TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO ĐỒ ÁN THỰC HÀNH
MÔN AN TOÀN BẢO MẬT HTTT

GVHD: Thầy Lương Vĩ Minh
Cô Tiết Gia Hồng

Nhóm: ATBM-A-11

Mục lục

I. Thông tin nhóm:

MSSV	Họ tên
21127035	Huỳnh Sơn Hà
21127041	Lý Nhật Hào
21127500	Lê Văn Dương

II. Phân công:

Phân hệ 1:

Chức năng	**Phân công**	**Độ hoàn thành**
Đăng nhập vào ứng dụng	Lý Nhật Hào	100%
Xem danh sách tài khoản người dùng trong hệ thống Oracle DB Server.	Huỳnh Sơn Hà	100%
Xem thông tin về quyền (privileges) của mỗi user/ role trên các đối tượng dữ liệu.	Huỳnh Sơn Hà	100%
Cho phép Tạo mới, Xóa, Sửa (hiệu chỉnh) user hoặc role.	Lê Văn Dương	100%
Cấp quyền cho user, cấp quyền cho role, cấp role user.	Lê Văn Dương	100%
Quá trình cấp quyền có tùy chọn là có cho phép người được cấp quyền có thể cấp quyền đó user/ role khác hay không (có chỉ định WITH GRANT OPTION hay không).	Huỳnh Sơn Hà	100%
Quyền select, update phải cho phép phân quyền đến mức cột; quyền insert, delete thì không.	Lê Văn Dương	100%
Cho phép thu hồi quyền hạn từ user/role.	Lý Nhật Hào	100%
Cho phép kiểm tra quyền của các chủ thể vừa được cấp quyền.	Lý Nhật Hào	100%

Phân hệ 2:

Yêu cầu	Chức năng	Phân công	Độ hoàn thà
Chung	Viết data.sql để tạo các bảng (nhansu, sinhvien, hocphan, khmo, phancong, dangky, thongbao) và thêm dữ liệu cho các bảng	Lý Nhật Hào	100%
	Viết setup.sql để tạo user ADM và ADMIN_OLS cùng với các thiết lập cần thiết cho các user này	Lê Văn Dương	100%
	Viết proc tạo user cho nhân sự và sinh viên, proc gán role cho các user (policies.sql)	Huỳnh Sơn Hà	100%
1	Chính sách CS#1: viết script phục vụ cho chính sách này (policies.sql) áp dụng cho các nhân viên cơ bản	Lý Nhật Hào	100%
	Chính sách CS#2: viết script phục vụ cho chính sách này (policies.sql) áp dụng cho các giảng viên	Huỳnh Sơn Hà	100%
	Chính sách CS#3: viết script phục vụ cho chính sách này (policies.sql) áp dụng cho các giáo vụ	Lê Văn Dương	100%
	Chính sách CS#4: viết script phục vụ cho chính sách này (policies.sql) áp dụng cho các trưởng đơn vị	Lý Nhật Hào	100%
	Chính sách CS#5: viết script phục vụ cho chính sách này (policies.sql) áp dụng cho các trưởng khoa	Huỳnh Sơn Hà	100%
	Chính sách CS#6: viết script phục vụ cho chính sách này (policies.sql) áp dụng cho các sinh viên	Lê Văn Dương	100%
	Viết các form chức năng liên quan đến nhân viên cơ bản và chính sách của họ	Lý Nhật Hào	100%
	Viết các form chức năng liên quan đến giảng viên và chính sách của họ	Huỳnh Sơn Hà	100%
	Viết các form chức năng liên quan đến giáo vụ và chính sách của họ	Lê Văn Dương	100%
	Viết các form chức năng liên quan đến trưởng đơn vị và chính sách của họ	Lý Nhật Hào	100%
	Viết các form chức năng liên quan đến trưởng khoa và chính sách của họ	Huỳnh Sơn Hà	100%
	Viết các form chức năng liên quan đến sinh viên và chính sách của họ	Lê Văn Dương	100%
2	Tạo ols policy kèm với các level, compartment và group (ols.sql)	Huỳnh Sơn Hà	100%
	Câu a: Viết script tạo label cho dữ liệu sẽ đc gán nhãn trong bảng thông báo, thêm dữ liệu và label của nó vào bảng thongbao, gắn label cho user tương ứng với yêu cầu đề (ols.sql)	Lý Nhật Hào	100%
	Câu b: Viết script tạo label cho dữ liệu sẽ đc gán nhãn trong bảng thông báo, thêm dữ liệu và label của nó vào bảng thongbao, gắn label cho user tương ứng với yêu cầu đề (ols.sql)	Huỳnh Sơn Hà	100%
	Câu c: Viết script tạo label cho dữ liệu sẽ đc gán nhãn trong bảng thông báo, thêm dữ liệu và label của nó vào bảng thongbao, gắn label cho user tương ứng với yêu cầu đề (ols.sql)	Lê Văn Dương	100%
	Câu d: Viết script tạo label cho dữ liệu sẽ đc gán nhãn trong bảng thông báo, thêm dữ liệu và label của nó vào bảng thongbao, gắn label cho user tương ứng với yêu cầu đề (ols.sql)	Lý Nhật Hào	100%

	Câu e: Viết script tạo label cho dữ liệu sẽ đc gán nhãn trong bảng thông báo, thêm dữ liệu và label của nó vào bảng thongbao, gắn label cho user tương ứng với yêu cầu đề (ols.sql)	Huỳnh Sơn Hà	100%
	Câu f: Viết script tạo label cho dữ liệu sẽ đc gán nhãn trong bảng thông báo, thêm dữ liệu và label của nó vào bảng thongbao, gắn label cho user tương ứng với yêu cầu đề (ols.sql)	Lê Văn Dương	100%
	Câu g: Viết script tạo label cho dữ liệu sẽ đc gán nhãn trong bảng thông báo, thêm dữ liệu và label của nó vào bảng thongbao, gắn label cho user tương ứng với yêu cầu đề (ols.sql)	Huỳnh Sơn Hà	100%
	Câu h1,2,3: Viết script tạo label cho dữ liệu sẽ đc gán nhãn trong bảng thông báo, thêm dữ liệu và label của nó vào bảng thongbao, gắn label cho user tương ứng với yêu cầu đề (ols.sql)	Lý Nhật Hào	100%
	Viết form để xem bảng thông báo cho các người dùng	Lê Văn Dương	100%
3	Viết form ứng dụng cho phần Standard audit: cho phép người dùng chọn audit trên bảng, view, function, proc với 2 lựa chọn khi thành công hoặc khi không thành công	Lê Văn Dương	100%
	Viết script fine-grained audit cho câu 3a: tạo policy	Lý Nhật Hào	100%
	Viết script fine-grained audit cho câu 3b: tạo policy	Huỳnh Sơn Hà	100%
	Viết form kiểm tra đọc dữ liệu nhật ký hệ thống cho cả hai audit	Lê Văn Dương	100%
4	Tìm hiểu về các phương pháp thực hiện sao lưu và phục hồi dữ liệu	Lý Nhật Hào	100%
	Hiện thực trên oracle (backup&recovery.sql)	Lê Văn Dương	100%
	Đánh giá ưu khuyết điểm các phương pháp	Huỳnh Sơn Hà	100%
	Kết luận	Lý Nhật Hào	100%

III. Hướng dẫn cài đặt trước khi sử dụng ứng dụng:

- Đầu tiên ta tạo 1 pdb (trương hợp nhóm em là tạo pdb có tên là pdbqlnb) bằng database configuration assistant
- Kết nối vào root với sys sau đó chạy file setup.sql:
+ Trong file sẽ kiểm tra ols đã đăng ký, bật chưa
+ nếu chưa thì bật lên, đăng ký, shutdown immediate và start up
+ Mở khóa user LBACSYS
+ Mở PDBQLNB
+ Chuyển sang sử dụng PDBQLNB
+Tạo user ADM với các quyền cần thiết để chứa tạo các bảng, policies
+Tạo user ADMIN_OLS với các quyền cần thiết để tạo chính sách ols
- Chạy file data.sql để tạo các bảng trong cơ sở dữ liệu kèm với insert data vào các bảng
- Chạy file policies.sql:
+ Bên trong chạy các proc để tạo user cho nhân sự và sinh viên kèm với các procedure gán role
+ Cài đặt cho các chính sách từ CS#1 đen CS#6
- Chạy file ols.sql để tạo chính sách ols và cài đặt label cho câu a đến h
- Chạy file fine_grained_audit.sql để tạo các chính sách audit cho câu a và b

LƯU Ý:
- Phần standard audit nhóm cài đặt trực tiếp trên form nên file standard_audit.sql chỉ chứa các câu lệnh được sử dụng cho phần cài đặt standrd audit chứ không phải để chạy toàn bộ chức năng (chi tiết có thể xem ở phần yêu cầu 3 dành cho audit).
- Tương tự, file rman_backupnrecovery.sql cũng chỉ chứ các khối lệnh liên quan đến việc thực hiện sao lưu và phục hồi nên chi tiết về thực hiện chức năng này có thể coi ở phần yêu cầu 4 ý 3 dành cho thực hành sao lưu và khôi phục trên oracle.

IV. Phân hệ 1:

Các yêu cầu của phân hệ 1 đã được hoành thành cài đặt trên ứng dụng, có thể thông qua sử dụng tài khoản **ADM** với password là **a** để vào giao diện admin để sử dụng các chức năng của yêu cầu 1

V. Phân hệ 2:

1. Yêu cầu 1: Cấp quyền truy cập:

a. Chính sách CS#1: Người có VAITRO là Nhân viên cơ bản

- Tạo và gán role rl_nhanviencoban cho các user nhân viên sau đó gán cho role này quyền select trên các bảng SINHVIEN, ĐƠNVỊ, HOCPHAN, KHMO.
- Với yêu cầu Xem dòng dữ liệu của chính mình trong quan hệ NHANSU, có thể chỉnh sửa số điện thoại (ĐT) của chính mình (nếu số điện thoại có thay đổi): Nhóm tạo view v_nhan_vien_co_ban với điều kiện manv = SYS_CONTEXT('USERENV','SESSION_USER') để giới hạn lại dữ liệu chỉ của user này kèm với with check option để hỗ trợ tuân thủ điều kiện trên khi update dữ liệu
- Gán quyền select, update trên v_nhan_vien_co_ban cho rl_nhanviencoban

b. Chính sách CS#2: Người có VAITRO là Giảng viên

- Gán rl_nhanviencoban cho các user giảng viên

- Tạo và gán rl_giangvien cho các user giảng viên
- Với yêu cầu Xem dữ liệu phân công giảng dạy liên quan đến bản thân mình (PHANCONG):
Nhóm tạo view v_giang_vien_phan_cong với điều kiện manv = SYS_CONTEXT ('USERENV','SESSION_USER') để giới hạn lại dữ liệu chỉ của user này.
- Với yêu cầu Xem dữ liệu trên quan hệ ĐANGKY liên quan đến các lớp học phần mà giảng viên được phân công giảng dạy:
Nhóm tạo view v_giang_vien_DANGKY với điều kiện magv = SYS_CONTEXT ('USERENV','SESSION_USER') để giới hạn lại dữ liệu chỉ của user này kèm với with check option để hỗ trợ tuân thủ điều kiện trên khi update dữ liệu
- Gán quyền select trên view v_giang_vien_DANGKY cho rl_giangvien
- Với yêu cầu Cập nhật dữ liệu tại các trường liên quan điểm số (trong quan hệ ĐANGKY) của các sinh viên có tham gia lớp học phần mà giảng viên đó được phân công giảng dạy:
Nhóm gán quyền update(DIEMTH, DIEMQT, DIEMCK, DIEMTK) trên view v_giang_vien_DANGKY cho rl_giangvien

c. Chính sách CS#3: Người có VAITRO là Giáo vụ

- Gán rl_nhanviencoban cho các user giáo vụ
- Tạo và gán rl_giaovu cho các user giáo vụ sau đó gán quyền xem, insert, update cho role này trên các bảng SINHVIEN, ĐƠNVỊ, HOCPHAN, KHMO.
- Với yêu cầu Xem dữ liệu trên toàn bộ quan hệ PHANCONG:
Gán quyền select trên phancong cho rl_giaovu
- Với yêu cầu chỉ được sửa trên các dòng dữ liệu phân công liên quan các học phần do "Văn phòng khoa" phụ trách phân công giảng dạy, thừa hành người trưởng đơn vị tương ứng là trưởng khoa:
Nhóm tạo view v_giao_vu_phan_cong bằng cách ghep bảng phân công và học phần với điều kiện hp.mahp = pc.mahp and hp.madv = 'VPK' để giới hạn lại dữ liệu của nhưng học phần của VPK kèm với with check option để hỗ trợ tuân thủ điều kiện trên khi update dữ liệu
- Gán quyền select, update trên v_giao_vu_phan_cong cho rl_giaovu (Vì khi muốn thực hiện update hay delete có kèm theo mệnh đề where thì cần phải kèm theo cả quyền select)
- Với yêu cầu Xóa hoặc Thêm mới dữ liệu trên quan hệ ĐANGKY theo yêu cầu của sinh viên trong khoảng thời gian còn cho hiệu chỉnh đăng ký:
Cảm thấy khó khăn trong việc cài đặt view vì có khá nhiều điều kiện bên trong nên nhóm quyết định sử dụng vpd cho yêu cầu này:
Nhóm tạo function f_hieu_chinh_dang_ky: lấy thời gian hiện tại để biết đc HK hiện tại là bao nhiêu cũng như biết đc năm hiện tại. Tùy vào học kỳ hiện tại mà tính sự chênh lệch giữa ngày hiện tại và ngày bắt đầu học kỳ tương ứng để để ghép vào trong vị từ trả về: RETURN 'HK = ' || HK || ' AND NAM = ' || YE || 'AND 0 < ' || DIFF || ' AND ' || DIFF || '<= 14'; (học kỳ và năm hiện tại và sự chênh lệch >0 và không quá 14 ngày kể từ ngày bắt đầu học kỳ)
- Add_policy với update_check = true
- Gán select, insert, delete trên dangky cho rl_giaovu (Vì khi muốn thực hiện update hay delete có kèm theo mệnh đề where thì cần phải kèm theo cả quyền select)

d. Chính sách CS#4: Người có VAITRO là Trưởng đơn vị

- Gán rl_giangvien và rl_nhanviencoban cho các user trưởng đơn vị (do gvien có quyền của nvcb)
- Tạo và gán rl_truongdonvi cho các user trưởng đơn vị

- Với yêu cầu Thêm, Xóa, Cập nhật dữ liệu trên quan hệ PHANCONG, đối với các học phần được phụ trách chuyên môn bởi đơn vị mà mình làm trưởng:
Nhóm thấy nếu tạo view cho yêu cầu này thì view sẽ được ghép bởi hai bảng nên không thể thực hiện insert bảng phân công thông qua view này nên nhóm quyết định xài vpd chi=o yêu cầu này:
Cài function F_TDV_PHANCONG: Kiểm tra nếu là giáo vụ thì trả về vị từ 'MAHP IN (SELECT MAHP FROM ' || P_SCHEMA || '.HOCPHAN WHERE MADV = "VPK")'; để đảm bảo chính sách 3. Nếu là trưởng đơn vị thì trả về vị từ: 'MAHP IN (SELECT MAHP FROM ' || P_SCHEMA || '.HOCPHAN WHERE MADV = (SELECT MADV FROM ' || P_SCHEMA || '.DONVI WHERE TRGDV = ''' || USR || '''))'; Để thỏa yêu cầu này.
- Add policy với function này và gán quyền select, insert, delete, update trên phân công cho rl_truongdonvi
- Với yêu cầu Được xem dữ liệu phân công giảng dạy của các giảng viên thuộc các đơn vị mà mình làm trưởng:
Nhóm cài view v_truongdonvi_phancong ghép các bảng nhân sự, đơn vị, phân công để lọc những giáo viên do user hiện tại làm tdv.
- Gán quyền select on view v_truongdonvi_phancong cho rl_truongdonvi

e. Chính sách CS#5: Người có VAITRO là Trưởng khoa

- Gán rl_giangvien và rl_nhanviencoban cho các user trưởng khoa (do gvien có quyền của nvcb)
- Tạo và gán rl_truongkhoa cho các user trưởng khoa
- Với yêu cầu Thêm, Xóa, Cập nhật dữ liệu trên quan hệ PHANCONG đối với các học phần quản lý bởi đơn vị "Văn phòng khoa":
Sửa lại function F_TDV_PHANCONG của trưởng đơn vị: Thêm trường hợp của Trưởng khoa vào
- Gán quyền select, insert, delete, update trên phân công cho rl_truongkhoa
- Gán quyền select, insert, delete, update trên nhân sự cho rl_truongkhoa
- Với yêu cầu Được quyền Xem (không giới hạn) dữ liệu trên toàn bộ lược đồ CSDL:
Nhóm viết procedure gán quyền select trên tất cả bảng cho rl_truongkhoa.

f. Chính sách CS#6: Người có VAITRO là Sinh viên:

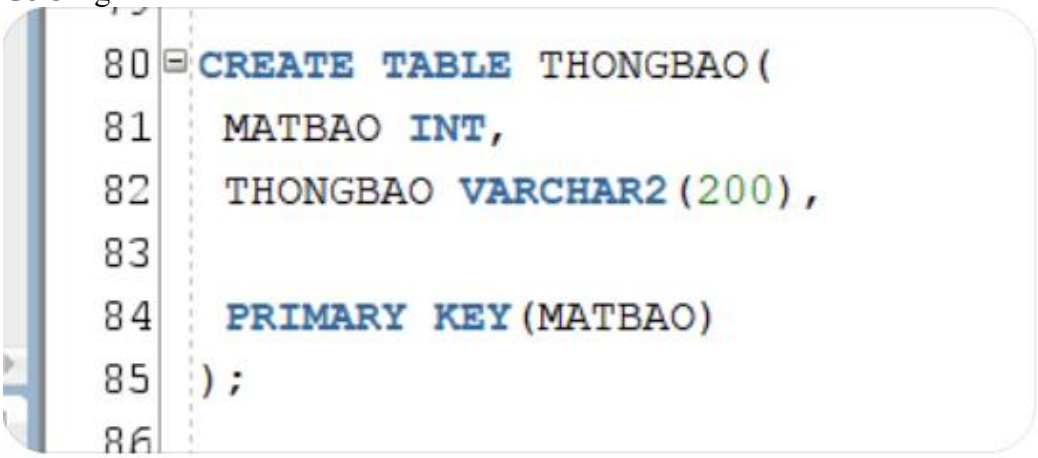
- Tạo và gán rl_sinhvien cho các user sinh viên
- Với yêu cầu Sinh viên chỉ được xem thông tin của chính mình, được Chỉnh sửa thông tin địa chỉ (ĐCHI) và số điện thoại liên lạc (ĐT) của chính sinh viên:
Nhóm cài function F_SINHVIEN: kiểm tra nếu là sinh viên thì trả về vị từ 'MASV = ''' || USR || ''' '; để giới hạn dữ liệu chỉ của sinh viên này
- Thực hiện add policy select trên sinh viên với function F_SINHVIEN
- Thực hiện add policy update trên cột DT và DCHI trên sinh viên với function F_SINHVIEN
- gán quyền select, update(diachi, DT) trên bảng sinhvien cho rl_sinhvien
- Với yêu cầu Xem danh sách tất cả học phần (HOCPHAN), kế hoạch mở môn (KHMO) của chương trình đào tạo mà sinh viên đang theo học:
Nhóm cài function F_SINHVIEN_KHMO: kiểm tra nếu là sinh viên thì thêm vị từ 'MACT = (SELECT MACT FROM SINHVIEN WHERE MASV = ''' || USR || ''')' để giới hạn các khóa học thuộc cùng mact của sinh viên. Add_policy select với funtion trên vào bảng khmo
Nhóm cài function F_SINHVIEN_HOCPHAN: kiểm tra nếu là sinh viên thì thêm vị từ 'MAHP IN (SELECT MAHP FROM KHMO WHERE MACT = (SELECT MACT FROM SINHVIEN

WHERE MASV = '" || USR || "')' để giới hạn các học phần thuộc cùng mact của sinh viên.
Add_policy select với funtion trên vào bảng học phần
- Gán quyền select trên khmo, học phần cho rl_sinhvien
- Với yêu cầu Thêm, Xóa các dòng dữ liệu đăng ký học phần (ĐANGKY) liên quan đến chính sinh viên đó trong học kỳ của năm học hiện tại (nếu thời điểm hiệu chỉnh đăng ký còn hợp lệ):
Nhóm sửa lại function f_hieu_chinh_dang_ky: thêm trường hợp nếu là sinh viên thì thêm điều kiện 'MASV = '" || USR || "' AND ' trước || 'HK = ' || HK || ' AND NAM = ' || YE || 'AND 0 < ' || DIFF || ' AND ' || DIFF || '<= 14';
- Với yêu cầu Sinh viên được Xem tất cả thông tin trên quan hệ ĐANGKY tại các dòng dữ liệu liên quan đến chính sinh viên:
Nhóm cài function F_SINHVIEN_DANGKY: Nếu là sinh viên thì trả về vị từ 'MASV = '" || USR || "" để giới hạn dữ liệu chỉ của sinh viên này.
- Gán quyền select, insert, delete, update(MASV,MAGV, MAHP, HK, NAM, MACT) trên dangky cho rl_sinhvien.

2. Yêu cầu 2: Vận dụng mô hình điều khiển truy cập OLS

Các bước chuẩn bị:

Có bảng THONGBAO như sau:



```
80  CREATE TABLE THONGBAO(
81    MATBAO INT,
82    THONGBAO VARCHAR2(200),
83
84    PRIMARY KEY(MATBAO)
85  );
86
```

Nhóm cài đặt policy như sau:

```
BEGIN SA_SYSDBA.CREATE_POLICY(
    POLICY_NAME   => 'P_THONGBAO',
    COLUMN_NAME   => 'DATA_LABEL'
); END;
/
```

Nhóm tạo các level, compartment và group như sau:

```
BEGIN
    SA_COMPONENTS.CREATE_LEVEL('P_THONGBAO',600, 'TKH', 'TRUONG KHOA');
    SA_COMPONENTS.CREATE_LEVEL('P_THONGBAO',500, 'TDV', 'TRUONG DON VI');
    SA_COMPONENTS.CREATE_LEVEL('P_THONGBAO',400, 'GV',  'GIANG VIEN');
    SA_COMPONENTS.CREATE_LEVEL('P_THONGBAO',300, 'GVU', 'GIAO VU');
    SA_COMPONENTS.CREATE_LEVEL('P_THONGBAO',200, 'NV',  'NHAN VIEN');
    SA_COMPONENTS.CREATE_LEVEL('P_THONGBAO',100, 'SV',  'SINH VIEN');

    SA_COMPONENTS.CREATE_COMPARTMENT('P_THONGBAO', 10, 'HTTT','HE THONG THONG TIN');
    SA_COMPONENTS.CREATE_COMPARTMENT('P_THONGBAO', 20, 'CNPM', 'CONG NGHE PHAN MEM');
    SA_COMPONENTS.CREATE_COMPARTMENT('P_THONGBAO', 30, 'KHMT', 'KHOA HOC MAY TINH');
    SA_COMPONENTS.CREATE_COMPARTMENT('P_THONGBAO', 40, 'CNTT', 'CONG NGHE TRI THUC');
```

```
    SA_COMPONENTS.CREATE_COMPARTMENT('P_THONGBAO', 50, 'TGMT', 'THI GIAC MAY TINH');
    SA_COMPONENTS.CREATE_COMPARTMENT('P_THONGBAO', 60, 'MMT', 'MANG MAY TINH VA VIEN
THONG');


    SA_COMPONENTS.CREATE_GROUP('P_THONGBAO', 700, 'CS1', 'CO SO 1', NULL);
    SA_COMPONENTS.CREATE_GROUP('P_THONGBAO', 800, 'CS2', 'CO SO 2', NULL);

END;
```

Với TKH là trưởng khoa, TDV là trưởng đơn vị, GV là giảng viên, GVU là giáo vụ, NV là nhân viên cơ bản, SV là sinh viên

Sau đó nhóm khởi tạo một số nhãn để có thể gắn những nhãn này cho dữ liệu trong bảng THONGBAO.

Nhóm tính label_tag bằng cách: label_tag = level * 100 + (sum(compartments)*10) + (sum(groups) * 1)

VD: `SA_LABEL_ADMIN.CREATE_LABEL('P_THONGBAO', '63600', 'TKH:HTTT,CNPM,KHMT,CNTT,TGMT,MMT:CS1,CS2');`

Khi apply_table_policy, cột DATA_LABEL sẽ tự được thêm vào ở cuối bảng THONGBAO:

```
BEGIN

    SA_POLICY_ADMIN.APPLY_TABLE_POLICY(
        POLICY_NAME   => 'P_THONGBAO',
        SCHEMA_NAME   => 'ADM',
        TABLE_NAME    => 'THONGBAO',
        TABLE_OPTIONS   => 'LABEL_DEFAULT,READ_CONTROL'
    );
END;
```

Tiếp theo tiến hành thêm các dữ liệu kèm nahwns của chúng vào bảng THONGBAO

VD: `INSERT INTO ADM.THONGBAO  VALUES(1, 'tbao cho truong khoa tat ca bo mon hai cs:', CHAR_TO_LABEL('P_THONGBAO', 'TKH:HTTT,CNPM,KHMT,CNTT,TGMT,MMT:CS1,CS2'));`

Sau đó ta tiến hành gán label cho các user

a. Hãy gán nhãn cho người dùng là Trưởng khoa có thể đọc được toàn bộ thông báo.

Gán nhãn cho TKH005 nhãn: `'TKH:HTTT,CNPM,KHMT,CNTT,TGMT,MMT:CS1,CS2'`

b. Hãy gán nhãn cho các Trưởng bộ môn phụ trách Cơ sở 2 có thể đọc được toàn bộ thông báo. dành cho trưởng bộ môn không phân biệt vị trí địa lý.

Gán nhãn cho các usr là tdv của cơ sở 2: `'TDV:HTTT,CNPM,KHMT,CNTT,TGMT,MMT:CS1,CS2'`

c. Hãy gán nhãn cho 01 Giáo vụ có thể đọc toàn bộ thông báo dành cho giáo vụ

Gán nhãn cho GVU000: `'GVU:HTTT,CNPM,KHMT,CNTT,TGMT,MMT:CS1,CS2'`

d. Hãy cho biết nhãn của dòng thông báo t1 để t1 được phát tán (đọc) bởi tất cả Trưởng đơn vị.

Nhãn của T1 là: `'GVU:HTTT,CNPM,KHMT,CNTT,TGMT,MMT:CS1,CS2'`

e. Hãy cho biết nhãn của dòng thông báo t2 để phát tán t2 đến Sinh viên thuộc ngành HTTT học ở Cơ sở 1.

Nhãn của T2: `'SV:HTTT:CS1'`

f. Hãy cho biết nhãn của dòng thông báo t3 để phát tán t3 đến Trưởng bộ môn KHMT ở Cơ sở 1.

Nhãn cũa T3: `'TDV:KHMT:CS1'`

g. Cho biết nhãn của dòng thông báo t4 để phát tán t4 đến Trưởng bộ môn KHMT ở Cơ sở 1 và Cơ sở 2.

Nhãn cũa T3: `'TDV:KHMT:CS1,CS2'`

h1. Các giảng viên đọc được hết thông báo của 2 cơ sở không quan trọng bộ môn

Gán nhãn cho các giảng viên: `'GV::CS1,CS2'`

h2. Nhân viên cơ sở 1 đọc được hết thông báo của cơ sơ mình làm việc không quan trọng bộ môn

Gán nhãn cho các nhân viên cơ sở 1: `'NV::CS1'`

h3. Sinh viên hai ngành cntt và cnpm có thể đọc thông báo của nhau ở cả hai cơ sở

Gán nhãn cho các sinh viên ngành cntt hoặc cnpm: `'SV:CNPM,CNTT:CS1,CS2'`

3. Yêu cầu 3: Ghi nhật ký hệ thống

Standard audit:

Nhóm ghi vào trg file standard_audit.sql những câu lệnh cần thiết để kích hoạt và hủy kích hoạt audit, đọc nhật ký truy vết, xóa nhật ký:

VD:
```
AUDIT ALL ON ADM.DANGKY BY ACCESS --Kích hoạt
NOAUDIT ALL ON ADM.DANGKY; --Hủy kích hoạt
select username, EXTENDED_TIMESTAMP ,obj_name, action_name, sql_text
from dba_audit_trail – Câu lệnh để đọc truy vết nhật ký.
DELETE FROM sys.aud$; -- xóa nhật ký
```

Nhóm thực hiện chức năng thêm audit trực tiếp trên ứng dụng winform:

- Yêu cầu người dùng chọn bảng, view, function hoặc procedure muốn audit
- Yêu cầu người dùng chọn 1 trg 2 hoặc cả 2 lựa chọn: khi thành công, khi không thành công
- Tiến hành ghép câu lệnh sql và thực hiện audit

Fine-grained audit:

a. Hành vi Cập nhật quan hệ ĐANGKY tại các trường liên quan đến điểm số nhưng người đó không thuộc vai trò Giảng viên.

```
BEGIN
  DBMS_FGA.ADD_POLICY(
   object_schema      => 'ADM',
   object_name        => 'DANGKY',
   policy_name        => 'p_AUDIT_ON_DANGKY',
   enable             =>  TRUE,
   AUDIT_COLUMN  => 'DIEMTH,DIEMQT,DIEMCK,DIEMTK',
   STATEMENT_TYPES  => 'UPDATE',
   audit_condition  => '1=CheckGV(SYS_CONTEXT(''USERENV'', ''SESSION_USER''))',
   audit_trail        =>  DBMS_FGA.DB_EXTENDED
   );
END;
/
```

Sử dụng Dbms_fga.add_policy để thêm chính sách audit lên bảng Dangky khi update các trưởng liên quan tới điểm điều kiện để ghi nhật ký là người thực hiện update không phải là giáo viên. Bọn em sử dụng 1 user là GVU000 để update điểm của 1 sv nhưng do vaitro giáo vụ không có quyền update trên bảng dangky nên bọn em sẽ cấp quyền update trên dangky cho user GVU000 này.
Khi dùng user GVU000 để update điểm trên dangky thì sẽ bị ghi vào nhật ký

b Hành vi của người dùng này có thể đọc trên trường PHUCAP của người khác ở quan hệ NHANSU.

```
BEGIN
  DBMS_FGA.ADD_POLICY(
   object_schema      => 'ADM',
   object_name        => 'NHANSU',
   policy_name        => 'p_AUDIT_ON_NHANSU',
   enable             =>  TRUE,
   AUDIT_COLUMN  => 'PHUCAP',
   STATEMENT_TYPES  => 'SELECT',
   AUDIT_CONDITION  => 'MANV <> SYS_CONTEXT(''USERENV'', ''SESSION_USER'')',
   audit_trail        =>  DBMS_FGA.DB_EXTENDED
   );
END;
/
```

Sử dụng Dbms_fga.add_policy để thêm chính sách audit lên bảng Nhansu khi select trưởng PHUCAP điều kiện để ghi nhật ký là người người bị truy xuất không phải người đang sử dụng
Khi sử dụng trưởng khoa để select lên bảng nhân sự thì sẽ xem được phụ cấ của những nhân viên khác, do đó hành động của trưởng khoa sẽ được lưu vào nhật ký

Để kiểm tra nhật ký của fga ta dùng câu truy vấn sau:

```
SELECT * FROM sys.FGA_LOG$;
```

Để xóa thì ta dùng câu lệnh sau:

```
DELETE FROM FGA_LOG$;
```

4. Yêu cầu 4: Sao lưu và phục hồi dữ liệu:

a. Các phương pháp sao lưu:

- **Full Backup**: Phương pháp sao lưu đầy đủ tạo một bản sao đầy đủ của toàn bộ cơ sở dữ liệu Oracle. Điều này bao gồm dữ liệu, bảng, chỉ mục và các đối tượng khác. Full backup đảm bảo khả năng phục hồi đầy đủ, nhưng yêu cầu tài nguyên lớn và thời gian sao lưu lâu.
    - **Ưu điểm:**
        - **Đảm bảo sao lưu toàn bộ dữ liệu**: Phương pháp Full Backup sao lưu toàn bộ dữ liệu, bao gồm cả dữ liệu mới và đã tồn tại. Điều này đảm bảo rằng không có dữ liệu nào bị bỏ sót trong quá trình sao lưu.
        - **Phục hồi dữ liệu dễ dàng**: Sao lưu Full Backup cho phép phục hồi dữ liệu nhanh chóng và dễ dàng vì tất cả các thông tin cần thiết đã được sao lưu.
    - **Khuyết điểm:**

        - **Tốn nhiều dung lượng lưu trữ**: Do sao lưu toàn bộ dữ liệu, phương pháp Full Backup yêu cầu dung lượng lưu trữ lớn hơn so với các phương pháp khác.
        - **Tốn thời gian**: Quá trình sao lưu Full Backup có thể tốn nhiều thời gian, đặc biệt khi có nhiều dữ liệu cần sao lưu.
- **Incremental Backup**: Phương pháp sao lưu lẻ chỉ sao lưu những phần của dữ liệu đã thay đổi kể từ lần sao lưu trước đó. Có ba loại incremental backup: level 0, level 1 và level 2. Level 0 sao lưu toàn bộ cơ sở dữ liệu, trong khi level 1 và level 2 chỉ sao lưu các phần đã thay đổi từ lần sao lưu trước đó. Điều này giúp tiết kiệm dung lượng lưu trữ và thời gian sao lưu.
    - **Ưu điểm:**
        - **Tiết kiệm dung lượng lưu trữ**: Phương pháp Incremental Backup chỉ sao lưu các thay đổi dữ liệu kể từ lần sao lưu trước đó. Điều này giúp tiết kiệm dung lượng lưu trữ so với Full Backup.
        - **Thời gian sao lưu nhanh**: Vì chỉ sao lưu các thay đổi, quá trình sao lưu Incremental Backup thường nhanh hơn so với Full Backup.
    - **Khuyết điểm:**
    - **Phức tạp trong việc phục hồi:** Phục hồi dữ liệu từ các sao lưu Incremental Backup phức tạp hơn so với Full Backup. Cần phải sử dụng các bản sao lưu trước đó và xử lý các thay đổi để tái tạo dữ liệu gốc.
    - **Cần quản lý các bản sao lưu:** Vì Incremental Backup phụ thuộc vào các bản sao lưu trước đó, việc quản lý và bảo quản các bản sao lưu là quan trọng để đảm bảo khả năng phục hồi dữ liệu.
- **Hot Backup:** Sao lưu Hot là phương pháp sao lưu dữ liệu trong khi cơ sở dữ liệu Oracle vẫn hoạt động và chạy trên hệ thống. Điều này cho phép tiếp tục truy cập và sử dụng cơ sở dữ liệu trong quá trình sao lưu. Trong quá trình sao lưu Hot, bản ghi log (redo log) của Oracle được sử dụng để ghi lại các thay đổi dữ liệu trong thời gian sao lưu. Điều này đảm bảo rằng dữ liệu được sao lưu là nhất quán và đầy đủ.
    - **Ưu điểm:**
        - **Không gián đoạn hoạt động**: Trong quá trình sao lưu Hot, cơ sở dữ liệu vẫn hoạt động và có thể truy cập. Điều này đảm bảo tính liên tục của hệ thống và không ảnh hưởng đến người dùng và ứng dụng.

        - **Tính nhất quán của dữ liệu:** Với việc sử dụng bản ghi log, sao lưu Hot đảm bảo tính nhất quán của dữ liệu trong quá trình sao lưu.
    - **Khuyết điểm:**
        - **Yêu cầu tài nguyên hệ thống**: Quá trình sao lưu Hot có thể tạo ra tải nặng cho hệ thống, đòi hỏi tài nguyên máy chủ và băng thông mạng đủ lớn.
        - **Không an toàn trong một số trường hợp:** Trong một số tình huống đặc biệt, sao lưu Hot có thể không đảm bảo tính toàn vẹn của dữ liệu, chẳng hạn như khi có sự cố hệ thống xảy ra trong quá trình sao lưu.
- **Cold Backup:**Sao lưu Cold là phương pháp sao lưu dữ liệu khi cơ sở dữ liệu Oracle đã được tắt hoàn toàn, không hoạt động trên hệ thống. Trước khi thực hiện sao lưu, cơ sở dữ liệu Oracle được dừng hoạt động hoàn toàn. Điều này đảm bảo tính nhất quán của dữ liệu trong quá trình sao lưu. Tất cả các tệp và thư mục liên quan đến cơ sở dữ liệu Oracle được sao lưu, bao gồm bản sao của bản ghi log, bảng không gian, control file và các thành phần khác. Sau khi sao lưu hoàn tất, cơ sở dữ liệu Oracle được khởi động lại để tiếp tục hoạt động.
    - **Ưu điểm:**
        - **Đảm bảo tính toàn vẹn của dữ liệu:** Với sao lưu Cold, hệ thống dừng hoạt động hoàn toàn trước khi thực hiện sao lưu. Điều này đảm bảo tính toàn vẹn của dữ liệu được sao lưu.
        - **Không ảnh hưởng đến hiệu suất hệ thống**: Vì hệ thống dừng hoạt động trong quá trình sao lưu Cold, không có tác động đến hiệu suất và tài nguyên của hệ thống.
    - **Khuyết điểm:**
        - **Gián đoạn hoạt động:** Quá trình sao lưu Cold yêu cầu dừng hoạt động của hệ thống, dẫn đến gián đoạn và không thể truy cập dữ liệu trong thời gian sao lưu.
        - **Thời gian sao lưu lâu**: Do cần dừng hoạt động của hệ thống, quá trình sao lưu Cold thường mất nhiều thời gian hơn so với các phương pháp khác.

* Để quá trình back up được thực hiện tự động thì nhóm đã cài đặt 1 file backup.rman để lưu những câu lệnh thực hiện backup và 1 file auto_backup.bat để khai báo tên database, đường dẫn đến rman trong ứng dụng oracle và đoạn code thực hiện chạy file backup.rman.
Sau đó, nhóm sử dụng ứng dụng task scheduler có sẵn của window để set up kịch bản thực hiện chạy file auto_backup.bat vào khung giờ đã chọn trc (có thể hằng ngày, hằng tuần,…)

b. Các phương pháp phục hồi:

- **Phục hồi đầy đủ (Complete Recovery)**: Phương pháp này sử dụng bản sao lưu đầy đủ và tất cả các bản ghi log sau đó để khôi phục cơ sở dữ liệu từ một điểm trong quá khứ đến thời điểm lỗi. Quá trình này đảm bảo tính toàn vẹn và nhất quán của dữ liệu, nhưng thời gian phục hồi có thể lâu.
- **Phục hồi điểm trong thời gian (Point-in-Time Recovery)**: Phương pháp này cho phép phục hồi cơ sở dữ liệu đến một điểm trong quá khứ cụ thể. Để thực hiện phục hồi điểm trong thời gian, chúng ta sử dụng bản sao lưu đầy đủ, các bản ghi log và các bản ghi log bổ sung (**archived_log**) sau thời điểm bản sao lưu. Quá trình này cho phép khôi phục dữ liệu đến một trạng thái được chỉ định trước khi sự cố xảy ra.

c. Thực hiện trên Oracle:

- bật audit trên bảng, view muốn truy vết, ở đây là view v_giang_vien_DANGKY
- chạy trong terminal

login sys: sqlplus / as sysdba

```
archive log list;
shutdown immediate;
startup mount;
alter database archivelog; --bật chế độ archivelog cho database
```

- vô terminal -> su dung rman

--rman target /

- tiến hành back up database:

```
BACKUP DATABASE PLUS ARCHIVELOG FORMAT 'D:/backup/bu_%u';
```

- Log vào user gv1002 để tạo vài update

```
update adm.v_giang_vien_DANGKY set diemtk=6.90 --update 1

update adm.v_giang_vien_DANGKY set diemtk=9.60 --update 2
```

- Ta vào terminal để xem các file backup và tag của chúng bằng câu lệnh:

LIST BACKUP OF DATABASE SUMMARY;

- Sau đó ta vào lại oracle bằng user adm để xem nhật ký lưu vết của 2 lần update trên để lấy giá trị scn của chúng bằng câu lệnh:

```
SELECT * FROM dba_audit_trail order by EXTENDED_TIMESTAMP desc;
```

- Sau khi có đc scn của update 1 và 2, nếu muốn recovery database cho đến lần update thứ 2 tức là sau khi update 1 thành công thì ta chạy block lệnh sau trong terminal:

```
RUN {
    SHUTDOWN IMMEDIATE;
    STARTUP MOUNT;
    SET UNTIL SCN 74053317; --Đây là scn của lần update thứ 2
    RESTORE DATABASE FROM TAG TAG20240430T221302; -- Đây là tag của file backup
    RECOVER DATABASE;
    ALTER DATABASE OPEN RESETLOGS;
}
```

- Sau đó chạy database lên, truy vấn audit ta sẽ chỉ thấy ghi vết của lần update 1 th còn lần 2 thì ko còn nữa.

d. Kết luận:

Trong phiên bản Oracle hiện tại, RMAN được coi là công cụ mạnh mẽ và tốt nhất cho việc sao lưu và phục hồi dữ liệu. RMAN không chỉ giúp tối ưu và cải thiện quy trình, mà còn cung cấp nhiều tính năng mạnh mẽ để đảm bảo sự an toàn và giảm thiểu rủi ro mất dữ liệu. Đồng thời, để tận dụng RMAN một cách hiệu quả, cần xây dựng chiến lược bài bản có chủ ý thông qua những phương pháp mạnh mẽ sẽ được đề cập bên dưới:

Việc thực hiện sao lưu toàn bộ (Hot backup - Full recovery) không nên được thực hiện 1 cách quá thường xuyên, mà chỉ nên thực hiện theo các chu kỳ cố định cụ thể, chẳng hạn sau mỗi cột mốc quan trọng theo tháng, quý, theo năm hoặc sự kiện thời gian đặc biệt. Nên lựa chọn thời điểm ít hoạt động để giảm thiểu tính không nhất quán. Ngoài ra, có thể sử dụng Cold backup khi có cơ hội tắt hệ thống hoàn toàn, chẳng hạn để thực hiện bảo trì hoặc chuẩn bị di chuyển.

Trên thực tế, Incremental backup thường được ưu tiên hàng đầu nhờ tính đa dạng và linh hoạt. Nếu môi trường ổn định và không đặt nặng yêu cầu phục hồi dữ liệu, lựa chọn Differential backup có thể được áp dụng được tốt không kém.

Cuối cùng, việc sao lưu và phục hồi với RMAN có thể được tự động hóa bằng cách viết các tập lệnh (batch file) xác định quy trình thực hiện và sử dụng lịch chạy (Windows Task Scheduler) để lập lịch thực hiện tự động theo định kỳ.