**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA HỆ THỐNG THÔNG TIN**



**CƠ SỞ DỮ LIỆU PHÂN TÁN**

**BÁO CÁO BÀI TẬP LỚN 1:**

**THIẾT KẾ CƠ SỞ DỮ LIỆU PHÂN TÁN**

**Lớp: IS211.M11.HTCL**

**Giảng viên hướng dẫn:** ThS.Thái Bảo Trân

Thầy Nguyễn Minh Nhựt

**Nhóm: 9**

**Sinh viên thực hiện:**

Huỳnh Minh Thư 19522304

Phan Phạm Quỳnh Hoa 19521520

Nguyễn Thị Phương Nhi 19520814

***TP. Hồ Chí Minh, ngày 05 tháng 12 năm 2021***

Contents

[**Yêu cầu 1:** Thiết kế CSDL Phân tán trên môi trường máy ảo, radmin 3](#_Toc90233736)

[**I:** **CẤU TRÚC BẢNG** 3](#_Toc90233737)

[**II: KIẾN TRÚC PHÂN MẢNH** 4](#_Toc90233738)

[**III: KIẾN TRÚC PHÂN QUYỀN** 5](#_Toc90233739)

[**IV: CÁC CÂU TRUY VẤN** 8](#_Toc90233740)

[**Yêu cầu 2:** Viết hàm, thủ tục, ràng buộc toàn vẹn truy vấn trên môi trường phân tán 13](#_Toc90233741)

[**I: HÀM VÀ THỦ TỤC TRÊN MÔI TRƯỜNG PHÂN TÁN** 13](#_Toc90233742)

[**II: RÀNG BUỘC TOÀN VẸN** 15](#_Toc90233743)

[**Yêu cầu 3:** Demo các mức cô lập (ISOLATION LEVEL) trong môi trường phân tán 17](#_Toc90233744)

[**Yêu cầu 4:** Tối ưu hóa truy vấn trên môi trường phân tán 21](#_Toc90233745)

[**Tài liệu tham khảo** 27](#_Toc90233746)

# **Yêu cầu 1:** Thiết kế CSDL Phân tán trên môi trường máy ảo, radmin

## **I:** **CẤU TRÚC BẢNG**

**Nội dung**

* Quản lý các chi nhánh thuộc Mochi Hotel
* Mochi Hotel có 3 khách sạn tại Việt Nam
* Lược đồ cơ sở dữ liệu của mỗi khách sạn như sau:

**KHACHSAN** (MAKS, TENKS, THANHPHO, SDT)

Mỗi khách sạn có mã khách sạn, tên khách sạn, thành phố và số điện thoại của khách sạn.

**PHONGBAN** (MAPB, TENPB)

Mỗi phòng ban có mã phòng ban, tên phòng ban.

**CHUCVU** (MACV, TENCV)

Mỗi chức vụ có một mã chức vụ (MACV), tên chức vụ (TENCV)

**NHANVIEN** (MANV, TENNV, MACV, MAPB, MAKS, NGAYBATDAU, NGAYKETTHUC, TRANGTHAI, LUONG)

Mỗi nhân viên có một mã nhân viên (MANV) duy nhất để phân biệt với các nhân viên khác, tên nhân viên (TENNV), mã chức vụ (MACV), mã phòng ban (MAPB) và mã khách sạn (MAKS) nhân viên làm việc, ngày nhân viên bắt đầu vào làm, ngày nghỉ việc (nếu có), trạng thái (TRANGTHAI) 0 đang làm việc hay 1 đã nghỉ việc và mức lương (LUONG) của người đó.

**LOAIPHONG** (MALP, GIA, MOTA)

Mỗi loại phòng có một mã loại phòng (MALP), giá loại phòng(GIA) và mô tả (MOTA) (phòng VIP, phòng đôi, phòng đơn, phòng gia đình).

**PHONG** (MAP, SOPHONG, MAKS, MALP, MOTA)

Mỗi phòng có một mã phòng duy nhất (MAP), số phòng (SOPHONG), mã khách sạn (MAKS), mã loại phòng (MALP) và mô tả phòng (phòng tầng mấy, có view biển hay kh…)

**KHACHHANG** (MAKH, TENKH, CMND, SODT, QUOCTICH, GIOITINH)

Mỗi khách hàng có một mã khách hàng duy nhất, tên khách hàng, chứng minh nhân dân, quốc tịch, giới tính(NAM/NỮ)

**DATPHONG** (MADP,MAKS, MAKH, MAP, NGAYBATDAU, NGAYKETTHUC, MANV)

Mỗi khách hàng khi đặt phòng tại một khách sạn được mô tả thông qua bảng DATPHONG bao gồm mã đạt phòng (MADP),mã khách sạn(MAKS), mã khách hàng đặt (MAKH), mã phòng(MAP), ngày check in (NGAYBATDAU), ngày check out (NGAYKETTHUC), mã nhân viên nhận đặt phòng (MANV).

**HOADON** (MAHD, MADP, NGHD, THANHTIEN)

Mỗi hóa đơn có một mã hóa đơn (MAHD) duy nhất, mã đặt phòng (MADP), ngày tạo hóa đơn (NGHD), thành tiền (THANHTIEN)

## **II: KIẾN TRÚC PHÂN MẢNH**

Quan hệ KHACHSAN phân mảnh ngang chính theo thành phố:

KHACHSAN1 = σ ThanhPho = 'TP Ha Giang’ KHACHSAN

KHACHSAN2 = σ ThanhPho = 'TP Dien Bien Phu’ KHACHSAN

Quan hệ PHONG phân mảnh ngang dẫn xuất như sau:

PHONG1 = PHONG MaKS KHACHSAN1

PHONG2 = PHONG MaKS KHACHSAN2

Quan hệ NHANVIEN phân mảnh ngang dẫn xuất như sau:

NHANVIEN1 = NHANVIEN MaKS KHACHSAN1

NHANVIEN2 = NHANVIEN MaKS KHACHSAN2

Quan hệ DATPHONG phân mảnh ngang dẫn xuất như sau:

DATPHONG1 = DATPHONG MaKS KHACHSAN1

DATPHONG2 = DATPHONG MaKS KHACHSAN2

Quan hệ KHACHHANG được nhân bản ở tất cả khách sạn.

## **III: KIẾN TRÚC PHÂN QUYỀN**

**Chi nhánh 1:** Có 3 quyền

GIAMDOC: Xem được thông tin tất cả các quan hệ chi nhánh 1, chi nhánh 2, Connect.

QUANLY: Xem được NHANVIEN, CHUCVU, HOADON, PHONG, KHACHHANG của chi nhánh 1 , 2.

LETAN: Xem được HOADON, KHACHHANG, DATPHONG của chi nhánh 1.

**Chi nhánh 2:** Có 2 quyền

QUANLY: Xem được NHANVIEN, CHUCVU, HOADON, PHONG của chi nhánh 1,2.

LETAN: Xem được HOADON, KHACHHANG, DATPHONG của chi nhánh 1.

1. **Khách sạn 1** 
   1. **Tạo User ban đầu: Khách sạn**

**CREATE** USER KS1 **IDENTIFIED** BY KS1;

**GRANT** CONNECT, DBA **TO** KS1;

* 1. **Tạo quyền**

**CREATE** ROLE r\_GIAMDOC;

**CREATE** ROLE r\_QUANLY;

**CREATE** ROLE r\_LETAN;

* 1. **Phân quyền**
* **Giám đốc**

GRANT SELECT ON KS1.KHACHSAN TO r\_GIAMDOC;

GRANT SELECT ON KS1.PHONGBAN TO r\_GIAMDOC;

GRANT SELECT ON KS1.CHUCVU TO r\_GIAMDOC;

GRANT SELECT ON KS1.NHANVIEN TO r\_GIAMDOC;

GRANT SELECT ON KS1.LOAIPHONG TO r\_GIAMDOC;

GRANT SELECT ON KS1.PHONG TO r\_GIAMDOC;

GRANT SELECT ON KS1.KHACHHANG TO r\_GIAMDOC;

GRANT SELECT ON KS1.DATPHONG TO r\_GIAMDOC;

GRANT SELECT ON KS1.HOADON TO r\_GIAMDOC;

* **Quản lý**

GRANT SELECT ON KS1.CHUCVU TO r\_QUANLY;

GRANT SELECT, INSERT, UPDATE, DELETE ON KS1.NHANVIEN TO r\_QUANLY;

GRANT SELECT ON KS1.PHONG TO r\_QUANLY;

GRANT SELECT ON KS1.KHACHHANG TO r\_QUANLY;

GRANT SELECT ON KS1.DATPHONG TO r\_QUANLY;

GRANT SELECT ON KS1.HOADON TO r\_QUANLY;

* **Lễ tân**

GRANT SELECT ON KS1.KHACHHANG TO r\_LETAN;

GRANT SELECT ON KS1.DATPHONG TO r\_LETAN;

GRANT SELECT ON KS1.HOADON TO r\_LETAN;

* 1. **Tạo các User Giám đốc, Quản lý, Lễ tân**
* **Giám đốc**

CREATE USER GIAMDOC IDENTIFIED BY GIAMDOC;

GRANT CONNECT TO GIAMDOC;

GRANT CREATE DATABASE LINK TO GIAMDOC;

GRANT r\_GIAMDOC TO GIAMDOC;

* **Quản lý**

CREATE USER QUANLY IDENTIFIED BY QUANLY;

GRANT CONNECT TO QUANLY;

GRANT CREATE DATABASE LINK TO QUANLY;

GRANT r\_QUANLY TO QUANLY;

* **Lễ tân**

CREATE USER LETAN IDENTIFIED BY LETAN;

GRANT CONNECT TO LETAN;

GRANT r\_LETAN TO LETAN;

1. **Khách sạn 2** 
   1. **Tạo User ban đầu: Khách sạn 2**

**CREATE** USER KS2 **IDENTIFIED** BY KS2;

**GRANT** CONNECT, **DBA** TO KS2;

* 1. **Tạo quyền**

**CREATE ROLE** r\_GIAMDOC;

**CREATE ROLE** r\_QUANLY;

**CREATE ROLE** r\_LETAN;

* 1. **Phân quyền**
* **Cấp quyền cho r\_GIAMDOC**

GRANT SELECT ON KS2.KHACHSAN TO r\_GIAMDOC;

GRANT SELECT ON KS2.PHONGBAN TO r\_GIAMDOC;

GRANT SELECT ON KS2.CHUCVU TO r\_GIAMDOC;

GRANT SELECT ON KS2.NHANVIEN TO r\_GIAMDOC;

GRANT SELECT ON KS2.LOAIPHONG TO r\_GIAMDOC;

GRANT SELECT ON KS2.PHONG TO r\_GIAMDOC;

GRANT SELECT ON KS2.KHACHHANG TO r\_GIAMDOC;

GRANT SELECT ON KS2.DATPHONG TO r\_GIAMDOC;

GRANT SELECT ON KS2.HOADON TO r\_GIAMDOC;

* **Cấp quyền cho r\_QUANLY**

GRANT SELECT ON KS2.CHUCVU TO r\_QUANLY;

GRANT SELECT, INSERT, UPDATE, DELETE ON KS2.NHANVIEN TO r\_QUANLY;

GRANT SELECT ON KS2.PHONG TO r\_QUANLY;

GRANT SELECT ON KS2.KHACHHANG TO r\_QUANLY;

GRANT SELECT ON KS2.DATPHONG TO r\_QUANLY;

GRANT SELECT ON KS2.HOADON TO r\_QUANLY;

* **Cấp quyền cho r\_LETAN**

GRANT SELECT ON KS2.KHACHHANG TO r\_LETAN;

GRANT SELECT ON KS2.DATPHONG TO r\_LETAN;

GRANT SELECT ON KS2.HOADON TO r\_LETAN;

* 1. **Tạo các User Giám đốc, Quản lý, Lễ tân**
* **Giám đốc**

CREATE USER GIAMDOC IDENTIFIED BY GIAMDOC;

GRANT CONNECT TO GIAMDOC;

GRANT CREATE DATABASE LINK TO GIAMDOC;

GRANT r\_GIAMDOC TO GIAMDOC;

* **Quản lý**

CREATE USER QUANLY IDENTIFIED BY QUANLY;

GRANT CONNECT TO QUANLY;

GRANT CREATE DATABASE LINK TO QUANLY;

GRANT r\_QUANLY TO QUANLY;

* **Lễ tân**

CREATE USER LETAN IDENTIFIED BY LETAN;

GRANT CONNECT TO LETAN;

GRANT r\_LETAN TO LETAN;

## **IV: CÁC CÂU TRUY VẤN**

* Các câu truy vấn tâm đắc sẽ được đánh dấu bằng biểu tượng .

**Khách sạn 1**

|  |  |
| --- | --- |
|  | **Câu 1:** Tài khoản lễ tân:  In ra danh sách các hóa đơn được xuất ra trong ngày 22/03/2021 hoặc hóa đơn có khách hàng mang quốc tịch ‘Bhutan’ đặt.  *Ý nghĩa:*  *Truy vết hóa đơn của các khách hàng đến khách sạn vào 1 ngày cụ thể hoặc tìm khách hàng theo quốc tịch.* |
|  | SELECT \*  FROM KS2.HOADON@KS2\_QL  WHERE NGHD = '22/03/2021'  UNION  SELECT hd.\*  FROM KS2.HOADON@KS2\_QL hd, KS2.DATPHONG@KS2\_QL dp, KS2.KHACHHANG@KS2\_QL kh  WHERE hd.MADP = dp.MADP AND dp.MAKH = kh.MAKH AND kh.QUOCTICH = 'Bhutan'; |
|  | **Câu 2:** Tài khoản quản lý:  Tìm khách hàng đã từng đặt tất cả các loại phòng tại cả 2 chi nhánh.  *Ý nghĩa:*  *Tìm những khách hàng tiềm năng, khách hàng sử dụng nhiều dịch vụ của khách sạn sau đó nâng hạng khách hàng, tặng các gói ưu đãi,…* |
|  | SELECT \*  FROM KS1.KHACHHANG kh  WHERE NOT EXISTS  (SELECT \*  FROM KS1.LOAIPHONG  WHERE MALP NOT IN  (SELECT DISTINCT p.MALP  FROM KS1.DATPHONG dp JOIN KS1.PHONG p  ON dp.MAP = p.MAP  WHERE kh.MAKH = dp.MAKH))  UNION  SELECT \*  FROM KS2.KHACHHANG@KS2\_GD kh  WHERE NOT EXISTS  (SELECT \*  FROM KS2.LOAIPHONG@KS2\_GD  WHERE MALP NOT IN  (SELECT DISTINCT p.MALP  FROM KS2.DATPHONG@KS2\_GD dp JOIN  KS2.PHONG@KS2\_GD p ON dp.MAP = p.MAP  WHERE kh.MAKH = dp.MAKH)); |
|  | **Câu 3:** Tài khoản quản lý:  In ra danh sách nhân viên LỄ TÂN (MACV=’CV2’) chưa nhận đơn đặt phòng trong tháng 3 năm 2021 của chi nhánh 2.  *Ý nghĩa:*  *Đánh giá hiệu suất làm việc của nhân viên cuối tháng, dùng để tính lương hoặc xem xét thưởng phạt.* |
|  | SELECT MANV, TENNV  FROM KS2.NHANVIEN@KS2\_QL  WHERE MACV = 'CV2'  MINUS  SELECT nv.MANV, TENNV  FROM KS2.NHANVIEN@KS2\_QL nv JOIN KS2.DATPHONG@KS2\_QL dp  ON nv.MANV = dp.MANV  WHERE EXTRACT (MONTH FROM dp.NGAYBATDAU) = '03' AND  EXTRACT (YEAR FROM dp.NGAYBATDAU) = '2021' AND  nv.MACV = 'CV2'; |
|  | **Câu 4:** Tài khoản quản lý:  Tìm nhân viên (MANV, TENNV) có số lần nhận đặt phòng nhiều nhất trong tháng 9 năm 2021.  *Ý nghĩa:*  *Đánh giá năng lực nhân viên, xem xét tăng lương, tặng thưởng hay thăng chức.* |
|  | SELECT NV.MANV, TENNV, COUNT(MADP) AS "Tong lan dat phong"  FROM KS1.NHANVIEN nv, KS1.DATPHONG dp  WHERE nv.MANV = dp.MANV AND EXTRACT (MONTH FROM dp.NGAYBATDAU) = '09' AND EXTRACT (YEAR FROM dp.NGAYBATDAU) = '2021'  GROUP BY NV.MANV, TENNV  HAVING COUNT(MADP) >= ALL (SELECT COUNT(MADP)  FROM KS1.NHANVIEN nv, KS1.DATPHONG dp  WHERE nv.MANV = dp.MANV AND EXTRACT (MONTH FROM dp.NGAYBATDAU) = '09' AND EXTRACT (YEAR FROM dp.NGAYBATDAU) = '2021'  GROUP BY NV.MANV, TENNV)  UNION  SELECT NV.MANV, TENNV, COUNT(MADP) AS "Tong lan dat phong"  FROM KS2.NHANVIEN@KS2\_QL nv1, KS2.DATPHONG@KS2\_QL dp1  WHERE nv1.MANV = dp1.MANV AND EXTRACT (MONTH FROM dp1.NGAYBATDAU) = '09' AND EXTRACT (YEAR FROM dp1.NGAYBATDAU) = '2021'  GROUP BY NV1.MANV, TENNV  HAVING COUNT(MADP) >= ALL (SELECT COUNT(MADP)  FROM KS2.NHANVIEN@KS2\_QL nv1, KS2.DATPHONG@KS2\_QL dp1  WHERE nv1.MANV = dp1.MANV AND EXTRACT (MONTH FROM dp1.NGAYBATDAU) = '09' AND EXTRACT (YEAR FROM dp1.NGAYBATDAU) = '2021'  GROUP BY NV1.MANV, TENNV) |
| Like Bee | **Câu 5:** Tài khoản giám đốc:  Trong mỗi loại phòng tìm phòng được đặt nhiều nhất trong năm 2021 của chi nhánh 2.  *Ý nghĩa:*  *Kiểm tra và nâng cấp trang thiết bị, các loại phòng được đặt nhiều có thể đầu tư xây thêm để đáp ứng nhu cầu cho khách hàng* |
|  | SELECT P.MALP, P.MAP, COUNT(DP.MADP) soLanDat  FROM KS2.DATPHONG@KS2\_QL DP, KS2.PHONG@KS2\_QL P  WHERE DP.MAP = P.MAP AND EXTRACT (YEAR FROM NGAYBATDAU)  ='2021'  GROUP BY P.MALP, P.MAP  HAVING COUNT(DP.MADP) >= ALL (SELECT COUNT(DP.MADP)  soLanDat  FROM KS2.DATPHONG@KS2\_QL DP, KS2.PHONG@KS2\_QL PH  WHERE DP.MAP =PH.MAP AND EXTRACT (YEAR FROM NGAYBATDAU)  ='2021'  GROUP BY PH.MALP, PH.MAP  HAVING P.MALP = PH.MALP); |
| Like Bee | **Câu 6:** Tài khoản giám đốc:  Tính doanh thu trung bình của khách sạn theo từng tháng trong năm 2021 sắp xếp theo doanh thu từ thấp đến cao. Của chi nhánh 1 và chi nhánh 2.  *Ý nghĩa:*  *Thống kê doanh thu của khách sạn theo từng tháng trong năm, các tháng có doanh thu cao là những tháng thu hút lượng khách hàng lớn, cần đầu tư hơn về quy trình CSKH và những dự án marketing cho khách sạn.* |
|  | SELECT DP.MAKS, EXTRACT(MONTH FROM NGHD),  AVG(HD.THANHTIEN) AS "TB DOANH THU"  FROM KS1.HOADON HD, KS1.DATPHONG DP, KS1.KHACHSAN KS  WHERE HD.MADP = DP.MADP AND DP.MAKS = KS.MAKS AND  EXTRACT (YEAR FROM NGHD) = '2021'  GROUP BY DP.MAKS, TENKS, EXTRACT(MONTH FROM NGHD)  ORDER BY 'TB DOANH THU' DESC  UNION  SELECT DP.MAKS, EXTRACT(MONTH FROM NGHD),  AVG(HD.THANHTIEN) AS "TB DOANH THU"  FROM KS2.HOADON@KS2\_GD HD, KS2.DATPHONG@KS2\_GD DP,  KS1.KHACHSAN@KS2\_GD KS  WHERE HD.MADP = DP.MADP AND DP.MAKS = KS.MAKS AND EXTRACT (YEAR FROM NGHD) = '2021'  GROUP BY DP.MAKS, TENKS, EXTRACT(MONTH FROM NGHD)  ORDER BY 'TB DOANH THU' DESC; |

**Khách sạn 2**

|  |
| --- |
| **Câu 7:** Tài khoản quản lý:  Tìm các phòng đã được đặt trong ngày 20/11/2021 và thông tin kh đã đặt phòng đó (makh, tenkh) tại cả 2 chi nhánh.  *Ý nghĩa:*  *Tìm khách hàng đã đặt phòng nhằm xin feedback về trải nghiệm trong ngày đó.* |
| SELECT DP.MAP, DP.MAKS,KH.MAKH, TENKH  FROM KS2.DATPHONG DP, KS2.KHACHHANG KH  WHERE DP.MAKH = KH.MAKH AND NGAYBATDAU = '20/11/2021'  UNION  SELECT DP1.MAP, DP1.MAKS,KH1.MAKH, TENKH  FROM KS1.DATPHONG@KS1\_QL DP1, KS1.KHACHHANG@KS1\_QL KH1  WHERE DP1.MAKH = KH1.MAKH AND NGAYBATDAU = '20/11/2021' |
| **Câu 8:** Tài khoản quản lý:  Tìm khách hàng đã từng đặt phòng ở cả chi nhánh 1 và chi nhánh 2.  *Ý nghĩa:*  *Bộ phận cskh xin feedback của khách hàng xem trải nghiệm ở chi nhánh nào tốt hơn để rút kinh nghiệm, nâng cấp chất lượng phục vụ.* |
| SELECT kh.\*  FROM KS2.DATPHONG dp JOIN KS2.KHACHHANG kh ON dp.MAKH = kh.MAKH  INTERSECT  SELECT kh1.\*  FROM KS1.DATPHONG@KS1\_QL dp1 JOIN KS1.KHACHHANG@KS1\_QL kh1 ON dp1.MAKH = kh1.MAKH; |
| **Câu 9:** Tài khoản quản lý:  Tính tổng số lần được đặt của mỗi phòng trong năm 2021, ở khách sách 1 và khách sạn 2.  *Ý nghĩa:*  *Thống kê xem nhu cầu loại phòng mà khách hàng ưu thích để nâng cấp mở rộng đáp ứng nhu cầu khách hàng.* |
| SELECT MAP, COUNT(MADP) soLanDat  FROM KS2.DATPHONG  WHERE EXTRACT (YEAR FROM NGAYBATDAU) = '2021'  GROUP BY MAP  UNION  SELECT MAP, COUNT(MADP) soLanDat  FROM KS1.DATPHONG@KS1\_QL  WHERE EXTRACT (YEAR FROM NGAYBATDAU) = '2021'  GROUP BY MAP |
| **Câu 10 :** Tài khoản quản lý:  In ra danh sách thông tin các phòng (MAP, SOPHONG) không được đặt trong tháng 12/2021 ở khách sạn 1 và khách sạn 2.  *Ý nghĩa:*  *Thống kê xem nhu cầu loại phòng trong tháng để cải thiện chất lượng phục vụ từng phòng trong tháng cao điểm.* |
| SELECT MAP, SOPHONG  FROM KS2.PHONG  MINUS  SELECT p.MAP, SOPHONG  FROM KS2.PHONG p JOIN KS2.DATPHONG dp ON p.MAP = dp.MAP  WHERE EXTRACT (MONTH FROM NGAYBATDAU) = '12' AND EXTRACT  (YEAR FROM NGAYBATDAU) = '2021'  UNION  SELECT MAP, SOPHONG  FROM KS1.PHONG@KS1\_QL  MINUS  SELECT p1.MAP, SOPHONG  FROM KS1.PHONG@KS1\_QL p1 JOIN KS1.DATPHONG@KS1\_QL dp1  ON p1.MAP = dp1.MAP  WHERE EXTRACT (MONTH FROM NGAYBATDAU) = '12' AND EXTRACT  (YEAR FROM NGAYBATDAU) = '2021' |

# **Yêu cầu 2:** Viết hàm, thủ tục, ràng buộc toàn vẹn truy vấn trên môi trường phân tán

## **I: HÀM VÀ THỦ TỤC TRÊN MÔI TRƯỜNG PHÂN TÁN**

|  |
| --- |
| **FUNCTION (Hàm)**  Viết hàm nhập vào chứng minh nhân dân của khách hàng, năm in ra tổng số lần đặt phòng của khách hàng trong năm đó. |
| CREATE OR REPLACE FUNCTION  findKH (v\_cmnd in varchar2, v\_year NUMBER)  RETURN NUMBER  AS  sum1 NUMBER;  sum2 NUMBER;  BEGIN  sum1:=0; sum2:=0;  SELECT COUNT(MADP) INTO sum1  FROM KS2.KHACHHANG kh JOIN KS2.DATPHONG dp ON  kh.MAKH=dp.MAKH  WHERE CMND = v\_cmnd AND EXTRACT (YEAR FROM NGAYBATDAU) = v\_year;  SELECT COUNT(MADP) INTO sum2  FROM KS1.KHACHHANG@KS1\_GD kh1 JOIN  KS1.DATPHONG@KS1\_GD dp1 ON kh1.MAKH=dp1.MAKH  WHERE CMND = v\_cmnd AND EXTRACT (YEAR FROM NGAYBATDAU)  = v\_year;  RETURN sum1+sum2;  --EXCEPTION WHEN NO\_DATA\_FOUND THEN RETURN 0;  END;  --gọi hàm  DECLARE  v\_cmnd varchar2(10);  v\_year varchar2(4);  kq number;  BEGIN  v\_cmnd := '131813584';  v\_year := '2021';  kq := findKH(v\_cmnd, v\_year);  DBMS\_OUTPUT.PUT\_LINE('khach hang co CMND: ' || v\_cmnd || ' trong nam:' || v\_year || ' dat phong:' || kq || ' lan');  END; |
| **PROCEDURE (Thủ tục)**  Viết procedure tăng lương cho các nhân viên có số lần đặt phòng nhiều nhất trong tháng 9 năm 2021 và in ra tên, lương sau khi update của các nhân viên đó. |
| CREATE OR REPLACE PROCEDURE  raise\_salary (amount IN NUMBER, nv\_name OUT VARCHAR2,  salary OUT NUMBER) IS  CURSOR top IS  SELECT NV.MANV, COUNT(MADP) AS "Tong lan dat phong"  FROM KS2.NHANVIEN@KS2\_QL nv, KS2.DATPHONG@KS2\_QL dp  WHERE nv.MANV = dp.MANV AND EXTRACT (MONTH FROM  dp.NGAYBATDAU) = '09' AND EXTRACT (YEAR FROM  dp.NGAYBATDAU) = '2021'  GROUP BY NV.MANV  HAVING COUNT(MADP) >= ALL (SELECT COUNT(MADP)  FROM KS2.NHANVIEN@KS2\_QL nv, KS2.DATPHONG@KS2\_QL dp  WHERE nv.MANV = dp.MANV AND EXTRACT (MONTH FROM  dp.NGAYBATDAU) = '09' AND EXTRACT (YEAR FROM  dp.NGAYBATDAU) = '2021'  GROUP BY NV.MANV);    nv\_top top%ROWTYPE;    BEGIN  OPEN top;  LOOP  FETCH top INTO nv\_top;  EXIT WHEN top%NOTFOUND;    UPDATE KS2.NHANVIEN@KS2\_QL SET LUONG = LUONG + amount  WHERE MANV = nv\_top.MANV;  SELECT TENNV, LUONG INTO nv\_name, salary  FROM KS2.NHANVIEN@KS2\_QL WHERE MANV = nv\_top.MANV;    DBMS\_OUTPUT.PUT\_LINE( 'Employee ID: ' ||nv\_top.MANV ||  ' Name: ' || nv\_name || ' Salary: ' || salary );  END LOOP;  CLOSE top;  END raise\_salary;  ---- Gọi hàm  DECLARE  salary number;  nvname VARCHAR2(25);  BEGIN  raise\_salary(2000000, nvname,salary);  END; |

## **II: RÀNG BUỘC TOÀN VẸN**

|  |
| --- |
| **TRIGGER**  Khách hàng đặt một phòng không được trùng vào khoảng thời gian do khách hàng khác đặt trước cùng phòng đó. |
| **Nội dung:**  ∀k1, k2 ∈ DATPHONG (k1.MAP = k2.MAP → (k1.NGAYBATDAU, k1.NGAYKETTHUC ∉ [k2.NGAYBATDAU, k2.NGAYKETTHUC]) ∧ k2.NGAYBATDAU ∉ [k1.NGAYBATDAU, k1.NGAYKETTHUC] |
| **Bối cảnh:** DATPHONG |
| **Bảng tầm ảnh hưởng:** |
| CREATE OR REPLACE TRIGGER TRG\_DATPHONG\_UPD\_INS  BEFORE INSERT OR UPDATE ON DATPHONG  FOR EACH ROW  DECLARE  temp NUMBER := 0;  v\_ngbd KS2.DATPHONG.NGAYBATDAU%TYPE;  v\_ngkt KS2.DATPHONG.NGAYKETTHUC%TYPE;  CURSOR cur IS  SELECT KS2.DATPHONG.NGAYBATDAU,  KS2.DATPHONG.NGAYKETTHUC  FROM KS2.DATPHONG  WHERE KS2.DATPHONG.MAP = :NEW.MAP;  BEGIN  OPEN cur;  LOOP  FETCH cur INTO v\_ngbd, v\_ngkt;  EXIT WHEN cur%NOTFOUND;    IF(:NEW.NGAYBATDAU >= v\_ngbd AND :NEW.NGAYBATDAU  <= v\_ngkt)  THEN  temp := temp + 1;  ELSIF  (:NEW.NGAYKETTHUC >= v\_ngbd AND  :NEW.NGAYKETTHUC <= v\_ngkt) THEN  temp := temp + 1;  ELSIF  (:NEW.NGAYBATDAU <= v\_ngbd AND  :NEW.NGAYKETTHUC >= v\_ngkt)  THEN  temp := temp + 1;  END IF;  END LOOP;  IF (temp >0 )  THEN  RAISE\_APPLICATION\_ERROR  (-20987, 'DA CO KHACH DAT PHONG VAO NGAY: ' || v\_ngbd);  ELSE DBMS\_OUTPUT.PUT\_LINE('DAT PHONG THANH CONG');  END IF;  END; |

# **Yêu cầu 3:** Demo các mức cô lập (ISOLATION LEVEL) trong môi trường phân tán

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PHANTOM READ** | | | | |
| **Time** | **KS1** | **Output** | **KS2** | **Output** |
| t0 | ALTER SESSION SET ISOLATION\_LEVEL= READ COMMITTED; | Session altered. |  |  |
| t1 | SELECT \* FROM KS1.NHANVIEN; | 11 rows selected |  |  |
| t2 |  |  | INSERT INTO KS1.NHANVIEN@KS1\_QL VALUES ('NV30', 'Quynh Nhu', 'CV2', 'PB3', 'KS1', '08/11/2021', NULL, '0', '15000000');  Commit; | 1 row inserted. |
| t3 | SELECT \* FROM KS1.NHANVIEN; | 12 rows selected |  |  |
| **CÁCH KHẮC PHỤC** | | | | |
| t0 | SET TRANSACTION ISOLATION LEVEL SERIALIZABLE; | Session altered. |  |  |
| t1 | SELECT \* FROM KS1.NHANVIEN; | 12 rows selected |  |  |
| t2 |  |  | DELETE FROM KS1.NHANVIEN@KS1\_QL WHERE MANV = 'NV30'; Commit | 1 row deleted. |
| t3 | SELECT \* FROM KS1.NHANVIEN; | 12 rows selected |  |  |
| t4 | Commit;  SELECT \* FROM KS1.NHANVIEN; | 11 rows selected |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UNREPEATABE** | | | | |
| **Time** | **KS1** | **Output** | **KS2** | **Output** |
| t0 | ALTER SESSION SET ISOLATION\_LEVEL= READ COMMITTED; | Session altered. |  |  |
| t1 | SELECT MADP,MP FROM Datphong WHERE MADP = 'DP24'; | MADP MAP  -----------  DP24 P3 |  |  |
| t2 |  |  | Update KS1.Datphong@KS1\_QL set MAP = 'P4' WHERE MADP = 'DP24';  COMMIT; | 1 row updated |
| t3 | SELECT MADP,MP FROM Datphong WHERE MADP = 'DP24'; | MADP MAP  -----------  DP24 P4 |  |  |
| **CÁCH KHẮC PHỤC** | | | | |
| t0 | SET TRANSACTION ISOLATION LEVEL SERIALIZABLE; | Session altered. |  |  |
| t1 | SELECT MADP,MP FROM Datphong WHERE MADP = 'DP24'; | MADP MAP  -----------  DP24 P3 |  |  |
| t2 |  |  | Update KS1.Datphong@KS1\_QL set MAP = 'P4' WHERE MADP = 'DP24';  COMMIT; | 1 row updated |
| t3 | SELECT MADP,MP FROM Datphong WHERE MADP = 'DP24'; | MADP MAP  -----------  DP24 P3 |  |  |
| t4 | Commit;  SELECT \* FROM KS1.NHANVIEN; | MADP MAP  -----------  DP24 P4 |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **LOST UPDATE** | | | | |
| **Time** | **KS1** | **Output** | **KS2** | **Output** |
| t0 | ALTER SESSION SET ISOLATION\_LEVEL= READ COMMITTED; | Session altered. | ALTER SESSION SET ISOLATION\_LEVEL= READ COMMITTED; | Session altered. |
| t1 | SELECT \* FROM KS1.NHANVIEN; |  |  |  |
| t2 | Update KS1.NHANVIEN SET MACV = 'CV1', MAPB = 'PB1', LUONG = 5000000 WHERE MANV = 'NV1'; | 1 row updated. |  |  |
| t3 |  |  | Update KS1.NHANVIEN@KS1\_QL SET MACV = 'CV1', MAPB = 'PB1', LUONG = 44000000 WHERE MANV = 'NV1';  Commit; | 1 row updated |
| t4 | SELECT MACV,MAPB, LUONG FROM KS1.NHANVIEN WHERE MANV = ‘NV1’ | MACV MAPB LUONG  -----------------------  CV1 PB1 44000000 |  |  |
| **CÁCH KHẮC PHỤC** | | | | |
| t0 | SET TRANSACTION ISOLATION LEVEL SERIALIZABLE; | Session altered. | SET TRANSACTION ISOLATION LEVEL SERIALIZABLE; | Session altered. |
| t1 | Update KS1.NHANVIEN SET MACV = 'CV1', MAPB = 'PB1', LUONG = 5000000 WHERE MANV = 'NV1'; | 1 row updated |  |  |
| t2 |  |  | Update KS1.NHANVIEN@KS1\_QL SET MACV = 'CV1', MAPB = 'PB1', LUONG = 44000000 WHERE MANV = 'NV1';  Commit; |  |
| t3 | Commit; |  |  | Error report -  ORA-08177: can't serialize access for this transaction |
| t4 | SELECT MACV,MAPB, LUONG FROM KS1.NHANVIEN WHERE MANV = ‘NV1’ | MACV MAPB LUONG  -----------------------  CV1 PB1 5000000 |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEADLOCK** | | | | |
| **Time** | **KS1** | **Output** | **KS2** | **Output** |
| t0 | ALTER SESSION SET ISOLATION\_LEVEL= READ COMMITTED; | Session altered. | ALTER SESSION SET ISOLATION\_LEVEL= READ COMMITTED; | Session altered. |
| t1 | SELECT \* FROM KS1.PHONGBAN; |  |  |  |
| t2 | UPDATE KS1.PHONGBAN SET TENPB = 'Phong Dieu Hanh' WHERE MAPB = 'PB1'; | 1 row updated. | UPDATE KS1.PHONGBAN@KS1\_GD SET TENPB = 'Phong Ke Toan' WHERE MAPB = 'PB2'; | 1 row updated. |
| t3 | UPDATE KS1.PHONGBAN SET TENPB = 'Phong Tai Chinh' WHERE MAPB = 'PB2'; |  | UPDATE KS1.PHONGBAN@KS1\_GD SET TENPB = 'Phong Hanh Chinh' WHERE MAPB = 'PB1'; |  |
| t4 | Commit; | ORA-00060: deadlock detected while waiting for resource |  |  |
| t5 |  |  | Commit; | 1 row updated. |
| t6 | SELECT \* FROM KS1.PHONGBAN; | MAPB TENPB  --------------------  PB1 Phong Hanh Chinh  PB2 Phong Ke Toan  PB3 Phong CSKH |  |  |
| **CÁCH KHẮC PHỤC** | | | | |
| t0 | SET TRANSACTION ISOLATION LEVEL SERIALIZABLE; | Session altered. | SET TRANSACTION ISOLATION LEVEL SERIALIZABLE; | Session altered. |
| t1 | SELECT \* FROM KS1.PHONGBAN; |  |  |  |
| t2 | UPDATE KS1.PHONGBAN SET TENPB = 'Phong Dieu Hanh' WHERE MAPB = 'PB1'; | 1 row updated. | UPDATE KS1.PHONGBAN@KS1\_GD SET TENPB = 'Phong Ke Toan' WHERE MAPB = 'PB2'; | 1 row updated. |
| t3 | UPDATE KS1.PHONGBAN SET TENPB = 'Phong Tai Chinh' WHERE MAPB = 'PB2'; |  | UPDATE KS1.PHONGBAN@KS1\_GD SET TENPB = 'Phong Hanh Chinh' WHERE MAPB = 'PB1'; |  |
| t4 |  | ORA-00060: deadlock detected while waiting for resource |  |  |
| t5 | Commit; |  |  | Error report -  ORA-08177: can't serialize access for this transaction |
| t6 |  |  | Commit; |  |
| t7 | SELECT \* FROM KS1.PHONGBAN; | MAPB TENPB  --------------------  PB1 Phong Dieu Hanh PB2 Phong Ke Toan  PB3 Phong CSKH |  |  |

# **Yêu cầu 4: Tối ưu hóa truy vấn trên môi trường phân tán**

|  |
| --- |
| **Yêu cầu:** Thực hiện một câu truy vấn đơn giản chưa được tối ưu :  Lấy thông tin khách hàng (KH.MaKH, TenKH, CMND, DP.MaDP, DP.MaP, NGAYBATDAU) đã đặt loại phòng đơn tại khách sạn 'Ma Vi' và có quốc tịch 'Viet Nam' |
| SELECT KH.MaKH, TenKH, CMND, DP.MaDP, DP.MaP, NGAYBATDAU  FROM DATPHONG DP, KHACHHANG KH, KHACHSAN KS, PHONG P, LOAIPHONG LP  WHERE DP.MAKH = KH.MAKH AND DP.MAKS = KS.MAKS  AND DP.MAP = P.MAP AND LP.MALP = P.MALP  AND LP.MOTA = 'Don' AND TENKS = 'Ma Vi' AND QUOCTICH = 'Viet Nam' |
| **CHẠY EXPLAIN** |
| SELECT /\*+ GATHER\_PLAN\_STATISTICS \*/ KH.MaKH, TenKH, CMND, DP.MaDP, DP.MaP, NGAYBATDAU  FROM DATPHONG DP, KHACHHANG KH, KHACHSAN KS, PHONG P, LOAIPHONG LP  WHERE DP.MAKH = KH.MAKH AND DP.MAKS = KS.MAKS  AND DP.MAP = P.MAP AND LP.MALP = P.MALP  AND LP.MOTA = 'Don' AND TENKS = 'Ma Vi' AND QUOCTICH = 'Viet Nam'  SELECT \* FROM TABLE(DBMS\_XPLAN.display\_cursor(format=>'ALLSTATS LAST'));  **Kết quả Explain** |
| Graphical user interface, application, table  Description automatically generated |
| **Cây truy vấn tối ưu toàn cục** |
| Diagram  Description automatically generated |
| **Câu lệnh tối ưu** |
| SELECT C.MAP, C.MAKS, C.MAKH, NGAYBATDAU, TENKH, CMND FROM  (SELECT B.MAP, B.MAKS, MAKH, NGAYBATDAU FROM  (SELECT DP.MAP, MAKS, MAKH, NGAYBATDAU FROM  (SELECT P.MAP FROM (SELECT MALP FROM LOAIPHONG WHERE MOTA = 'Don') LP INNER JOIN (SELECT MAP, MALP FROM PHONG) P ON LP.MALP = P.MALP) A  INNER JOIN (SELECT MADP, MAP, MAKS, MAKH, NGAYBATDAU FROM DATPHONG) DP ON DP.MAP = A.MAP) B  INNER JOIN (SELECT MAKS FROM KHACHSAN WHERE TENKS='Ma Vi') KS ON B.MAKS = KS.MAKS ) C  INNER JOIN (SELECT MAKH, TENKH, CMND FROM KHACHHANG WHERE QUOCTICH = 'Viet Nam') KH ON C.MAKH = KH.MAKH |
| **CHẠY EXPLAIN** |
| SELECT /\*+ GATHER\_PLAN\_STATISTICS \*/ C.MAP, C.MAKS, C.MAKH, NGAYBATDAU, TENKH, CMND FROM  (SELECT B.MAP, B.MAKS, MAKH, NGAYBATDAU FROM  (SELECT DP.MAP, MAKS, MAKH, NGAYBATDAU FROM  (SELECT P.MAP FROM (SELECT MALP FROM LOAIPHONG WHERE MOTA = 'Don') LP INNER JOIN (SELECT MAP, MALP FROM PHONG) P ON LP.MALP = P.MALP) A  INNER JOIN (SELECT MADP, MAP, MAKS, MAKH, NGAYBATDAU FROM DATPHONG) DP ON DP.MAP = A.MAP) B  INNER JOIN (SELECT MAKS FROM KHACHSAN WHERE TENKS='Ma Vi') KS ON B.MAKS = KS.MAKS ) C  INNER JOIN (SELECT MAKH, TENKH, CMND FROM KHACHHANG WHERE QUOCTICH = 'Viet Nam') KH ON C.MAKH = KH.MAKH  SELECT \* FROM TABLE(DBMS\_XPLAN.display\_cursor(format=>'ALLSTATS LAST')); |
| **Kết quả Explain** |
| **Tối ưu hóa câu truy vấn trên môi trường phân tán**  **Table  Description automatically generated** |
| **Tối ưu hóa câu truy vấn trên môi trường phân tán** |
| *Ta có chương trình định vị cho các quan hệ như sau:*  KHACHSAN: KS = KS1 KS2  PHONG: P = P1 P2  DATPHONG: DP = DP1 DP2  Quan hệ LOAIPHONG LP và KHACHHANG KH được nhân bản. |
| *Cây truy vấn đại số quan hệ trên các mảnh*  Diagram  Description automatically generated  *Xét nhánh 1 ta có:*  TenKS = ‘Ma Vi’ (KS1 KS2)  TenKS = ‘Ma Vi’ (KS2) TenKS = ‘Ma Vi’ (KS1)  TenKS = ‘Ma Vi’ (KS2)  TenKS = ‘Ma Vi’ (KS2) KS2  Xét phép kết ở quan hệ KS2 và nhánh 2 ta loại bỏ được các mảnh không cần thiết là mảnh DP1, giữ lại mảnh DP2  Tương tự ở nhánh 3, ta loại bỏ được các mảnh P1, giữ lại mảnh P2  *Vậy cây truy vấn được rút gọn trên các mảnh phân tán là:*  Diagram  Description automatically generated |
| **Viết lại câu Query trên môi trường phân tán** |
| **Tại chi nhánh KS2**  SELECT C.MAP, C.MAKS, C.MAKH, NGAYBATDAU, TENKH, CMND FROM  (SELECT B.MAP, B.MAKS, MAKH, NGAYBATDAU FROM  (SELECT DP.MAP, MAKS, MAKH, NGAYBATDAU FROM  (SELECT P.MAP FROM (SELECT MALP FROM KS2.LOAIPHONG WHERE MOTA = 'Don') LP INNER JOIN (SELECT MAP, MALP FROM KS2.PHONG) P ON LP.MALP = P.MALP) A  INNER JOIN (SELECT MADP, MAP, MAKS, MAKH, NGAYBATDAU FROM KS2.DATPHONG) DP ON DP.MAP = A.MAP) B  INNER JOIN (SELECT MAKS FROM KS2.KHACHSAN WHERE TENKS='Ma Vi') KS ON B.MAKS = KS.MAKS ) C  INNER JOIN (SELECT MAKH, TENKH, CMND FROM KS2.KHACHHANG WHERE QUOCTICH = 'Viet Nam') KH ON C.MAKH = KH.MAKH |

# **Tài liệu tham khảo**

* **Oracle8 Concepts - Release 8.0**

Chap 30: Distributed Databases

<https://docs.oracle.com/cd/A64702_01/doc/server.805/a58227/ch21.htm>

* **Oracle® Database Concepts 10g Release 1 (10.1)**

13 Data Concurrency and Consistency

<https://docs.oracle.com/cd/B13789_01/server.101/b10743/consist.htm>