TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THỰC PHẨM THÀNH PHỐ HỒ CHÍ MINH

KHOA CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN HỌC PHẦN

HỆ QUẢN TRỊ CỞ SỞ DỮ LIỆU ORACLE

Giảng viên hướng dẫn:

Cô ***Nguyễn Phương Hạc***

Thành viên:

***Nguyễn Quang Trung - 2001206954***

***Nguyễn Duy Thịnh - 2001207328***

***Nguyễn Tri Long - 200117037***

Thành Hồ, tháng 2 năm 2023

# **Thông tin học phần**

Tên học phần: Hệ quản trị cơ sở dữ liệu Oracle

Mã học phần: 010100175401

Giảng viên hướng dẫn: Nguyễn Phương Hạc

Thời gian: Thứ 6 tiết 1-3

Thành viên:

Nguyễn Quang Trung - 2001206954

Nguyễn Duy Thịnh - 2001207328

Nguyễn Tri Long - 200117037

# Mục Lục

|  |  |
| --- | --- |
| Tuần | Nội dung báo cáo |
| 2 | * Xác định thành viên tham gia, tên, nội dung phân công, đánh giá các nội dung hoàn thành/ không hoàn thành, mục tiêu của đồ án * Mô tả nội dung chi tiết của ứng dụng (đồ án) * Phân tích và thiết kế mô hình quan hệ (từ 4-6 bảng) * Mô tả các yêu cầu ràng buộc của đồ án * Mô tả các chức năng nghiệp vụ trong đồ án (mỗi người khoảng 2-3 nghiệp vụ) * Mô tả danh mục nhóm quyền (2- 3 nhóm quyền), quyền (được và không được thực hiện trên bảng nào), profile trong đồ án * Mô tả các nghiệp vụ được cài đặt ở dạng giao tác trong đồ án * Thiết kế giao diện hệ thống, giao diện các chức năng theo yêu cầu đồ án (các yêu cầu trong nội dung ở bên dưới, bắt đầu từ tuần 3) * Kết nối ứng dụng với Oracle cục bộ * Thiết kế chức năng đăng nhập, đăng xuất. |
| 3 | * Thiết kế cơ sở dữ liệu trên Oracle. Lưu ý dữ liệu phải đa dạng, nhiều kiểu dữ liệu khác nhau, có cả âm thanh và hình ảnh * Kết nối ứng dụng với Oracle từ xa * Thiết kế chức năng xem một số thông tin tùy chọn của sga, pga, process, instance, database, datafile, control files, spfile * Thiết kế chức năng (1) kiểm tra một số thông tin của session: số lượng session, sid, serial, username, ứng dụng nào đang đăng nhập…; (2) hủy một session đang đăng nhập; (3) xem các process ứng với các session đang đăng nhập. * Cho phép xem các tablespace, tên tablespace tương ứng thuộc quyền sở hữu của một user nào đó được nhập vào   + (1) hiển thị tất cả datafile, (2) hiển thị chi tiết thông tin liên quan đến từng datafiles trong cơ sở dữ liệu: tên, id, tablespace chứa datafile đó   + (1) Tạo tablespace trên một hoặc nhiều datafile do người dùng lựa chọn; (2) Bổ sung datafile vào tablespace đã tồn tại |
| 4 | Thiết kế chức năng:   * Xem thông tin liên quan đến một user nào đó được chỉ định, ít nhất phải có các thông tin cơ bản như: tên user, ngày tạo, ngày hết hạn, trạng thái user, lần đăng nhập cuối, profile của user. * Thiết kế giao diện khi người dùng đăng nhập luôn có hiển thị thông tin tên người dùng, lần đăng nhập cuối cùng tại một góc màn hình * Thiết kế chức năng (1) Kiểm tra các policy hiện có (2) Thiết lập ít nhất 4 policy trên các bảng và các cột do sv tùy chọn phù hợp với đồ án (3) xuất báo cáo quá trình audit dữ liệu của các policy có liên quan đã thiết lập (4) Thiết lập audit liên quan đến một user nào đó * Cảnh báo tự động khi có sự truy cập hoặc thao tác quá mức quy định với dữ lấy được từ audit |
| 5 | Thiết lập   * (1) Nhiều loại profile khác nhau trong csdl; (2) cho phép người dùng tạo mới profile với tham số như thời gian sử dụng, số lần thay đổi password, khóa tài khoản nếu sai password n lần, thời gian tài khoản bị khóa khi nhập sai password… * (1) cho phép người dùng tạo, sửa user với profile được chọn lựa trong danh sách profile đang có; (2) cho phép xóa user; (3) xóa profile * Chức năng cho phép người dùng phân quyền theo các lựa chọn từ danh sách; (3) cho phép thu hồi quyền user * (1) Chức năng tạo và xóa nhóm quyền; (2) chức năng phân quyền cho nhóm quyền |
| 6 | Thiết lập   * Sao lưu phục hồi dữ liệu với các kiểu sao lưu đa dạng * Export và import dữ liệu * Cài đặt các chức năng tìm kiếm, tổng hợp dữ liệu có sử dụng cursor ngầm định, tường minh, có tham số, for update, where current of. Mỗi sinh viên làm ít nhất 2 câu truy vấn có sử dụng đầy đủ các kỹ thuật này |
| 7 | * Cài đặt các chức năng truy vấn, tìm kiếm, cảnh báo có sử dụng các hàm có sẵn trong Oracle, các cấu trúc điều khiển kết hợp với procedure, function, package. Mỗi sinh viên làm ít nhất 2 câu |
| 8 | * Cài đặt đầy đủ các loại trigger: before, after, instead of, compound, sự kiện hệ thống (start, shutdown, log off). Mỗi sinh viên làm ít nhất 2 trigger * Kết hợp trigger, thủ tục, hàm… để cài đặt chức năng hệ thống tự động tạo người dùng mới với profile mặc định, cách thức đặt tên theo quy định khi nhập mới thông tin liên quan đến người dùng. Ví dụ: khi nhập mới một sinh viên “nguyen thi lan”, hệ thống tự động tạo tài khoản đăng nhập “lannt”; nếu có 2 người trùng tên thì tự động chèn số thứ tự thêm vào sau như “lannt1”, “lannt2” * Cài đặt phân quyền tự động cho người dùng được tạo mới * Cài đặt tự động ghi nhận dữ liệu thêm, xóa, sửa của (một) bảng được xem là quan trọng |
| 9 | * Cài đặt các giao tác đã phân tích trong bảng mô tả, có sử dụng các mức cô lập phù hợp với giao tác * Thiết kế chức năng tìm kiếm session, user đang block dữ liệu; chức năng chọn lựa và xóa các session nào đang block dữ liệu * Nội dung tìm hiểu thêm cộng điểm: Áp dụng mô hình bảo mật cơ sở dữ liệu, chọn một trong 2 kỹ thuật trong đồ án:   + Sử dụng VPD (Virtual Private Database)   + Sử dụng OLS (Oracle Label Security) |

# Tuần 2 17.02.2023

## Nội dung báo cáo

- Xác định thành viên tham gia, tên, nội dung phân công, đánh giá các nội dung hoàn thành/ không hoàn thành, mục tiêu của đồ án

- Mô tả nội dung chi tiết của ứng dụng (đồ án)

- Phân tích và thiết kế mô hình quan hệ (từ 4-6 bảng)

- Mô tả các yêu cầu ràng buộc của đồ án

- Mô tả các chức năng nghiệp vụ trong đồ án (mỗi người khoảng 2-3 nghiệp

vụ)

- Mô tả danh mục nhóm quyền (2- 3 nhóm quyền), quyền (được và không

được thực hiện trên bảng nào), profile trong đồ án

- Mô tả các nghiệp vụ được cài đặt ở dạng giao tác trong đồ án

- Thiết kế giao diện hệ thống, giao diện các chức năng theo yêu cầu đồ án

(các yêu cầu trong nội dung ở bên dưới, bắt đầu từ tuần 3)

- Kết nối ứng dụng với Oracle cục bộ

- Thiết kế chức năng đăng nhập, đăng xuất.

## Nội dung thực hiện

### Chi tiết đề tài

Phần mềm quản lý quán cafe là một công cụ hỗ trợ hiệu quả cho các chủ quán và nhân viên trong việc quản lý các hoạt động kinh doanh của quán. Phần mềm quản lý quán cafe có nhiều tính năng như: quản lý bàn, order đồ uống, tính tiền, quản lý nguyên liệu, quản lý nhân sự, quản lý doanh thu, quản lý khách hàng, quản lý kho, quản lý chi phí, quản lý khuyến mãi, quản lý báo cáo, quản lý cài đặt... Phần mềm quản lý quán cafe giúp quán tiết kiệm thời gian, chi phí, nâng cao hiệu quả kinh doanh, tăng sự hài lòng của khách hàng và nhân viên. Phần mềm quản lý quán cafe có nhiều loại phù hợp với từng mô hình quán như: Polaris FnB, Cukcuk , POS365, iPOS...

### Phân tích chức năng

* Quản lý menu: cho phép quản lý danh sách các sản phẩm (đồ uống, đồ ăn) có trong quán, bao gồm thông tin về tên, mô tả, giá cả, hình ảnh.
* Quản lý đơn hàng: cho phép nhân viên quản lý và xử lý đơn hàng của khách hàng. Điều này bao gồm việc tạo đơn hàng, thêm / xóa các món trong đơn hàng, tính tổng tiền và lưu trữ lịch sử đơn hàng.
* Quản lý khách hàng: cho phép quản lý thông tin khách hàng, bao gồm thông tin liên hệ, địa chỉ, lịch sử mua hàng và các thông tin khác.
* Quản lý nhân viên: cho phép quản lý thông tin của các nhân viên, bao gồm tên, địa chỉ, số điện thoại, lịch sử làm việc và các thông tin khác.
* Quản lý kho: cho phép quản lý số lượng sản phẩm trong kho, thông báo khi còn ít sản phẩm và tự động đặt hàng khi cần thiết.

### Mô hình quan hệ

* Đối tượng: Khách hàng

Tính năng: Đặt hàng, thanh toán, lịch sử đặt hàng

Quan hệ: Khách hàng có thể đặt hàng và thanh toán cho nhiều đơn hàng, mỗi đơn hàng có thể có nhiều sản phẩm

* Đối tượng: Sản phẩm

Tính năng: Tên sản phẩm, mô tả, giá, loại sản phẩm

Quan hệ: Mỗi sản phẩm có thể thuộc nhiều loại sản phẩm khác nhau, mỗi đơn hàng có thể bao gồm nhiều sản phẩm

* Đối tượng: Nhân viên

Tính năng: Tên, chức vụ, lương, lịch làm việc

Quan hệ: Mỗi nhân viên thuộc về một hoặc nhiều chức vụ khác nhau, mỗi đơn hàng có thể được phục vụ bởi nhiều nhân viên khác nhau

* Đối tượng: Đơn hàng

Tính năng: Mã đơn hàng, thời gian đặt hàng, trạng thái đơn hàng, tổng giá trị đơn hàng

Quan hệ: Mỗi đơn hàng thuộc về một khách hàng, có nhiều sản phẩm, và được phục vụ bởi nhiều nhân viên khác nhau

* Đối tượng: Bàn

Tính năng: Mã bàn, số lượng khách, trạng thái bàn (đang sử dụng, trống, đang dọn dẹp)

Quan hệ: Mỗi bàn có thể được sử dụng bởi nhiều khách hàng khác nhau, và được phục vụ bởi nhiều nhân viên khác nhau

* Đối tượng: Thực đơn

Tính năng: Tên thực đơn, mô tả, giá, danh sách sản phẩm thuộc thực đơn

Quan hệ: Mỗi thực đơn có thể bao gồm nhiều sản phẩm khác nhau

* Đối tượng: Loại sản phẩm

Tính năng: Tên loại sản phẩm, mô tả

Quan hệ: Mỗi loại sản phẩm có thể bao gồm nhiều sản phẩm khác nhau

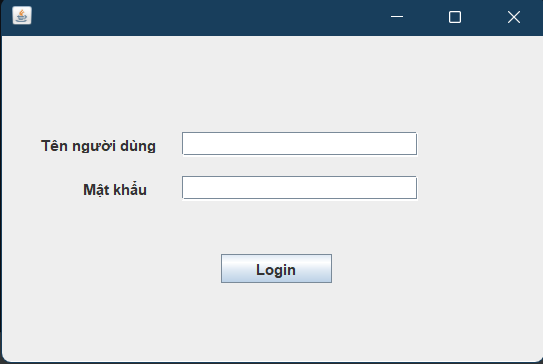
### Ràng buộc đồ án

* Hạn chế kỹ thuật:
* Hệ thống phải hỗ trợ đa nền tảng và có thể hoạt động trên các thiết bị khác nhau.
* Hệ thống phải có khả năng xử lý đồng thời nhiều yêu cầu từ nhiều người dùng.
* Ràng buộc chức năng:
* Khách hàng phải có thể đặt món trực tuyến, thanh toán trực tuyến và theo dõi lịch sử đặt hàng của mình.
* Nhân viên phải có thể quản lý đơn hàng, kiểm tra trạng thái bàn và xử lý các yêu cầu của khách hàng.
* Quản lý phải có thể quản lý các sản phẩm, nhân viên, đơn hàng và bàn trong quán.
* Ràng buộc về dữ liệu:
* Hệ thống phải có khả năng lưu trữ và quản lý dữ liệu của khách hàng, sản phẩm, nhân viên, đơn hàng và bàn.
* Dữ liệu phải được lưu trữ một cách an toàn, đảm bảo bảo mật thông tin của khách hàng.
* Ràng buộc về giao diện:
* Giao diện hệ thống phải đơn giản, dễ sử dụng và thân thiện với người dùng.
* Giao diện phải hiển thị đầy đủ thông tin về đơn hàng, sản phẩm và trạng thái bàn.
* Ràng buộc về hiệu suất:
* Hệ thống phải đảm bảo độ trễ tối đa trong việc xử lý các yêu cầu của khách hàng và nhân viên.
* Hệ thống phải đảm bảo tốc độ xử lý nhanh chóng và hiệu quả để giảm thiểu thời gian chờ đợi của khách hàng.

### Chức năng nghiệp vụ

* Quản lý menu: Cho phép quản lý viên thêm, sửa, xóa các sản phẩm trong menu của quán. Quản lý viên cũng có thể tạo các danh mục sản phẩm, quản lý đơn vị tính và giá của từng sản phẩm.
* Quản lý đơn hàng: Hệ thống cho phép khách hàng chọn sản phẩm từ menu, lựa chọn bàn và thực hiện đặt hàng. Nhân viên sẽ nhận được thông báo về đơn hàng mới và tiến hành xử lý. Hệ thống cũng cung cấp các tính năng quản lý đơn hàng, bao gồm xem đơn hàng, xác nhận đơn hàng, hủy đơn hàng, cập nhật trạng thái đơn hàng, và xuất hóa đơn.
* Quản lý bàn: Hệ thống cho phép quản lý viên quản lý các bàn trong quán, bao gồm thêm, sửa, xóa bàn. Nhân viên sẽ tiến hành gán đơn hàng vào các bàn tương ứng và cập nhật trạng thái bàn (trống, đã đặt, đang phục vụ, đã thanh toán).
* Quản lý khách hàng: Hệ thống cho phép quản lý viên quản lý thông tin khách hàng, bao gồm thêm, sửa, xóa thông tin khách hàng. Khách hàng có thể tạo tài khoản để lưu trữ thông tin và lịch sử đặt hàng.
* Quản lý nhân viên: Hệ thống cho phép quản lý viên quản lý thông tin nhân viên, bao gồm thêm, sửa, xóa thông tin nhân viên. Quản lý viên cũng có thể phân quyền truy cập cho nhân viên và quản lý thời gian làm việc của nhân viên.
* Thống kê và báo cáo: Hệ thống cung cấp các báo cáo thống kê về doanh thu, lợi nhuận, số lượng đơn hàng và các sản phẩm được bán chạy nhất. Hệ thống cũng cung cấp các báo cáo khác như báo cáo khách hàng và báo cáo nhân viên.

### Giao diện đăng nhập, kết nối CSDL



# Tuần 3 24.02.2023

## Nội dung báo cáo

Thiết kế cơ sở dữ liệu trên Oracle. Lưu ý dữ liệu phải đa dạng, nhiều kiểu dữ liệu khác nhau, có cả âm thanh và hình ảnh

- Kết nối ứng dụng với Oracle từ xa

- Thiết kế chức năng xem một số thông tin tùy chọn của sga, pga, process,instance, database, datafile, control files, spfile

- Thiết kế chức năng (1) kiểm tra một số thông tin của session: số lượng

session, sid, serial, username, ứng dụng nào đang đăng nhập…; (2) hủy một

session đang đăng nhập; (3) xem các process ứng với các session đang đăng

nhập.

- Cho phép xem các tablespace, tên tablespace tương ứng thuộc quyền sở hữu

của một user nào đó được nhập vào

o (1) hiển thị tất cả datafile, (2) hiển thị chi tiết thông tin liên quan

đến từng datafiles trong cơ sở dữ liệu: tên, id, tablespace chứa

datafile đó

o (1) Tạo tablespace trên một hoặc nhiều datafile do người dùng lựa

chọn; (2) Bổ sung datafile vào tablespace đã tồn tại.

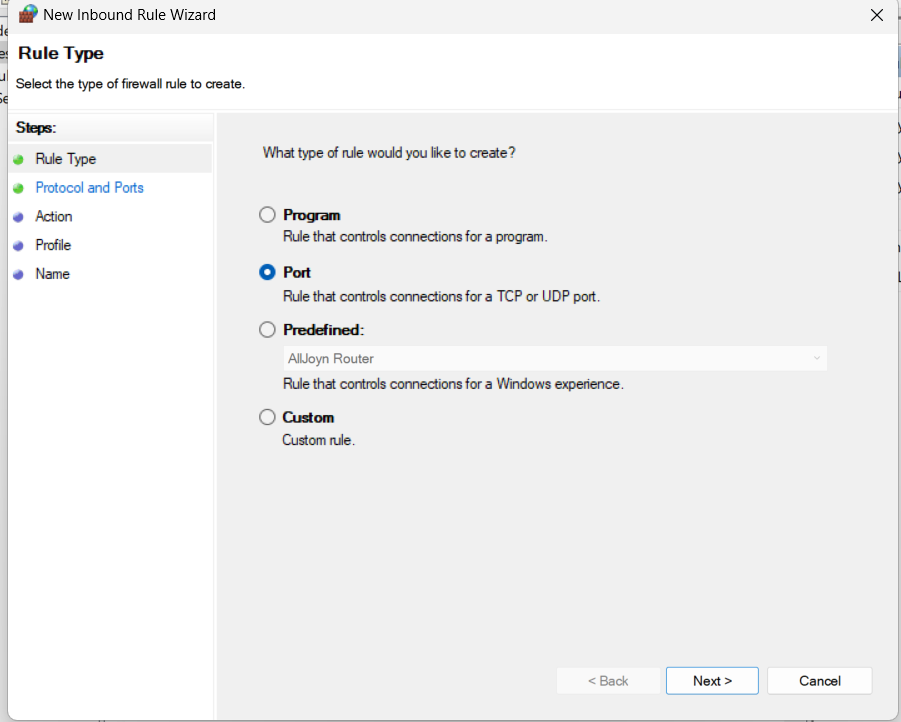
## Nội dung thực hiện

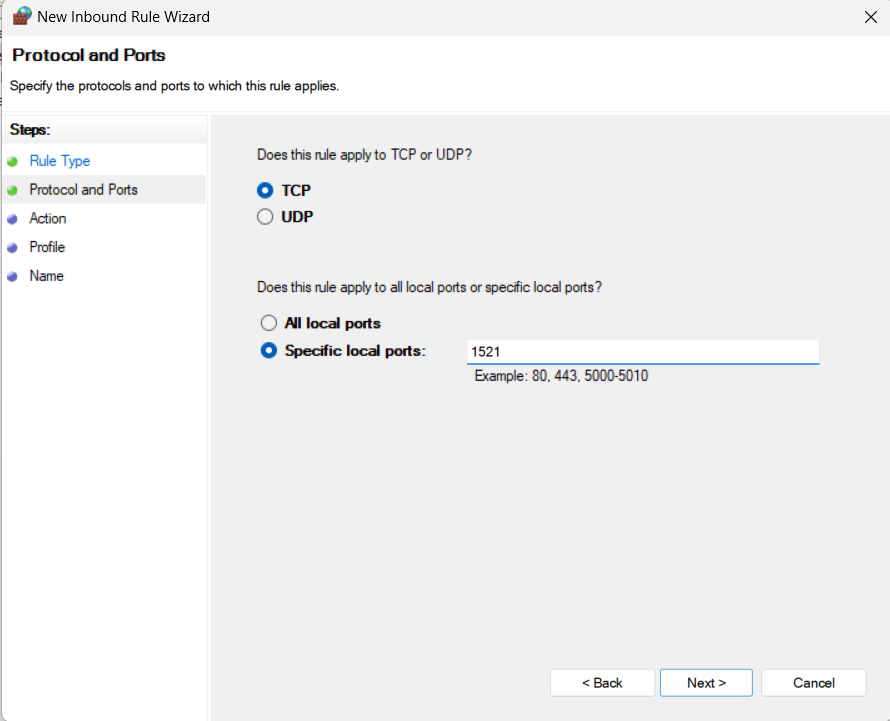
### Kết nối Oracle từ xa (Oracle remote connection)

**Bước 1** : Mở (Tưởng lửa Bảo vệ Windows với Bảo mật Nâng cao( Windows Defender Firewall with Advanced Security) và nhấn vào tab Inbound Rules và Chọn New Rule

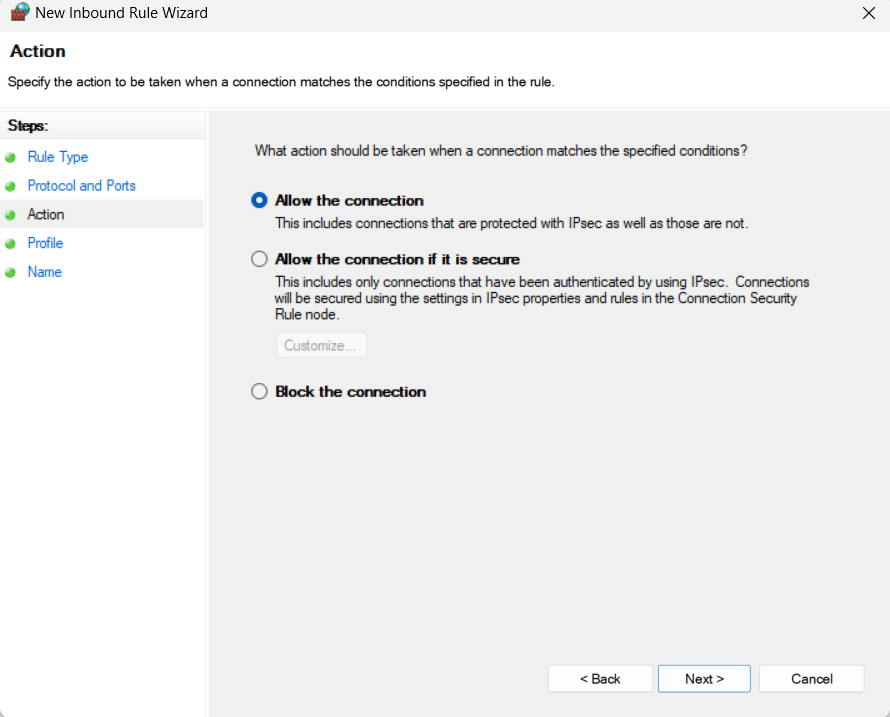
Ảnh có chứa văn bản

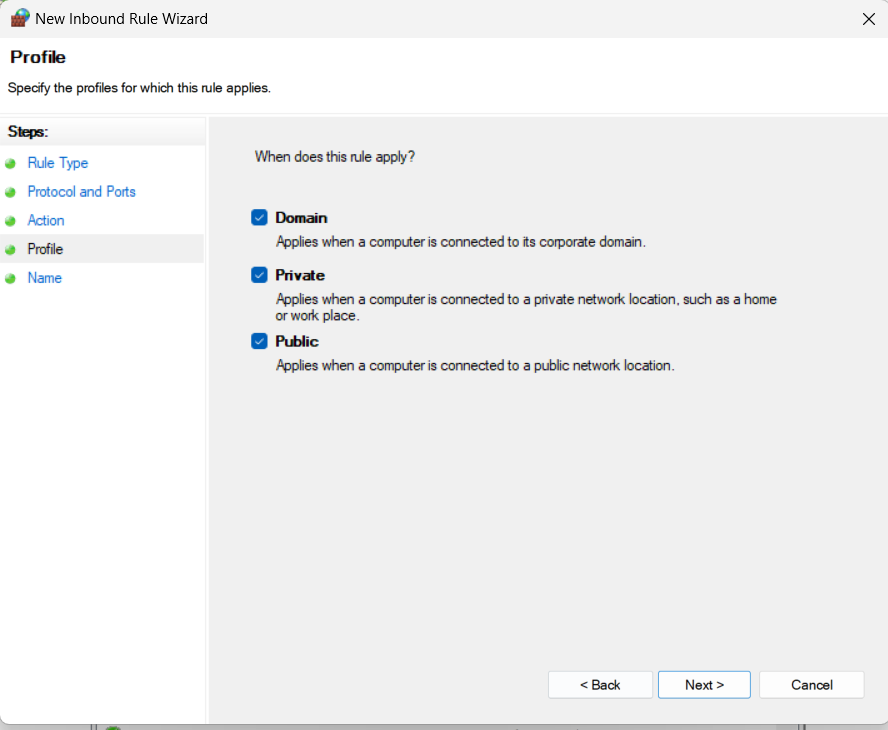
Mô tả được tạo tự động

**Bước 2 :** 



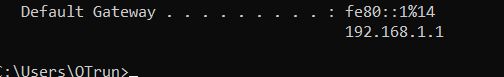
Đặt số cổng là 1521

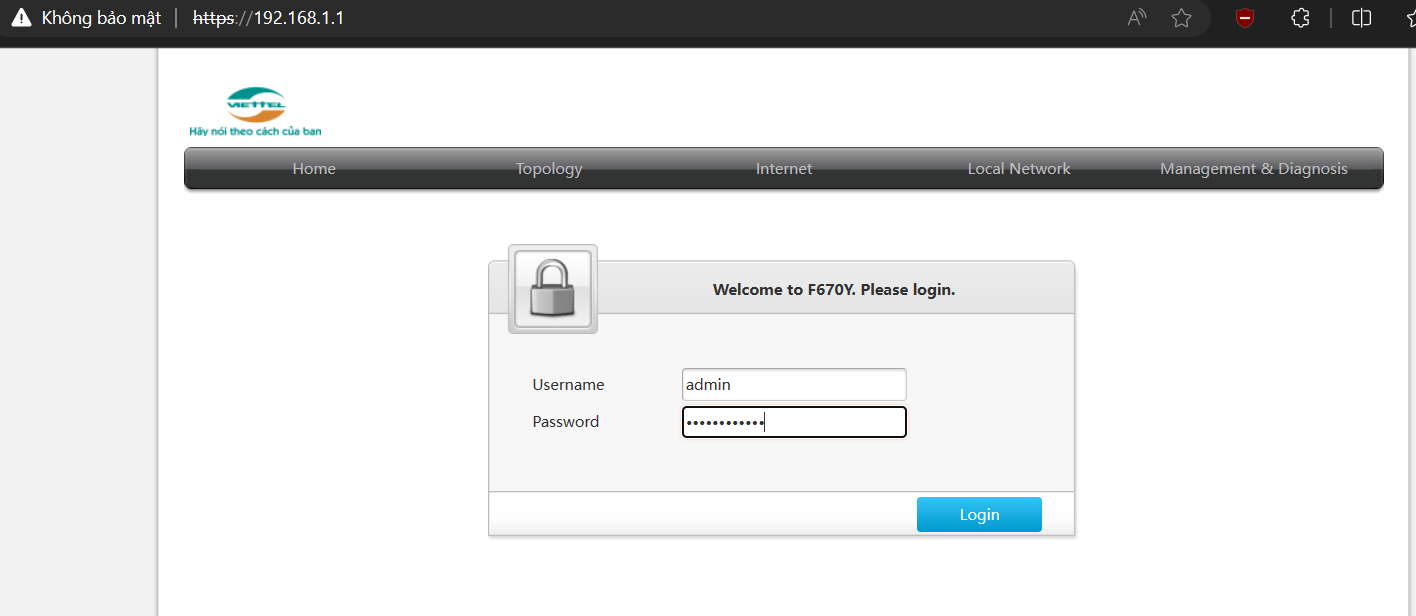




**Bước 3:** Truy cập vào router (thường là 192.168.1.1

Vào /ipconfig phần Default gataway xem IP router





Ảnh có chứa văn bản

Mô tả được tạo tự động

Vào Internet -> Security -> Port Forwarding (Tùy router)

Name: Đặt tên cho nó

Protocol: TCP

WAN Conenction: Để mặc định của thiết đặt

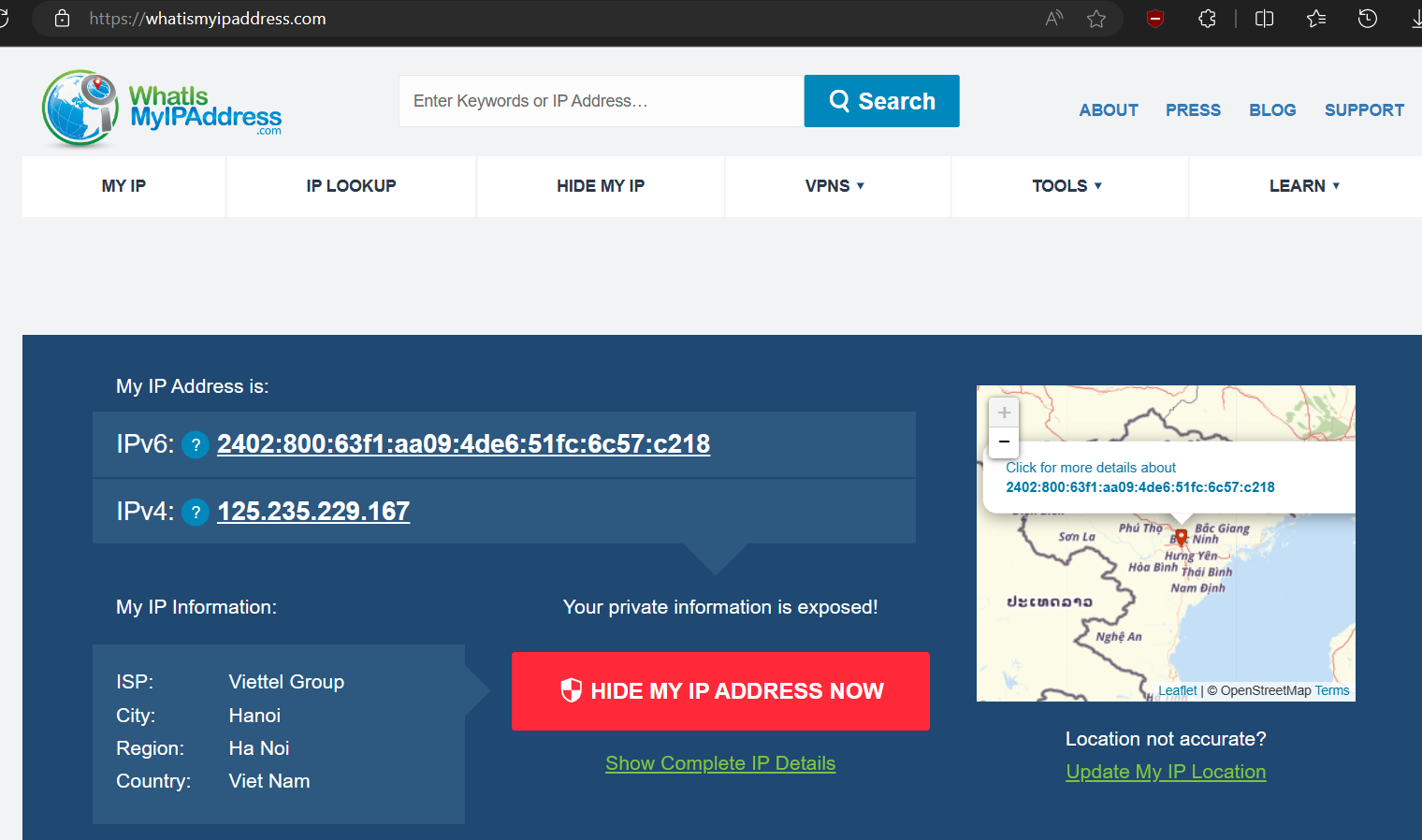
WAN Host IP Address: Để nguyên

LAN host: Địa chỉ IP của máy chứa Oracle DB( Vào ipconfig xem mục IpV4 Address)

WAN port: Cổng cần mở, Nhập 1521 của Oracle mặc định

LAN host port: Cũng nhập 1521

Bước 4: Lên google gõ what my ip address tìm IP Công khai



Vào trang <https://ping.eu/port-chk/> kiểm tra xem cổng đã mở chưa

Ảnh có chứa văn bản

Mô tả được tạo tự động

**Bước 5:** Tiến hành đăng nhập với địa chỉ IP và cổng đã mở

### Xem thông tin tài nguyên (View resources)

1. --xem thông tin sga
2. SELECT \* FROM V$SGA;
3. --xem thông tin về buffer cache
4. SELECT \* FROM V$BUFFER\_POOL\_STATISTICS;
5. --xem thông tin về pga
6. SELECT \* FROM V$PROCESS\_MEMORY;
7. --xem thong tin instance
8. SELECT INSTANCE\_NUMBER,INSTANCE\_NAME,HOST\_NAME,VERSION\_FULL,STARTUP\_TIME,STATUS,DATABASE\_STATUS  FROM V$INSTANCE;
9. --xem thong tin ve process
10. SELECT PID,PROGRAM,USERNAME,TERMINAL,CPU\_USED,EXECUTION\_TYPE FROM V$PROCESS;
11. --xem thong tin ve database
12. SELECT \* FROM V$DATABASE;
13. --xem thong tin ve datafile
14. SELECT \* FROM V$DATAFILE;
15. --xem thong tin ve control file
16. SELECT \* FROM V$CONTROLFILE;
17. --xem thong tin ve spfile
18. SHOW PARAMETER SPFILE;

Ảnh có chứa bàn

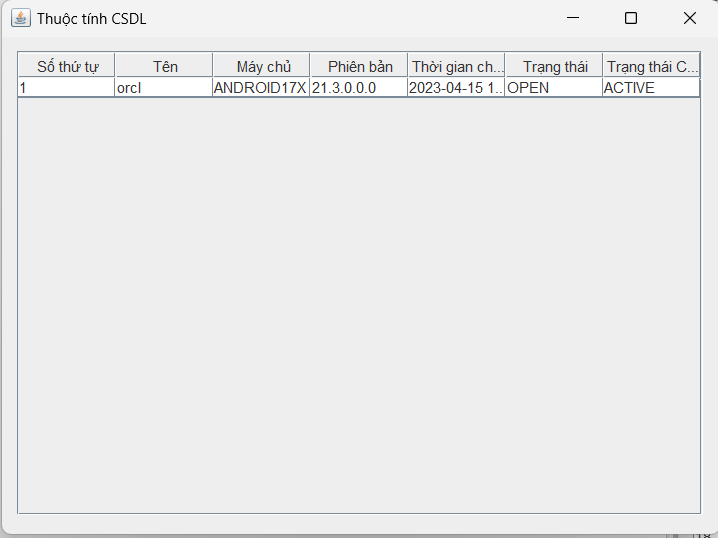
Mô tả được tạo tự động

Ảnh có chứa văn bản, tòa nhà

Mô tả được tạo tự động

Ảnh có chứa bàn

Mô tả được tạo tự động



### Quản lý phiên (Session manager)

1. --dem so luong session
2. SELECT COUNT(\*) FROM V$SESSION;
3. SELECT SID, SERIAL#, PROGRAM, USERNAME, STATUS FROM V$SESSION;
4. SELECT USERNAME, COUNT(\*) FROM V$SESSION GROUP BY USERNAME;
5. SELECT SID, SERIAL#, PROGRAM, USERNAME, STATUS,MACHINE FROM V$SESSION;
6. SELECT DISTINCT PROGRAM FROM V$SESSION;
7. SELECT DISTINCT MACHINE, PROGRAM FROM V$SESSION WHERE USERNAME='<tên đăng nhập của bạn>';
8. ---huy session
9. ALTER SYSTEM KILL SESSION '[sid],[serial#]' IMMEDIATE;
10. ---xem các process ứng với các session đang đăng nhập
11. SELECT s.sid, s.serial#, p.spid, s.program
12. FROM v$session s JOIN v$process p ON s.paddr = p.addr;

Ảnh có chứa bàn

Mô tả được tạo tự động

### Tablespace

**Xem các tablespace thuộc quyền sở hữu của một người dùng cụ thể**

SELECT DISTINCT tablespace\_name

FROM dba\_segments

WHERE owner = 'user\_name';

**Xem tablespace**

SELECT tablespace\_name, owner

FROM dba\_tablespaces;

*Thay dba bằng user nếu không có quyền dba*

**Hiện tất cả datafile**

SELECT file\_id,file\_name, tablespace\_name, bytes/1024/1024 as size\_mb

FROM dba\_data\_files;

**Tạo tablespace với 2 datafiles**

CREATE TABLESPACE example\_ts

DATAFILE '/u01/oracle/datafile1.dbf' SIZE 100M,

'/u02/oracle/datafile2.dbf' SIZE 200M

AUTOEXTEND ON NEXT 10M MAXSIZE 500M;

**Bổ sung datafile vào tablespace**

ALTER TABLESPACE USER\_DATA

ADD DATAFILE '/u03/oradata/user\_data\_03.dbf' SIZE 500M;

**Xóa datafile**

ALTER TABLESPACE USER\_DATA

DROP DATAFILE '/u03/oradata/user\_data\_03.dbf';

**Xóa tablespace**

DROP TABLESPACE USER\_DATA;

**Xóa tablespace và datafile 1 lượt**

DROP TABLESPACE USER\_DATA INCLUDING CONTENTS AND DATAFILES;

# Tuần 4 03.03.2023

## Nội dung báo cáo

Thiết kế chức năng:

* Xem thông tin liên quan đến một user nào đó được chỉ định, ít nhất phải có các thông tin cơ bản như: tên user, ngày tạo, ngày hết hạn, trạng thái user, lần đăng nhập cuối, profile của user.
* Thiết kế giao diện khi người dùng đăng nhập luôn có hiển thị thông tin tên người dùng, lần đăng nhập cuối cùng tại một góc màn hình
* Thiết kế chức năng (1) Kiểm tra các policy hiện có (2) Thiết lập ít nhất 4 policy trên các bảng và các cột do sv tùy chọn phù hợp với đồ án (3) xuất báo cáo quá trình audit dữ liệu của các policy có liên quan đã thiết lập (4) Thiết lập audit liên quan đến một user nào đó
* Cảnh báo tự động khi có sự truy cập hoặc thao tác quá mức quy định với dữ lấy được từ audit

## Nội dung thực hiện

### Xem thông tin người dùng (View user info)

SELECT

USERNAME,

CREATED,

EXPIRY\_DATE,

ACCOUNT\_STATUS,

LAST\_LOGIN,

PROFILE

FROM

DBA\_USERS

WHERE

USERNAME = 'ten\_user';

Ảnh có chứa bàn

Mô tả được tạo tự động

### Thiết kế giao diện khi người dùng đăng nhập luôn có hiển thị thông tin tên người dùng, lần đăng nhập cuối cùng tại một góc màn hình

* Tạo 2 label để hiển thị username và lần đăng nhập cuối cùng

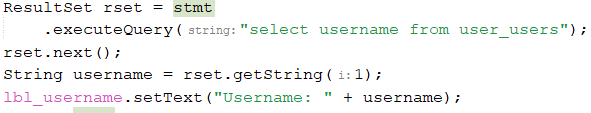
**Graphical user interface, application

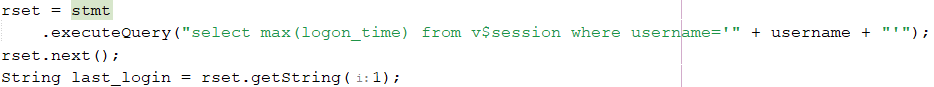
Description automatically generated**

* Trong sự kiện **formWindowOpened**, tạo kết nối đến Oracle



* Để lấy thông tin **username** của user hiện tại, ta sẽ truy vấn vào bảng **user\_users** bằng lệnh “**SELECT username FROM user\_users**”.



* Để lấy thông tin lần ***đăng nhập cuối cùng*** của user hiện tại, ta sẽ truy vấn vào bảng ***v$session***, lấy lần đăng nhập mới nhất bằng lệnh
* “**SELECT MAX(logon\_time) FROM v$session WHERE username=(username)**”
* 
* \* Lưu ý: phải gán quyền đọc bảng v$session cho user hiện tại bằng lệnh “**GRANT SELECT ON V\_$SESSION TO (username)**”
* Gán dữ liệu cho label để hiển thị lên giao diện

### Thiết lập 4 chính sách (Set up 4 policies)

1. Policy để ẩn dữ liệu nhạy cảm:

Chính sách này có thể được áp dụng cho bảng hoặc các cột chứa thông tin nhạy cảm mà không được tiết lộ cho mọi người. Để áp dụng chính sách này, ta có thể sử dụng câu lệnh sau:

DBMS\_RLS.ADD\_POLICY (

object\_schema => 'schema\_name',

object\_name => 'table\_name',

policy\_name => 'sensitive\_data\_policy',

function\_schema => 'policy\_function\_schema',

policy\_function => 'policy\_function',

statement\_types => 'SELECT, UPDATE',

enable => TRUE,

static\_policy => FALSE);

Trong đó, 'schema\_name' là tên schema của bảng, 'table\_name' là tên bảng, 'sensitive\_data\_policy' là tên của chính sách, 'policy\_function\_schema' là schema chứa hàm chính sách, 'policy\_function' là tên hàm chính sách và 'SELECT, UPDATE' chỉ định các loại câu lệnh cho phép được sử dụng.

1. Policy để giới hạn quyền truy cập:

Chính sách này có thể được sử dụng để giới hạn quyền truy cập đến bảng hoặc các cột chỉ cho những người dùng cụ thể hoặc nhóm người dùng. Để áp dụng chính sách này, ta có thể sử dụng câu lệnh sau:

DBMS\_RLS.ADD\_POLICY (

object\_schema => 'schema\_name',

object\_name => 'table\_name',

policy\_name => 'access\_policy',

function\_schema => 'policy\_function\_schema',

policy\_function => 'policy\_function',

statement\_types => 'SELECT',

enable => TRUE,

static\_policy => FALSE);

Trong đó, 'schema\_name' là tên schema của bảng, 'table\_name' là tên bảng, 'access\_policy' là tên của chính sách, 'policy\_function\_schema' là schema chứa hàm chính sách, 'policy\_function' là tên hàm chính sách và 'SELECT' chỉ định loại câu lệnh cho phép được sử dụng.

1. Policy để kiểm tra ràng buộc dữ liệu:

Chính sách này được sử dụng để kiểm tra ràng buộc dữ liệu trước khi chèn hoặc cập nhật dữ liệu trong bảng hoặc các cột cụ thể. Chính sách này sẽ đảm bảo rằng dữ liệu được chèn hoặc cập nhật phù hợp với các ràng buộc được định nghĩa trước đó trong hệ thống. Để áp dụng chính sách này, ta có thể sử dụng câu lệnh sau:

DBMS\_RLS.ADD\_POLICY (

object\_schema => 'schema\_name',

object\_name => 'table\_name',

policy\_name => 'constraint\_policy',

function\_schema => 'policy\_function\_schema',

policy\_function => 'policy\_function',

statement\_types => 'INSERT,UPDATE',

enable => TRUE,

static\_policy => FALSE);

Trong đó, 'schema\_name' là tên schema của bảng, 'table\_name' là tên bảng, 'constraint\_policy' là tên của chính sách, 'policy\_function\_schema' là schema chứa hàm chính sách, 'policy\_function' là tên hàm chính sách và 'INSERT,UPDATE' chỉ định loại câu lệnh được phép sử dụng.

1. Policy để giới hạn số lượng dòng được truy vấn:

Chính sách này có thể được sử dụng để giới hạn số lượng dòng được truy vấn trong bảng hoặc các cột cụ thể. Điều này có thể hữu ích để giảm tải hệ thống và bảo vệ dữ liệu. Để áp dụng chính sách này, ta có thể sử dụng câu lệnh sau:

DBMS\_RLS.ADD\_POLICY (

object\_schema => 'schema\_name',

object\_name => 'table\_name',

policy\_name => 'row\_limit\_policy',

function\_schema => 'policy\_function\_schema',

policy\_function => 'policy\_function',

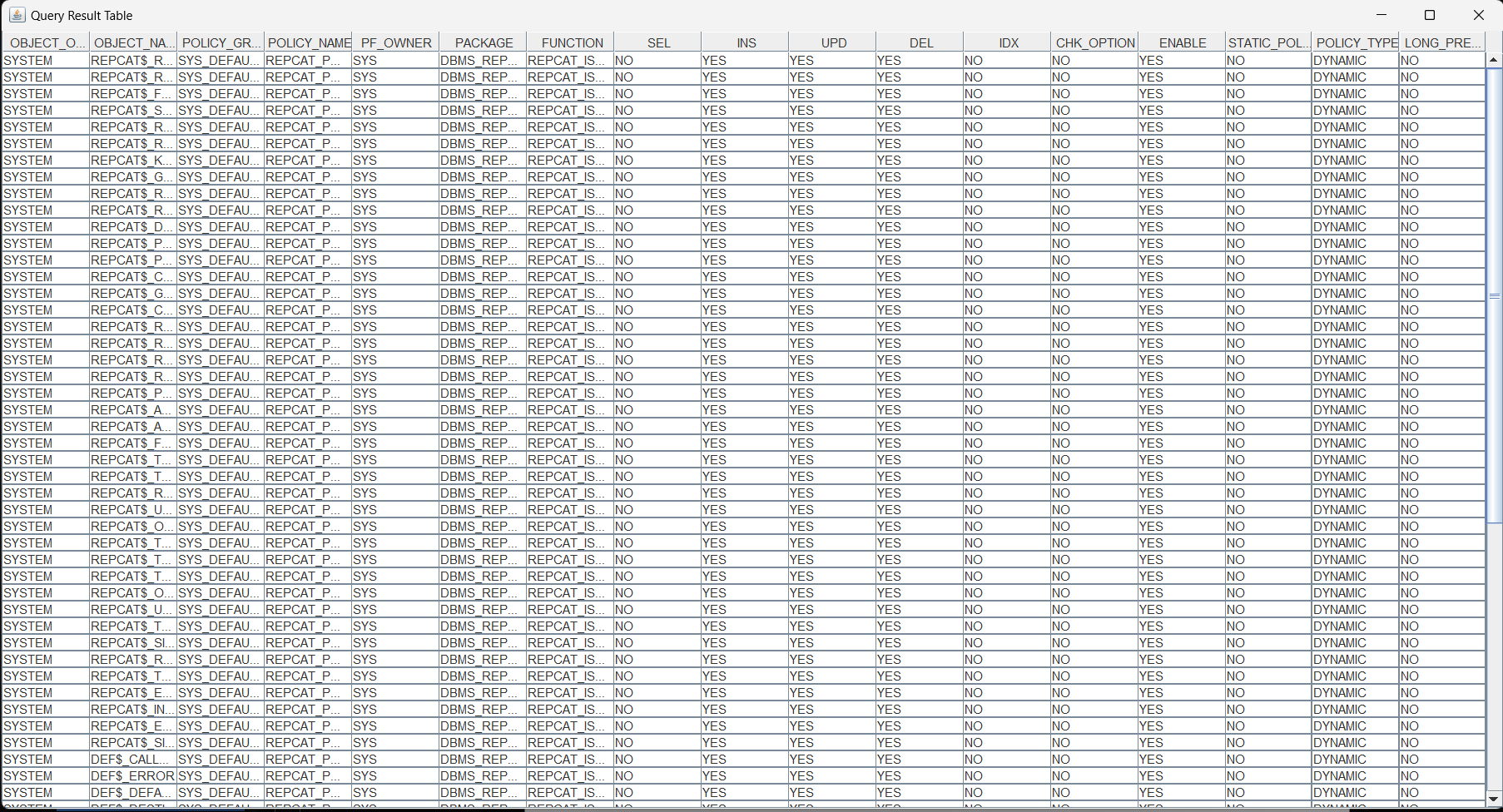
statement\_types => 'SELECT',

enable => TRUE,

static\_policy => FALSE);

Trong đó, 'schema\_name' là tên schema của bảng, 'table\_name' là tên bảng, 'row\_limit\_policy' là tên của chính sách, 'policy\_function\_schema' là schema chứa hàm chính sách, 'policy\_function' là tên hàm chính sách và 'SELECT' chỉ định loại câu lệnh cho phép được sử dụng.

### 3.Kiểm tra các chính sách (Policy result)



Code

----Buoi4-----

---xem thong tin user---

SELECT

    username,

    created,

    expiry\_date,

    account\_status,

    last\_login,

    profile

FROM

    dba\_users

WHERE

    username = 'SYS';

---kiem tra policy hien co

SELECT policy\_name, object\_owner, object\_name, policy\_type, enable

FROM dba\_policies;

---Báo cáo tất cả các audit trail entries cho các policy đã thiết lập:

SELECT \* FROM dba\_audit\_trail;

AUDIT ALL BY user;

AUDIT ALL BY user;

AUDIT ALL ON schema.object BY user;

# Tuần 5 09.03.2023

## Nội dung báo cáo

- (1) Nhiều loại profile khác nhau trong csdl; (2) cho phép người dùng tạo mới 3 profile với tham số như thời gian sử dụng, số lần thay đổi password, khóa tài khoản nếu sai password n lần, thời gian tài khoản bị khóa khi nhập sai password…

- (1) cho phép người dùng tạo, sửa user với profile được chọn lựa trong danh sách profile đang có; (2) cho phép xóa user; (3) xóa profile

- Chức năng cho phép người dùng phân quyền theo các lựa chọn từ danh sách; (3) cho phép thu hồi quyền user

- (1) Chức năng tạo và xóa nhóm quyền; (2) chức năng phân quyền cho nhóm quyền.

## 2. Nội dung thực hiện

### Quản lý người dùng (User Manager)

Khi tạo người dùng trong Oracle, nếu bạn không muốn sử dụng tiền tố c##, bạn có thể bỏ qua tiền tố này bằng cách thiết lập tham số COMMON\_USER\_PREFIX thành một chuỗi rỗng trước khi tạo người dùng.

Bạn có thể sử dụng câu lệnh sau để thiết lập tham số COMMON\_USER\_PREFIX thành chuỗi rỗng:

ALTER SYSTEM SET COMMON\_USER\_PREFIX='' SCOPE=SPFILE;

Tạo người dùng

CREATE USER admin

IDENTIFIED BY 123

DEFAULT TABLESPACE users

TEMPORARY TABLESPACE temp;

Gán quyền SYSDBA

GRANT SYSDBA TO username container=all;

Thu hồi quyền SYSDBA

REVOKE SYSDBA FROM username;

Gán quyền kết nối CSDL

GRANT CREATE SESSION TO user1;

Xóa người dùng

DROP USER username;

**Xem danh sách user**

Select username from DBA\_USERS;

Select username from USER\_USERS;

Ảnh có chứa bàn

Mô tả được tạo tự động

### Quản lý vai trò (Role Manager)

Tạo vai trò

CREATE ROLE role\_name;

Thêm quyền cho vai trò

GRANT select, insert, update ON table\_name TO role\_name;

Gán vai trò cho người dùng

GRANT role\_name TO user\_name;

Hủy gán vai trò cho người dùng

REVOKE role\_name FROM username;

Xóa vai trò

DROP ROLE role\_name;

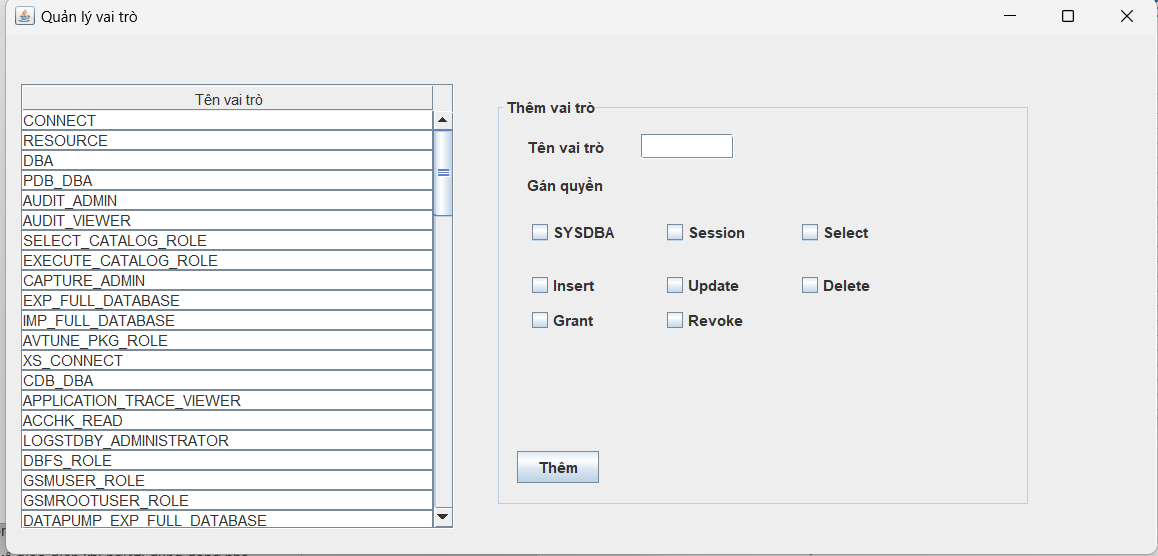
Gán quyền SYSDBA cho vai trò

GRANT SYSDBA TO role\_name;

**Xem danh sách vai trò**

SELECT role FROM dba\_roles;

SELECT role FROM user\_roles;



### Quản lý hồ sơ (Profile Manager)

**Tạo profile**

* Profile với giới hạn tài nguyên và thời gian đăng nhập

CREATE PROFILE limited\_resources LIMIT

SESSIONS\_PER\_USER 3

CPU\_PER\_SESSION 1000

CPU\_PER\_CALL 100

CONNECT\_TIME 30

IDLE\_TIME 10;

* Profile thay đổi mật khẩu định kỳ

CREATE PROFILE change\_password LIMIT

PASSWORD\_LIFE\_TIME 60

PASSWORD\_GRACE\_TIME 7

PASSWORD\_REUSE\_TIME 365

PASSWORD\_REUSE\_MAX 10;

* Profile với giới hạn số lần đăng nhập không hợp lệ, và chặn nếu sai quá n lần

CREATE PROFILE lock\_account LIMIT

FAILED\_LOGIN\_ATTEMPTS 5

PASSWORD\_LOCK\_TIME 10;

**Gán profile cho người dùng**

ALTER USER <tên user> PROFILE <tên profile>;

**Gán profile cho vai trò**

ALTER ROLE <Tên role> PROFILE <tên profile>;

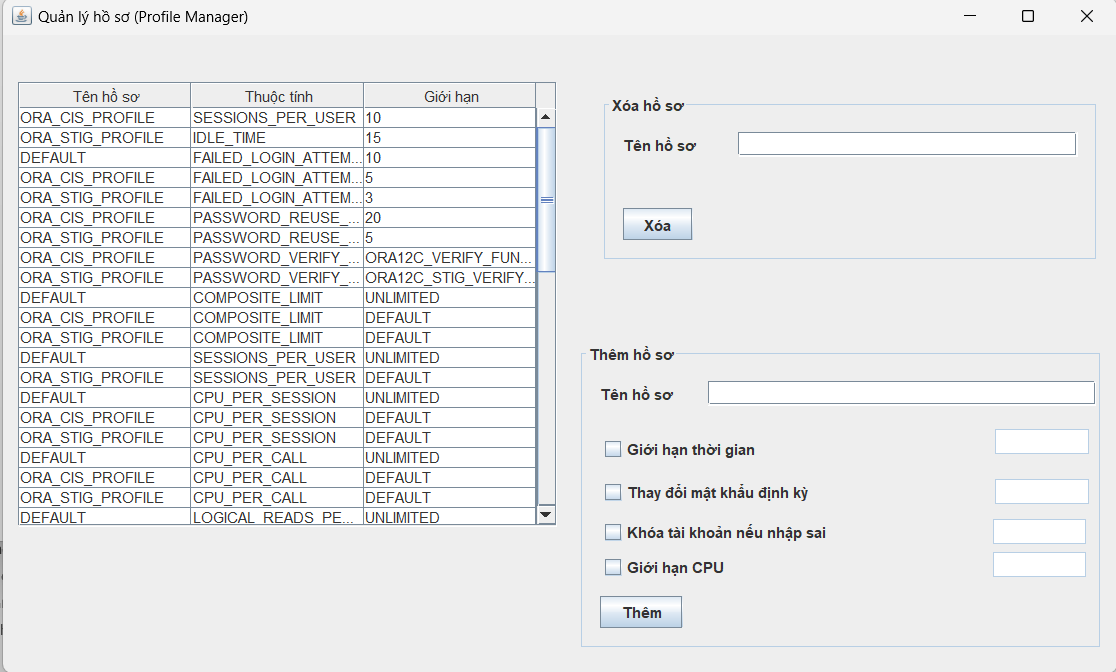
**Xóa profile**

DROP PROFILE lock\_account;

**Xem danh sách profile**

SELECT profile FROM USER\_PROFILES;

SELECT profile FROM DBA\_PROFILES;



# Tuần 6 16.03.2023

## Nội dung báo cáo

Thiết lập

- Sao lưu phục hồi dữ liệu với các kiểu sao lưu đa dạng(Trung- Hoàn thành)

- Export và import dữ liệu (Long-Hoàn thành)

- Cài đặt các chức năng tìm kiếm, tổng hợp dữ liệu có sử dụng cursor ngầm

định, tường minh, có tham số, for update, where current of. Mỗi sinh viên

làm ít nhất 2 câu truy vấn có sử dụng đầy đủ các kỹ thuật này(Hoàn thành)

## Nội dung thực hiện

### Ví dụ sử dụng cursor ngầm và định để lấy thông tin sinh viên có mã số SV001:

DECLARE @StudentID NVARCHAR(10) = 'SV001'

DECLARE @StudentName NVARCHAR(50)

DECLARE @StudentAddress NVARCHAR(100)

DECLARE @StudentPhone NVARCHAR(20)

DECLARE student\_cursor CURSOR LOCAL STATIC READ\_ONLY

FOR SELECT StudentName, StudentAddress, StudentPhone

FROM Students

WHERE StudentID = @StudentID

OPEN student\_cursor

FETCH NEXT FROM student\_cursor INTO @StudentName, @StudentAddress, @StudentPhone

IF @@FETCH\_STATUS = 0

BEGIN

PRINT 'Student name: ' + @StudentName

PRINT 'Student address: ' + @StudentAddress

PRINT 'Student phone: ' + @StudentPhone

END

CLOSE student\_cursor

DEALLOCATE student\_cursor

**Ví dụ sử dụng cursor tường minh và có tham số để lấy thông tin các sinh viên có điểm trung bình lớn hơn hoặc bằng 8:**

DECLARE @AvgScore FLOAT = 8.0

DECLARE @StudentID NVARCHAR(10)

DECLARE @StudentName NVARCHAR(50)

DECLARE @StudentAddress NVARCHAR(100)

DECLARE @StudentPhone NVARCHAR(20)

DECLARE student\_cursor CURSOR LOCAL FORWARD\_ONLY READ\_ONLY FOR

SELECT StudentID, StudentName, StudentAddress, StudentPhone

FROM Students

WHERE AvgScore >= @AvgScore

OPEN student\_cursor

FETCH NEXT FROM student\_cursor INTO @StudentID, @StudentName, @StudentAddress, @StudentPhone

WHILE @@FETCH\_STATUS = 0

BEGIN

PRINT 'Student ID: ' + @StudentID

PRINT 'Student name: ' + @StudentName

PRINT 'Student address: ' + @StudentAddress

PRINT 'Student phone: ' + @StudentPhone

FETCH NEXT FROM student\_cursor INTO @StudentID, @StudentName, @StudentAddress, @StudentPhone

END

CLOSE student\_cursor

DEALLOCATE student\_cursor

**Ví dụ sử dụng cursor for update để cập nhật điểm của các sinh viên có mã số lớn hơn SV005:**

DECLARE @StudentID NVARCHAR(10)

DECLARE @Score FLOAT

DECLARE student\_cursor CURSOR LOCAL FORWARD\_ONLY READ\_ONLY FOR

SELECT StudentID, Score

FROM Scores

WHERE StudentID > 'SV005'

FOR UPDATE OF Score

OPEN student\_cursor

FETCH NEXT FROM student\_cursor INTO @StudentID, @Score

WHILE @@FETCH\_STATUS = 0

BEGIN

SET @Score = @Score \* 1.1 -- tăng điểm lên 10%

UPDATE Scores SET Score = @Score WHERE CURRENT OF student\_cursor

FETCH NEXT FROM student\_cursor INTO @StudentID, @Score

END

CLOSE student\_cursor

DEALLOCATE student\_cursor

### Nhập và xuất dữ liệu

Oracle Data Pump là một công cụ để export và import dữ liệu trong Oracle Database.

Data Pump có 2 tập lệnh là expdp và impdp. Trong đó, expdp sẽ export dữ liệu trong Oracle Database ra thành dump file. Và impdp sẽ sử dụng dump file đó và import lại vào Oracle Database.

Dump file có thể chứa dữ liệu của table, index hay dữ liệu cấu trúc (metadata).

**Export dữ liệu**

* Bước đầu tiên, để có thể Export hay Import dữ liệu, cần tạo 1 directory trên database để chứa các dump file.

**mkdir C:\oracle\export**

Text

Description automatically generated

* Kết nối vào SQL Plus

**sqlplus / as sysdba**



* Tạo 1 đối tượng thư mục “dmpdir”

**CREATE OR REPLACE DIRECTORY dmpdir AS ‘C:\oracle\export’;**

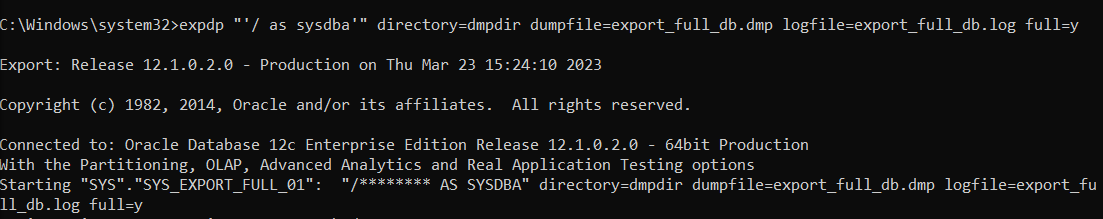
GRANT READ,WRITE ON DIRECTORY dmpdir TO nt;

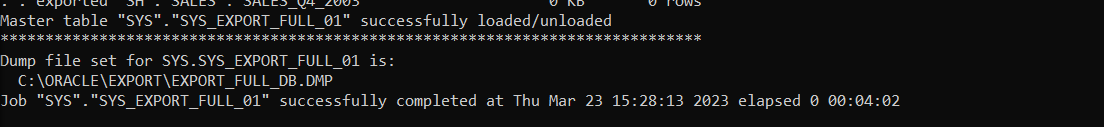
Text

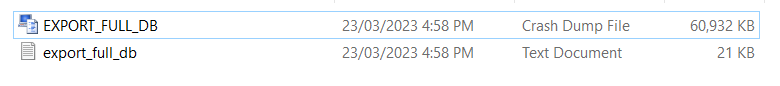
Description automatically generated with medium confidence

* Export toàn bộ database:

***expdp "'/ as sysdba'" directory=dmpdir dumpfile=export\_full\_db.dmp logfile=export\_full\_db.log full=y***







Trong đó:

directory: Tên đối tượng thư mục vừa tạo ở bước trên

dumpfile: Tên dump file chứa các dữ liệu sẽ export

logfile: Tên file nhật ký ghi lại quá trình export

+ nologfile=yes: Không lưu log

full=y: Export toàn bộ dữ liệu của database

\*Lưu ý: Thoát khỏi sqlplus trước

* Export schema

***expdp "'/ as sysdba'" directory=dmpdir dumpfile=export\_schema.dmp logfile=export\_schema.log schemas=nt***

schemas: Tên các schema bạn muốn export dữ liệu. Bạn có thể export nhiều schema 1 lúc, tên schema cách nhau bởi dấu phẩy. VD: schemas=SOE,SCOTT

Text

Description automatically generated



* Export table

***expdp "'/ as sysdba'" directory=dmpdir dumpfile=export\_tables.dmp logfile=export\_tables.log tables=nt.Demo***

***Text

Description automatically generated***

******

* Export metadata only

Bạn có thể chỉ export dữ liệu cấu trúc, các procedure, function của database với option content=metadata\_only

***expdp "'/ as sysdba'" directory=dmpdir dumpfile=export\_tables\_metadata.dmp logfile=export\_tables\_metadata.log tables=nt.Demo content=metadata\_only***

***Text

Description automatically generated***

Text

Description automatically generated

Bạn cũng làm tương tự nếu chỉ muốn export data, không bao gồm metadata

***expdp "'/ as sysdba'" directory=dmpdir dumpfile=export\_tables\_data.dmp logfile=export\_tables\_data.log tables=nt.Demo content=data\_only***

***Text

Description automatically generated***

Chú ý: Nếu không chỉ ra mệnh đề content, cả metadata và data đều sẽ được export.

**Import dữ liệu**

* Nếu bạn import full database, bao gồm cả các schema như SYS, SYSTEM, bạn có thể gặp rất nhiều lỗi, do conflict với nhiều bảng hệ thống.

Do đó, bạn nên tránh import full database hoặc full schema SYS, SYSTEM.

* Import Database

***impdp "'/ as sysdba'" directory= dmpdir dumpfile=export\_full\_db.dmp logfile=import\_full\_db.log full=y;***

* Import schema

***impdp "'/ as sysdba'" directory=dmpdir dumpfile=export\_schema.dmp logfile=import\_schema.log schemas=nt***

***Text

Description automatically generated***

Bạn có thể đổi tên schema trong quá trình import với option remap\_schema như sau:

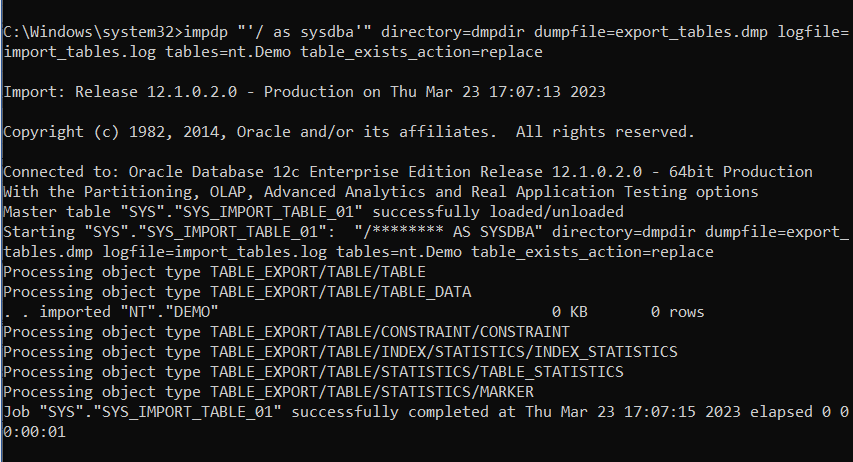
***impdp "'/ as sysdba'" directory=dmpdir dumpfile=export\_schema.dmp logfile=import\_schema.log schemas=nt remap\_schema=nt:nt1***

Text

Description automatically generated

* Import table

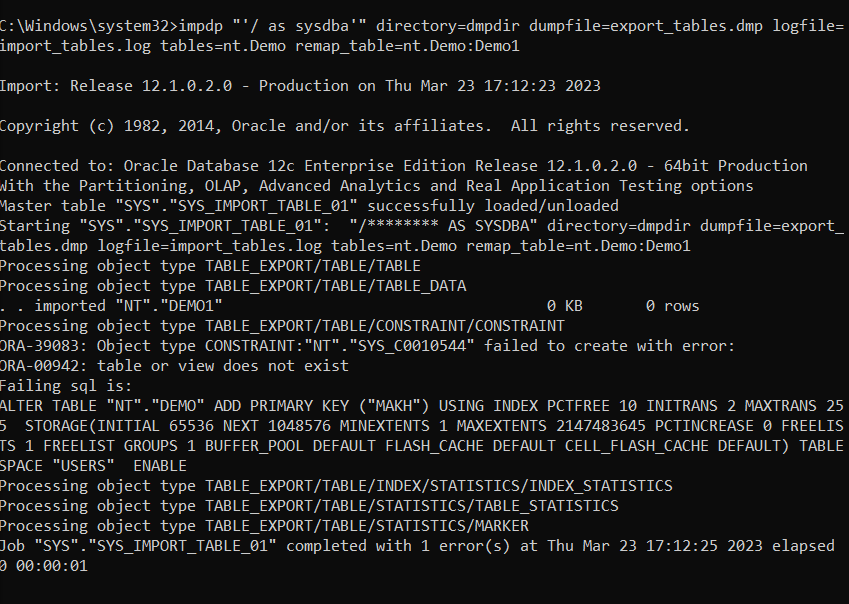
***impdp "'/ as sysdba'" directory=dmpdir dumpfile=export\_tables.dmp logfile=import\_tables.log tables=nt.Demo***

****

* + Nếu đã có table trên database đích, bạn có thể thêm option *table\_exists\_action=replace* để ghi đè lên các đối tượng đã có.

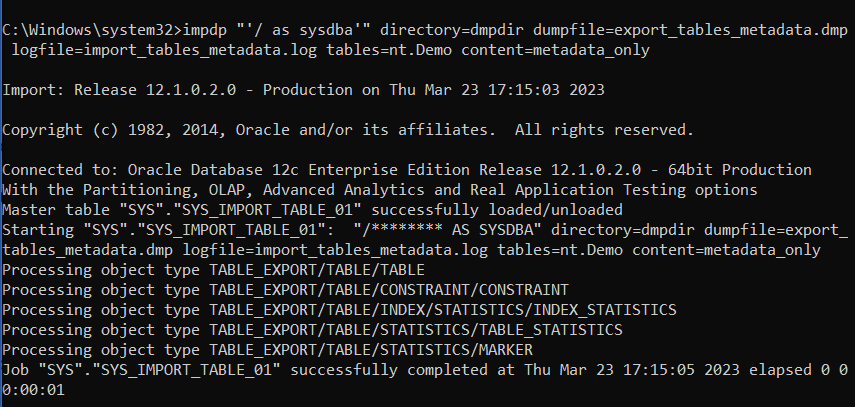
Bạn cũng có thể đổi tên tables trong quá trình import với option remap\_tables:

***impdp "'/ as sysdba'" directory=dmpdir dumpfile=export\_tables.dmp logfile=import\_tables.log tables=nt.Demo remap\_table=nt.Demo:Demo1***

****

* Import metadata, data:

***impdp "'/ as sysdba'" directory=dmpdir dumpfile=export\_tables\_metadata.dmp logfile=import\_tables\_metadata.log tables=nt.Demo content=metadata\_only***

****

***impdp "'/ as sysdba'" directory=dmpdir dumpfile=export\_tables\_metadata.dmp logfile=import\_tables\_data.log tables=nt.Demo content=data\_only***

***Text

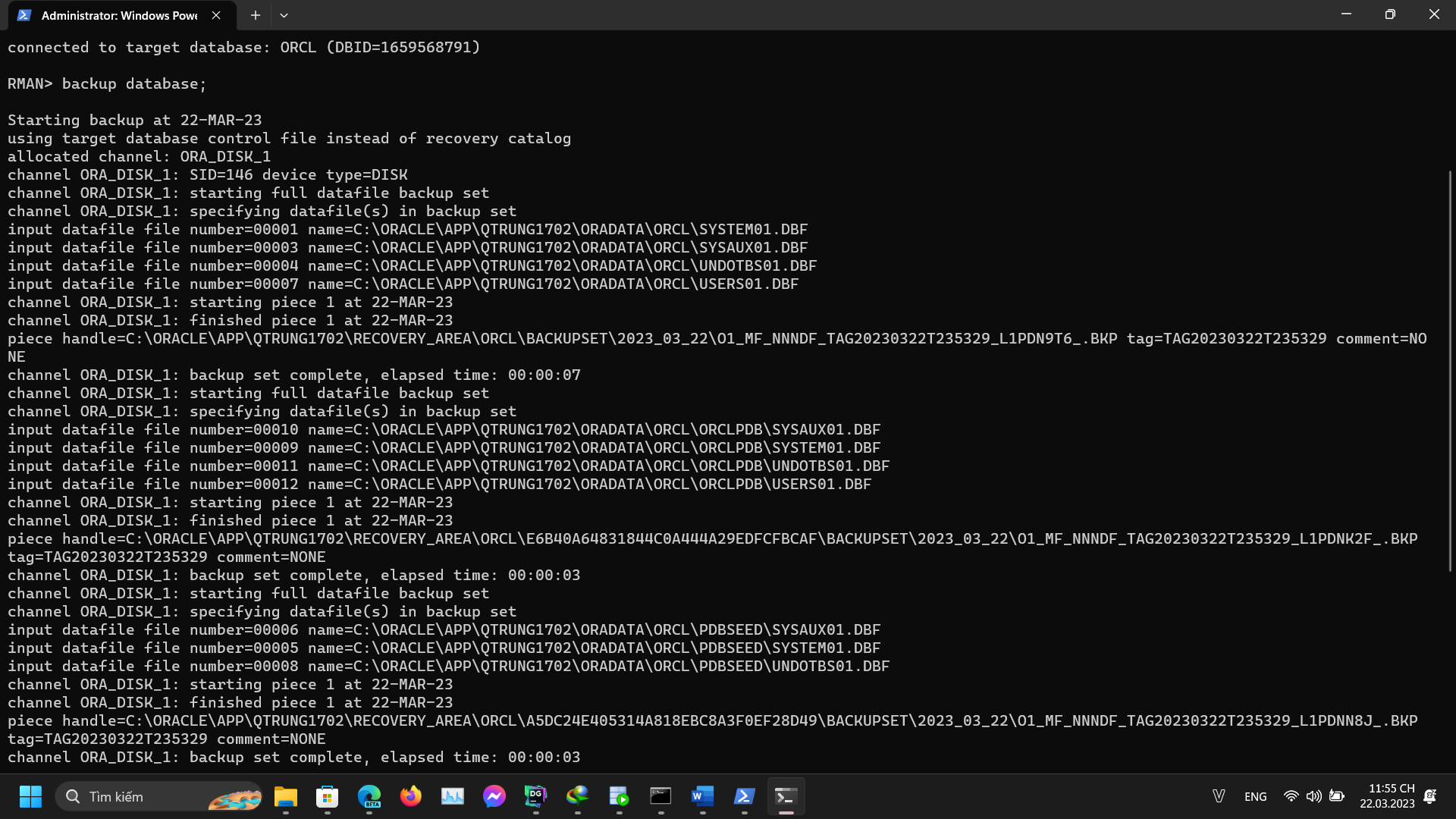
Description automatically generated***

### Sao lưu và khôi phục CSDL với RMAN

RMAN (Recovery Manager) là một công cụ dành cho việc sao lưu và khôi phục dữ liệu trong Oracle. RMAN có thể được sử dụng để sao lưu toàn bộ cơ sở dữ liệu Oracle hoặc một phần của cơ sở dữ liệu. Sau đây là các bước để sao lưu và khôi phục Oracle bằng RMAN:

B1: Vào Terminal gõ rman target /

B2: Gõ lệnh Backup database;



RMAN> backup database;

Starting backup at 22-MAR-23

using target database control file instead of recovery catalog

allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=400 device type=DISK

RMAN-00571: ===========================================================

RMAN-00569: =============== ERROR MESSAGE STACK FOLLOWS ===============

RMAN-00571: ===========================================================

RMAN-03002: failure of backup command at 03/22/2023 23:40:24

RMAN-06149: cannot BACKUP DATABASE in NOARCHIVELOG mode

Lỗi "cannot BACKUP DATABASE in NOARCHIVELOG mode" xuất hiện khi bạn cố gắng sao lưu cơ sở dữ liệu Oracle trong chế độ NOARCHIVELOG. NOARCHIVELOG mode là chế độ hoạt động của cơ sở dữ liệu Oracle trong đó nó không sử dụng bất kỳ redo log hoặc archive log nào để bảo vệ dữ liệu trong trường hợp xảy ra sự cố.

Để khắc phục lỗi này, bạn cần thực hiện một trong các hành động sau:

Chuyển đổi cơ sở dữ liệu sang chế độ ARCHIVELOG:

RMAN> SHUTDOWN IMMEDIATE;

database closed

database dismounted

Oracle instance shut down

RMAN> STARTUP MOUNT;

connected to target database (not started)

Oracle instance started

database mounted

Total System Global Area 1191181288 bytes

Fixed Size 9853928 bytes

Variable Size 452984832 bytes

Database Buffers 721420288 bytes

Redo Buffers 6922240 bytes

RMAN> ALTER DATABASE ARCHIVELOG;

Statement processed

RMAN> ALTER DATABASE OPEN;

Statement processed

Ảnh có chứa văn bản

Mô tả được tạo tự động

B3: Gõ lại lệnh Backup database;

-Bạn có thể chỉ định tên và vị trí cho tệp sao lưu bằng câu lệnh sau:

RMAN> backup database format '/u01/app/backup/oracle\_backup\_%U.bkp';

B4: Kiểm tra xem sao lưu có thành công không

RMAN> list backup;

### Khôi phục Oracle bằng RMAN

Bước 1: Khởi động RMAN và kết nối với cơ sở dữ liệu Oracle

rman target /

Bước 2: Khôi phục cơ sở dữ liệu

RMAN> run {

allocate channel ch1 type disk;

restore database;

recover database;

}

Ảnh có chứa văn bản

Mô tả được tạo tự động

Bước 3: Mở cơ sở dữ liệu

RMAN> alter database open;

Lưu ý: Trong quá trình khôi phục, RMAN sẽ yêu cầu bạn nhập tên và vị trí của các tệp sao lưu. Nếu bạn muốn khôi phục một tệp sao lưu cụ thể, bạn có thể sử dụng câu lệnh sau:

RMAN> restore database from '/u01/app/backup/oracle\_backup\_1.bkp';

Ảnh có chứa văn bản

Mô tả được tạo tự động

# Tuần 7: 24.03.2023

## Nội dung báo cáo

Cài đặt các chức năng truy vấn, tìm kiếm, cảnh báo có sử dụng các hàm có

sẵn trong Oracle, các cấu trúc điều khiển kết hợp với procedure, function,

package. Mỗi sinh viên làm ít nhất 2 câu

## Nội dung thực hiện

### Bảng (Table)

CREATE TABLE KHACHHANG   
(   
 MAKH CHAR(10) NOT NULL Primary Key ,  
 TENKH NVARCHAR2(50),  
 DIENTHOAI CHAR(10),  
 DIACHI NVARCHAR2(250)  
);  
CREATE TABLE NHANVIEN   
(   
 MANV CHAR(10) NOT NULL PRIMARY KEY ,  
 TENVN NVARCHAR2(50),  
 DIENTHOAI CHAR(10),  
 DIACHI NVARCHAR2(250),  
 CHUCVU NVARCHAR2(10)  
);  
CREATE TABLE MATHANG   
(   
 MAHANG CHAR(10) NOT NULL,   
 TENHANG NVARCHAR2(50),  
 NSX NVARCHAR2(50),  
 GIABAN FLOAT,  
 TRANGTHAI nvarchar2(20),  
 CONSTRAINT PK\_MATHANG PRIMARY KEY (MAHANG)  
);  
CREATE TABLE HOADON   
(   
 MAHD CHAR(10) NOT NULL,   
 MANV CHAR(10),  
 MAKH CHAR(10),  
 NGAYLAP DATE,  
 MAHANG CHAR(10),  
 TONGTIEN FLOAT,  
 CONSTRAINT PK\_HD PRIMARY KEY (MAHD),  
 FOREIGN KEY (MANV) REFERENCES NHANVIEN(MANV)  
);  
CREATE TABLE CHITIETHD   
(   
 MAHD CHAR(10) NOT NULL,   
 MAHANG CHAR(10),  
 TENHANG NVARCHAR2(50),  
 SOLUONG INT,  
 GIABAN FLOAT,  
 THANHTIEN FLOAT,  
 CONSTRAINT PK\_CTHD PRIMARY KEY (MAHD),  
 FOREIGN KEY (MAHD) REFERENCES HOADON(MAHD),  
 FOREIGN KEY (MAHANG) REFERENCES MATHANG(MAHANG)  
);

### Ràng buộc (constraint)

### Thủ tục (Procedure)

--Thủ tuc de tao mot don hang moi

CREATE OR REPLACE PROCEDURE TaoDonHang (

    p\_ID\_KhachHang IN NUMBER,

    p\_NgayDatHang IN DATE,

    p\_ChiTietDonHang IN ChiTietDonHang%ROWTYPE,

    p\_TongTien OUT NUMBER

) IS

    v\_ID\_DonHang NUMBER;

BEGIN

    -- Tạo một đơn hàng mới

    INSERT INTO DonHang (ID\_KhachHang, NgayDatHang, TongTien)

    VALUES (p\_ID\_KhachHang, p\_NgayDatHang, 0)

    RETURNING ID\_DonHang INTO v\_ID\_DonHang;

    -- Thêm các sản phẩm trong chi tiết đơn hàng

    INSERT INTO ChiTietDonHang (ID\_DonHang, ID\_SanPham, SoLuong, DonGia)

    VALUES (v\_ID\_DonHang, p\_ChiTietDonHang.ID\_SanPham, p\_ChiTietDonHang.SoLuong, p\_ChiTietDonHang.DonGia);

    -- Cập nhật tổng tiền của đơn hàng

    UPDATE DonHang

    SET TongTien = (SELECT SUM(SoLuong \* DonGia) FROM ChiTietDonHang WHERE ID\_DonHang = v\_ID\_DonHang)

    WHERE ID\_DonHang = v\_ID\_DonHang;

    -- Trả về tổng tiền của đơn hàng

    SELECT TongTien INTO p\_TongTien FROM DonHang WHERE ID\_DonHang = v\_ID\_DonHang;

END TaoDonHang;

--Tao procedure de them moi khach hang

CREATE OR REPLACE PROCEDURE them\_KhachHang (

p\_HoTen IN KhachHang.HoTen%TYPE,

p\_Email IN KhachHang.Email%TYPE,

p\_SoDienThoai IN KhachHang.SoDienThoai%TYPE,

p\_DiaChi IN KhachHang.DiaChi%TYPE

)

IS

BEGIN

INSERT INTO KhachHang (HoTen, Email, SoDienThoai, DiaChi)

VALUES (p\_HoTen, p\_Email, p\_SoDienThoai, p\_DiaChi);

COMMIT;

END;

### Hàm (Function)

--Tao function de tinh tong tien cua don hang

CREATE OR REPLACE FUNCTION tinh\_TongTienDonHang (

p\_ID\_DonHang IN DonHang.ID\_DonHang%TYPE

)

RETURN NUMBER

IS

v\_TongTien NUMBER(10,2);

BEGIN

SELECT SUM(SoLuong \* DonGia) INTO v\_TongTien

FROM ChiTietDonHang

WHERE ID\_DonHang = p\_ID\_DonHang;

RETURN v\_TongTien;

END;

### Gói (Package)

--Tạo package để quản lý các chức năng liên quan đến đơn hàng:

CREATE OR REPLACE PACKAGE pk\_DonHang AS

    PROCEDURE them\_DonHang (

        p\_ID\_KhachHang IN DonHang.ID\_KhachHang%TYPE,

        p\_NgayDatHang IN DonHang.NgayDatHang%TYPE,

        p\_TongTien IN DonHang.TongTien%TYPE

    );

    FUNCTION tim\_KhachHangTheoTen (

        p\_TenKhachHang IN KhachHang.HoTen%TYPE

    )

    RETURN SYS\_REFCURSOR;

    FUNCTION tinh\_TongTienDonHang (

        p\_ID\_DonHang IN DonHang.ID\_DonHang%TYPE

    )

    RETURN NUMBER;

END pk\_DonHang;

/

CREATE OR REPLACE PACKAGE BODY pk\_DonHang AS

    PROCEDURE them\_DonHang (

        p\_ID\_KhachHang IN DonHang.ID\_KhachHang%TYPE,

        p\_NgayDatHang IN DonHang.NgayDatHang%TYPE,

        p\_TongTien IN DonHang.TongTien%TYPE

    ) AS

    BEGIN

        INSERT INTO DonHang (ID\_KhachHang, NgayDatHang, TongTien)

        VALUES (p\_ID\_KhachHang, p\_NgayDatHang, p\_TongTien);

    END them\_DonHang;

    --Hàm tìm khach hang theo ten

    FUNCTION tim\_KhachHangTheoTen (

        p\_TenKhachHang IN KhachHang.HoTen%TYPE

    ) RETURN SYS\_REFCURSOR AS

        cur SYS\_REFCURSOR;

    BEGIN

        OPEN cur FOR SELECT \* FROM KhachHang WHERE HoTen = p\_TenKhachHang;

        RETURN cur;

    END tim\_KhachHangTheoTen;

    --Hàm tính tổng tiền đơn hàng

    FUNCTION tinh\_TongTienDonHang (

        p\_ID\_DonHang IN DonHang.ID\_DonHang%TYPE

    ) RETURN NUMBER AS

        v\_TongTien NUMBER(10,2);

    BEGIN

        SELECT SUM(SoLuong \* DonGia) INTO v\_TongTien

        FROM ChiTietDonHang

        WHERE ID\_DonHang = p\_ID\_DonHang;

        RETURN v\_TongTien;

    END tinh\_TongTienDonHang;

END pk\_DonHang;

# Tuần 8 31.03.2023

## Nội dung báo cáo

- Cài đặt đầy đủ các loại trigger: before, after, instead of, compound, sự kiện hệ thống (start, shutdown, log off). Mỗi sinh viên làm ít nhất 2 trigger

- Kết hợp trigger, thủ tục, hàm… để cài đặt chức năng hệ thống tự động tạo người dùng mới với profile mặc định, cách thức đặt tên theo quy định khi nhập mới thông tin liên quan đến người dùng. Ví dụ: khi nhập mới một sinh viên “nguyen thi lan”, hệ thống tự động tạo tài khoản đăng nhập “lannt”; nếu có 2 người trùng tên thì tự động chèn số thứ tự thêm vào sau như “lannt1”, “lannt2”

- Cài đặt phân quyền tự động cho người dùng được tạo mới

- Cài đặt tự động ghi nhận dữ liệu thêm, xóa, sửa của (một) bảng được xem là quan trọng

## Nội dung thực hiện

### Kích hoạt tự động (Trigger)

--Tạo cảnh báo khi tổng tiền của một đơn hàng vượt quá 10.000 đồng

CREATE TRIGGER trg\_TongTienDonHang

AFTER INSERT OR UPDATE ON DonHang

FOR EACH ROW

DECLARE

v\_TongTien NUMBER(10,2);

BEGIN

SELECT SUM(SoLuong \* DonGia) INTO v\_TongTien

FROM ChiTietDonHang

WHERE ID\_DonHang = :NEW.ID\_DonHang;

IF v\_TongTien > 10000 THEN

DBMS\_OUTPUT.PUT\_LINE('Tong tien don hang vuot qua 10000!');

END IF;

END;

---tự tạo user khi nhân viên được thêm

CREATE OR REPLACE TRIGGER create\_user\_trigger

AFTER INSERT ON NhanVien

FOR EACH ROW

DECLARE

  new\_username VARCHAR2(50);

BEGIN

  -- Tạo tên đăng nhập cho user mới bằng cách kết hợp họ và tên đệm của nhân viên

  new\_username := :new.TENNV || '\_' || :new.employee\_id;

  -- Tạo user mới với tên đăng nhập và mật khẩu mặc định là "password"

  EXECUTE IMMEDIATE 'CREATE USER ' || new\_username || ' IDENTIFIED BY password';

  -- Cấp quyền truy cập cho user mới vào các bảng và đối tượng trong cơ sở dữ liệu

  EXECUTE IMMEDIATE 'GRANT SELECT, INSERT, UPDATE, DELETE ON employees TO ' || new\_username;

  EXECUTE IMMEDIATE 'GRANT SELECT, INSERT, UPDATE, DELETE ON departments TO ' || new\_username;

  EXECUTE IMMEDIATE 'GRANT SELECT, INSERT, UPDATE, DELETE ON jobs TO ' || new\_username;

END;

* Tự xóa user khi xóa nhân viên

CREATE OR REPLACE TRIGGER delete\_user\_trigger

AFTER DELETE ON NhanVien

FOR EACH ROW

DECLARE

  username\_to\_delete VARCHAR2(50);

BEGIN

  -- Lấy tên đăng nhập của user tương ứng với nhân viên bị xóa

  SELECT TENNV || '\_' || MANV INTO username\_to\_delete

  FROM NhanVien

  WHERE MANV = :old.employee\_id;

  -- Xóa user có tên đăng nhập tương ứng

  EXECUTE IMMEDIATE 'DROP USER ' || username\_to\_delete;

END;

### Phân quyền tự động (Automatic authorization)

Để cài đặt phân quyền tự động cho người dùng được tạo mới trong Oracle database, bạn có thể sử dụng tính năng "Default Roles" của Oracle.

Bước 1: Tạo một vai trò mặc định (default role) cho các người dùng mới được tạo ra. Ví dụ, bạn có thể tạo một vai trò "user\_role" và cấp các quyền cần thiết cho vai trò này.

CREATE ROLE user\_role;

GRANT SELECT, INSERT, UPDATE, DELETE ON your\_table TO user\_role;

Bước 2: Kích hoạt tính năng "Default Roles" bằng cách sử dụng câu lệnh sau:

ALTER SYSTEM SET DEFAULT\_ROLE = 'user\_role' IDENTIFIED BY your\_password;

Bước 3: Cấp quyền cho người dùng để kích hoạt vai trò mặc định khi đăng nhập. Bạn có thể sử dụng câu lệnh sau để cấp quyền này cho một người dùng cụ thể:

GRANT SET DEFAULT ROLE TO your\_user;

Bước 4: Thử nghiệm tính năng "Default Roles" bằng cách tạo một người dùng mới và đăng nhập vào Oracle database. Người dùng mới này sẽ được kích hoạt vai trò mặc định "user\_role" và có thể sử dụng các quyền được cấp cho vai trò này.

CREATE USER new\_user IDENTIFIED BY new\_password;

GRANT CONNECT TO new\_user;

-------Phân quyền tự động-----

CREATE OR REPLACE TRIGGER new\_user\_trigger

AFTER CREATE ON SCHEMA

DECLARE

    v\_user varchar2(100);

BEGIN

    -- Lấy tên người dùng vừa được tạo

    v\_user := ora\_dict\_obj\_name;

    -- Cấp quyền SELECT cho người dùng mới

    EXECUTE IMMEDIATE 'GRANT SELECT, INSERT, UPDATE, DELETE ON <table\_name> TO ' || v\_user;

END;

### Ghi lại thay đổi quan trọng

1. Kích hoạt tính năng Audit trong Oracle DB bằng cách thực hiện lệnh sau:

Ảnh có chứa văn bản

Mô tả được tạo tự động

1. Sau đó khởi động lại DB

SQL> SHUTDOWN IMMEDIATE;

SQL> STARTUP;

1. Tạo một policy Audit cho bảng cần theo dõi bằng lệnh sau:

AUDIT INSERT, UPDATE, DELETE ON admin.sanpham BY ACCESS;

1. Kiểm tra xem policy Audit đã được tạo thành công bằng lệnh

SQL> SHOW PARAMETER AUDIT\_TRAIL;

Ảnh có chứa bàn

Mô tả được tạo tự động

1. Để xem bảng ghi

SELECT \*

FROM DBA\_AUDIT\_TRAIL

WHERE Ú = 'admin'

AND OBJECT\_NAME = 'SANPHAM';

# Tuần 9 07.04.2023

## 1. Nội dung thực hiện

- Cài đặt các giao tác đã phân tích trong bảng mô tả, có sử dụng các mức cô lập phù hợp với giao tác

- Thiết kế chức năng tìm kiếm session, user đang block dữ liệu; chức năng chọn lựa và xóa các session nào đang block dữ liệu

- Nội dung tìm hiểu thêm cộng điểm: Áp dụng mô hình bảo mật cơ sở dữ liệu, chọn một trong 2 kỹ thuật trong đồ án:

o Sử dụng VPD (Virtual Private Database)

o Sử dụng OLS (Oracle Label Security)

## 2. Nội dung thực hiện

### Giao tác

### Thao tác khi đang khóa dữ liệu

### Bảo mật cơ sở dữ liệu