Ở khóa nodejs thì mình dùng package mongoose, còn ở khóa mongdb thì mình dùng package mongdb và 2 package này có sự khác nhau như phía dưới. Mặc dù cả hai đều sử dụng cùng với node js

**MongoDB (Package mongodb)**

* **mongodb** là một package chính thức của MongoDB dành cho Node.js, cho phép bạn kết nối và tương tác trực tiếp với cơ sở dữ liệu MongoDB từ ứng dụng Node.js.
* **Cung cấp API cấp thấp**: Gói mongodb cung cấp các phương thức để thực hiện các thao tác CRUD (Create, Read, Update, Delete) như insertOne, find, updateOne, deleteOne, và nhiều hơn nữa. Các phương thức này tương ứng trực tiếp với các lệnh MongoDB mà bạn sẽ sử dụng trong MongoDB shell.
* **Tương tác trực tiếp**: Làm việc với gói mongodb nghĩa là bạn tương tác trực tiếp với MongoDB mà không có bất kỳ lớp trừu tượng nào khác. Điều này cung cấp cho bạn sự linh hoạt cao, nhưng cũng yêu cầu bạn phải quản lý các chi tiết như xác thực, quản lý kết nối, và thực hiện các truy vấn phức tạp một cách thủ công.
* **Sử dụng cho các dự án phức tạp**: Gói mongodb thường được sử dụng khi bạn cần kiểm soát chi tiết các hoạt động với MongoDB hoặc khi bạn đang làm việc với một dự án phức tạp đòi hỏi sự tinh chỉnh trong việc tương tác với cơ sở dữ liệu.

**Mongoose**

* **Mongoose** là một Object Data Modeling (ODM) library cho MongoDB và Node.js. Nó cung cấp một lớp trừu tượng trên MongoDB, cho phép bạn làm việc với MongoDB theo cách giống như làm việc với cơ sở dữ liệu quan hệ (SQL).
* **Schemas và Models**: Mongoose cho phép bạn định nghĩa các schema, từ đó tạo ra các model. Schema định nghĩa cấu trúc của các tài liệu trong MongoDB, bao gồm các loại dữ liệu, các quy tắc xác thực, và các phương thức tùy chỉnh. Model được tạo từ schema và đại diện cho một collection trong MongoDB.
* **Validation và Middleware**: Mongoose cung cấp các tính năng như xác thực dữ liệu tự động, middleware trước và sau khi lưu tài liệu, và các công cụ hữu ích khác để quản lý dữ liệu và logic nghiệp vụ.
* **Quan hệ dữ liệu**: Mặc dù MongoDB là một cơ sở dữ liệu NoSQL, Mongoose cung cấp các công cụ để quản lý mối quan hệ giữa các tài liệu, chẳng hạn như tham chiếu (reference) và nhúng tài liệu (embedding documen

**So sánh MongoDB (mongodb) và Mongoose**

| **Tiêu chí** | **MongoDB (Package mongodb)** | **Mongoose** |
| --- | --- | --- |
| **Mức độ trừu tượng** | Cấp thấp, tương tác trực tiếp với MongoDB. | Cấp cao, cung cấp một lớp trừu tượng trên MongoDB. |
| **Kiểm soát** | Cung cấp sự kiểm soát chi tiết và trực tiếp trên các thao tác với cơ sở dữ liệu. | Trừu tượng hóa nhiều chi tiết, tập trung vào cấu trúc dữ liệu và logic nghiệp vụ. |
| **Cấu trúc dữ liệu** | Không có schema, dữ liệu có thể linh hoạt. | Dùng schema để định nghĩa cấu trúc dữ liệu, giúp dễ dàng quản lý và xác thực. |
| **Validation** | Phải tự viết mã để xác thực dữ liệu. | Cung cấp sẵn tính năng xác thực dữ liệu theo schema. |
| **Middleware** | Không có khái niệm middleware. | Hỗ trợ middleware trước và sau các hoạt động như save, validate, remove. |
| **Quan hệ dữ liệu** | Phải quản lý thủ công các mối quan hệ giữa các tài liệu. | Hỗ trợ cơ chế tham chiếu và nhúng tài liệu để quản lý mối quan hệ giữa các tài liệu. |
| **Độ phức tạp** | Yêu cầu hiểu biết sâu hơn về MongoDB. | Dễ sử dụng hơn với các tính năng tích hợp sẵn. |
| **Hiệu suất** | Hiệu suất cao hơn do ít lớp trừu tượng. | Hiệu suất có thể thấp hơn một chút do có lớp trừu tượng, nhưng vẫn rất hiệu quả cho hầu hết các ứng dụng. |

**Khi nào nên sử dụng cái nào?**

* **Sử dụng mongodb**:
  + Khi bạn cần sự linh hoạt tối đa và muốn kiểm soát từng chi tiết của các truy vấn và thao tác với cơ sở dữ liệu.
  + Khi bạn đã quen thuộc với MongoDB và cần tối ưu hóa hiệu suất cho các dự án phức tạp.
* **Sử dụng Mongoose**:
  + Khi bạn muốn có một công cụ giúp quản lý dữ liệu một cách chặt chẽ với các quy tắc xác thực, schema rõ ràng, và các tính năng quản lý mối quan hệ dữ liệu.
  + Khi bạn đang phát triển các ứng dụng thông thường mà không cần sự phức tạp của việc tương tác trực tiếp với MongoDB.
  + Khi bạn muốn tiết kiệm thời gian bằng cách sử dụng các tính năng tiện lợi như middleware, virtuals, và methods của Mongoose.