

2019-2020 Dönemi Veri Tabanı Yönetimi Dersi (2. Grup - UK) Laboratuvarı 2. Quizi Cevap Anahtarı

25.11.2019

NOT: Teslim edeceğiniz .sql uzantılı dosyanın adına öğrenci numaranızı veriniz.

Süre: 80 dk

Soru 1: Her çalışanın kaç tane akrabası olduğunu bulan ve çalışan ismine göre alfabetik olarak sıralayan SQL sorgusunu yazınız.

```
SELECT fname, COUNT(*) FROM employee, dependent WHERE ssn = essn GROUP BY fname ORDER BY fname;
```

Soru 2: Çalışanlarının ortalama maaşı 40000'den fazla olan departman isimlerini bulan SQL sorgusunu yazınız.

```
SELECT dname, AVG(salary) FROM employee, department WHERE dno = dnumber GROUP BY dname HAVING AVG(salary) > 40000;
```

Soru 3: *Departmanlarda çalışan kişilerin ortalama maaşlarının, departmanlarda yapılan proje sayısına bölümüne ağırlıklı maaş denilmektedir.* İsmi verilen bir departmanın ağırlıklı maaşı, şirketin en düşük maaşa sahip kadın çalışanın maaşından daha yüksekse, o kadın çalışanın maaşını numarası verilen departmanın ağırlıklı maaşı olarak belirleyen PL/pgSQL fonksiyonunu yazınız.

```
CREATE OR REPLACE FUNCTION soru3(bolum department.dname%TYPE) RETURNS VOID AS '  
DECLARE  
    ort_maas real;  
    pro_say integer;  
    agirlikli_maas real;  
    en_dusuk_f integer;  
    kadin employee.ssn%TYPE;  
BEGIN  
    SELECT AVG(salary) INTO ort_maas FROM employee, department  
        WHERE dno = dnumber AND dname = bolum;  
    SELECT COUNT(*) INTO pro_say FROM department, project  
        WHERE dnum = dnumber AND dname = bolum;  
    IF pro_say = 0 THEN  
        agirlikli_maas := 1;  
    ELSE  
        agirlikli_maas := ort_maas / pro_say;  
    END IF;  
    SELECT MIN(salary) INTO en_dusuk_f FROM employee WHERE sex = "F";  
    SELECT ssn INTO kadin FROM employee WHERE salary = en_dusuk_f AND sex = "F";  
    IF agirlikli_maas > en_dusuk_f THEN  
        UPDATE employee SET salary = agirlikli_maas WHERE ssn = kadin;  
    END IF;  
END;  
' LANGUAGE plpgsql;  
--SELECT soru3('Headquarters');
```

Soru 4: En yaşlı kişinin çalıştığı departmanda verilen bir değerden daha fazla kişi çalışıyorsa ve o departmanda yapılan projeler verilen bir değerden az ise, o departmanda departman yöneticisinden daha yaşlı olan erkek çalışanlarının projede/projelerde çalışmış olduğu bilgisini silen ve yine o departmandaki en düşük maaşı alan kadın hariç diğer kadın çalışanların maaşlarına verilen bir oranda indirim yapan PL/pgSQL fonksiyonunu yazınız.

```
CREATE OR REPLACE FUNCTION soru4(yasli_deger integer, proje_deger integer, indirim real) RETURNS
VOID AS '
DECLARE
    yasli_calisan integer;
    yasli_dep employee.dno%TYPE;
    yapilan_proje integer;
    dep_yon employee.ssn%TYPE;
BEGIN
    SELECT dno INTO yasli_dep FROM employee WHERE bdate IN
        (SELECT MIN(bdate) FROM employee);
    SELECT COUNT(*) INTO yasli_calisan FROM employee WHERE dno = yasli_dep;
    IF yasli_deger < yasli_calisan THEN
        SELECT COUNT(*) INTO yapilan_proje FROM project WHERE dnum = yasli_dep;
        IF yapilan_proje < proje_deger THEN
            SELECT mgrssn INTO dep_yon FROM department WHERE dnumber = yasli_dep;
            DELETE FROM works_on WHERE essn IN
                (SELECT e.ssn FROM employee e, employee dm, department d
                 WHERE dm.ssn = dep_yon AND e.dno = d.dnumber
                 AND d.mgrssn = dm.ssn AND dm.bdate > e.bdate AND sex = "M");
            UPDATE employee SET salary = salary - (salary * indirim)/100
                WHERE dno = yasli_dep AND sex = "F" AND ssn NOT IN
                (SELECT ssn FROM employee WHERE dno = yasli_dep AND
                 sex = "F" ORDER BY salary LIMIT 1);
        END IF;
    END IF;
END;
' LANGUAGE plpgsql;
--SELECT soru4(5, 5, 5);
```