UBND TỈNH BÌNH DƯƠNG

**TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT**

**BÁO CÁO GIỮA KÌ**

**ĐIỆN TOÁN ĐÁM MÂY**

**ĐỀ TÀI: TÌM HIỂU VỀ CƠ SỞ DỮ LIỆU ORIENTDB**

Giảng viên hướng dẫn: ThS. Nguyễn Đình Thọ

Nhóm : G12

Thành viên nhóm: Bùi Lâm Quý

Phạm Phong Hào

Đinh Thị Quỳnh

Nguyễn Minh Kha

Lớp: D17PM02

***Bình Dương 04/2021***

**MỤC LỤC**

[CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN VỀ NOSQL 1](#_Toc69643815)

[1.1. Cơ sở dữ liệu NoSQL là gì 1](#_Toc69643816)

[1.2. Cơ sở dữ liệu NoSQL (phi quan hệ) hoạt động như thế nào 1](#_Toc69643817)

[1.3. Vì sao bạn nên sử dụng cơ sở dữ liệu NoSQL 2](#_Toc69643818)

[1.4. Các loại cơ sở dữ liệu NoSQL 2](#_Toc69643819)

[1.5. Sự khác biệt giữa cơ sở dữ liệu SQL (quan hệ) với cơ sở dữ liệu NoSQL (phi quan hệ) 4](#_Toc69643820)

[CHƯƠNG 2. TÌM HIỂU VỀ ORIENTDB 8](#_Toc69643821)

[2.1. OrientDB là gì 8](#_Toc69643822)

[2.2. Lịch sử phát triển 8](#_Toc69643823)

[2.3. Các phiên bản và giấy phép 9](#_Toc69643824)

[2.4. Tính năng và tốc độ của OrientDB 9](#_Toc69643825)

[2.5. OrientDB hỗ trợ tốt cho doanh nghiệp 12](#_Toc69643826)

[CHƯƠNG 3. CÀI ĐẶT VÀ SỬ DỤNG 13](#_Toc69643827)

[3.1. Cài đặt JDK 13](#_Toc69643828)

[3.2. Cài đặt OrientDB 13](#_Toc69643829)

[TÀI LIỆU THAM KHẢO 17](#_Toc69643830)

**DANH MỤC HÌNH**

[**Hình 1.1:** Cơ sở dữ liệu NoSQl 1](#_Toc69643831)

[**Hình 1.2:** Các mô hình SQL và NoSQL database 7](#_Toc69643832)

[**Hình 2.1:** Cơ sở dữ liệu OrientDB 8](#_Toc69643833)

[**Hình 2.2:** Graph OrientDB 11](#_Toc69643834)

[**Hình 3.1:** Cài đặt JDK 13](#_Toc69643835)

[**Hình 3.2:** Tải OrientDB 14](#_Toc69643836)

[**Hình 3.3:** Cài đặt OrientDB 14](#_Toc69643837)

[**Hình 3.4:** File server.bat sau khi cấu hình thành công 15](#_Toc69643838)

[**Hình 3.5:** Giao diện trang chủ OrientDB 15](#_Toc69643839)

[**Hình 3.6:** Tạo database 16](#_Toc69643840)

[**Hình 3.7:** Giao diện database khi mới tạo 16](#_Toc69643841)

**LỜI MỞ ĐẦU**

Cơ sở dữ liệu đồ thị là một vấn đề mới mà chúng ta cần phải nghiên cứu

hiện nay. Đỉnh cao của ứng dụng cơ sở dữ liệu đồ thị là dùng để lưu trữ dữ liệu

của một hay nhiều mạng xã hội một cách dễ dàng và hiệu quả. Nếu như cơ sở dữ

liệu truyề thống biểu diễn bởi các bẳng và mối quan hệ giữa chúng thông qua khóa

ngoại, thì cơ sở dữ liệu đồ thị biểu diễn trực quan hơn bởi các node, thể hiện các

mối quan hệ khác nhau giữa các node, các cạnh, dựa trên các quan hệ đó để tính

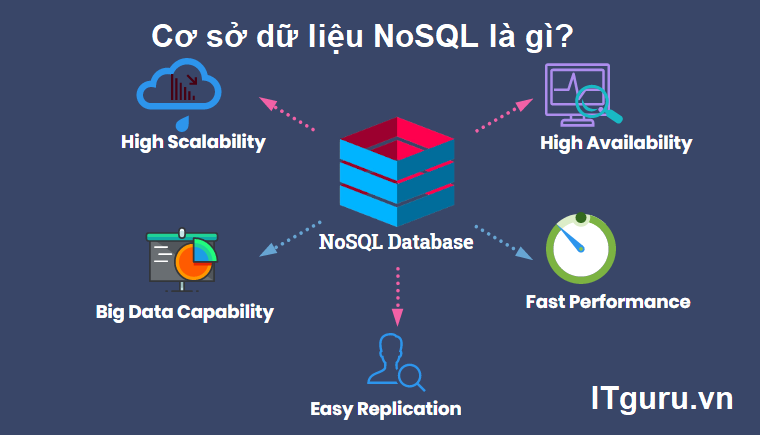
toán các lượi ích kinh tế khác. Có nhiều cơ sở dữ liệu đồ thị khác nhau, tuy nhiên

Các sản phẩm NoSQL thế hệ thứ 2 tiên tiến như OrientDB là tương lai: cung cấp nhiều chức năng và tính linh hoạt hơn, đồng thời đủ mạnh để thay thế DBMS đang hoạt động của bạn.

# GIỚI THIỆU TỔNG QUAN VỀ NOSQL

## Cơ sở dữ liệu NoSQL là gì

Cơ sở dữ liệu NoSQL là Cơ sở dữ liệu được xây dựng dành riêng cho mô hình dữ liệu và có sơ đồ linh hoạt để xây dựng các ứng dụng hiện đại. Cơ sở dữ liệu NoSQL được công nhận rộng rãi vì khả năng dễ phát triển, chức năng cũng như hiệu năng ở quy mô lớn. Trang này có các tài nguyên giúp bạn hiểu thêm về cơ sở dữ liệu NoSQL và bắt đầu sử dụng.



* + - * 1. Cơ sở dữ liệu NoSQl

## Cơ sở dữ liệu NoSQL (phi quan hệ) hoạt động như thế nào

Cơ sở dữ liệu NoSQL sử dụng nhiều mô hình dữ liệu để truy cập và quản lý dữ liệu. Các loại cơ sở dữ liệu này được tối ưu hóa dành riêng cho các ứng dụng yêu cầu mô hình dữ liệu linh hoạt có lượng dữ liệu lớn và độ trễ thấp, có thể đạt được bằng cách giảm bớt một số hạn chế về tính nhất quán của dữ liệu của các cơ sở dữ liệu khác.

Ví dụ:

* Trong cơ sở dữ liệu quan hệ, hồ sơ về một cuốn sách thường được phân tách (hay còn gọi là "chuẩn hóa") và lưu trữ trong các bảng tách biệt nhau, còn mối quan hệ được quy định bằng các ràng buộc khóa ngoại và khóa chính. Trong ví dụ này, bảng Sách có các cột cho ISBN, Tên sách và Số phiên bản, bảng Tác giả có các cột cho ID tác giả và Tên tác giả và cuối cùng, bảng Tác giả–ISBN có các cột cho ID tác giả và ISBN. Mô hình quan hệ được thiết kế để cho phép các cơ sở dữ liệu này thực thi tính toàn vẹn tham chiếu giữa nhiều bảng trong cơ sở dữ liệu, được chuẩn hóa để giảm dư thừa và thường được tối ưu hóa cho mục đích lưu trữ.
* Trong cơ sở dữ liệu NoSQL, hồ sơ về một cuốn sách thường được lưu trữ dưới dạng văn bản JSON. Với từng quyển sách, mục, ISBN, Tên sách, Số phiên bản, Tên tác giả và ID tác giả được lưu trữ dưới dạng thuộc tính trong một văn bản duy nhất. Trong mô hình này, dữ liệu được tối ưu hóa cho việc phát triển trực quan và khả năng thay đổi quy mô theo chiều ngang.

## Vì sao bạn nên sử dụng cơ sở dữ liệu NoSQL

Cơ sở dữ liệu NoSQL là lựa chọn cực kỳ thích hợp cho nhiều ứng dụng hiện đại, ví dụ như di động, web và trò chơi đòi hỏi phải sử dụng cơ sở dữ liệu cực kỳ thiết thực, linh hoạt, có khả năng thay đổi quy mô và hiệu năng cao để đem đến cho người dùng trải nghiệm tuyệt vời.

* **Linh hoạt**: Cơ sở dữ liệu NoSQL thường cung cấp các sơ đồ linh hoạt giúp công đoạn phát triển nhanh hơn và có khả năng lặp lại cao hơn. Mô hình dữ liệu linh hoạt biến cơ sở dữ liệu NoSQL thành lựa chọn lý tưởng cho dữ liệu không được tổ chức thành cấu trúc hoặc có cấu trúc chưa hoàn chỉnh.
* **Khả năng thay đổi quy mô**: Cơ sở dữ liệu NoSQL thường được thiết kế để tăng quy mô bằng cách sử dụng các cụm phần cứng được phân phối thay vì tăng quy mô bằng cách bổ sung máy chủ mạnh và tốn kém. Một số nhà cung cấp dịch vụ đám mây xử lý các hoạt động này một cách không công khai dưới dạng dịch vụ được quản lý đầy đủ.
* **Hiệu năng cao**: Cơ sở dữ liệu NoSQL được tối ưu hóa theo các mô hình dữ liệu cụ thể và các mẫu truy cập giúp tăng hiệu năng cao hơn so với việc cố gắng đạt được mức độ chức năng tương tự bằng cơ sở dữ liệu quan hệ.
* **Cực kỳ thiết thực**: Cơ sở dữ liệu NoSQL cung cấp các API và kiểu dữ liệu cực kỳ thiết thực được xây dựng riêng cho từng mô hình dữ liệu tương ứng.

## Các loại cơ sở dữ liệu NoSQL

**Khóa–giá trị**: Cơ sở dữ liệu khóa–giá trị có khả năng phân mảnh cao và cho phép thay đổi quy mô theo chiều ngang ở các quy mô lớn mà các loại hình cơ sở dữ liệu khác không thể làm được. Mô hình dữ liệu khóa–giá trị được dùng cho trò chơi, công nghệ quảng cáo và đặc biệt thích hợp cho IoT. Amazon DynamoDB được thiết kế để có độ trễ ổn định chỉ vài mili giây cho khối lượng công việc thuộc quy mô bất kỳ. Hiệu năng ổn định này là lý do chính để di chuyển tính năng Stories của Snapchat (kể cả khối lượng công việc ghi lưu trữ lớn nhất của Snapchat) sang DynamoDB.

**Tài liệu**: Trong mã ứng dụng, dữ liệu thường được biểu diễn dưới dạng một đối tượng hoặc văn bản dạng JSON vì đây là mô hình dữ liệu hiệu quả và trực quan cho các nhà phát triển. Cơ sở dữ liệu văn bản giúp nhà phát triển dễ dàng lưu trữ và truy vấn dữ liệu trong một cơ sở dữ liệu bằng cách sử dụng cùng một định dạng mô hình văn bản họ sử dụng trong mã ứng dụng của mình. Tính chất linh hoạt, bán cấu trúc và phân cấp của các văn bản và cơ sở dữ liệu văn bản cho phép chúng phát triển phù hợp với yêu cầu của ứng dụng. Mô hình văn bản phát huy hiệu quả với danh mục, hồ sơ người dùng và hệ thống quản lý nội dung, nơi từng văn bản là duy nhất và phát triển theo thời gian. Amazon DocumentDB (với khả năng tương thích MongoDB) và MongoDB là các cơ sở dữ liệu văn bản phổ biến cung cấp các API mạnh mẽ và trực quan để phát triển linh hoạt và lặp lại.

**Đồ thị**: Mục đích của cơ sở dữ liệu đồ thị là giúp việc dựng và chạy ứng dụng hoạt động với các bộ dữ liệu có khả năng kết nối cao trở nên dễ dàng. Cơ sở dữ liệu đồ thị thường được sử dụng cho các đồ thị tri thức, mạng xã hội, công cụ đề xuất và phát hiện lừa đảo. Amazon Neptune là dịch vụ cơ sở dữ liệu đồ thị được quản lý đầy đủ. Neptune hỗ trợ cả mô hình Đồ thị thuộc tính lẫn Framework mô tả tài nguyên (RDF), cung cấp cho người dùng lựa chọn hai API đồ thị: TinkerPop và RDF/SPARQL. Các cơ sở dữ liệu đồ thị phổ biến gồm có Neo4j và Giraph.

**Trong bộ nhớ**: Ứng dụng trò chơi và công nghệ quảng cáo đã sử dụng loại Cơ sở dữ liệu này trong nhiều trường hợp như bảng xếp hạng, kho lưu trữ phiên và phân tích thời gian thực yêu cầu thời gian phản hồi đạt mức micro giây và lưu lượng có thể đạt mức đỉnh điểm vào bất kỳ thời điểm nào. Amazon ElastiCache cung cấp Memcached và Redis, để phục vụ các khối lượng công việc có độ trễ thấp và thông lượng cao, chẳng hạn như không thể đáp ứng nhu cầu của McDonald’s bằng kho dữ liệu trên ổ đĩa. Amazon DynamoDB Accelerator (DAX) là ví dụ khác về kho dữ liệu được xây dựng chuyên dụng. DAX giúp DynamoDB đọc yêu cầu cường độ nhanh hơn.

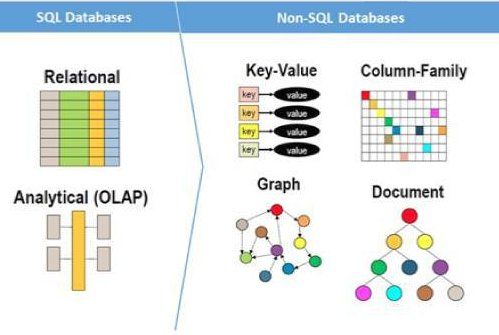
**Tìm kiếm**: Nhiều ứng dụng xuất nhật ký để giúp nhà phát triển khắc phục sự cố. Amazon Elasticsearch Service (Amazon ES) là Cơ sở dữ liệu được xây dựng chuyên dụng để cung cấp khả năng ảo hóa và phân tích dữ liệu do máy tạo gần như theo thời gian thực bằng cách tạo chỉ mục, tập hợp và tìm kiếm nhật ký và số liệu có cấu trúc chưa hoàn chỉnh. Amazon ES còn là công cụ tìm kiếm mạnh mẽ và có hiệu năng cao dành cho các trường hợp sử dụng tìm kiếm toàn văn bản. Expedia đang sử dụng trên 150 miền Amazon ES, 30 TB dữ liệu và 30 tỷ văn bản cho nhiều chức năng đóng vai trò quan trọng, từ theo dõi và khắc phục sự cố trong khâu vận hành đến tối ưu hóa quá trình theo dấu và định giá chồng ứng dụng được phân phối.

## Sự khác biệt giữa cơ sở dữ liệu SQL (quan hệ) với cơ sở dữ liệu NoSQL (phi quan hệ)

Trong nhiều thập kỷ, mô hình dữ liệu chiếm ưu thế về lượng sử dụng để phát triển ứng dụng là mô hình dữ liệu quan hệ được sử dụng trong các cơ sở dữ liệu quan hệ, ví dụ như Oracle, DB2, SQL Server, MySQL và PostgreSQL. Mãi cho đến cuối những năm 2000, các mô hình dữ liệu khác mới bắt đầu được đưa vào áp dụng và sử dụng nhiều hơn. Để phân biệt và phân loại các cơ sở dữ liệu và mô hình dữ liệu mới này, từ "NoSQL" đã được đặt ra. Thông thường, thuật ngữ "NoSQL" được sử dụng tương đương với "phi quan hệ".

Mặc dù có nhiều loại cơ sở dữ liệu NoSQL với nhiều tính năng đa dạng, bảng sau sẽ minh họa một số khác biệt giữa cơ sở dữ liệu SQL và NoSQL.

|  |  |  |
| --- | --- | --- |
|  | Cơ sở dữ liệu quan hệ | Cơ sở dữ liệu NoSQL |
| **Khối lượng công việc tối ưu** | Cơ sở dữ liệu quan hệ được thiết kế dành cho các ứng dụng xử lý giao dịch trực tuyến (OLTP) trong giao dịch có độ ổn định cao và thích hợp để xử lí phân tích trực tuyến (OLAP). | Các cơ sở dữ liệu NoSQL được thiết kế cho các mẫu truy cập dữ liệu, bao gồm các ứng dụng có độ trễ thấp. Cơ sở dữ liệu tìm kiếm NoSQL được thiết kế để phục vụ phân tích dữ liệu có cấu trúc chưa hoàn chỉnh. |
| **Mô hình dữ liệu** | Mô hình quan hệ chuẩn hóa dữ liệu vào bảng được hình thành từ hàng và cột. Sơ đồ quy định rõ ràng bảng, hàng, cột, chỉ mục, mối quan hệ giữa các bảng và các thành tố cơ sở dữ liệu khác. Cơ sở dữ liệu sẽ thực thi tính toàn vẹn tham chiếu trong mối quan hệ giữa các bảng. | Các cơ sở dữ liệu NoSQL cung cấp nhiều mô hình dữ liệu khác nhau như khóa-giá trị, tài liệu và biểu đồ, được tối ưu hóa để đạt hiệu năng và quy mô tối ưu. |
| **Thuộc tính ACID** | Cơ sở dữ liệu quan hệ có các thuộc tính mang tính nguyên tố, nhất quán, tách biệt và bền vững (ACID):   * Tính nguyên tố đòi hỏi giao dịch phải được thực thi đầy đủ hoặc hoàn toàn không thực hiện. * Tính nhất quán đòi hỏi rằng khi giao dịch được thực hiện, dữ liệu phải nhất quán với sơ đồ cơ sở dữ liệu. * Tính tách biệt đòi hỏi rằng các giao dịch diễn ra đồng thời phải được thực thi tách biệt với nhau. * Tính bền vững đòi hỏi phải có khả năng phục hồi từ tình trạng hư hỏng hệ thống hoặc mất điện đột ngột về trạng thái đã biết cuối cùng. | Cơ sở dữ liệu NoSQL thường phải đánh đổi bằng cách nới lỏng một số thuộc tính ACID này của cơ sở dữ liệu quan hệ để có mô hình dữ liệu linh hoạt hơn có khả năng thay đổi quy mô theo chiều ngang. Việc này biến các cơ sở dữ liệu NoSQL thành lựa chọn tuyệt vời cho các trường hợp sử dụng cần thông lượng cao, độ trễ thấp cần thay đổi quy mô theo chiều ngang vượt qua giới hạn của một phiên bản duy nhất. |
| **Hiệu năng** | Hiệu năng thường phụ thuộc vào hệ thống con của ổ đĩa. Thông thường, việc tối ưu hóa các truy vấn, chỉ mục và cấu trúc bảng bắt buộc phải được thực hiện để đạt mức hiệu năng tối đa. | Hiệu năng thường được xem là chức năng của kích cỡ cụm phần cứng ngầm, độ trễ mạng và ứng dụng đưa ra lệnh gọi. |
| **Quy mô** | Cơ sở dữ liệu quan hệ thường tăng quy mô bằng cách tăng năng lực điện toán của phần cứng hoặc tăng quy mô bằng cách thêm bản sao của khối lượng công việc chỉ đọc. | Cơ sở dữ liệu NoSQL thường có tính phân mảnh cao do các mẫu truy cập khóa-giá trị có khả năng tăng quy mô bằng cách sử dụng kiến trúc được phân phối để tăng thông lượng, đem đến hiệu năng ổn định với quy mô gần như không giới hạn. |
| **API** | Yêu cầu lưu trữ và truy xuất dữ liệu được truyền đạt bằng cách sử dụng các truy vấn nhất quán với ngôn ngữ truy vấn có cấu trúc (SQL). Các truy vấn này được phân tích và thực thi bởi cơ sở dữ liệu quan hệ. | API trên cơ sở đối tượng cho phép các nhà phát triển ứng dụng dễ dàng lưu trữ và truy xuất cấu trúc dữ liệu trong bộ nhớ. Khóa phân mảnh tìm kiếm các cặp khóa-giá trị, tập hợp cột hoặc văn bản có cấu trúc chưa hoàn chỉnh có chứa đối tượng và thuộc tính của ứng dụng được xếp theo chuỗi. |



* + - * 1. Các mô hình SQL và NoSQL database

# TÌM HIỂU VỀ ORIENTDB

## OrientDB là gì

OrientDB là một hệ quản trị cơ sở dữ liệu NoSQLmã nguồn mở được viết bằng Java . Đây là cơ sở dữ liệu đa mô hình , hỗ trợ các mô hình graph , document , key / value và object , nhưng các mối quan hệ được quản lý như trong cơ sở dữ liệu đồ thị với các kết nối trực tiếp giữa các bản ghi. Nó hỗ trợ các chế độ giản đồ, lược đồ đầy đủ và lược đồ hỗn hợp. Nó có một hệ thống cấu hình bảo mật mạnh mẽ dựa trên người dùng và vai trò, đồng thời hỗ trợ truy vấn với Gremlin cùng với SQL được mở rộng để duyệt đồ thị. OrientDB sử dụng một số cơ chế lập chỉ mục dựa trên B-tree và Băm mở rộng , cơ chế cuối cùng được gọi là "chỉ mục băm", có kế hoạch triển khai LSM-tree và chỉ mục dựa trên cây Fractal . Mỗi bản ghi có Phím thay thế cho biết vị trí của bản ghi bên trong Danh sách mảng , liên kết giữa các bản ghi được lưu trữ dưới dạng giá trị đơn lẻ của vị trí bản ghi được lưu trữ bên trong liên kết giới thiệu hoặc B- cây vị trí bản ghi (còn gọi là ID bản ghi hoặc RID) cho phép duyệt nhanh (với độ phức tạp O (1) ) của các mối quan hệ một-nhiều và nhanh chóng thêm / xóa các liên kết mới. OrientDB là cơ sở dữ liệu đồ thị phổ biến thứ ba theo xếp hạng cơ sở dữ liệu đồ thị của DB-Engines , tính đến tháng 9 năm 2017.



* + - * 1. Cơ sở dữ liệu OrientDB

## Lịch sử phát triển

OrientDB ban đầu được tác giả vào năm 2010. Luca đã viết nó như một bản viết lại Java của lớp liên tục nhanh của cơ sở dữ liệu Orient ODBMS (ban đầu được phát triển vào năm 1999), ban đầu được viết bằng C ++. Trong suốt năm 2012–2014, công cụ lưu trữ đã được Andrey Lomakin phát triển lại. Nó đã có một tên mới "plocal" viết tắt của "địa phương được phân trang". Tên này ngụ ý rằng công cụ lưu trữ mới dựa trên khái niệm về việc chia nhỏ các tệp dữ liệu theo các trang và trang được coi như một đơn vị nguyên tử duy nhất của sự thay đổi. Kể từ năm 2012, dự án được tài trợ bởi (trước đây là Orient Technologies LTD), một công ty hoạt động vì lợi nhuận với Luca là Giám đốc điều hành và người sáng lập. Vào năm 2013, Andrey Lomakin đã gia nhập công ty với tư cách là kỹ sư trưởng R&D và đồng sở hữu của công ty.

Lần đầu tiên từ "đa mô hình" được liên kết với cơ sở dữ liệu là vào ngày 30 tháng 5 năm 2012 tại Cologne, Đức , trong bài phát biểu quan trọng của Luca Garulli "Áp dụng NoSQL - Bước tiếp theo là gì?". Luca Garulli đã hình dung ra sự phát triển của các sản phẩm NoSQL thế hệ 1 thành các sản phẩm mới với nhiều tính năng hơn có thể được sử dụng bởi nhiều trường hợp sử dụng. OrientDB là sản phẩm đầu tiên bao gồm các tài liệu, biểu đồ, khóa-giá trị, không gian địa lý và mô hình phản ứng trong cùng một sản phẩm, ở cấp cốt lõi. Điều này có nghĩa là nhiều mô hình đã được tích hợp vào lõi mà không cần sử dụng các lớp. Vì lý do này, OrientDB là cơ sở dữ liệu đa mô hình "Native" .

OrientDB đã được các phương tiện truyền thông đưa tin và là người chiến thắng giải thưởng InfoWorld Bossie năm 2015.

Vào ngày 15 tháng 9 năm 2017, Công ty OrientDB LTD đã được mua lại bởi CallidusCloud một công ty đại chúng được giao dịch trên NASDAQ .

Vào ngày 30 tháng 1 năm 2018, SAP (company) đã mua lại CallidusCloud với giá 2,4 tỷ đô la . và do đó OrientDB hiện được hỗ trợ bởi SAP (company) .Wikipedia site:vi.wikiarabi.org

## Các phiên bản và giấy phép

OrientDB Community Edition miễn phí cho mọi hoạt động sử dụng (giấy phép Apache 2). Phần mềm nguồn mở được xây dựng dựa trên cộng đồng các nhà phát triển. Các tính năng như chia tỷ lệ theo chiều ngang, khả năng chịu lỗi, phân cụm, sharding và sao chép không bị vô hiệu hóa trong OrientDB Community Edition.

OrientDB Enterprise Edition là phần mở rộng thương mại của OrientDB Community Edition được tạo ra để xử lý các trường hợp sử dụng mạnh mẽ và khắt khe hơn. OrientDB Enterprise Edition bao gồm các tính năng bổ sung như trình biên dịch truy vấn, cấu hình phân cụm phân tán, ghi số liệu, màn hình trực tiếp, Teleporter (công cụ di chuyển) và cảnh báo có thể định cấu hình.

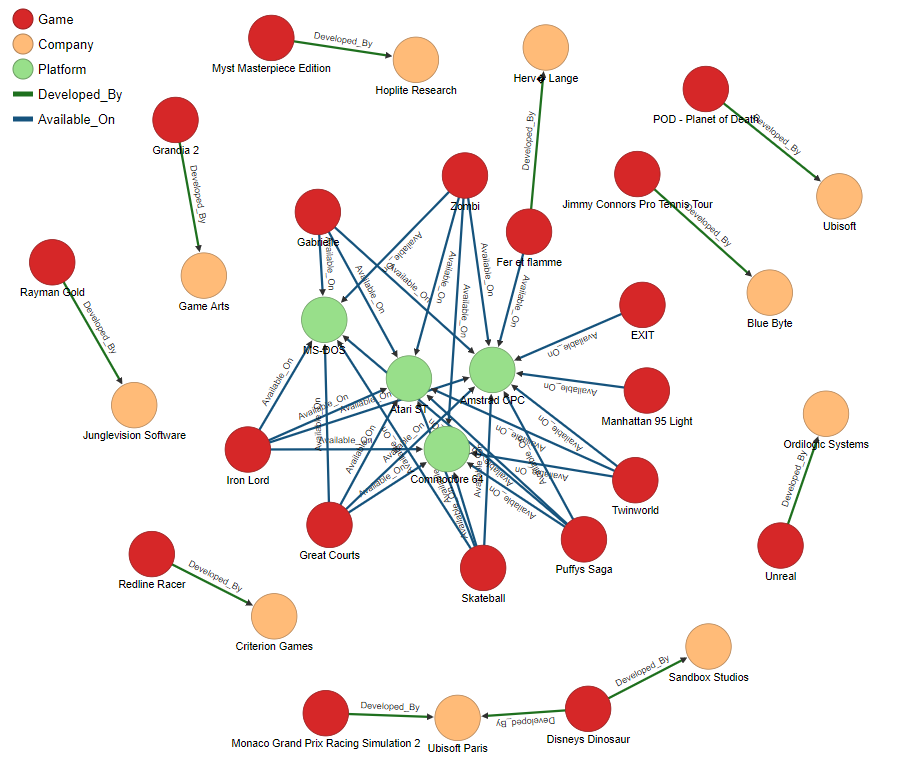
## Tính năng và tốc độ của OrientDB

Tính năng

* Cài đặt nhanh. OrientDB có thể được cài đặt và chạy trong vòng chưa đầy 60 giây
* Hoàn toàn giao dịch: hỗ trợ các giao dịch ACID đảm bảo rằng tất cả các giao dịch cơ sở dữ liệu được xử lý đáng tin cậy và trong trường hợp có sự cố, tất cả các tài liệu đang chờ xử lý sẽ được phục hồi và cam kết.
* Mô hình dữ liệu có cấu trúc đồ thị : quản lý nguyên bản đồ thị. Hoàn toàn tuân thủ khung máy tính đồ thị mã nguồn mở Apache TinkerPop Gremlin (trước đây được gọi là Blueprints).
* SQL : hỗ trợ các truy vấn SQL có phần mở rộng để xử lý các mối quan hệ không có SQL join , quản lý cây và đồ thị của các tài liệu được kết nối.
* Công nghệ web: hỗ trợ nguyên bản giao thức HTTP , RESTful và JSON các thư viện hoặc thành phần bổ sung .
* Distributed: hỗ trợ đầy đủ cho sao chép đa tổng thể bao gồm các cụm phân tán theo địa lý.
* Chạy mọi nơi: được triển khai bằng cách sử dụng Java thuần túy cho phép nó chạy trên Linux , OS X , Windows hoặc bất kỳ hệ thống nào có chế độ cục bộ JVM .
* Embeddable: để sử dụng cơ sở dữ liệu bỏ qua Người phục vụ. Hoàn hảo cho các trường hợp cơ sở dữ liệu được nhúng.
* Giấy phép Apache 2 : luôn miễn phí cho mọi trường hợp sử dụng. Không cần phí hoặc tiền bản quyền để sử dụng nó.
* Máy chủ đầy đủ có dung lượng khoảng 512 MB.
* Hỗ trợ thương mại có sẵn từ OrientDB.
* Đối sánh mẫu: Được giới thiệu trong phiên bản 2.2, câu lệnh Match truy vấn cơ sở dữ liệu theo cách khai báo, sử dụng đối sánh mẫu.
* Các tính năng bảo mật được giới thiệu trong OrientDB 2.2 cung cấp một khuôn khổ có thể mở rộng để thêm trình xác thực bên ngoài, xác thực mật khẩu, nhập LDAP của vai trò và người dùng cơ sở dữ liệu, khả năng kiểm tra nâng cao và nhật ký hệ thống ủng hộ. OrientDB Enterprise Edition cung cấp hỗ trợ Kerberos (giao thức) xác thực trình duyệt đầy đủ SPNEGO . Khi nói đến mã hóa cơ sở dữ liệu, bắt đầu từ phiên bản 2.2, OrientDB có thể mã hóa các bản ghi trên đĩa. Điều này ngăn người dùng trái phép truy cập nội dung cơ sở dữ liệu hoặc thậm chí bỏ qua bảo mật của OrientDB.
* Teleporter: Cho phép nhanh chóng nhập cơ sở dữ liệu quan hệ vào OrientDB trong vài bước đơn giản.
* Sẵn sàng cho đám mây: Có thể triển khai OrientDB trên đám mây và hỗ trợ các nhà cung cấp sau: Amazon Web Services, Microsoft Azure, CenturyLink Cloud, Jelastic, DigitalOceanWikipedia site:vi.wikiarabi.org

Tốc độ

* OrientDB được thiết kế ngay từ đầu với hiệu suất là đặc điểm kỹ thuật quan trọng. Nó nhanh trên cả thao tác đọc và ghi. Lưu trữ lên đến 120.000 bản ghi mỗi giây
* Không tham gia nữa: các mối quan hệ là các liên kết vật lý đến các bản ghi.
* Sử dụng RAM tốt hơn.
* Duyệt qua các phần của hoặc toàn bộ cây và biểu đồ của bản ghi trong mili giây.
* Tốc độ duyệt không bị ảnh hưởng bởi kích thước cơ sở dữ liệu.



* + - * 1. Graph OrientDB

## OrientDB hỗ trợ tốt cho doanh nghiệp

Trong khi hầu hết các DBMS NoSQL được sử dụng làm cơ sở dữ liệu thứ cấp, thì OrientDB đủ mạnh và linh hoạt để được sử dụng như một DBMS hoạt động. Mặc dù OrientDB Community Edition miễn phí cho mục đích sử dụng thương mại, các ứng dụng mạnh mẽ cần có các chức năng cấp doanh nghiệp để đảm bảo an ninh dữ liệu và hiệu suất hoàn hảo. OrientDB Enterprise Edition cung cấp cho bạn tất cả các tính năng của phiên bản cộng đồng của chúng tôi cùng với:

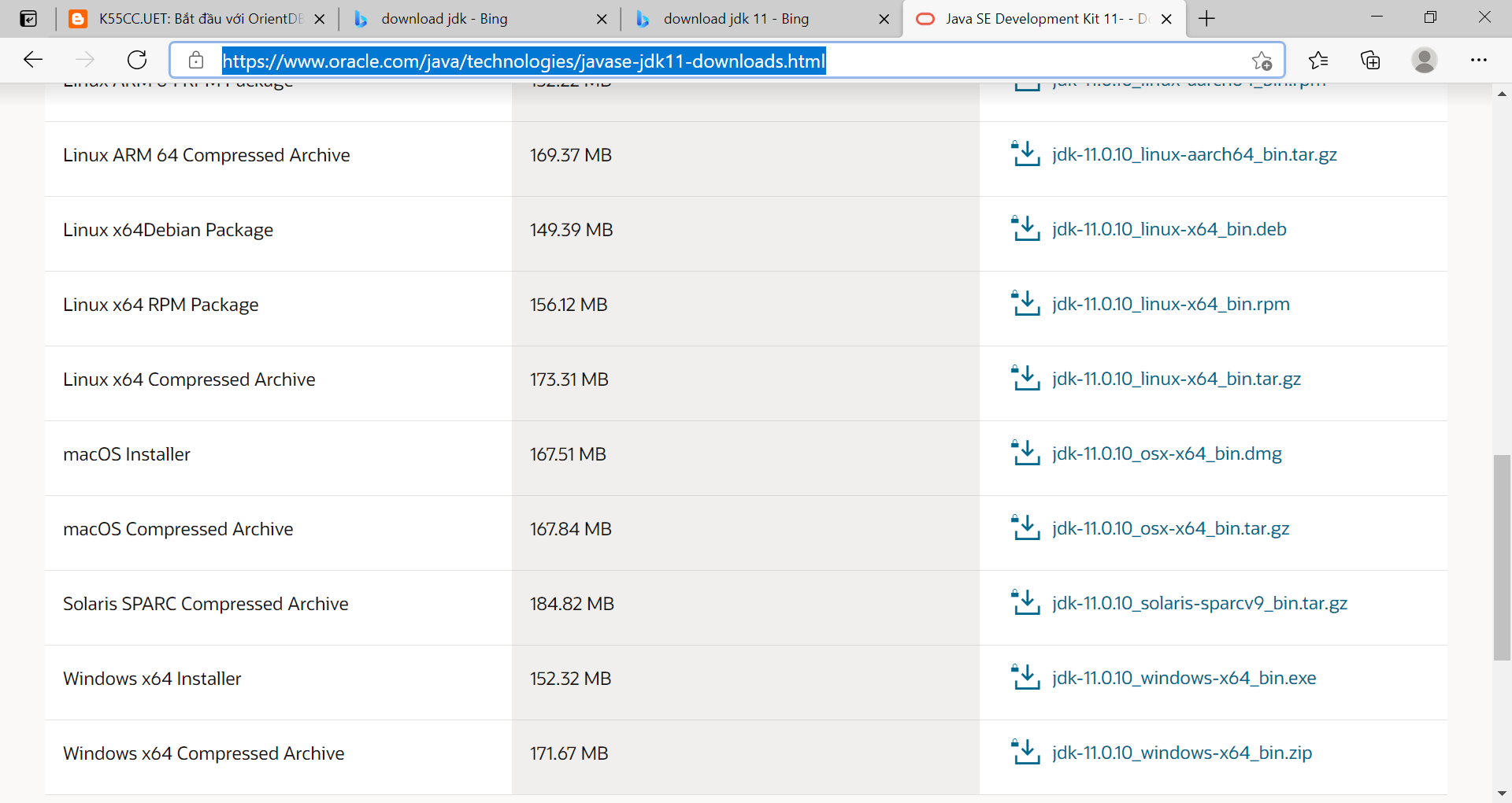
* Sao lưu tăng dần
* Bảo mật chưa từng có
* Hỗ trợ 24x7
* Hồ sơ truy vấn
* Cấu hình phân cụm phân tán
* Ghi lại số liệu
* Màn hình trực tiếp với các cảnh báo có thể định cấu hình

# CÀI ĐẶT VÀ SỬ DỤNG

## Cài đặt JDK

Có khá nhiều phiên bản JDK và OrientDB yêu cầu phiên bản JDK thấp nhất là phiên bản 1.6

Có thể cài đặt JDK phiên bản 11 tại website (<https://www.oracle.com/java/technologies/javase-jdk11-downloads.html>)

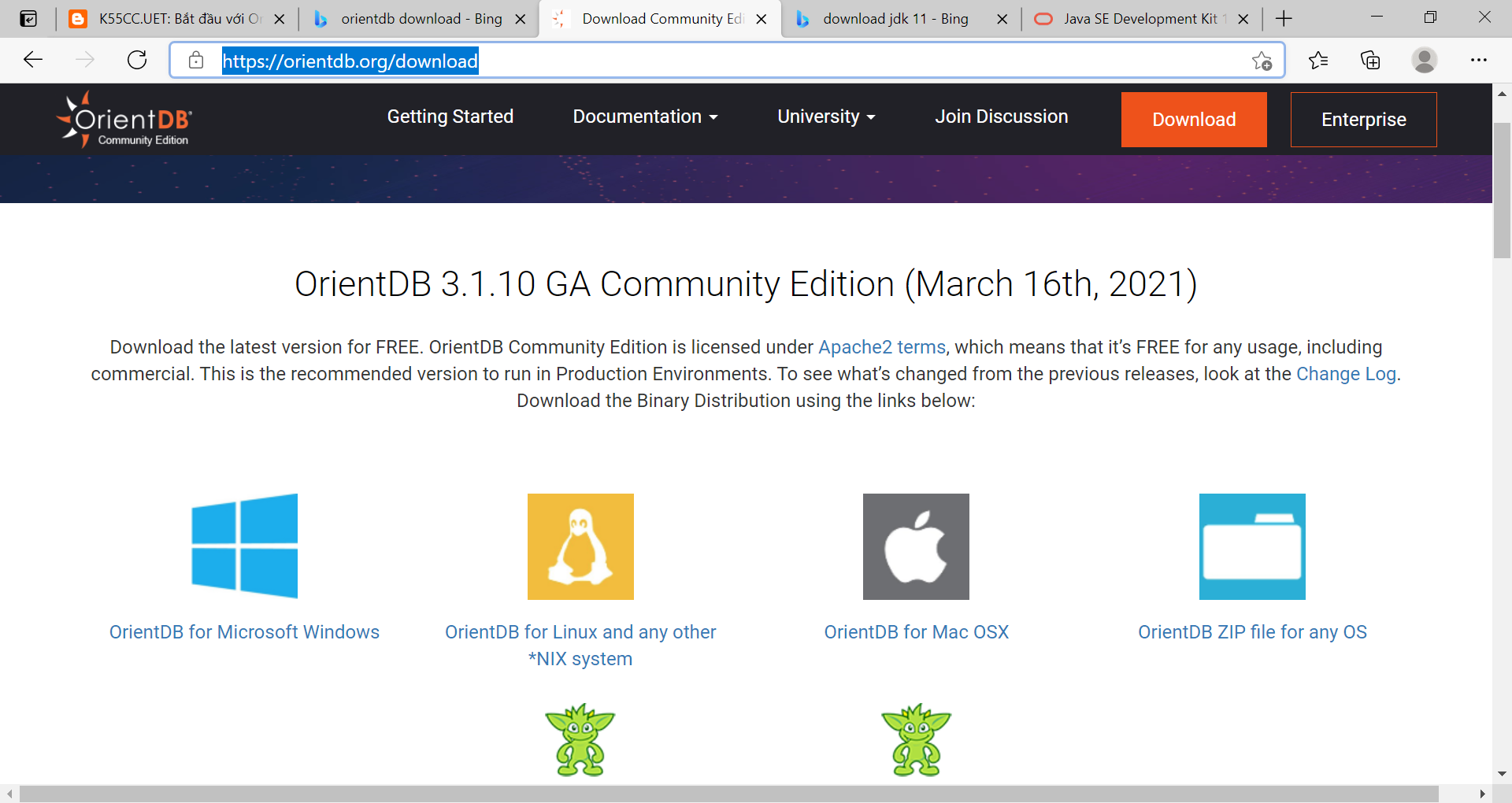


* + - * 1. Cài đặt JDK

Sau khi cài đặt JDK chúng ta hãy bắt đầu đến OrientDB

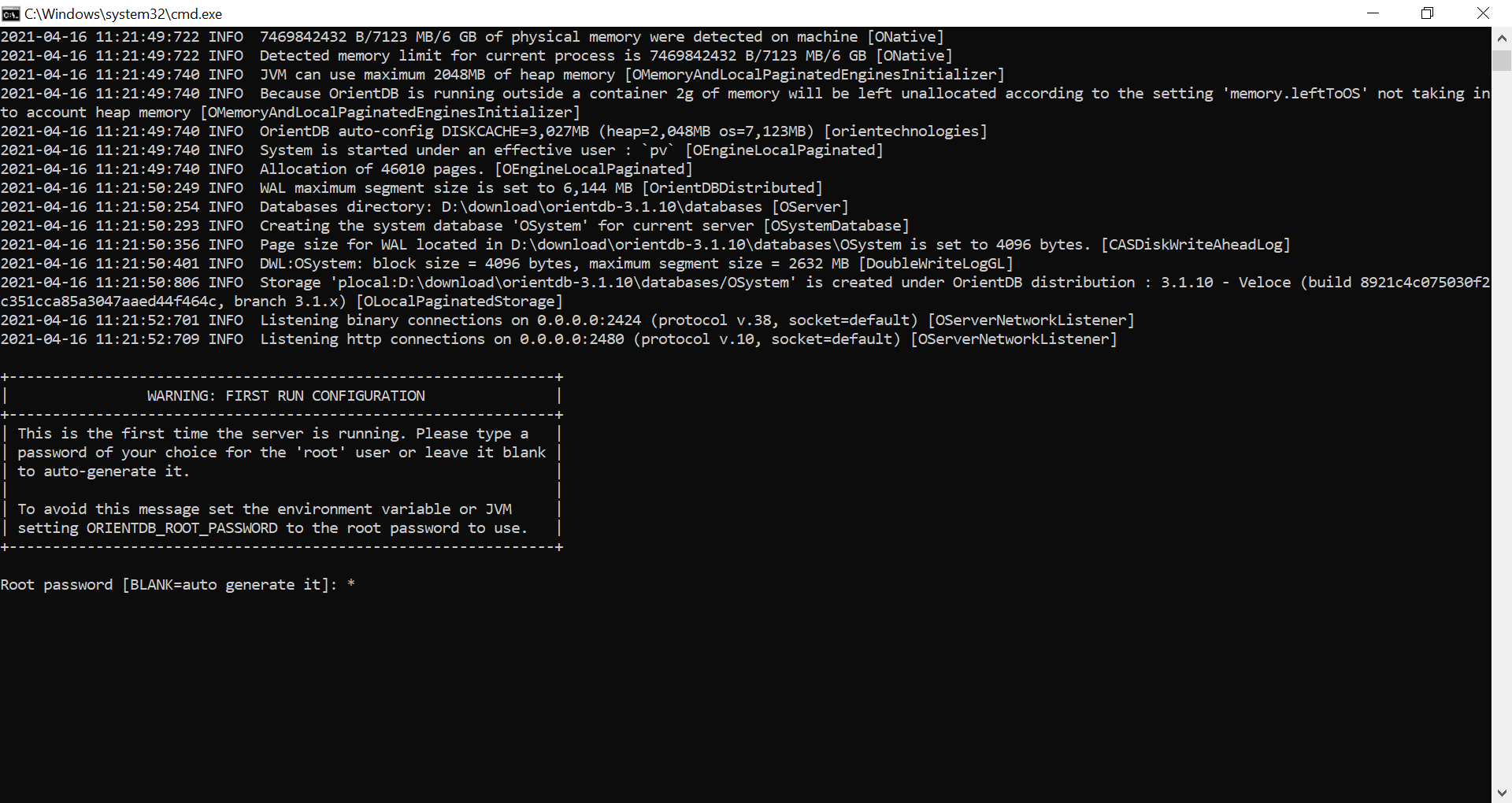
## Cài đặt OrientDB

Ta có thể tải phiên bản OrientDB mới nhất tại (<https://orientdb.org/download>)



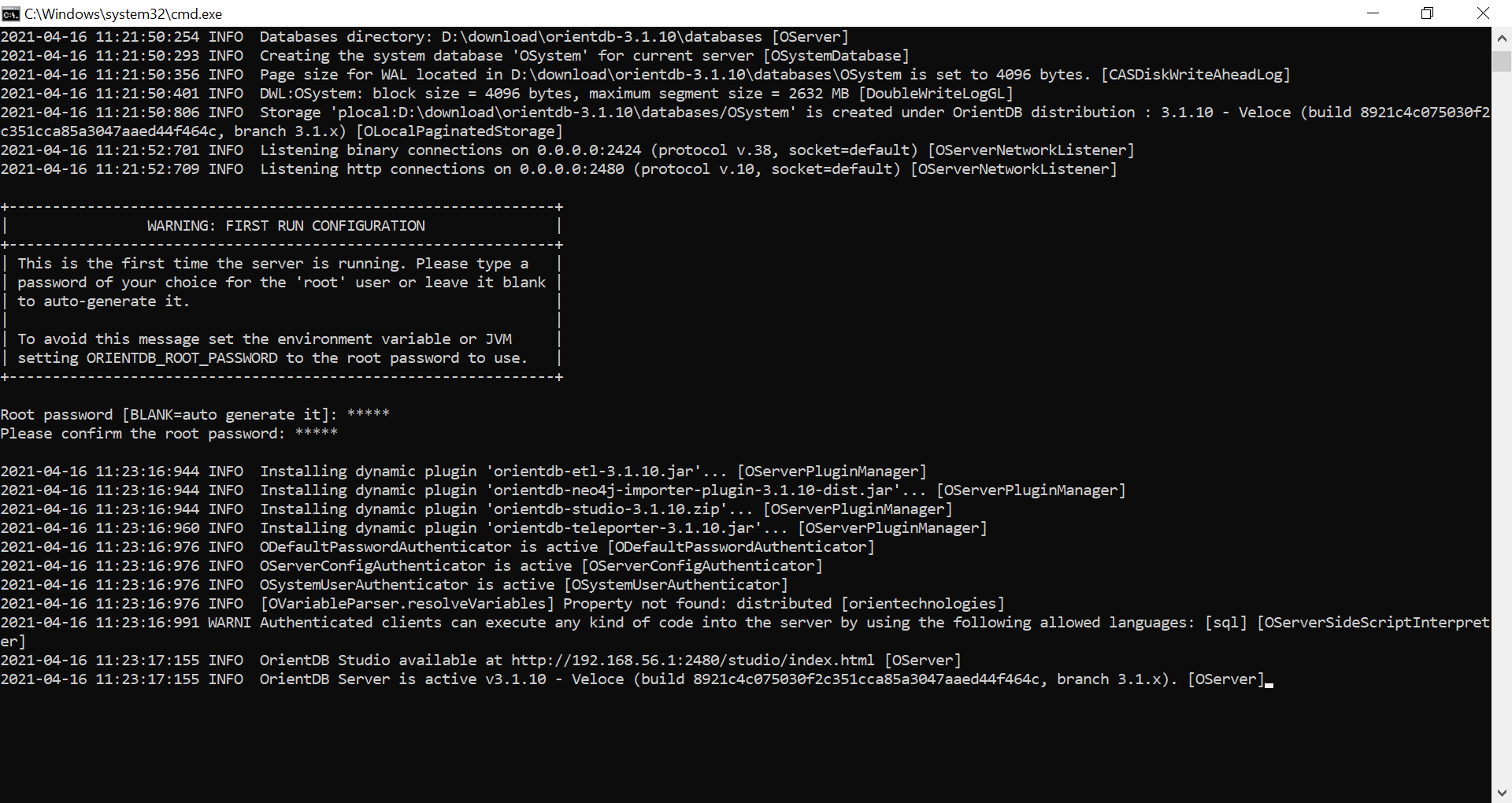
* + - * 1. Tải OrientDB

Sau khi tải về ta giải nén file và mở thư mục lên vào thư mục bin và mở file server.bat cho windown và server.sh cho linux



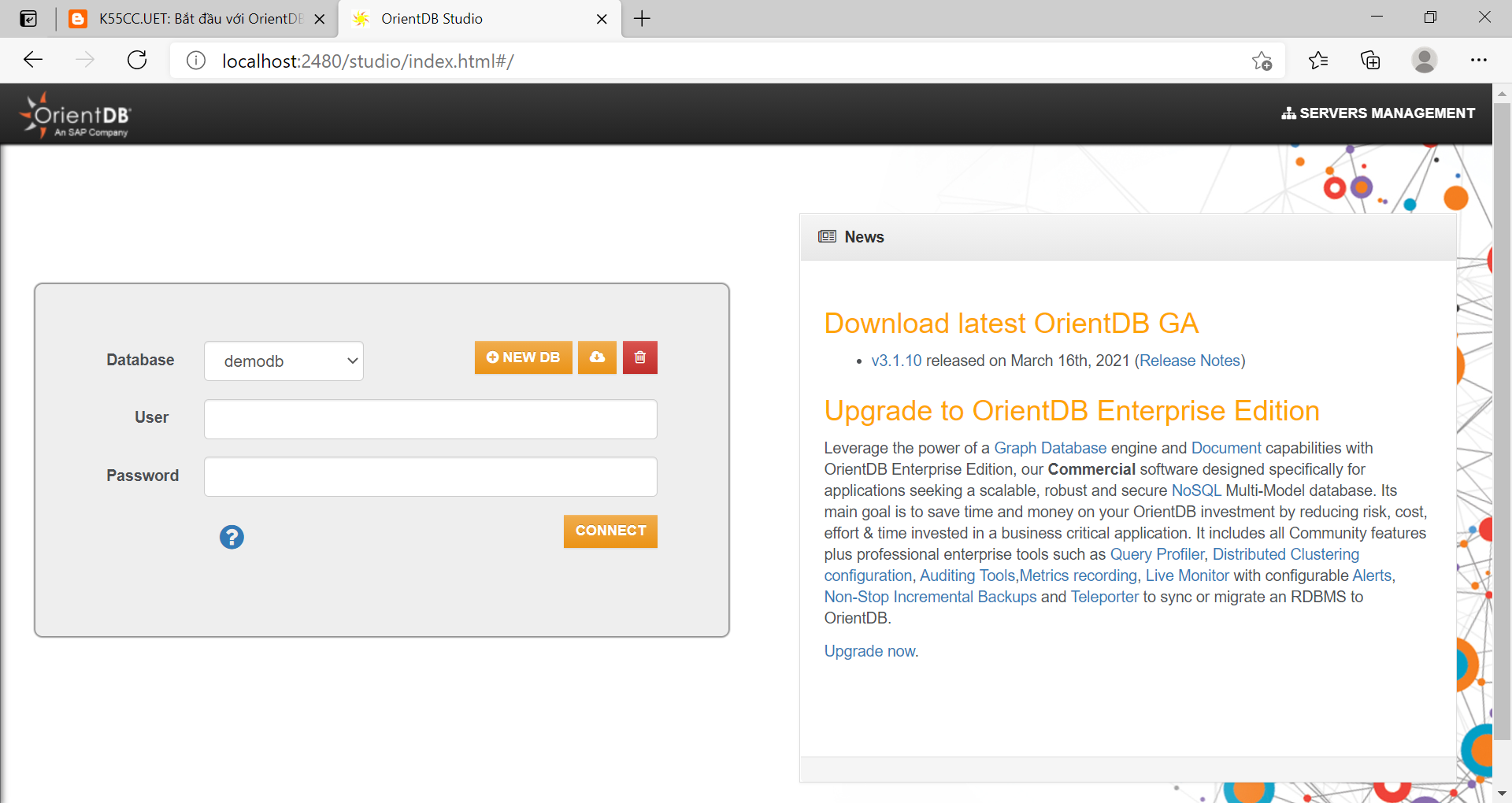
* + - * 1. Cài đặt OrientDB

Sau khi mở file chạy xong chúng ta sẽ nhập mật khẩu



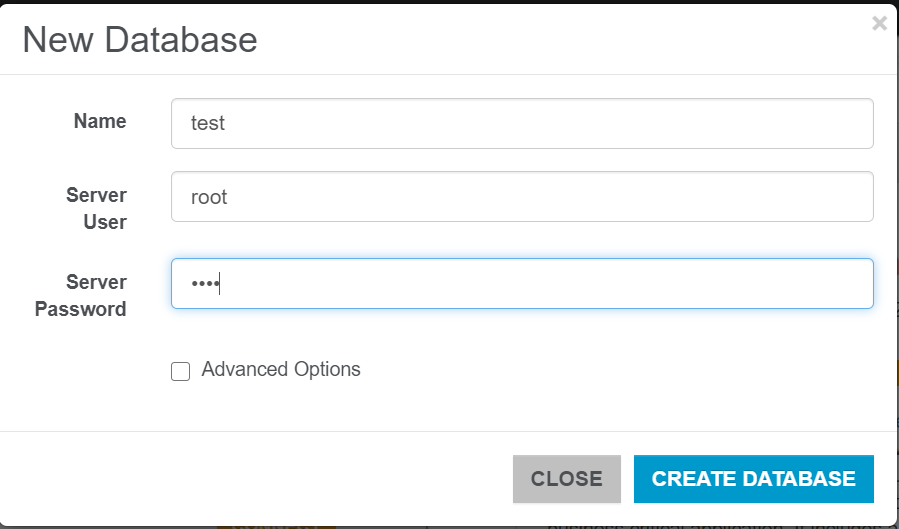
* + - * 1. File server.bat sau khi cấu hình thành công

Sau khi cấu hình thành công chúng ta mở browser và vào đường dẫn <http://localhost:2480/>

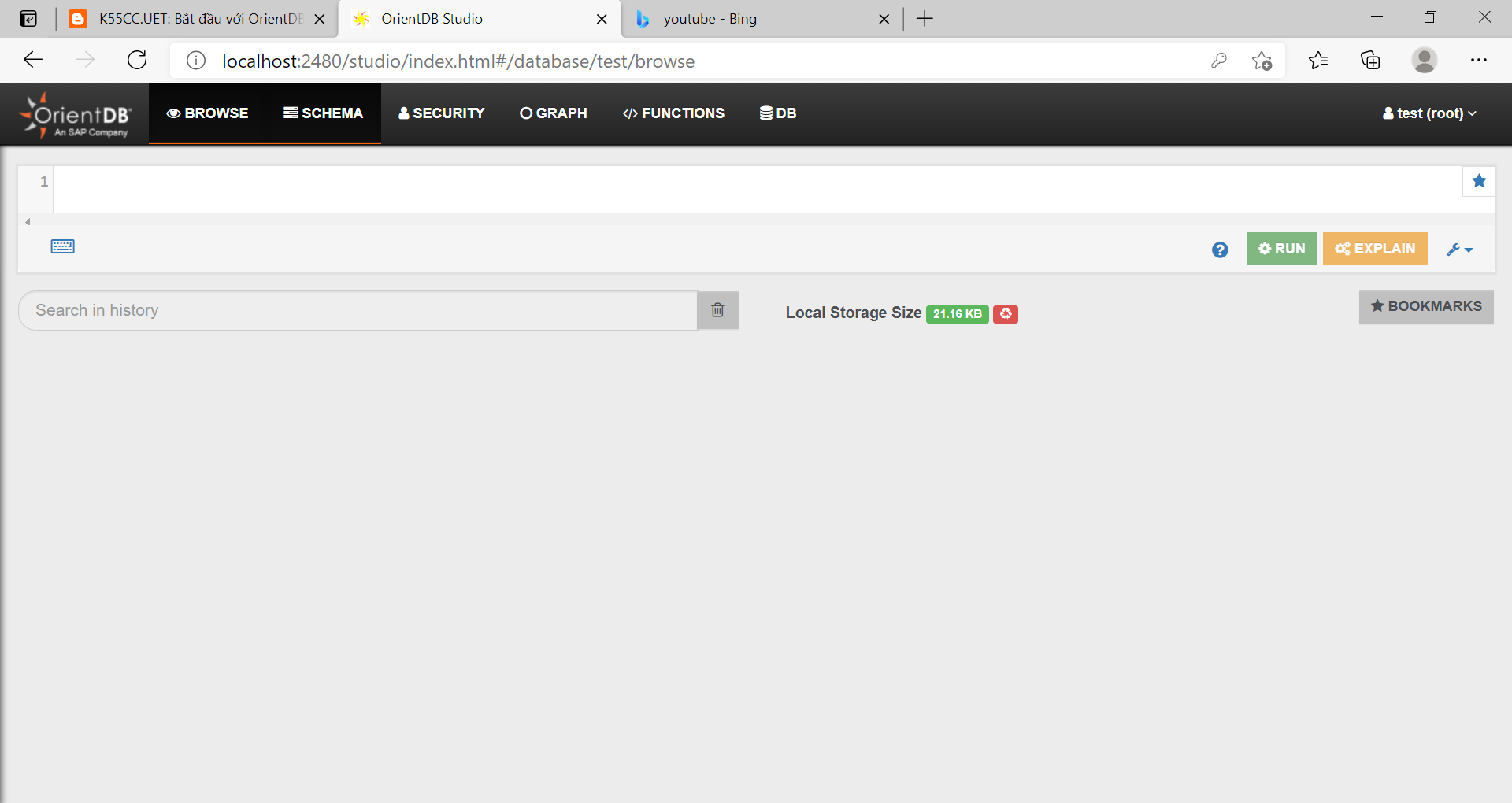


* + - * 1. Giao diện trang chủ OrientDB

Nhập tài khoản mật khẩu và tạo DB test



* + - * 1. Tạo database



* + - * 1. Giao diện database khi mới tạo

đó là những bước cài đặt cơ bản em sẽ demo sâu hơi trong video demo

TÀI LIỆU THAM KHẢO

<https://aws.amazon.com/vi/nosql/>

<https://vi.wikipedia.org/wiki/Wikipedia>

<https://k55cc.blogspot.com/2013/09/bat-au-voi-orientdb.html?fbclid=IwAR0hszCdGznp9yDmR2ixWoc_lflN8VN1dHrF5WkId6aQfedjqzLnrzTs-uI>