môi trường làm việc**

**Cài đặt MongoDB:**

Bước 1: Truy cập: <https://www.mongodb.com/try/download/community>

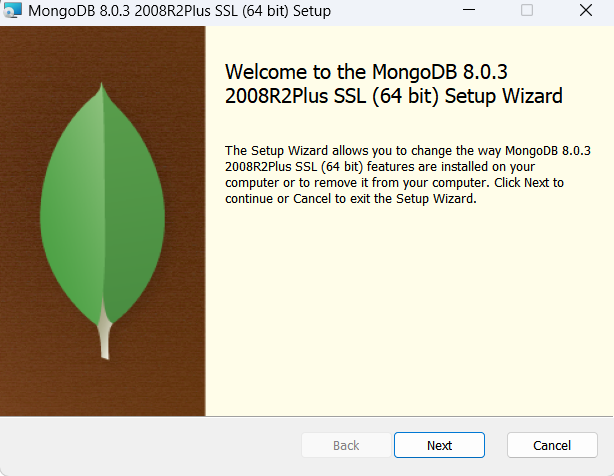
*Hình 1**. Cài đặt MongoDB Community Server*

Bước 2: Tải xuống MongoDB Commnunity Server và MongoDB Shell:



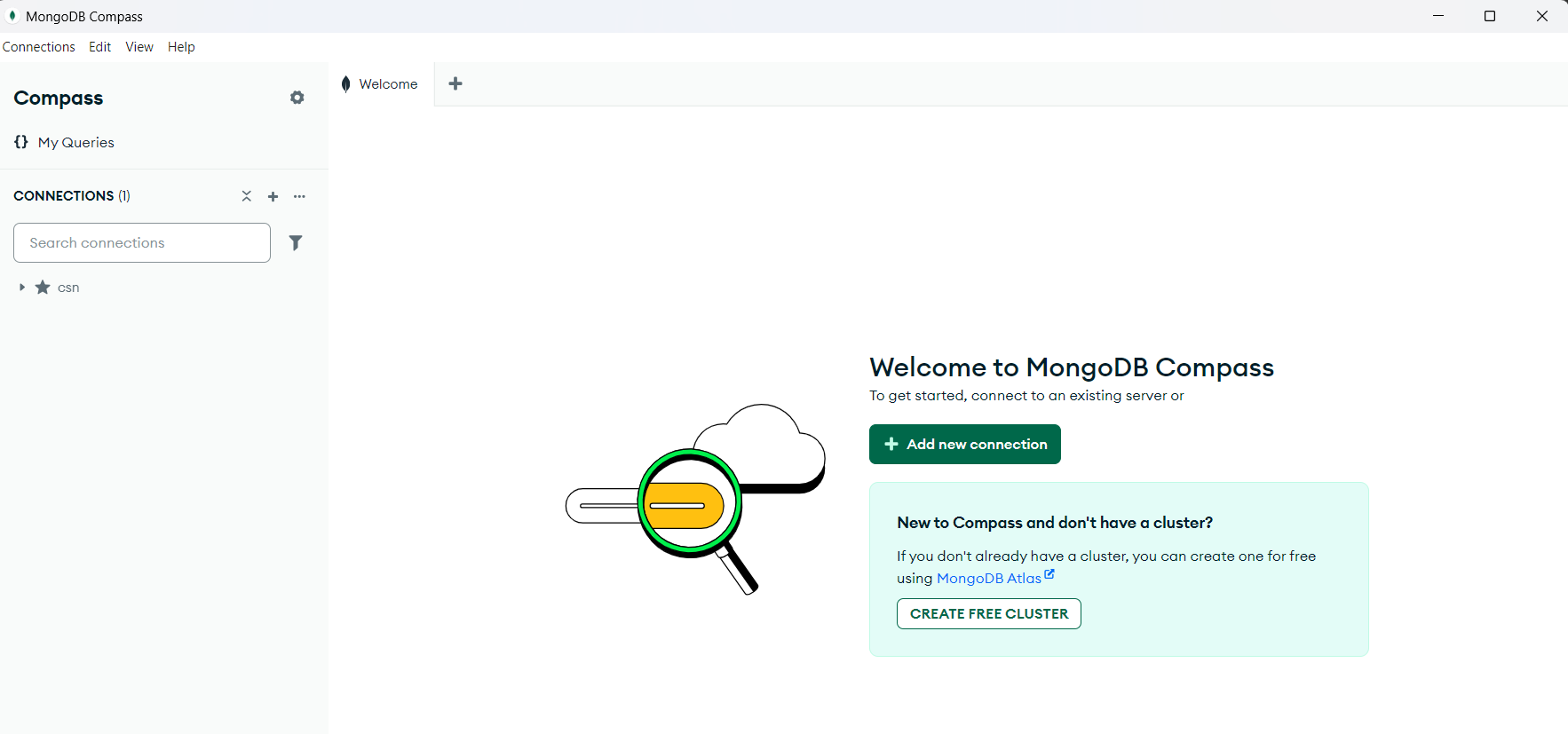
*Hình 2**. Cài đặt MongoDB Shell*

Bước 3: Tiến hành xác nhận tiến trình cài đặt:



*Hình 3**. Setup cài đặt MongoDB*

Bước 4: Hoàn thành cài đặt MongoDB Compass:



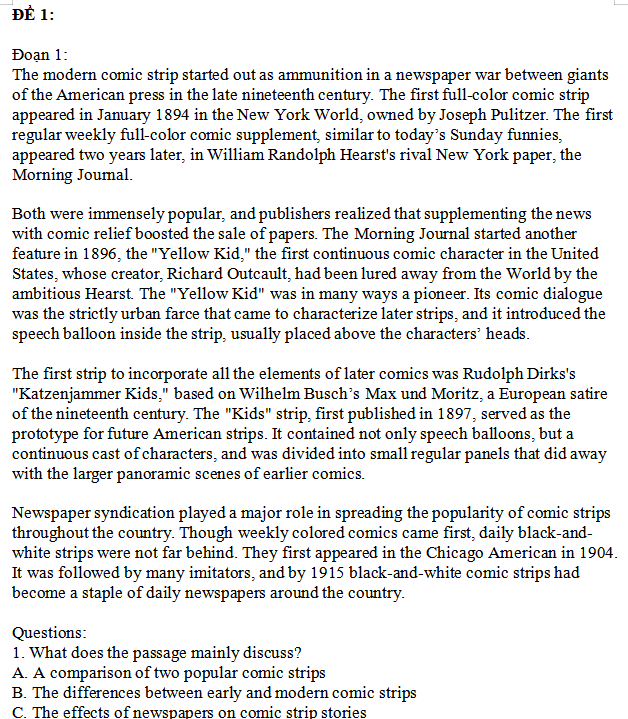
*Hình 4**. Giao diện MongoDB Compass*

# CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

1. **Chọn dữ liệu mẫu thử**

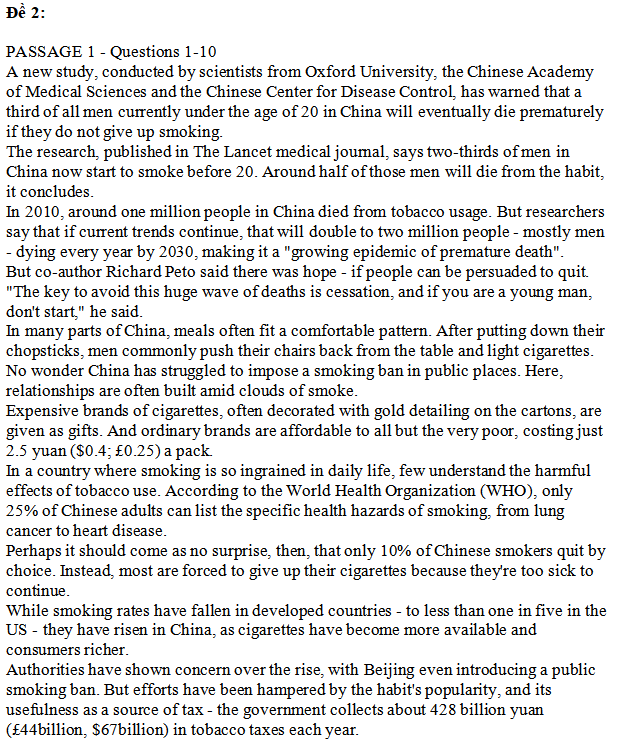
Chọn 3 đề thi tiếng Anh Reading dựa theo đề thi VSTEP đã ra:

Nội dung đề:



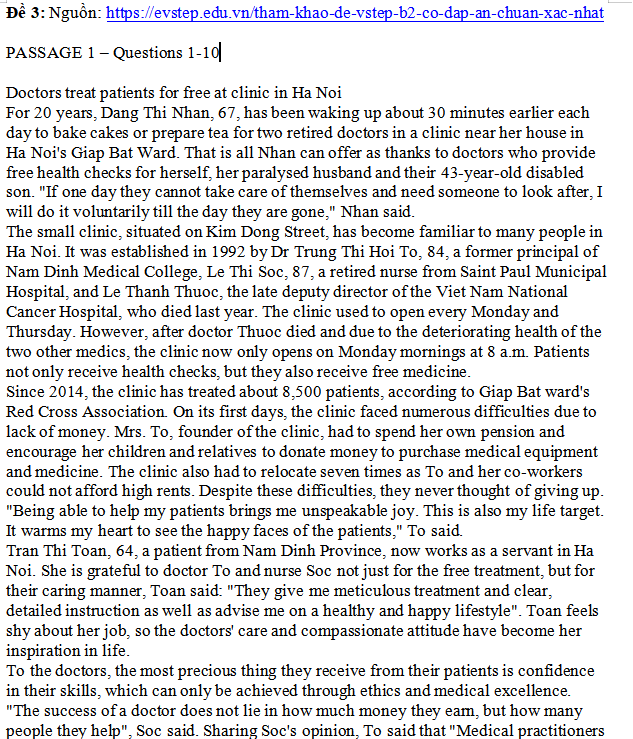
*Hình 5**. Minh họa đề 1*

Nguồn tài liệu: <https://evstep.edu.vn/de-thi-vstep-b1-co-dap-va-phuong-phap-on-luyen>



*Hình 6**. Minh họa đề 2*

Nguồn tài liệu: <https://evstep.edu.vn/tham-khao-de-vstep-b2-co-dap-an-chuan-xac-nhat>



*Hình 7**. Minh họa đề 3*

Nguồn tài liệu: <https://evstep.edu.vn/tham-khao-de-vstep-b2-co-dap-an-chuan-xac-nhat>

1. **Nhập dữ liệu mẫu thử**

Tổng quát hệ thống gồm 1 Collection với 15 Documents. Mỗi Documents chứa thông tin 1 thí sinh, đề thi và kết quả thi của thí sinh đó. Sau đây là mẫu cấu trúc cơ bản của đề 3 dựa trên đề thi có sẵn:

{

"\_id": {

"$oid": "675d5be80cfb7d2a236c249b"

},

"id": 14,

"ngay\_thi": {

"$date": "2024-01-02T00:00:00.000Z"

},

"ten": "Phan Thanh Phương",

"email": "11023phuongthognminh@gmail.com",

"mat\_khau": "\*\*\*",

"ngay\_tao": {

"$date": "2021-01-15T00:00:00.000Z"

},

"diem\_so": 3,

"dethi": {

"ma\_de": 3,

"noi\_dung": {

"doan\_mot": {

"noi\_dung": "Doctors treat patients for free at clinic in Ha Noi\nFor 20 years, Dang Thi Nhan, 67, has been waking up about 30 minutes earlier each day to bake cakes or prepare tea for two retired doctors in a clinic near her house in Ha Noi's Giap Bat Ward. That is all Nhan can offer as thanks to doctors who provide free health checks for herself, her paralysed husband and their 43-year-old disabled son. \"If one day they cannot take care of themselves and need someone to look after, I will do it voluntarily till the day they are gone,\" Nhan said.\nThe small clinic, situated on Kim Dong Street, has become familiar to many people in Ha Noi. It was established in 1992 by Dr Trung Thi Hoi To, 84, a former principal of Nam Dinh Medical College, Le Thi Soc, 87, a retired nurse from Saint Paul Municipal Hospital, and Le Thanh Thuoc, the late deputy director of the Viet Nam National Cancer Hospital, who died last year. The clinic used to open every Monday and Thursday. However, after doctor Thuoc died and due to the deteriorating health of the two other medics, the clinic now only opens on Monday mornings at 8 a.m. Patients not only receive health checks, but they also receive free medicine.\nSince 2014, the clinic has treated about 8,500 patients, according to Giap Bat ward's Red Cross Association. On its first days, the clinic faced numerous difficulties due to lack of money. Mrs. To, founder of the clinic, had to spend her own pension and encourage her children and relatives to donate money to purchase medical equipment and medicine. The clinic also had to relocate seven times as To and her co-workers could not afford high rents. Despite these difficulties, they never thought of giving up. \"Being able to help my patients brings me unspeakable joy. This is also my life target. It warms my heart to see the happy faces of the patients,\" To said.\nTran Thi Toan, 64, a patient from Nam Dinh Province, now works as a servant in Ha Noi. She is grateful to doctor To and nurse Soc not just for the free treatment, but for their caring manner, Toan said: \"They give me meticulous treatment and clear, detailed instruction as well as advise me on a healthy and happy lifestyle\". Toan feels shy about her job, so the doctors' care and compassionate attitude have become her inspiration in life.\nTo the doctors, the most precious thing they receive from their patients is confidence in their skills, which can only be achieved through ethics and medical excellence. \"The success of a doctor does not lie in how much money they earn, but how many people they help\", Soc said. Sharing Soc's opinion, To said that \"Medical practitioners should not consider their profession as a tool to get rich. They should not benefit from their patients' pain. Patients come first, not money.\"",

"cau\_mot": {

"noi\_dung": "How old was Dang Thi Nhan when she first started to bake cakes or prepare tea for the doctors?",

"A": "47",

"B": "67",

"C": "64",

"D": "47",

"dap\_an\_dung": "A",

"dap\_an\_thi\_sinh": "D"

},

"cau\_hai": {

"noi\_dung": "The word \"they\" in paragraph 1 refers to ...............",

"A": "two doctors",

"B": "husband and son",

"C": "health checks",

"D": "cakes",

"dap\_an\_dung": "A",

"dap\_an\_thi\_sinh": "A"

},

**…**

"cau\_muoi": {

"noi\_dung": "According to paragraph 5, the softwoods in the South posed which of the following problems for southern potash?",

"A": "The softwoods were not very plentiful.",

"B": "The softwoods could not be used to build houses.",

"C": "The softwoods were not very rich in potash.",

"D": "The softwoods were not very useful for making potash.",

"dap\_an\_dung": "C",

"dap\_an\_thi\_sinh": "C" }}}}}

1. **Đặt và chạy các câu truy vấn**
2. **Đếm tổng số thí sinh tham gia thi**

Câu truy vấn:

db.Thi\_sinh.countDocuments({})

Kết quả:

|  |
| --- |
| < 15 |

1. **Hiển thị các thí sinh thi ngày 2**

Câu truy vấn:

db.Thi\_sinh.countDocuments({"ngay\_thi": ISODate("2024-01-02T00:00:00Z")})

Kết quả:

|  |
| --- |
| < 5 |

1. **Đếm số thí sinh làm đề số 3**

Câu truy vấn:

db.Thi\_sinh.countDocuments({"dethi.ma\_de":3})

Kết quả:

|  |
| --- |
| < 5 |

1. **Tìm các thí sinh thi đề 3 và sắp xếp theo ngày thi gần nhất**

Câu truy vấn:

db.Thi\_sinh.find({“dethi.ma\_de”: 3}).sort({“ngay\_thi”:-1})

Kết quả:

|  |
| --- |
| < {  \_id: ObjectId(‘6749cb5aca1365681254fb98’), id: 14, ngay\_thi: 2024-01-02T00:00:00.000Z, ten: ‘Phan Thanh Phương’, … |

1. **Tìm và hiển thị nội dung câu số 7 của đoạn 3 trong mã đề 2**

Câu truy vấn:

db.Thi\_sinh.aggregate([{$match: {"dethi.ma\_de": 2}},

{$project: {\_id: 0, "noi\_dung\_cau\_bay": "$dethi.noi\_dung.doan\_ba.cau\_bay"},

{$group: {\_id: null, nd\_cau\_bay\_doan\_ba: { $first: "$noi\_dung\_cau\_bay" }}},

{$project: {\_id: 0, nd\_cau\_bay\_doan\_ba: 1}}] )

Kết quả:

|  |
| --- |
| < { nd\_cau\_bay\_doan\_ba: {  noi\_dung: ‘The word “incorporate” is closest in meaning to’,  A: ‘affect’,  B: ‘create’,  C: ‘combine’,  D: ‘mention’,  dap\_an\_dung: ‘C’,  dap\_an\_thi\_sinh: ‘A’ }  } |

1. **Lấy thông tin của thí sinh theo tên**

Câu truy vấn:

db.Thi\_sinh.find({ ten: "Phan Thanh Phương" }, { \_id: 0, ten: 1, email: 1, dethi: 1 })

Kết quả:

|  |
| --- |
| < { ten: ‘Phan Thanh Phương’, email: [‘10102002phuong@gmail.com,](mailto:‘10102002phuong@gmail.com,) dethi: {  ma\_de: 3,  noi\_dung: {  doan\_mot: {  noi\_dung: ‘Technology has revolutionized the way we live, work, …’  … |

1. **Hiển thị chỉ nội dung đề 2**

Câu truy vấn:

db.Thi\_sinh.aggregate([{$match: {"dethi.ma\_de": 2}},

{$project: {\_id: 0, "noi\_dung\_de\_2": "$dethi.noi\_dung"}},

{$group: {\_id: null, noi\_dung\_de\_2: { $first: "$noi\_dung\_de\_2" }}},

{$project: {\_id: 0, noi\_dung\_de\_2: 1}}] )

Kết quả:

|  |
| --- |
| < { noi\_dung\_de\_2: {  doan\_mot: {  noi\_dung: ‘A new study, conducted by scientists from Oxford University  cau\_mot: {  noi\_dung: ‘How many Chinese men start to smoke before the age of 20?  A: ‘one-third’,  B: ‘two-thirds’,  C: ‘half’,  D: ‘all of them’,  dap\_an\_dung: ‘B’,  dap\_an\_thi\_sinh: ‘B’  },  cau\_hai: {  … |

1. **Hiển thị tên và mã đề của các thí sinh thi ngày 1**

Câu truy vấn:

db.Thi\_sinh.find(

{"ngay\_thi": ISODate("2024-01-01T00:00:00Z")},

{\_id:0, "ten":1, "dethi.ma\_de":1})

Kết quả:

|  |
| --- |
| < { ten: ‘Tô Chí Nguyên’, dethi: {  ma\_de: 1 }  }  { ten: ‘Nguyễn Thiện Nhân’, dethi: {  ma\_de: 1 }  }  … |

1. **Đếm số câu sai của thí sinh Tô Chí Nguyên**

Câu truy vấn:

db.Thi\_sinh.aggregate([

{$match: {"ten": "Tô Chí Nguyên"}},

{$project: {"ten": 1, "sai\_doan\_mot": { $size: {$filter: {input: [

{$ne: ["$dethi.noi\_dung.doan\_mot.cau\_mot.dap\_an\_dung", "$dethi.noi\_dung.doan\_mot.cau\_mot.dap\_an\_thi\_sinh"]},

{$ne: ["$dethi.noi\_dung.doan\_mot.cau\_hai.dap\_an\_dung", "$dethi.noi\_dung.doan\_mot.cau\_hai.dap\_an\_thi\_sinh"]},

{$ne: ["$dethi.noi\_dung.doan\_mot.cau\_ba.dap\_an\_dung", "$dethi.noi\_dung.doan\_mot.cau\_ba.dap\_an\_thi\_sinh"]},

**…**

"$dethi.noi\_dung.doan\_bon.cau\_chin.dap\_an\_thi\_sinh"]},

{$ne: ["$dethi.noi\_dung.doan\_bon.cau\_muoi.dap\_an\_dung", "$dethi.noi\_dung.doan\_bon.cau\_muoi.dap\_an\_thi\_sinh"]}

], cond: {$eq: [true, "$$this"] }}}}}},

{$project: { "ten": 1, "tong\_cau\_sai": { $add: [ "$sai\_doan\_mot", "$sai\_doan\_hai", "$sai\_doan\_ba", "$sai\_doan\_bon"]}, "\_id": 0 } } ] )

Kết quả:

|  |
| --- |
| < { ten: ‘Tô Chí Nguyên’, tong\_cau\_sai: 12  } |

1. **Hiển thị các câu đúng của thí sinh Kiến Nguyễn Khánh Đoan**

Câu truy vấn:

Dựa vào câu truy vấn 9, thêm hàm $subtract để trừ đi số câu sai:

{ $project: { "ten": 1, "so\_cau\_dung":

{$subtract: [ 40,

{$add: ["$sai\_doan\_mot", "$sai\_doan\_hai", "$sai\_doan\_ba", "$sai\_doan\_bon"]}]}, "\_id": 0}}

Kết quả:

|  |
| --- |
| < { ten: ‘Kiến Nguyễn Khánh Đoan’, so\_cau\_dung: 20  } |

1. **Hiển thị tên các thí sinh và điểm số**

Câu truy vấn:

db.Thi\_sinh.find({},{\_id:0, “ten”: 1, “diem\_so”: 1})

Kết quả:

|  |
| --- |
| < { ten: ‘Tô Chí Nguyên’, diem\_so: 7  }  { ten: ‘Nguyễn Thiện Nhân’, diem\_so: 8  }  { ten: ‘Cam Văn Sáng’, diem\_so: 6.5  }  … |

1. **Tìm các thí sinh thi trượt (điểm dưới 4.0)**

Câu truy vấn:

db.Thi\_sinh.find({"diem\_so": {$lt:4}},{\_id: 0, "ten": 1, "diem\_so": 1})

Kết quả:

|  |
| --- |
| < { ten: ‘Phạm Trung Kiên’, diem\_so: 3.75  }  { ten: ‘Phan Thanh Phương’, diem\_so: 3  } |

1. **Đếm số câu đúng ở đoạn 2 của thí sinh Cam Văn Sáng**

Câu truy vấn:

db.Thi\_sinh.aggregate([

{$match: {"ten": "Cam Văn Sáng"}},

{ $project: { "ten": 1, "sai\_doan\_hai": { $size: { $filter: { input: [ {$ne: ["$dethi.noi\_dung.doan\_hai.cau\_mot.dap\_an\_dung", "$dethi.noi\_dung.doan\_hai.cau\_mot.dap\_an\_thi\_sinh"]},

{$ne: ["$dethi.noi\_dung.doan\_hai.cau\_hai.dap\_an\_dung", "$dethi.noi\_dung.doan\_hai.cau\_hai.dap\_an\_thi\_sinh"]},

{$ne: ["$dethi.noi\_dung.doan\_hai.cau\_ba.dap\_an\_dung", "$dethi.noi\_dung.doan\_hai.cau\_ba.dap\_an\_thi\_sinh"]},

{$ne: ["$dethi.noi\_dung.doan\_hai.cau\_bon.dap\_an\_dung", "$dethi.noi\_dung.doan\_hai.cau\_bon.dap\_an\_thi\_sinh"]},

{$ne: ["$dethi.noi\_dung.doan\_hai.cau\_nam.dap\_an\_dung", "$dethi.noi\_dung.doan\_hai.cau\_nam.dap\_an\_thi\_sinh"]},

{$ne: ["$dethi.noi\_dung.doan\_hai.cau\_sau.dap\_an\_dung", "$dethi.noi\_dung.doan\_hai.cau\_sau.dap\_an\_thi\_sinh"]},

{$ne: ["$dethi.noi\_dung.doan\_hai.cau\_bay.dap\_an\_dung", "$dethi.noi\_dung.doan\_hai.cau\_bay.dap\_an\_thi\_sinh"]},

{$ne: ["$dethi.noi\_dung.doan\_hai.cau\_tam.dap\_an\_dung", "$dethi.noi\_dung.doan\_hai.cau\_tam.dap\_an\_thi\_sinh"]},

{$ne: ["$dethi.noi\_dung.doan\_hai.cau\_chin.dap\_an\_dung", "$dethi.noi\_dung.doan\_hai.cau\_chin.dap\_an\_thi\_sinh"]},

{$ne: ["$dethi.noi\_dung.doan\_hai.cau\_muoi.dap\_an\_dung", "$dethi.noi\_dung.doan\_hai.cau\_muoi.dap\_an\_thi\_sinh"]}

], cond: {$eq: [true, "$$this"]}}}}}},

{ $project: { "ten": 1, "tong\_cau\_sai\_doan\_hai": {$add: [ "$sai\_doan\_hai"]}, "\_id": 0 }}] )

Kết quả:

|  |
| --- |
| < { ten: ‘Cam Văn Sáng’, tong\_cau\_sai\_doan\_hai: 2  } |

1. **Tìm các thí sinh có họ tên bắt đầu bằng chữ T có điểm trên 7.25**

Câu truy vấn:

db.Thi\_sinh.find(

    { "ten": { $regex: "^T" }, "diem\_so": { $gt: 7.25 }},

{ "ten": 1, "diem\_so": 1, "\_id": 0})

Kết quả:

|  |
| --- |
| < { ten: ‘Tạ Nhật Khoa’, diem\_so: 7.5  }  { ten: ‘Trần Cao Trí’, diem\_so: 8.75  } |

1. **Hiển thị nội dung đoạn 4 của đề 3 và đề 2**

Câu truy vấn:

db.Thi\_sinh.aggregate([

{ $match: { "dethi.ma\_de": { $in: [2, 3] }}},

{ $group: { "\_id": "$dethi.ma\_de", "noi\_dung\_doan\_bon": { $first: "$dethi.noi\_dung.doan\_bon.noi\_dung" }}},

{ $project: { "ma\_de": "$\_id", "noi\_dung\_doan\_bon": 1, "\_id": 0 }},

{ $sort: { "ma\_de": 1 }}])

Kết quả:

|  |
| --- |
| < { noi\_dung\_doan\_bon: ‘Antibiotics block the life cycle of bacteria that invade …’ ma\_de: 2  }  { noi\_dung\_doan\_bon: ‘Potash ( the old name for potassium carbonate) is one…’ ma\_de: 3  } |

# CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

**5.1 Kết luận**

**5.1.1 Kết quả đạt được**

**Thiết kế được 1 cơ sở dữ liệu:**

**Demo mô hình thử nghiệm:**

**Hoàn thành nội dung trong đề cương đề ra:**

**5.1.2 Hạn chế**

Bài thi thiếu thang đo cấp độ.

Cơ sở dữ liệu chưa có chiều sâu và đa dạng dữ liệu.

**5.2 Hướng phát triển**

Thêm nhiều thuộc tính, tính năng đánh giá năng lực thí sinh và cấp độ bài làm.

# DANH MỤC TÀI LIỆU THAM KHẢO

**Tài liệu là trang web:**

1. EVSTEP (2023), Đề thi Vstep B1 có đáp án và phương pháp luyện thi hiệu quả. Liên kết: <https://evstep.edu.vn/de-thi-vstep-b1-co-dap-va-phuong-phap-on-luyen>
2. EVSTEP (2023), Tham khảo đề Vstep B2 có đáp án chuẩn xác nhất. Liên kết: <https://evstep.edu.vn/tham-khao-de-vstep-b2-co-dap-an-chuan-xac-nhat>
3. PREP, Cấu trúc đề thi VSTEP 4 kỹ năng: Reading, Listening, Writing, Speaking. Liên kết: <https://prepedu.com/vi/blog/cau-truc-de-thi-vstep>
4. VIBLO [2021], Tổng quan về MongoDB. Liên kết: <https://viblo.asia/p/tong-quan-ve-mongodb-EoDkQoxqGbV>
5. vietJack, Cơ bản về MongoDB. Liên kết: https://vietjack.com/mongodb/index.jsp

**Tài liệu là sách báo:**

1. Phan Thị Phương Nam (2015), Tài liệu giảng dạy môn Hệ Quản Trị CSDL (lưu hành nội bộ), Trường đại học Trà Vinh.
2. Hà Thị Thúy Vi (2013), Tài liệu giảng dạy môn CSDL (lưu hành nội bộ), Trường đại học Trà Vinh.

# PHỤ LỤC