

Bài tập tuần 11-12

Tạo Function

1. Tạo hàm có tên `sum_salary` tính tổng lương của từng phòng ban với tham số truyền vào là mã phòng ban.
2. Tạo hàm có tên `name_con` cho biết tên quốc gia với mã quốc gia là tham số truyền vào.
3. Tạo hàm có tên là `annual_comp` tính thu nhập hàng năm với tham số truyền vào là lương tháng và hoa hồng. (lương hàng năm = lương tháng * 12 + (hoa hồng * lương tháng*12)).
4. Tạo hàm có tên `avg_salary` tính lương trung bình của từng phòng ban với tham số truyền vào là mã phòng ban.
5. Tạo hàm có tên là `time_work` tính số tháng làm việc của nhân viên, với tham số truyền vào là mã nhân viên

Tạo Package

6. Tạo package có tên `emp_info` chứa thủ tục `salary_table` cho biết thông tin về lương của nhân viên và một hàm có tên `sum_salary` tính tổng lương của các phòng ban với mã phòng ban là tham số truyền vào.
7. Tạo package có tên `employer` chứa thủ tục `employ_info` cho biết mã nhân viên, tên nhân viên, lương, hoa hồng, ngày thuê, tên người quản lý, tên công việc, tên phòng ban với mã nhân viên là tham số truyền vào.
8. Tạo package có tên là `job_pack` chứa 3 thủ tục tên là `add_job` để thêm công việc, `update_job` để cập nhật công việc, `del_job` để xóa công việc và 1 hàm có tên `q_job` để tìm

Trigger

9. Cài đặt ràng buộc toàn vẹn ngày thuê nhân viên phải nhỏ hơn hoặc bằng ngày hiện hành khi thêm mới hoặc cập nhật thông tin về nhân viên.
10. Cài đặt ràng buộc toàn vẹn `min_salary` luôn nhỏ hơn `max_salary` khi thêm mới hoặc cập nhật thông tin bảng công việc.
11. Cài đặt ràng buộc tên công việc phải duy nhất.
12. Cài đặt ràng buộc toàn vẹn ngày bắt đầu luôn nhỏ hơn hoặc bằng ngày kết thúc khi thêm mới hoặc cập nhật thông tin bảng `job_history`.
13. Cài đặt ràng buộc toàn vẹn lương và hoa hồng của nhân viên phải tăng chứ không giảm khi cập nhật nhân viên.
14. Cài đặt ràng buộc toàn vẹn Ngày vào làm phải trước ngày bắt đầu tham gia công việc.

```

create table statistic
(
    table_name varchar2(20),
    transaction_name varchar2(10),
    transaction_user varchar2(30),
    transaction_date date
)

CREATE OR REPLACE TRIGGER instructor_aud AFTER
UPDATE OR DELETE ON instructor
    DECLARE v_type VARCHAR2(10);
BEGIN
    IF updating THEN
        v_type:='Update';
    ELSE
        v_type:='Delete';
    END IF;
    UPDATE statistic
    SET transaction_user=USER,
        transaction_date =sysdate
    WHERE table_name  ='Employees'
    AND transaction_name=v_type;
    IF sql%notfound THEN
        INSERT INTO statistic
        VALUES('Instructor',v_type,USER,sysdate);
    END IF;
END;

```