

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA HỆ THỐNG THÔNG TIN



BÁO CÁO BÀI TẬP LỚN 2
CƠ CHẾ PHÂN TÁN TRÊN HỆ QUẢN TRỊ NOSQL

Hệ quản trị sử dụng: OrientDB

Lớp: IS211.M11.2

GVHD: Nguyễn Minh Nhựt

Nhóm: Nhóm 3 - Pun

Sinh viên thực hiện:

- Đặng Vũ Phương Uyên – 19520345
- Trần Kim Ngân – 19521890
- Lê Thị Ái Nhi – 19521963
- Nguyễn Thị Thu Phương – 19522066

TP. Hồ Chí Minh, Ngày 26 tháng 12 năm 2021

MỤC LỤC

1. Giới thiệu cơ sở dữ liệu NoSQL	1
2. Tổng quan OrientDB.....	2
3. Lịch sử hình thành.....	2
4. OrientDB các khái niệm cơ bản.....	4
4.1 Mô hình tài liệu	4
4.2 Mô hình đồ thị.....	5
4.3 Mô hình khóa/ giá trị.....	6
4.4 Mô hình đối tượng.....	6
4.4 Cơ chế phân tán	7
5. Cách cài đặt OrientDB trong Window	7
6. Khởi động OrientDB để làm việc trên Command Line	10
7. Tạo và thêm mới dữ liệu	11
7.1. Mô tả cơ sở dữ liệu.....	11
7.2. Tạo database	12
8. Thực hiện truy vấn giữa hai máy	14
9. Thao tác dữ liệu giữa hai máy	23

1. Giới thiệu cơ sở dữ liệu NoSQL

- Trong vài năm qua chúng ta đã thấy sự gia tăng của một loại cơ sở dữ liệu mới, đó là cơ sở dữ liệu NoSQL - mà đang thách thức sự thống trị của cơ sở dữ liệu quan hệ. Có một câu nói ví von như thế này: “Trên thế giới có âm thì phải có dương, có SQL thì cũng phải có NoSQL.” Nếu như SQL – Structured Query Language (Ngôn ngữ truy vấn có cấu trúc) được thiết kế tại IBM vào giữa những năm 1970, được ra mắt và phát triển bởi Oracle năm 1979, thì NoSQL (Ngôn ngữ truy vấn phi quan hệ) lại được sử dụng và phát triển trễ hơn vào cuối những năm 2000.
- NoSQL có nghĩa là Không Chỉ SQL, ngụ ý rằng khi thiết kế một giải pháp phần mềm hoặc sản phẩm, có nhiều hơn một cơ chế lưu trữ có thể được sử dụng dựa trên các nhu cầu. NoSQL không có một định nghĩa rõ ràng mà chúng ta có thể hiểu là mô hình cơ sở dữ liệu mới có các đặc điểm như sau:
 - Không sử dụng các mô hình quan hệ
 - Chạy tốt trên các cụm(server)
 - Chủ yếu là mã nguồn mở
 - Xây dựng cho web thế kỷ 21
 - Tối thiểu hóa lược đồ
- Chúng ta sẽ gặp vấn đề khó khăn khi có sự không tương ứng giữa cấu trúc dữ liệu quan hệ và cấu trúc dữ liệu được lưu trong bộ nhớ hệ thống. Tuy nhiên, sử dụng NoSQL cho phép chúng ta tiếp tục phát triển mà không cần chuyển đổi cấu trúc lưu trong bộ nhớ với cấu trúc dữ liệu quan hệ.
- NoSQL được chia làm 4 loại sau:
 - Loại key-value
 - Cơ sở dữ liệu (Document databases)
 - Cơ sở dữ liệu “column family”
 - Cơ sở dữ liệu đồ thị
- Dưới đây là một số lý do để xem xét việc sử dụng các cơ sở dữ liệu NoSQL.

- Để cải thiện năng suất lập trình bằng cách sử dụng một cơ sở dữ liệu phù hợp hơn với nhu cầu của ứng dụng.
- Để cải thiện hiệu suất truy cập dữ liệu thông qua một số sự kết hợp xử lý khối lượng dữ liệu lớn hơn, giảm độ trễ, và cải thiện thông.

2. Tổng quan OrientDB



Hình 1: Logo OrientDB

- OrientDB là một cơ sở dữ liệu NoSQL đa mô hình với sự hỗ trợ cho dữ liệu đồ thị và tài liệu, điểm đáng chú ý của OrientDB đó chính là nó được phát triển bằng ngôn ngữ lập trình Java nên có thể được cài đặt trên bất kỳ hệ điều hành nào các bạn muốn, đây là một lợi thế lớn so với các cơ sở dữ liệu NoSQL khác. Cơ sở dữ liệu OrientDB cũng mang đầy đủ các đặc tính của một cơ sở dữ liệu NoSQL
- Điểm mạnh của OrientDB đó chính là tốc độ thực thi, trả về kết quả các truy vấn rất nhanh, giúp tăng tốc độ ứng dụng của các bạn. Nếu các bạn đang thiết kế một ứng dụng với nhiều thao tác lấy dữ liệu hơn so với ghi dữ liệu thì OrientDB là một sự lựa chọn tốt để cải thiện hiệu năng ứng dụng.

3. Lịch sử hình thành



Hình 2: Tác giả tạo ra OrientDB - Luca Garulli

- Trong rất nhiều hệ quản trị cơ sở dữ liệu NoSQL thì OrientDB là một hệ quản trị cơ sở dữ liệu NoSQL mã nguồn mở được viết bằng Java. OrientDB ban đầu được viết bởi tác giả Luca Garulli vào năm 2010. Luca đã viết nó như một bản viết lại Java của fast persistent layer của cơ sở dữ liệu Orient ODBMS (ban đầu được phát triển bởi Luca Garulli vào năm 1999 bằng C ++). Trong suốt năm 2012–2014, công cụ lưu trữ đã được phát triển lại bởi Andrey Lomakin.



Hình 3: Andrey Lomakin - Người phát triển công cụ lưu trữ OrientDB

- Kể từ năm 2012, dự án được tài trợ bởi OrientDB LTD (trước đây là Orient Technologies LTD).
- Ngày 15 tháng 9 năm 2017, công ty OrientDB LTD đã được CallidusCloud - một công ty đại chúng được giao dịch trên NASDAQ mua lại.
- Ngày 30 tháng 1 năm 2018, có thông báo SAP (công ty) đã mua lại CallidusCloud với giá 2,4 tỷ đô la và do đó OrientDB hiện được hỗ trợ bởi SAP.
- Ngày 1 tháng 9 năm 2021, người sáng lập ban đầu Luca Garulli rời SAP (công ty) và chuyển dự án thành ArcadeDB sau khi SAP quyết định ngừng cung cấp hỗ trợ thương mại cho OrientDB.

4. OrientDB các khái niệm cơ bản

4.1 Mô hình tài liệu (Document model)

- Là dữ liệu được lưu trữ trong Tài liệu và nhóm Tài liệu được gọi là **Collection**. Về mặt kỹ thuật, tài liệu có nghĩa là một tập hợp các cặp khóa / giá trị hoặc còn được gọi là trường hoặc thuộc tính.

Mô hình quan hệ	Mô hình tài liệu	Mô hình tài liệu OrientDB
Bàn	Bộ sưu tập	Lớp hoặc Cụm
Hàng	Tài liệu	Tài liệu
Cột	Cặp khóa / giá trị	Trường tài liệu
Mối quan hệ	Không có sẵn	Liên kết

Hình 4: Mô tả mô hình tài liệu.

SELECT * FROM LOAIIPHONG

METADATA			PROPERTIES				IN	
@rid	@version	@class	MALOAIPH	TENLOAIPH	SONGUCOI	DONGIA	ThuocLoaiPhong	
#90.0	1	LoaiPhong	LPHONG0001	Standard Room	3	800000		
#91.0	4	LoaiPhong	LPHONG0002	Superior Room	3	1500000	#122.0 #127.0 #128.0	
#92.0	8	LoaiPhong	LPHONG0003	Deluxe Room	3	2000000	#123.0 #124.0 #125.0 #126.0 #122-1	[]
#93.0	1	LoaiPhong	LPHONG0004	Suite Room	3	1700000		
#94.0	3	LoaiPhong	LPHONG0005	Connecting Room	5	3000000	[]	

10 25 50 100 1000 5000

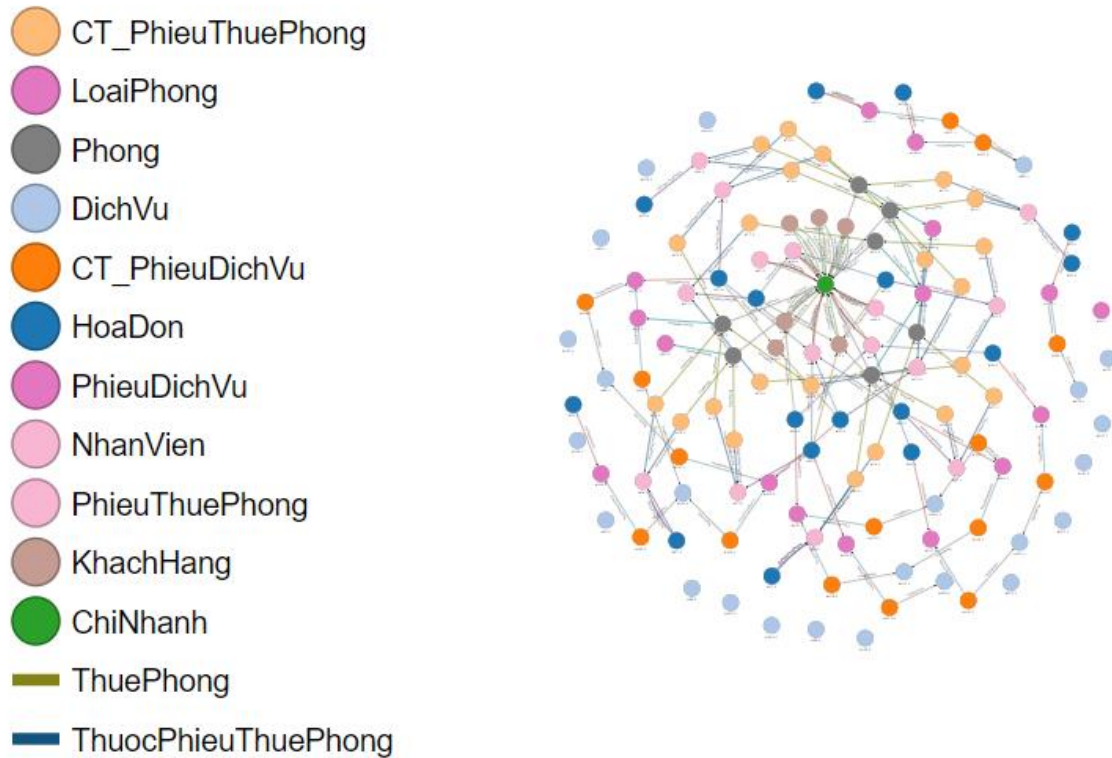
Query executed in 0.157 sec. Returned 5 record(s). Limit: 100 [\(CHANGE IT\)](#)

[Table](#) [Raw](#) [Explain](#)

Hình 5: Mô hình tài liệu trong OrientDB.

4.2 Mô hình đồ thị (Graph Model)

- Cấu trúc dữ liệu đồ thị là một mô hình dữ liệu có thể lưu trữ dữ liệu dưới dạng các Dọc (Nút) được kết nối với nhau bằng các Cạnh (Arcs). Ý tưởng về cơ sở dữ liệu đồ thị OrientDB đến từ đồ thị thuộc tính. Đỉnh và cạnh là các tạo tác chính của mô hình Đồ thị. Chúng chứa các thuộc tính, có thể làm cho chúng xuất hiện tương tự như các tài liệu.
 - Đỉnh - một thực thể có thể được liên kết với các đỉnh khác và có các thuộc tính bắt buộc sau:
 - Định danh duy nhất
 - Tập hợp các cạnh đến
 - Tập hợp các cạnh đi
 - Cạnh - một thực thể liên kết hai đỉnh và có các thuộc tính bắt buộc sau:
 - Định danh duy nhất
 - Liên kết đến một đỉnh đến (còn được gọi là đầu)
 - Liên kết đến một đỉnh gửi đi (còn được gọi là đuôi)
 - Nhãn xác định kiểu kết nối / mối quan hệ giữa đỉnh đầu và đỉnh đuôi
- Ngoài các thuộc tính bắt buộc, mỗi đỉnh hoặc cạnh cũng có thể chứa một tập hợp các thuộc tính tùy chỉnh.



Hình 6: Mô tả mô hình đồ thị

4.3 Mô hình khóa/ giá trị (Key/Value Model)

- Mô hình Khóa / Giá trị có nghĩa là dữ liệu có thể được lưu trữ dưới dạng cặp khóa / giá trị trong đó các giá trị có thể thuộc loại đơn giản và phức tạp. Nó có thể hỗ trợ các tài liệu và phân tử đồ thị dưới dạng giá trị.

4.4 Mô hình đối tượng (Object Model)

- Mô hình này đã được kế thừa bởi lập trình Hướng đối tượng và hỗ trợ **Inheritance** giữa các loại (loại phụ mở rộng các loại siêu), **Polymorphism** khi bạn tham chiếu đến một lớp cơ sở và **Direct binding** from / to Các đối tượng được sử dụng trong ngôn ngữ lập trình.

Mô hình quan hệ	Mô hình đồ thị	Mô hình đồ thị OrientDB
Bản	Lớp Vertex và Edge	Lớp mở rộng "V" (cho Vertex) và "E" (cho Edges)
Hàng	Đỉnh	Đỉnh
Cột	Thuộc tính Vertex và Edge	Thuộc tính Vertex và Edge
Mối quan hệ	Cạnh	Cạnh

Hình 6: Mô tả mô hình đối tượng

4.4 Cơ chế phân tán

- Trong Kiến trúc phân tán OrientDB, tất cả các nút đều là nút chính (multi-Master), trong khi trong hầu hết DBMS, bản sao hoạt động ở chế độ Master-Slave đồng nghĩa với việc chỉ có một nút chính và các Slave chỉ được sử dụng để đọc hoặc khi Master bị hỏng. Điều này cũng có nghĩa là tất cả các nút trong cụm đều là Master và có thể đọc và ghi vào cơ sở dữ liệu, nó cho phép mở rộng quy mô theo chiều ngang mà không bị tắc nghẽn như hầu hết các giải pháp RDBMS và NoSQL khác.
- Nhưng bắt đầu từ OrientDB v2.1, ta có thể gán vai trò của REPLICAS cho một số nút. Khi khởi động máy chủ OrientDB ở chế độ phân tán (bin / dserver.sh), nó sẽ tìm kiếm một cụm tồn tại. Nếu tồn tại, nút bắt đầu sẽ tham gia vào cụm, nếu không sẽ tạo một nút mới. Ta có thể có nhiều cụm trong mạng của mình, mỗi cụm có một "tên nhóm" khác nhau.

5. Cách cài đặt OrientDB trong Window

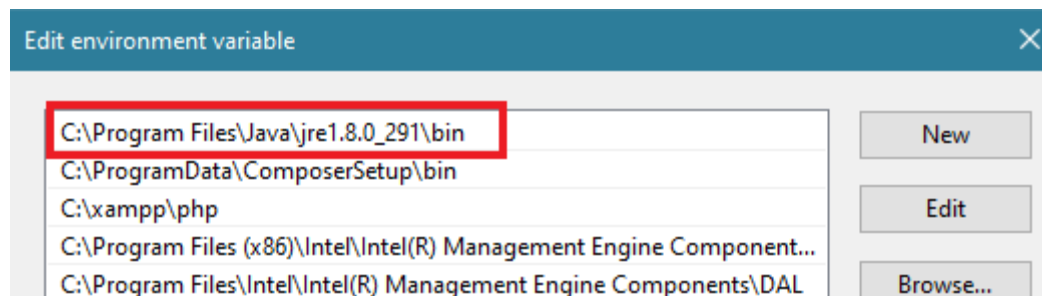
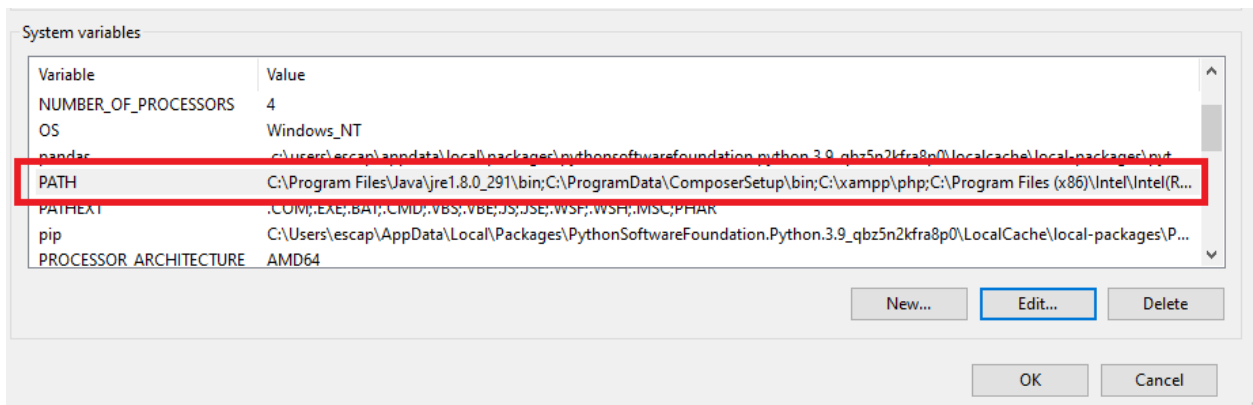
Bước 1 :

- Download OrientDB cho Windows : <https://orientdb.org/download>
- Download Java Development Kit (JDK) : <https://www.oracle.com/java/technologies/downloads/#java8-windows>

Bước 2 : Giải nén và tạo biến môi trường :

- Tiến hành giải nén thư mục OrientDB vừa mới download, sau đó di chuyển thư mục đã giải nén vào ổ C:\.

- Vào Environment Variables trong Window, tạo biến môi trường PATH chứa đường dẫn đến file bin của JDK.



Hình 7: Tạo biến môi trường cho JDK.

Bước 3 : Tiến hành chạy OrientDB, bao gồm 2 bước:

- **Khởi động Server :** Sử dụng lệnh sau để khởi động server.

```
> cd %ORIENTDB_HOME%\bin
> ./server.bat
```

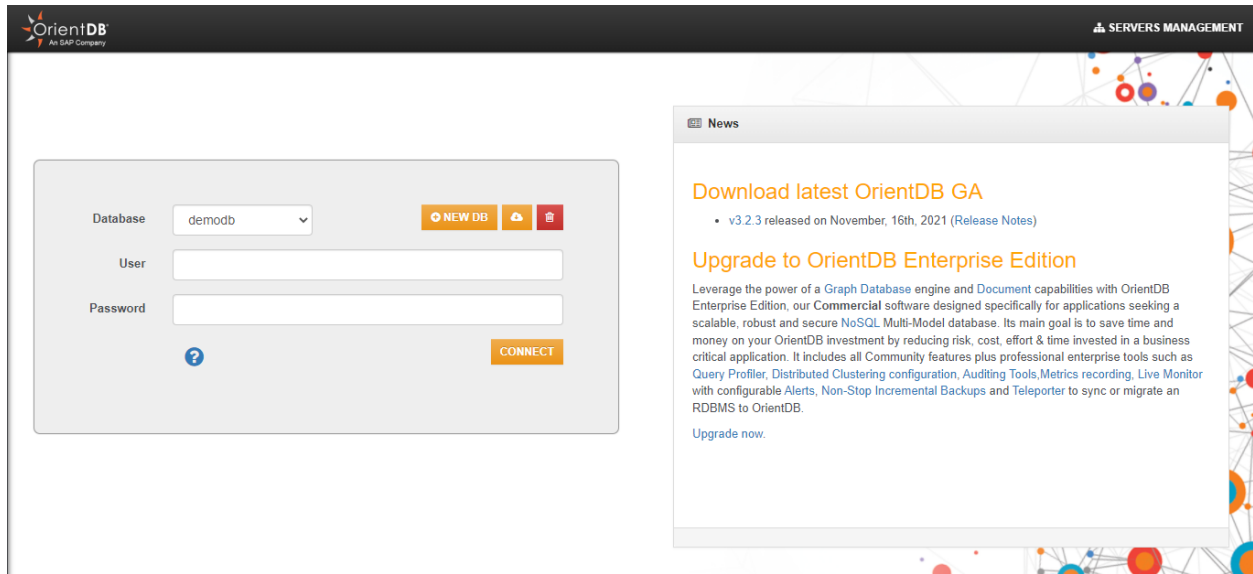
- Nếu chương trình cài đặt thành công thì sẽ hiện thị kết quả như sau:



- **Khởi động Studio** : Sau khi khởi động server thành công, nhập URL <http://localhost:2480/> trên trình duyệt web của máy tính sẽ kết quả như sau:



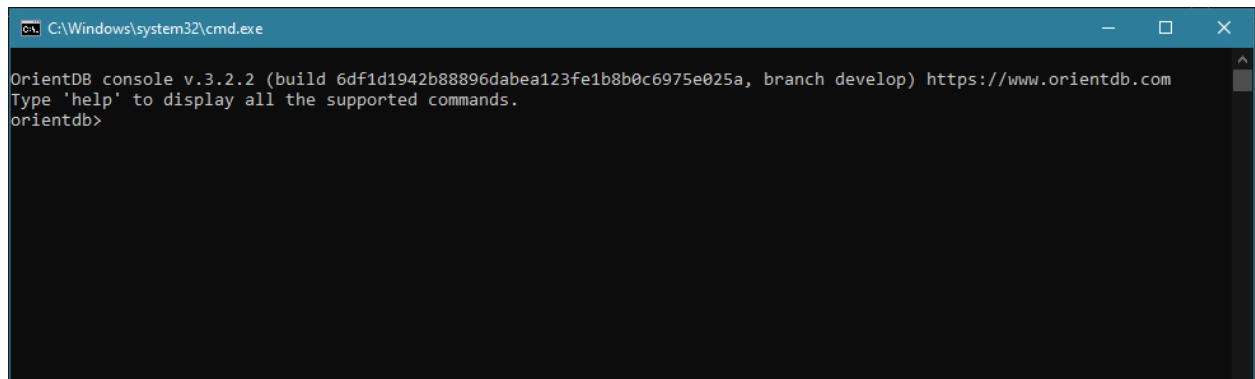
- **Khởi động Studio** : Sau khi khởi động server thành công, nhập URL <http://localhost:2480/> trên trình duyệt web của máy tính sẽ kết quả như sau:



Hình 10: Giao diện Studio của OrientDB sau khi cài đặt thành công.

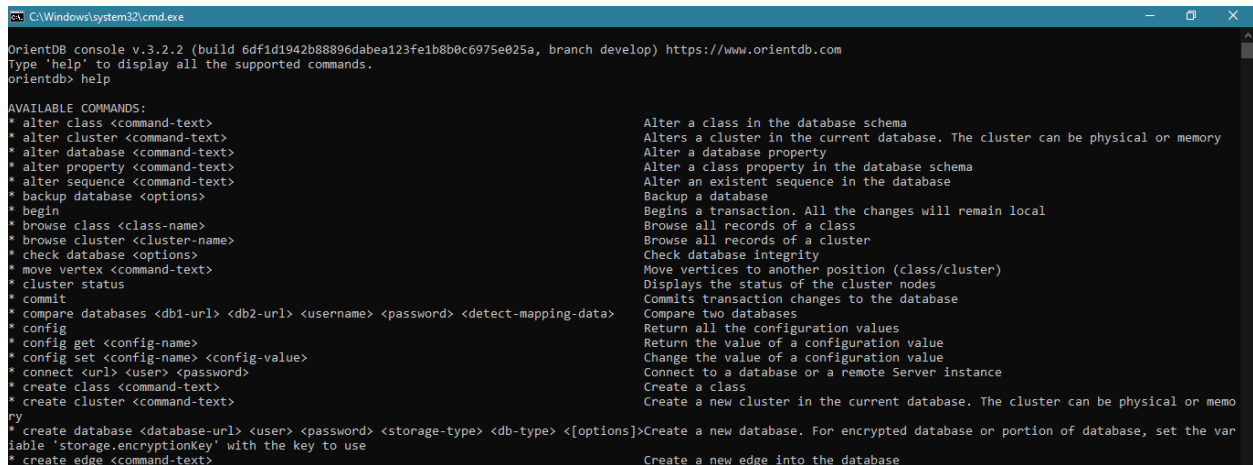
6. Khởi động OrientDB để làm việc trên Command Line

- Mở OrientDB trên CMD bằng cách chạy file console.bat trong thư mục bin (đảm bảo file server.bat đã được khởi động và đang chạy)



Hình 11: File console.bat sau khi khởi động.

- Gõ lệnh help để xem các cú pháp thao tác với OrientDB.



```
C:\Windows\system32\cmd.exe
OrientDB console v.3.2.2 (build 6df1d1942b88896dabea123fe1b8b0c6975e025a, branch develop) https://www.orientdb.com
Type 'help' to display all the supported commands.
orientdb> help

AVAILABLE COMMANDS:
* alter class <command-text>          Alter a class in the database schema
* alter cluster <command-text>        Alters a cluster in the current database. The cluster can be physical or memory
* alter database <command-text>       Alter a database property
* alter property <command-text>       Alter a class property in the database schema
* alter sequence <command-text>       Alter an existent sequence in the database
* backup database <options>           Backup a database
* begin                               Begins a transaction. All the changes will remain local
* browse class <class-name>           Browse all records of a class
* browse cluster <cluster-name>       Browse all records of a cluster
* check database <options>           Check database integrity
* move vertex <command-text>          Move vertices to another position (class/cluster)
* cluster status                      Displays the status of the cluster nodes
* commit                             Commits transaction changes to the database
* compare databases <db1-url> <db2-url> <username> <password> <detect-mapping-data> Compare two databases
* config                             Return all the configuration values
* config get <config-name>            Return the value of a configuration value
* config set <config-name> <config-value> Change the value of a configuration value
* connect <url> <user> <password>     Connect to a database or a remote Server instance
* create class <command-text>         Create a class
* create cluster <command-text>       Create a new cluster in the current database. The cluster can be physical or memo
ry
* create database <database-url> <user> <password> <storage-type> <db-type> <[options]> Create a new database. For encrypted database or portion of database, set the var
iable 'storage.encryptedKey' with the key to use
* create edge <command-text>         Create a new edge into the database
```

Hình 12: Danh sách các cú pháp thao tác sau khi gõ lệnh help.

- Sau khi khởi động thành công thì tiến hành thao tác với OrientDB.

5. Tạo và thêm mới dữ liệu

7.1. Mô tả cơ sở dữ liệu

- Khách sạn PUN có chi nhánh 1 ở Hà Nội và chi nhánh 2 ở Huế.
- Mô hình cơ sở dữ liệu quan hệ của mỗi chi nhánh như sau :

ChiNhanh (**MaCN**, TenCN, DiaChi);

LoaiPhong (**MaLoaiPH**, TenLoaiPH, DonGia, SoNguoi, GhiChu);

DichVu (**MaDV**, tenDV, GiaDV);

HoaDon (**MaHD**, TongTien, NgayTT, **MaNV**, **MaKH**, **MaPTP**, **MaPDV**);

KhachHang (**MaKH**, TenKH, CMND, QuocTich, NgSinh, DChi, LoaiKH, **MaCN**);

NhanVien (**MaNV**, HoTen, GioiTinh, SoDT, NgSinh, DChi, NgVL, Luong, **MaQL**, **MaCN**);

PhieuDichVu (**MaPDV**, TienDV);

PhieuThuePhong (**MaPTP**, TienTP);

Phong (**MaPH**, TinhTrang, GhiChu, MaLoaiPH, MaCN);

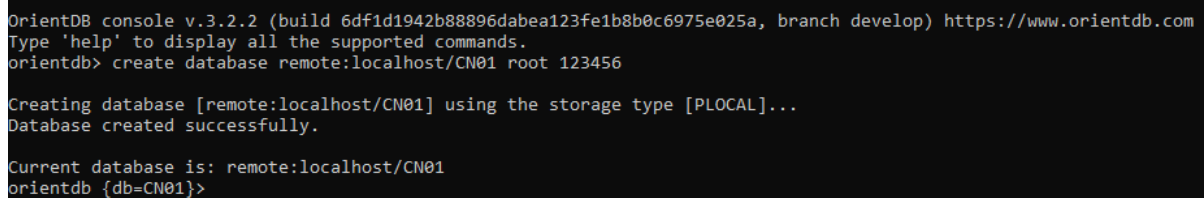
CT_PhieuDichVu (**MaPDV**, **MaDV**, SLDV, NgaySD);

CT_PhieuThuePhong (**MaPTP**, **MaPH**, NgayNP, NgayTPTT, PhuThu, SLNguoi);

7.2. Tạo database

- Gõ lệnh ***create database remote:localhost/CN01 root 123456***.

⇒ Lệnh này sẽ tạo ra 1 database mới tên CN01 với root là user và password là 123456



```
OrientDB console v.3.2.2 (build 6df1d1942b88896dabea123fe1b8b0c6975e025a, branch develop) https://www.orientdb.com
Type 'help' to display all the supported commands.
orientdb> create database remote:localhost/CN01 root 123456

Creating database [remote:localhost/CN01] using the storage type [PLOCAL]...
Database created successfully.

Current database is: remote:localhost/CN01
orientdb {db=CN01}>
```

Hình 13: Tạo database CN01.

- Tiến hành tạo class LoaiPhong trong database vừa tạo với câu lệnh:

create class LoaiPhong extends V

⇒ Câu lệnh này tạo một class tên LoaiPhong và được kế thừa từ lớp V (lớp chứa đối tượng) có sẵn của OrientDB.

- Tạo một property maloaiph trong class đã tạo bằng câu lệnh:

create property LoaiPhong.maloaiph string

⇒ Câu lệnh này tạo một property tên maloaiph trong class LoaiPhong với kiểu dữ liệu String

```
orientdb {db=CN01}> create class LoaiPhong extends V
Class created successfully.

orientdb {db=CN01}> create property LoaiPhong.maloaiph string
Property created successfully.
```

Hình 14: Tạo class LoaiPhong và property maLoaiPH.

- Thực hiện tương tự đối với các property còn lại của LoaiPhong.

```
create property LoaiPhong.maloaiph string
create property LoaiPhong.tenloaiph string
create property LoaiPhong.dongia decimal
create property LoaiPhong.songuoi integer
create property LoaiPhong.ghichu string
```

Hình 15: Các properties của class LoaiPhong.

- Tiến hành Insert dữ liệu vào class LoaiPhong

```
orientdb {db=CN01}> Insert into LOAIPHONG (MALOAIPH, TENLOAIPH,DONGIA,SONGUOI) values ('LPHONG0001','Standard Room',800000,3)
Inserted record '[LoaiPhong#26:0{MALOAIPH:LPHONG0001,TENLOAIPH:Standard Room,DONGIA:800000,SONGUOI:3} v1]' in 0.043000 sec(s).
```

Hình 16: Insert dữ liệu vào class LoaiPhong.

- Thực hiện insert tương tự với các dữ liệu còn lại

```
--LoaiPhong
Insert into LOAIPHONG (MALOAIPH, TENLOAIPH,DONGIA,SONGUOI) values ('LPHONG0001','Standard Room',800000,3)
Insert into LOAIPHONG (MALOAIPH, TENLOAIPH,DONGIA,SONGUOI) values ('LPHONG0002','Superior Room',1500000,3)
Insert into LOAIPHONG (MALOAIPH, TENLOAIPH,DONGIA,SONGUOI) values ('LPHONG0003','Deluxe Room',2000000,3)
Insert into LOAIPHONG (MALOAIPH, TENLOAIPH,DONGIA,SONGUOI) values ('LPHONG0004','Suite Room',1700000,3)
Insert into LOAIPHONG (MALOAIPH, TENLOAIPH,DONGIA,SONGUOI) values ('LPHONG0005','Connecting Room',3000000,5)
```

Hình 17: Dữ liệu Insert của bảng LoaiPhong.

- Sau khi insert xong thì truy vấn LoaiPhong để kiểm tra thử dữ liệu được nhập vào chưa

```
orientdb {db=CN01}> SELECT * FROM LOAIPHONG
```

#	@RID	@CLASS	MALOAPH	TENLOAPH	DONGIA	SONGUOI
0	#26:0	LoaiPhong	LPHONG0001	Standard Room	800000	3
1	#26:1	LoaiPhong	LPHONG0005	Connecting Room	3000000	5
2	#27:0	LoaiPhong	LPHONG0002	Superior Room	1500000	3
3	#28:0	LoaiPhong	LPHONG0003	Deluxe Room	2000000	3
4	#29:0	LoaiPhong	LPHONG0004	Suite Room	1700000	3

Hình 18: Kết quả truy vấn class LoaiPhong.

- Thực hiện tạo và insert dữ liệu tương tự với các class và property còn lại của CN01 và CN02.

6. Thực hiện truy vấn giữa hai máy

- Tiến hành cài đặt OrientDB trên máy 1 tương ứng với CN01 và máy 2 tương ứng với CN02 theo hướng dẫn cài đặt ở phần trước.
- Thiết lập kết nối mạng giữa hai máy bằng Radmin VPN
(Download: <https://www.radmin-vpn.com/>)
- Sau khi tiến hành ping thành công, thì ta bắt đầu kết nối đến database. Trong trường hợp này, ở máy 2 đã có một database là CN02 được tạo bởi user root, máy 1 sẽ tiến hành truy vấn đến máy 2.
- Để tạo kết nối, tại máy 1 ta nhập lệnh có cú pháp như sau:

connect remote:[địa chỉ IP của máy 2] / [tên database] [user] [password]

Trong trường hợp này sẽ là : *connect remote 26.167.204.199/CN02 root 050724*

Khi màn hình thông báo 'OK' là đã thực hiện kết nối thành công.

```
orientdb> connect remote:26.167.204.199/CN02 root 050724
Connecting to database [remote:26.167.204.199/CN02] with user 'root'...OK
```

Hình 19: Kết quả sau khi chạy câu lệnh connect máy 1 với máy 2.

Thực hiện truy vấn trực tiếp

Tại CN01:

- Top 3 dịch vụ được sử dụng nhiều nhất

Câu truy vấn :

```
SELECT *
FROM (
    MATCH{
        CLASS:DICHVU, AS: P
    }-ThuocDichVu-{
        CLASS:CT_PHIEUDICHVU, AS: CTPTDV
    }-ThuocPhieuDichVu-{
        CLASS:PHIEUDICHVU, AS: PTDV
    }
    RETURN P.MADV, COUNT(P.MADV) AS SOLANTHUE
    GROUP BY P.MADV
    ORDER BY SOLANTHUE DESC
    LIMIT 3
)
```

The screenshot displays a database query interface. At the top, a Cypher query is entered in a text area. Below the query, there are buttons for 'RUN' and 'EXPLAIN'. The interface also shows a search bar, local storage size (270.88 KB), and a bookmarks section. The query results are displayed in a table with two columns: P.MADV and SOLANTHUE.

P.MADV	SOLANTHUE
DV0006	3
DV0012	2
DV0010	2

Hình 20: Kết quả câu select.

Tại CN02:

- Tìm khách hàng có quốc tịch là Việt Nam thuê phòng với số lượng người trong phòng đó bằng với số người quy định của loại phòng đó

Câu truy vấn :

```
SELECT DISTINCT HD.MAKH, KH.TENKH
FROM (
    MATCH{
        CLASS:CT_PHIEUTHUEPHONG, AS:CTPTP, WHERE:(PHUTHU !=
0 )
        }-ThuocPhieuThuePhong-{
        CLASS:PHIEUTHUEPHONG, AS:PTP
        }-CoPhieuThuePhong-{
        CLASS:HOADON, AS:HD
        }-ThuocKhachHang-{
        CLASS:KHACHHANG, AS:KH, WHERE:(QUOCTICH = 'Việt Nam')}
    RETURN *)
```

The screenshot displays a query execution environment. At the top, a Cypher query is entered in a text area, matching the query provided in the text. Below the query area, there are buttons for 'RUN' and 'EXPLAIN'. A 'Local Storage Size' indicator shows 271.65 KB. Below this, a 'COMMAND' bar shows the executed query. At the bottom, a table titled 'PROPERTIES' displays the results of the query.

PROPERTIES	
HD.MAKH	KH.TENKH
KH0006	Phạm Thị Ngọc Nhi
KH0008	Huỳnh Thu Tuyết

Hình 21: Kết quả câu select.

➤ Thực hiện truy vấn giữa hai máy

Tại CN01 thực hiện truy vấn đến CN02 đồng thời ở CN02 cũng thực hiện các câu truy vấn có yêu cầu dưới đây:

1. *Thống kê tổng doanh thu của các khách hàng không phải là thành viên và kiểm tra xem khách hàng đó có tổng doanh thu có lớn hơn 50 triệu không? Nếu có thì xuất ra kết quả là Thành viên để phân loại khách hàng tiềm năng.*

Câu truy vấn :

```
SELECT *, IF(EVAL("REVENUE_KHACHHANG > 50000000"), 'Thanh vien' ,
'Vang lai') as RESULT_KHACHHANG
FROM (
    MATCH {
        CLASS: HOADON, AS: A
    }-ThuocKhachHang-{class: KHACHHANG, as: B}
    RETURN B.MAKH AS ID_KHACHHANG, B.TENKH AS
NAME_KHACHHANG, SUM(A.TONGTIEN) AS REVENUE_KHACHHANG
GROUP BY B.MAKH, B.TENKH )
```

Kết quả :

```
orientdb {db=CN02}> connect remote:localhost/CN02 root 123456
Disconnecting from the database [CN02]...OK
Connecting to database [remote:localhost/CN02] with user 'root'...OK
orientdb {db=CN02}> SELECT *, IF(EVAL("REVENUE_KHACHHANG > 50000000"), 'Thanh vien' , 'Vang lai') as RESULT_KHACHHANG FROM (MATCH {CLASS: HOADON, AS: A}-ThuocKhachHang-{class: KHACHHANG, as: B} RETURN B.MAKH AS ID_KHACHHANG, B.TENKH AS NAME_KHACHHANG, SUM(A.TONGTIEN) AS REVENUE_KHACHHANG GROUP BY B.MAKH, B.TENKH);
```

#	ID_KHACHHANG	NAME_KHACHHANG	REVENUE_KHACHHANG	RESULT_KHACHHANG
0	KH0006	Phạm Thị Ngọc Nhi	30750000	Vang lai
1	KH0007	Nguyễn Hoàng Tú	109316000	Thanh vien
2	KH0008	Huỳnh Thu Tuyet	37754000	Vang lai
3	KH0009	Hoàng Minh Tuấn	9518000	Vang lai
4	KH0010	Trần Tấn Trung	21439000	Vang lai

Hình 22: Kết quả câu truy vấn 1 ở chi nhánh 2.

```
orientdb {db=CN02}> SELECT *, IF(EVAL("REVENUE_KHACHHANG > 50000000"), 'Thành viên', 'Vang lai') as RESULT_KHACHHANG FROM (MATCH {CLASS: HOADON, AS: A}-ThuocKhachHang-
{class: KHACHHANG, as: B} RETURN B.MAKH AS ID_KHACHHANG, B.TENKH AS NAME_KHACHHANG, SUM(A.TONGTIEN) AS REVENUE_KHACHHANG GROUP BY B.MAKH, B.TENKH);
```

#	ID_KHACHHANG	NAME_KHACHHANG	REVENUE_KHACHHANG	RESULT_KHACHHANG
0	KH0006	Phạm Thị Ngọc Nhi	30750000	Vang lai
1	KH0007	Nguyễn Hoàng Tú	109316000	Thành viên
2	KH0008	Huỳnh Thu Tuyet	37754000	Vang lai
3	KH0009	Hoàng Minh Tuấn	9518000	Vang lai
4	KH0010	Trần T?n Trung	21439000	Vang lai

5 item(s) found. Query executed in 0.012 sec(s).

Hình 23: Kết quả câu truy vấn 1 ở chi nhánh 1.

2. Thống kê các số lượng thuê phòng của phòng đó để tiến hành đề ra các chiến dịch nhằm khắc phục tình trạng trống phòng quá lâu

Câu truy vấn :

```
SELECT *
FROM (
    MATCH{
        CLASS:PHONG, AS: P
    }-THUEPHONG-{
        CLASS:CT_PHIEUTHUEPHONG, AS: CTPTP
    }-ThuocPhieuThuePhong-{
        CLASS:PHIEUTHUEPHONG, AS: PTP}
    RETURN COUNT(P.MAPH) AS SOLANTHUE, P.MAPH
    GROUP BY P.MAPH
    ORDER BY SOLANTHUE ASC)
```

```
orientdb {db=CN02}> connect remote:localhost/CN02 root 123456
Disconnecting from the database [CN02]...OK

Connecting to database [remote:localhost/CN02] with user 'root'...OK
orientdb {db=CN02}> SELECT * FROM ( MATCH{ CLASS:PHONG, AS: P }-THUEPHONG-(CLASS:CT_PHIEUTHUEPHONG, AS: CTPTP)-ThuocPhieuThuePhong-(CLASS:PHIEUTHUEPHONG, AS: PTP) RETURN COUNT(P.MAPH) AS SOLANTHUE, P.MAPH GROUP BY P.MAPH ORDER BY SOLANTHUE ASC);
```

#	SOLANTHUE	P.MAPH
0	1	MP0002
1	1	MP0003
2	1	MP0007
3	2	MP0001
4	3	MP0006
5	3	MP0004
6	7	MP0005

Hình 24: Kết quả câu truy vấn 2 ở chi nhánh 2.

```
orientdb {db=CN02}> SELECT * FROM ( MATCH{ CLASS:PHONG, AS: P }-THUEPHONG-(CLASS:CT_PHIEUTHUEPHONG, AS: CTPTP)-ThuocPhieuThuePhong-(CLASS:PHIEUTHUEPHONG, AS: PTP) RETURN COUNT(P.MAPH) AS SOLANTHUE, P.MAPH GROUP BY P.MAPH ORDER BY SOLANTHUE ASC);
```

#	SOLANTHUE	P.MAPH
0	1	MP0002
1	1	MP0003
2	1	MP0007
3	2	MP0001
4	3	MP0006
5	3	MP0004
6	7	MP0005

7 item(s) found. Query executed in 0.026 sec(s).

Hình 25: Kết quả câu truy vấn 2 ở chi nhánh 1.

3. Tìm hóa đơn được lập vào ngày 23/01/2021 có mà có tính phí phụ thu của các phòng có đơn giá trên 1tr500 do nhân viên có số điện thoại là 0928493155 lập

Câu truy vấn:

```
SELECT DISTINCT *
FROM (
    MATCH{
        CLASS: NHANVIEN, AS:NV, WHERE:(SODT='0928493155')
    }-TuNhanVien-{
        CLASS: HOADON, AS: HD, WHERE:(NGAYTT='2021/01/23')
    }-CoPhieuThuePhong-{
        CLASS:PHIEUTHUEPHONG, AS:PTP
    }-ThuocPhieuThuePhong-{
        CLASS:CT_PHIEUTHUEPHONG, AS:CTPTP, WHERE:(PHUTHU > 0)
    }-ThuePhong-{
        CLASS:PHONG,AS:P
```

```

    }-ThuocLoaiPhong-{
        CLASS:LOAIPHONG, AS:LP, WHERE:(DONGIA>1500000)}
    RETURN HD.MAHD, HD.MAKH
    ORDER BY HD.MAHD
)

```

```

orientdb (db=CN02)> connect remote:localhost/CN02 root 123456

Disconnecting from the database [CN02]...OK

Connecting to database [remote:localhost/CN02] with user 'root'...OK
orientdb (db=CN02)> SELECT DISTINCT * FROM (MATCH(CLASS: NHANVIEN, AS:NV, WHERE:(SODT='0928493155'))-TuNhanVien-(CLASS: HOADON, AS: HD, WHERE:(NGAYTT='2021/01/23'))-CoPhieuThuePhong-(CLASS:PHIEUTHUEPHONG, AS:PTP)-ThuocPhieuThuePhong-(CLASS:CT_PHIEUTHUEPHONG, AS:CTPTP, WHERE:(PHUTHU > 0))-ThuePhong-(CLASS:PHONG, AS:P)-ThuocLoaiPhong-(CLASS:LOAIPHONG, AS:LP, WHERE:(DONGIA>1500000))RETURN HD.MAHD, HD.MAKH ORDER BY HD.MAHD);

-----+-----+
|# |HD.MAHD|HD.MAKH|
-----+-----+
|0 |HD0004 |KH0008 |
-----+-----+

```

Hình 26: Kết quả câu truy vấn 3 ở chi nhánh 2.

```

orientdb (db=CN02)> SELECT DISTINCT * FROM (MATCH(CLASS: NHANVIEN, AS:NV, WHERE:(SODT='0928493155'))-TuNhanVien-(CLASS: HOADON, AS: HD, WHERE:(NGAYTT='2021/01/23'))-CoPhieuThuePhong-(CLASS:PHIEUTHUEPHONG, AS:PTP)-ThuocPhieuThuePhong-(CLASS:CT_PHIEUTHUEPHONG, AS:CTPTP, WHERE:(PHUTHU > 0))-ThuePhong-(CLASS:PHONG, AS:P)-ThuocLoaiPhong-(CLASS:LOAIPHONG, AS:LP, WHERE:(DONGIA>1500000))RETURN HD.MAHD, HD.MAKH ORDER BY HD.MAHD);

-----+-----+
|# |HD.MAHD|HD.MAKH|
-----+-----+
|0 |HD0004 |KH0008 |
-----+-----+

1 item(s) found. Query executed in 0.009 sec(s).

```

Hình 27: Kết quả câu truy vấn 3 ở chi nhánh 1.

- Tại CN02 thực hiện truy vấn đến CN01 đồng thời ở CN01 cũng thực hiện các câu truy vấn có yêu cầu dưới đây:
 4. Top 3 khách hàng có giá trị hóa đơn cao nhất

```

SELECT *
FROM (
    MATCH {
        CLASS: HOADON, AS: A
    }-ThuocKhachHang-{class: KHACHHANG, as: B}
    RETURN B.MAKH AS ID_KHACHHANG, B.TENKH AS
NAME_KHACHHANG, SUM(A.TONGTIEN) AS TOTAL
    GROUP BY B.MAKH, B.TENKH
    ORDER BY TOTAL DESC
    LIMIT 3
)

```

```

orientdb {db=CN01}> connect remote:localhost/CN01 root 123456
Disconnecting from the database [CN01]...OK
Connecting to database [remote:localhost/CN01] with user 'root'...OK
orientdb {db=CN01}> SELECT * FROM (MATCH {CLASS: HOADON, AS: A}-ThuocKhachHang-{class: KHACHHANG, as: B} RETURN B.MAKH AS ID_KHACHHANG, B.TENKH AS NAME_KHACHHANG, SUM(A
.TONGTIEN) AS TOTAL GROUP BY B.MAKH, B.TENKH ORDER BY TOTAL DESC LIMIT 3);
+-----+-----+-----+
|#| ID_KHACHHANG|NAME_KHACHHANG|TOTAL|
+-----+-----+-----+
|0| KH0002      |Trần Kim Anh  |311510000|
|1| KH0001      |Trần Ngọc Tuấn Anh|55474000|
|2| KH0005      |Trần Kim Ngọc  |10000000|
+-----+-----+-----+

```

Hình 28: Kết quả câu truy vấn 4 ở chi nhánh 2.

```

orientdb {db=CN01}> SELECT * FROM (MATCH {CLASS: HOADON, AS: A}-ThuocKhachHang-{class: KHACHHANG, as: B} RETURN B.MAKH AS ID_KHACHHANG, B.TENKH AS NAME_KHACHHANG, SUM(A
.TONGTIEN) AS TOTAL GROUP BY B.MAKH, B.TENKH ORDER BY TOTAL DESC LIMIT 3);
+-----+-----+-----+
|#| ID_KHACHHANG|NAME_KHACHHANG|TOTAL|
+-----+-----+-----+
|0| KH0002      |Trần Kim Anh  |311510000|
|1| KH0001      |Trần Ngọc Tuấn Anh|55474000|
|2| KH0005      |Trần Kim Ngọc  |10000000|
+-----+-----+-----+
3 item(s) found. Query executed in 0.006 sec(s).

```

Hình 29: Kết quả câu truy vấn 4 ở chi nhánh 1.

5. Nhân viên lập nhiều hóa đơn nhất

```
SELECT DISTINCT *
FROM (
    MATCH{
        CLASS:NHANVIEN, AS: P
        }-TuNhanVien-{
        CLASS:HOADON, AS: M
        }
    RETURN P.HOTEN, COUNT(M.MAHD) AS SOLANLAP
    GROUP BY P.MANV
    ORDER BY SOLUONG DESC
LIMIT 1
)
```

```
orientdb {db=CN01}> SELECT DISTINCT * FROM (MATCH{CLASS:NHANVIEN, AS: P}-TuNhanVien-{CLASS:HOADON, AS: M} RETURN P.HOTEN, COUNT(M.MAHD) AS SOLANLAP GROUP BY P.MANV ORDER BY SOLUONG DESC LIMIT 1)
+-----+-----+
|#      |P.HOTEN      |SOLANLAP|
+-----+-----+
|0      |Nguyễn Th? H?ng Hoa|3      |
+-----+-----+
```

Hình 30: Kết quả câu truy vấn 5 ở chi nhánh 2.

```
orientdb {db=CN01}> SELECT DISTINCT * FROM (MATCH{CLASS:NHANVIEN, AS: P}-TuNhanVien-{CLASS:HOADON, AS: M} RETURN P.HOTEN, COUNT(M.MAHD) AS SOLANLAP GROUP BY P.MANV ORDER BY SOLUONG DESC LIMIT 1)
+-----+-----+
|#      |P.HOTEN      |SOLANLAP|
+-----+-----+
|0      |Nguyễn Th? H?ng Hoa|3      |
+-----+-----+
1 item(s) found. Query executed in 0.008 sec(s).
```

Hình 31: Kết quả câu truy vấn 5 ở chi nhánh 1.

6. Tìm thông tin khách hàng thuê phòng loại Deluxe Room vào tháng 2


```

SELECT DISTINCT KH.TENKH, KH.SDT, KH.DIACHI
FROM (
    MATCH
        {CLASS:KHACHHANG, AS:KH}-ThuocKhachHang-
        {CLASS:HOADON, AS:HD}
        -CoPhieuThuePhong-{
            CLASS:PHIEUTHUEPHONG, AS:PTP
        }-ThuocPhieuThuePhong-{CLASS:CT_PHIEUTHUEPHONG,
AS:CTPTP}-ThuePhong-{
            CLASS:PHONG, AS:P
        }-ThuocLoaiPhong-{
            CLASS:LOAIPHONG, AS:LP
        }
    RETURN *)
WHERE LP.TENLOAIPH="Deluxe Room" AND CTPTP.NGAYNP BETWEEN
"2021-02-01" AND "2021-02-28"
    
```

```

orientdb {db=CN01}> connect remote:localhost/CN01 root 123456
Disconnecting from the database [CN01]...OK
Connecting to database [remote:localhost/CN01] with user 'root'...OK
orientdb {db=CN01}> SELECT DISTINCT KH.TENKH, KH.SDT, KH.DIACHI FROM (MATCH{CLASS:KHACHHANG, AS:KH}-ThuocKhachHang-{CLASS:HOADON, AS:HD}-CoPhieuThuePhong-{CLASS:PHIEUTHUEPHONG, AS:PTP}-ThuocPhieuThuePhong-{CLASS:CT_PHIEUTHUEPHONG, AS:CTPTP}-ThuePhong-{CLASS:PHONG, AS:P}-ThuocLoaiPhong-{CLASS:LOAIPHONG, AS:LP}RETURN *)WHERE LP.TENLOAIPH="Deluxe Room" AND CTPTP.NGAYNP BETWEEN "2021-02-01" AND "2021-02-28"
+-----+-----+-----+
|# |KH.TENKH |KH.SDT|KH.DIACHI|
+-----+-----+-----+
|0 |Trần Kim Anh|123 |TPHCM |
+-----+-----+-----+
    
```

Hình 32: Kết quả câu truy vấn 6 ở chi nhánh 1.

```

orientdb {db=CN01}> SELECT DISTINCT KH.TENKH, KH.SDT, KH.DIACHI FROM (MATCH{CLASS:KHACHHANG, AS:KH}-ThuocKhachHang-{CLASS:HOADON, AS:HD}-CoPhieuThuePhong-{CLASS:PHIEUTHUEPHONG, AS:PTP}-ThuocPhieuThuePhong-{CLASS:CT_PHIEUTHUEPHONG, AS:CTPTP}-ThuePhong-{CLASS:PHONG, AS:P}-ThuocLoaiPhong-{CLASS:LOAIPHONG, AS:LP}RETURN *)WHERE LP.TENLOAIPH="Deluxe Room" AND CTPTP.NGAYNP BETWEEN "2021-02-01" AND "2021-02-28"
+-----+-----+-----+
|# |KH.TENKH |KH.SDT|KH.DIACHI|
+-----+-----+-----+
|0 |Trần Kim Anh|123 |TPHCM |
+-----+-----+-----+
1 item(s) found. Query executed in 0.018 sec(s).
    
```

Hình 33: Kết quả câu truy vấn 6 ở chi nhánh 2.

7. Thao tác dữ liệu giữa hai máy

➤ Thực hiện ở máy 1

Insert

```
orientdb {db=CN01}> SELECT * FROM NHANVIEN
```

#	@RID	@CLASS	LUONG	SODT	MAQL	MACN	HOTEN	in_TuNhanVien	MANV	out_ThuocKhachHang	out_CoNhanVien
0	#42:0	NhanVien	55000000	0779880707		CN0001	Nguy?n Th? H?ng Hoa	[#170:0,#172:0,#173:0]	NV0001	[#168:1]	[#202:0]
1	#43:0	NhanVien	15000000	0978304543	NV0001	CN0001	Tr?nh Th? Thu H??ng	[#174:0,#175:0,#176:0,#177:0]	NV0002	[#162:2]	[#203:0]
2	#44:0	NhanVien	15000000	0979604733	NV0001	CN0001	??ng Th?y Trang	[#170:1,#171:0]	NV0003	[#163:0]	[#204:0]
3	#45:0	NhanVien	15000000	0918537012	NV0001	CN0001	B?i Th? Th?y Trang		NV0004	[#166:0]	[#205:0]
4	#46:0	NhanVien	15000000	0967528671	NV0001	CN0001	Tr?n Th? Dung		NV0005	[#169:0]	[#206:0]

Hình 34: Kết quả truy vấn của NhanVien trước khi thực hiện lệnh Insert.

- Thực hiện Insert vào class NhanVien của máy 1 với một bản ghi có MaNV = ‘NV0006’ , HoTen = ‘Nguyễn Thanh Trúc’ có SoDT= ‘0921253664’ , Luong = ‘6000000’ và MaCN = ‘CN0001’.

```
orientdb {db=CN01}> SELECT * FROM NHANVIEN
```

#	@RID	@CLASS	LUONG	SODT	MAQL	MACN	HOTEN	in_TuNhanVien	MANV	out_ThuocKhachHang	out_CoNhanVien
0	#42:0	NhanVien	55000000	0779880707		CN0001	Nguy?n Th? H?ng Hoa	[#170:0,#172:0,#173:0]	NV0001	[#168:1]	[#202:0]
1	#42:1	NhanVien	6000000	0921253664		CN0001	Nguy?n Thanh Tr?c		NV0006		
2	#43:0	NhanVien	15000000	0978304543	NV0001	CN0001	Tr?nh Th? Thu H??ng	[#174:0,#175:0,#176:0,#177:0]	NV0002	[#162:2]	[#203:0]
3	#44:0	NhanVien	15000000	0979604733	NV0001	CN0001	??ng Th?y Trang	[#170:1,#171:0]	NV0003	[#163:0]	[#204:0]
4	#45:0	NhanVien	15000000	0918537012	NV0001	CN0001	B?i Th? Th?y Trang		NV0004	[#166:0]	[#205:0]
5	#46:0	NhanVien	15000000	0967528671	NV0001	CN0001	Tr?n Th? Dung		NV0005	[#169:0]	[#206:0]

Hình 35: Kết quả sau khi thực hiện lệnh Insert.

Update

```
orientdb {db=CN01}> SELECT * FROM NHANVIEN
```

#	@RID	@CLASS	LUONG	SODT	MAQL	MACN	HOTEN	in_TuNhanVien	MANV	out_ThuocKhachHang	out_CoNhanVien
0	#42:0	NhanVien	55000000	0779880707		CN0001	Nguy?n Th? H?ng Hoa	[#170:0,#172:0,#173:0]	NV0001	[#168:1]	[#202:0]
1	#42:1	NhanVien	6000000	0921253664		CN0001	Nguy?n Thanh Tr?c		NV0006		
2	#43:0	NhanVien	15000000	0978304543	NV0001	CN0001	Tr?nh Th? Thu H??ng	[#174:0,#175:0,#176:0,#177:0]	NV0002	[#162:2]	[#203:0]
3	#44:0	NhanVien	15000000	0979604733	NV0001	CN0001	??ng Th?y Trang	[#170:1,#171:0]	NV0003	[#163:0]	[#204:0]
4	#45:0	NhanVien	15000000	0918537012	NV0001	CN0001	B?i Th? Th?y Trang		NV0004	[#166:0]	[#205:0]
5	#46:0	NhanVien	15000000	0967528671	NV0001	CN0001	Tr?n Th? Dung		NV0005	[#169:0]	[#206:0]

Hình 36: Kết quả truy vấn của NhanVien trước khi thực hiện lệnh Update.

- Thực hiện Update tại class NhanVien của máy 2 với bản ghi có MaNV= ‘NV0006’ với dữ liệu được update là Luong = ‘75000000’

```
orientdb {db=CN01}> UPDATE NHANVIEN SET LUONG = '75000000' WHERE MANV = 'NV0006';

Updated record(s) ' [{count:1}] ' in 0.005000 sec(s).

orientdb {db=CN01}> SELECT * FROM NHANVIEN
```

#	@RID	@CLASS	LUONG	SODT	MAQL	MACN	HOTEN	in_TuNhanVien	MANV	out_ThuocKhachHang	out_CoNhanVien
0	#42:0	NhanVien	55000000	0779880707		CN0001	Nguy?n Th? H?ng Hoa	[#170:0,#172:0,#173:0]	NV0001	[#168:1]	[#202:0]
1	#42:1	NhanVien	75000000	0921253664		CN0001	Nguy?n Thanh Tr?c		NV0006		
2	#43:0	NhanVien	15000000	0978304543	NV0001	CN0001	Tr?nh Th? Thu H?ng	[#174:0,#175:0,#176:0,#177:0]	NV0002	[#162:2]	[#203:0]
3	#44:0	NhanVien	15000000	0979604733	NV0001	CN0001	??ng Thùy Trang	[#170:1,#171:0]	NV0003	[#163:0]	[#204:0]
4	#45:0	NhanVien	15000000	0918537012	NV0001	CN0001	Bùi Th? Thùy Trang		NV0004	[#166:0]	[#205:0]
5	#46:0	NhanVien	15000000	0967528671	NV0001	CN0001	Tr?n Th? Dung		NV0005	[#169:0]	[#206:0]

Hình 37: Kết quả sau khi thực hiện lệnh Update.

Delete

```
orientdb {db=CN01}> UPDATE NHANVIEN SET LUONG = '75000000' WHERE MANV = 'NV0006';

Updated record(s) ' [{count:1}] ' in 0.005000 sec(s).

orientdb {db=CN01}> SELECT * FROM NHANVIEN
```

#	@RID	@CLASS	LUONG	SODT	MAQL	MACN	HOTEN	in_TuNhanVien	MANV	out_ThuocKhachHang	out_CoNhanVien
0	#42:0	NhanVien	55000000	0779880707		CN0001	Nguy?n Th? H?ng Hoa	[#170:0,#172:0,#173:0]	NV0001	[#168:1]	[#202:0]
1	#42:1	NhanVien	75000000	0921253664		CN0001	Nguy?n Thanh Tr?c		NV0006		
2	#43:0	NhanVien	15000000	0978304543	NV0001	CN0001	Tr?nh Th? Thu H?ng	[#174:0,#175:0,#176:0,#177:0]	NV0002	[#162:2]	[#203:0]
3	#44:0	NhanVien	15000000	0979604733	NV0001	CN0001	??ng Thùy Trang	[#170:1,#171:0]	NV0003	[#163:0]	[#204:0]
4	#45:0	NhanVien	15000000	0918537012	NV0001	CN0001	Bùi Th? Thùy Trang		NV0004	[#166:0]	[#205:0]
5	#46:0	NhanVien	15000000	0967528671	NV0001	CN0001	Tr?n Th? Dung		NV0005	[#169:0]	[#206:0]

Hình 38: Kết quả truy vấn của NhanVien trước khi thực hiện lệnh Delete.

- Thực hiện Delete ở class NhanVien của máy 2 với bản ghi có MaNV = 'NV0006'

```
orientdb {db=CN01}> DELETE VERTEX FROM NHANVIEN WHERE MANV = 'NV0006';

Delete record(s) ' [{count:1}] ' in 0.015000 sec(s).

orientdb {db=CN01}> SELECT * FROM NHANVIEN
```

#	@RID	@CLASS	LUONG	SODT	MAQL	MACN	HOTEN	in_TuNhanVien	MANV	out_ThuocKhachHang	out_CoNhanVien
0	#42:0	NhanVien	55000000	0779880707		CN0001	Nguy?n Th? H?ng Hoa	[#170:0,#172:0,#173:0]	NV0001	[#168:1]	[#202:0]
1	#43:0	NhanVien	15000000	0978304543	NV0001	CN0001	Tr?nh Th? Thu H?ng	[#174:0,#175:0,#176:0,#177:0]	NV0002	[#162:2]	[#203:0]
2	#44:0	NhanVien	15000000	0979604733	NV0001	CN0001	??ng Thùy Trang	[#170:1,#171:0]	NV0003	[#163:0]	[#204:0]
3	#45:0	NhanVien	15000000	0918537012	NV0001	CN0001	Bùi Th? Thùy Trang		NV0004	[#166:0]	[#205:0]
4	#46:0	NhanVien	15000000	0967528671	NV0001	CN0001	Tr?n Th? Dung		NV0005	[#169:0]	[#206:0]

Hình 39: Kết quả sau khi thực hiện lệnh Delete.

➤ Thực hiện ở máy 2

Insert

```
orientdb> connect remote:26.167.204.199/CN02 root 050724
Connecting to database [remote:26.167.204.199/CN02] with user 'root'...OK
orientdb {db=CN02}> SELECT * FROM LOAIPHONG;
```

#	@RID	@CLASS	MALOAIPH	TENLOAIPH	DONGIA	SONGDOI	in_ThuocLoaiPhong	in_
0	#90:0	LoaiPhong	LPHONG0001	Standard Room	800000	3		
1	#91:0	LoaiPhong	LPHONG0002	Superior Room	1500000	3	[#122:0,#127:0,#128:0]	
2	#92:0	LoaiPhong	LPHONG0003	Deluxe Room	2000000	3	[#123:0,#124:0,#125:0,#126:0,#122:1]	[]
3	#93:0	LoaiPhong	LPHONG0004	Suite Room	1700000	3		
4	#94:0	LoaiPhong	LPHONG0005	Connecting Room	3000000	5	[]	

```
5 item(s) found. Query executed in 1.421 sec(s).
```

Hình 40: Kết quả truy vấn của NhanVien trước khi thực hiện lệnh Insert.

- Thực hiện Insert vào bảng LOAIPHONG của máy 2 với một bản ghi có MaLoaiPhong = 'LPHONG0006', TenLoaiPhong = 'Double Room' có DonGia = '1000000' và SoNguoi = '3'.

```
orientdb {db=CN02}> Insert into LOAIPHONG (MALOAIPH, TENLOAIPH,DONGIA,SONGDOI) values ('LPHONG0006','Double Room',1000000,3)
Inserted record '[LoaiPhong#91:1{MALOAIPH:LPHONG0006,TENLOAIPH:Double Room,DONGIA:1000000,SONGDOI:3} v1]' in 0.112000 sec(s).
orientdb {db=CN02}> SELECT * FROM LOAIPHONG
```

#	@RID	@CLASS	MALOAIPH	TENLOAIPH	SONGDOI	DONGIA	in_ThuocLoaiPhong	in_
0	#90:0	LoaiPhong	LPHONG0001	Standard Room	3	800000		
1	#90:1	LoaiPhong	LPHONG0005	Connecting Room	5	3000000		
2	#91:0	LoaiPhong	LPHONG0002	Superior Room	3	1500000	[#122:0,#128:0,#129:0]	
3	#91:1	LoaiPhong	LPHONG0006	Double Room	3	1000000		
4	#92:0	LoaiPhong	LPHONG0003	Deluxe Room	3	2000000	[#124:0,#125:0,#126:0,#127:0,#123:0]	[]
5	#93:0	LoaiPhong	LPHONG0004	Suite Room	3	1700000		
6	#94:0	LoaiPhong	LPHONG0005	Connecting Room	5	3000000	[]	

```
7 item(s) found. Query executed in 0.24 sec(s).
```

Hình 41: Kết quả sau khi thực hiện lệnh Insert.

Update

```
orientdb {db=CN02}> Insert into LOAIPHONG (MALOAIPH, TENLOAIPH,DONGIA,SONGUOI) values ('LPHONG0006','Double Room',1000000,3)
Inserted record '[LoaiPhong#91:1{MALOAIPH:LPHONG0006,TENLOAIPH:Double Room,DONGIA:1000000,SONGUOI:3} v1]' in 0.112000 sec(s).
orientdb {db=CN02}> SELECT * FROM LOAIPHONG
```

#	@RID	@CLASS	MALOAIPH	TENLOAIPH	SONGUOI	DONGIA	in_ThuocLoaiPhong	in_
0	#90:0	LoaiPhong	LPHONG0001	Standard Room	3	800000		
1	#90:1	LoaiPhong	LPHONG0005	Connecting Room	5	3000000		
2	#91:0	LoaiPhong	LPHONG0002	Superior Room	3	1500000	[#122:0,#128:0,#129:0]	
3	#91:1	LoaiPhong	LPHONG0006	Double Room	3	1000000		
4	#92:0	LoaiPhong	LPHONG0003	Deluxe Room	3	2000000	[#124:0,#125:0,#126:0,#127:0,#123:0]	[]
5	#93:0	LoaiPhong	LPHONG0004	Suite Room	3	1700000		
6	#94:0	LoaiPhong	LPHONG0005	Connecting Room	5	3000000	[]	

```
7 item(s) found. Query executed in 0.24 sec(s).
```

Hình 42: Kết quả truy vấn của NhanVien trước khi thực hiện lệnh Update.

- Thực hiện Update tại bảng LOAIPHONG của máy 2 với bản ghi có MaLoaiPhong = 'LPHONG0006' với dữ liệu được update là DonGia = 2500000

```
orientdb {db=CN02}> UPDATE LoaiPhong SET DONGIA = 2500000 WHERE MALOAIPH = 'LPHONG0006'
Updated record(s) '[{count:1}]' in 0.091000 sec(s).
orientdb {db=CN02}> SELECT * FROM LOAIPHONG
```

#	@RID	@CLASS	MALOAIPH	TENLOAIPH	SONGUOI	DONGIA	in_ThuocLoaiPhong	in_
0	#90:0	LoaiPhong	LPHONG0001	Standard Room	3	800000		
1	#90:1	LoaiPhong	LPHONG0005	Connecting Room	5	3000000		
2	#91:0	LoaiPhong	LPHONG0002	Superior Room	3	1500000	[#122:0,#128:0,#129:0]	
3	#91:1	LoaiPhong	LPHONG0006	Double Room	3	2500000		
4	#92:0	LoaiPhong	LPHONG0003	Deluxe Room	3	2000000	[#124:0,#125:0,#126:0,#127:0,#123:0]	[]
5	#93:0	LoaiPhong	LPHONG0004	Suite Room	3	1700000		
6	#94:0	LoaiPhong	LPHONG0005	Connecting Room	5	3000000	[]	

```
7 item(s) found. Query executed in 0.093 sec(s).
```

Hình 43: Kết quả sau khi thực hiện lệnh Update.

Delete

```
orientdb {db=CN02}> UPDATE LoaiPhong SET DONGIA = 2500000 WHERE MALOAIPH = 'LPHONG0006'

Updated record(s) ' [{count:1}] ' in 0.091000 sec(s).

orientdb {db=CN02}> SELECT * FROM LOAIPHONG
```

#	@RID	@CLASS	MALOAIPH	TENLOAIPH	SONGUOI	DONGIA	in_ThuocLoaiPhong	in_
0	#90:0	LoaiPhong	LPHONG0001	Standard Room	3	800000		
1	#90:1	LoaiPhong	LPHONG0005	Connecting Room	5	3000000		
2	#91:0	LoaiPhong	LPHONG0002	Superior Room	3	1500000	[#122:0,#128:0,#129:0]	
3	#91:1	LoaiPhong	LPHONG0006	Double Room	3	2500000		
4	#92:0	LoaiPhong	LPHONG0003	Deluxe Room	3	2000000	[#124:0,#125:0,#126:0,#127:0,#123:0]	[]
5	#93:0	LoaiPhong	LPHONG0004	Suite Room	3	1700000		
6	#94:0	LoaiPhong	LPHONG0005	Connecting Room	5	3000000	[]	

```
7 item(s) found. Query executed in 0.093 sec(s).
```

Hình 44: Kết quả truy vấn của NhanVien trước khi thực hiện lệnh Delete.

- Thực hiện Delete ở bảng LOAIPHONG của máy 2 với bản ghi có MaLoaiPhong = 'LPHONG0006'

```
orientdb {db=CN02}> DELETE VERTEX FROM LOAIPHONG WHERE MALOAIPH = 'LPHONG0006'

Delete record(s) ' [{count:1}] ' in 0.112000 sec(s).

orientdb {db=CN02}> SELECT * FROM LOAIPHONG
```

#	@RID	@CLASS	MALOAIPH	TENLOAIPH	SONGUOI	DONGIA	in_ThuocLoaiPhong	in_
0	#90:0	LoaiPhong	LPHONG0001	Standard Room	3	800000		
1	#90:1	LoaiPhong	LPHONG0005	Connecting Room	5	3000000		
2	#91:0	LoaiPhong	LPHONG0002	Superior Room	3	1500000	[#122:0,#128:0,#129:0]	
3	#92:0	LoaiPhong	LPHONG0003	Deluxe Room	3	2000000	[#124:0,#125:0,#126:0,#127:0,#123:0]	[]
4	#93:0	LoaiPhong	LPHONG0004	Suite Room	3	1700000		
5	#94:0	LoaiPhong	LPHONG0005	Connecting Room	5	3000000	[]	

```
6 item(s) found. Query executed in 0.092 sec(s).
```

Hình 45: Kết quả sau khi thực hiện lệnh Delete.

TÀI LIỆU THAM KHẢO

- [1] <https://orientdb.com/docs/last/introduction/>
- [2] <https://orientdb.com/docs/last/datamodeling/Tutorial-Document-and-graph-model.html/>
- [3] <https://gextend.net/threads/gioi-thieu-co-so-du-lieu-nosql-da-mo-hinh-orientdb.320/>
- [4] <https://orientdb.com/docs/last/gettingstarted/Tutorial-Run-the-server.html>
- [5] <https://orientdb.org/docs//2.0/orientdb.wiki/Tutorial-Setup-a-distributed-database.html>