**T.C.**

**ESKİŞEHİR OSMANGAZİ ÜNİVERSİTESİ**

**MÜHENDİSLİK VE MİMARLIK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**DERSİN ADI:**

VERİ TABANI YÖNETİM

SİSTEMLERİ LABORATUVARI

**DENEY #7**

**Doç. Ahmet YAZICI**

Gizem Taşcı - 152120131100

**ESKİŞEHİR**

ÇÖZÜMLER

**1.**

create database db\_lab7

create table tbl\_Employee(

Fname varchar(20),

Minit varchar(10),

Lname varchar(20),

Ssn varchar(20),

Addres varchar(50),

Sex varchar(5),

Salary decimal,

Super\_ssn varchar(20),

Dno int,

Bdate date,

Primary key (Ssn),

Foreign key (Super\_ssn) references tbl\_Employee (Ssn)

);

create table tbl\_Department(

Dname varchar(20),

Dnumber int,

Mgr\_ssn varchar(20),

Mgr\_start\_date date,

Primary key (Dnumber),

Foreign key (Mgr\_ssn) references tbl\_Employee(Ssn)

);

create table tbl\_Dept\_Locations(

Dlocations varchar(20),

Dnumber int,

Primary key(Dlocations),

Foreign key (Dnumber) references tbl\_Department(Dnumber)

);

create table tbl\_Project(

Pname varchar(20),

Pnumber int,

Plocation varchar(20),

Dnum int,

Primary key (Pnumber),

Foreign key (Dnum) references tbl\_Department(Dnumber)

);

create table tbl\_Works\_On(

Essn varchar(20),

Pno int,

Hourss time,

Foreign key (Essn) references tbl\_Employee(Ssn),

Foreign key (Pno) references tbl\_Project(Pnumber)

);

create table tbl\_Dependent(

Essn varchar(20),

Dependent\_name varchar (20),

Sex varchar(5),

Bdate date,

Relationship varchar(20),

Primary key(Dependent\_name),

Foreign key (Essn)references tbl\_Employee(Ssn)

);

insert into tbl\_Employee values('Mevlüt', null,'Yurttaş',1,'Ankara','Kadın',5000,1,01,'10.18.1993')

insert into tbl\_Employee values('Elif',null,'Karaayaz',2,'Ankara','Erkek',4000,2,02,'08.10.1994')

insert into tbl\_Employee values('Burak', null,'Yıldırım',3,'KAYSERİ','Erkek',2500,3,03,'05.25.1993')

insert into tbl\_Employee values('Özge', null,'Aslan',4,'Ankara','Kadın',1000,4,04,'02.12.1994')

insert into tbl\_Employee values('Ali', null,'Yılmaz',5,'Eskişehir','Erkek',3000,5,05,'03.20.1993')

insert into tbl\_Employee values('Akın',null,'Aydın',6,'Eskişehir','Erkek',1800,NULL,02,'05.16.1985')

insert into tbl\_Employee values('Emre',null,'Sarı',7,'İstanbul','Erkek',1800,NULL,03,'09.25.1988')

insert into tbl\_Employee values('Mustafa',null,'Can',8,'İstanbul','Erkek',1500,NULL,05,'01.01.1970')

insert into tbl\_Employee values('Ali',null,'Yurttaş',9,'Kütahya','Erkek',1800,NULL,03,'03.15.1961')

insert into tbl\_Employee values('Metehan',null,'Doğan',10,'Ankara','Erkek',1500,NULL,05,'08.13.1995')

insert into tbl\_department values('Arge', 01,NULL,'05.07.2014')

insert into tbl\_department values('A', 02,NULL,'10.18.2012')

insert into tbl\_department values('D', 03,null,'08.06.2011')

insert into tbl\_department values('Arge', 04,null,'12.01.2012')

insert into tbl\_department values('G', 05,NULL,'04.04.2012')

insert into tbl\_Dept\_Locations values('Eskişehir',01)

insert into tbl\_Dept\_Locations values('Ankara',02)

insert into tbl\_Dept\_Locations values('Kütahya',03)

insert into tbl\_Dept\_Locations values('Kayseri',04)

insert into tbl\_Dept\_Locations values('İstanbul',05)

insert into tbl\_Project values('Project1',101,'Eskişehir',01)

insert into tbl\_Project values('Project2',102,'Ankara',02)

insert into tbl\_Project values('Project3',103,'İstanbul',05)

insert into tbl\_Project values('Project4',104,'Eskişehir',01)

insert into tbl\_Project values('Project5',105,'Kayseri',04)

insert into tbl\_Project values('Project5',106,'Kayseri',04)

insert into tbl\_Works\_On values(1,101,'01:00:00')

insert into tbl\_Works\_On values(3,102,'02:00:00')

insert into tbl\_Works\_On values(2,105,'01:00:00')

insert into tbl\_Works\_On values(4,106,'01:00:00')

insert into tbl\_Works\_On values(5,104,'01:00:00')

insert into tbl\_Dependent values(1,'Mevlüt','Erkek','01.01.2005','Oğlu')

insert into tbl\_Dependent values(2,'Elif','Kadın','02.02.2006','Kızı')

insert into tbl\_Dependent values(3,'Burak','Erkek','03.03.2006','Oğlu')

insert into tbl\_Dependent values(3,'Özge','Kadın','04.04.2008','Kızı')

insert into tbl\_Dependent values(3,'Ali','Erkek','05.05.2004','Babası')

* Öncelikle ‘create’ komutu kullanılarak database oluşturuldu.
* Ardından tablolar ‘Primary Key’ ve ‘Foreign Key’ tanımlamalarıyla birlikte oluşturuldu .
* tbl\_Employee tablosundaki ‘Dno’ değişkeni ‘Foreign Key’ olarak tanımlanırken ‘Database Diagram’ kullanılarak tanımlandı . Çünkü tbl\_Employee ve tbl\_Department tabloları birbirlerine referans olduğundan ötürü ikisini birden sorgu ekranında tanımlama şansımız bulunmamaktadır.

**2.**

1. select E.Fname,E.Addres from tbl\_Employee E,tbl\_Department D where E.Dno=D.Dnumber and D.Dname='Arge'

* tbl\_Employee tablosundan ‘Arge’ departmanında çalışanların isimleri ve adresleri ‘select’ komutuyla getirildi.

1. create procedure DepartmentNumberControl (@Dnumber int)

as

select E.Lname,E.Addres,P.Pnumber,E.Bdate

from tbl\_Employee E,tbl\_Project P,tbl\_Works\_On W

where W.Pno=P.Pnumber and W.Essn=E.Ssn and P.Plocation='Eskişehir' and P.Dnum=@Dnumber and E.Ssn=E.Super\_ssn

go

exec DepartmentNumberControl @Dnumber=01

* ‘Department Number’ kontrolü yapılarak proje bölgesi ‘Eskişehir’ olan yöneticilerin soyisimleri , adresleri , doğum tarihleri ve çalıştıkları projenin numaraları tbl\_Employee ve tbl\_Project tablolarından ‘procedure’ kullanılarak getirildi .
* ‘procedure’ kullanılmasındaki amaç birden fazla sayıda şehir olmasından ötürü ‘DepartmentNumber’ın kontrolünü sağlamak ve farklı şehirlerdeki sonuçları görmekti .

1. select E.Fname

from tbl\_Employee E, tbl\_Dependent D

where E.Ssn=D.Essn and D.Dependent\_name=E.Fname

* Kendisiyle aynı isimli ‘Dependent’ı olan çalışanların ismi ‘select’ komutuyla getirildi.

1. select P.Pname,E.Fname,E.Dno

from tbl\_Employee E, tbl\_Works\_On W, tbl\_Project P

where W.Essn=E.Ssn and P.Pnumber=W.Pno order by E.Dno ,E.Lname

* Projelerde çalışanların isimleri , projelerin isimleri ve çalışanların departman numaraları ; çalışanların departman numarasına göre ve soy isimlerine göre sıralayarak ‘select’ ve ‘order by’ komutları kullanılarak getirildi.

1. select MAX(E.salary),MIN(E.salary),AVG(E.salary)

from tbl\_Department D, tbl\_Employee E

where D.Dnumber=E.Dno and D.Dname='Arge'

* ‘Arge’ departmanında çalışanlardan en yüksek maaşı , en düşük maaşı ve çalışanların maaşlarının ortalamaları ‘select’ komutuyla getirildi .

1. select P.Pname, W.Pno, COUNT (\*)

from tbl\_Employee E, tbl\_Project P, tbl\_Works\_On W

where W.Pno=P.Pnumber and W.Essn=E.Ssn

group by P.Pname,W.Pno

* Her bir proje için proje adı , proje numarası ve projedeki toplam çalışan sayısı ‘select’ ve ‘count’ komutları kullanılarak getirildi .
* ‘group by’ komutu ile proje ismi ile proje numarası gruplandırıldı ve ‘count’ komutu ile de kaç çalışan bulunduğu getirilmiş olundu .

1. create trigger InsertTrigger on tbl\_Department

after insert

as

declare @Dnumber int

select @Dnumber=D.Dnumber from tbl\_Department D

insert into tbl\_Dept\_Locations

values('temp',@Dnumber)

go

insert into tbl\_Department values('F', 06,NULL,'12.7.2012')

select\*from tbl\_Dept\_Locations

* tbl\_Department tablosuna veri eklendiğinde tbl\_Dept\_Locations tablosuna da veri eklenmesini tetikleyen ‘trigger’ ile tanımlanan bir değişken tbl\_Department tablosundan ‘Dnumber’ değişkenine eşitleniyor . Ardından tbl\_Dept\_Locations tablosuna da ‘insert komutuyla bu değişken üzerinden veri ekleniyor .
* Böylelikle tbl\_Department tablosuna bir veri eklendiğinde buradaki ‘Dnumber’ değişkeni otomatik olarak tbl\_Dept\_Locations tablosuna da ‘trigger’ yardımıyla eklenmiş oluyor .
* Ardından ‘select’ komutuyla eklenen verinin görülmesi sağlanıyor.

1. create trigger DeleteTrigger on tbl\_Employee

after delete

as

declare @Essn varchar(20)

select @Essn =E.Ssn from tbl\_Employee E

delete from tbl\_Dependent

where Essn=@Essn

go

* tbl\_Employee tablosundan bir veri silindiğinde tbl\_Dependent tablosundan da veri silinmesini tetikleyen ‘trigger’ ile tanımlanan bir değişken tbl\_Employee tablosundan E.Ssn değişkenine eşitleniyor . Ardından ‘delete’ komutuyla birlikte ‘where’ komutunda tbl\_Dependent tablosundaki ‘Essn’ değişkeninin tanımlanan değişkene eşit olup olmadığı kontrolü yapılarak veri siliniyor.
* Böylelikle tbl\_Employee tablosundan bir veri silindiğinde kontrol edilen değişkene göre tbl\_Dependent tablosundan da o kontrol edilen değişkenin bulunduğu veri silinmiş oluyor.

1. create table tbl\_Projet\_Log(

PLname varchar(20),

PLnumber int,

PLlocation varchar(20),

Dnum int,

);

create trigger UpdateTrigger on tbl\_project

after update

as

declare @PLname varchar(20)

declare @PLnumber int

declare @PLlocation varchar(20)

declare @Dnum int

select @PLname=P.Pname from tbl\_Project P

select @PLnumber=P.Pnumber from tbl\_Project P

select @PLlocation=P.Plocation from tbl\_Project P

select @Dnum=P.Dnum from tbl\_Project P

insert into tbl\_Projet\_Log

values(@PLname,@PLnumber,@PLlocation,@Dnum)

print'tbl\_Project is update'

go

* Öncelikle tbl\_Project ile aynı özelliklere sahip bir tbl\_Project\_Log tablosu oluşturuluyor . Burdaki amaç değiştirilen verilerin bir log dosyasında tutulmasını sağlamak.
* Ardından ‘update trigger’ ile tbl\_Project tablosundan bir veri update edildiğinde bu verinin tbl\_Project\_Log tablosuna kaydedilmesi ve ardından da bir tbl\_Project tablosundaki verinin update edildiğine dair bir uyarı mesajının verilmesi sağlanıyor .
* Önce oluşturulan yeni tablodaki özelliklerin tamamı için değişlenler tanımlanıyor .
* Ardından bu değişkenlerin hepsi tbl\_Project’teki değişkenlere eşitleniyor .
* Ardından bu değişkenlerle birlikte tbl\_Project\_Log tablosuna update edilmeden önceki veri ekleniyor ve ardından ‘print’ komutu ile de uyarı mesajı veriliyor .