Veri Tabanı Yönetimi ve Modellemesi

PROJE SUNUMU: 08253001 ALI VELI

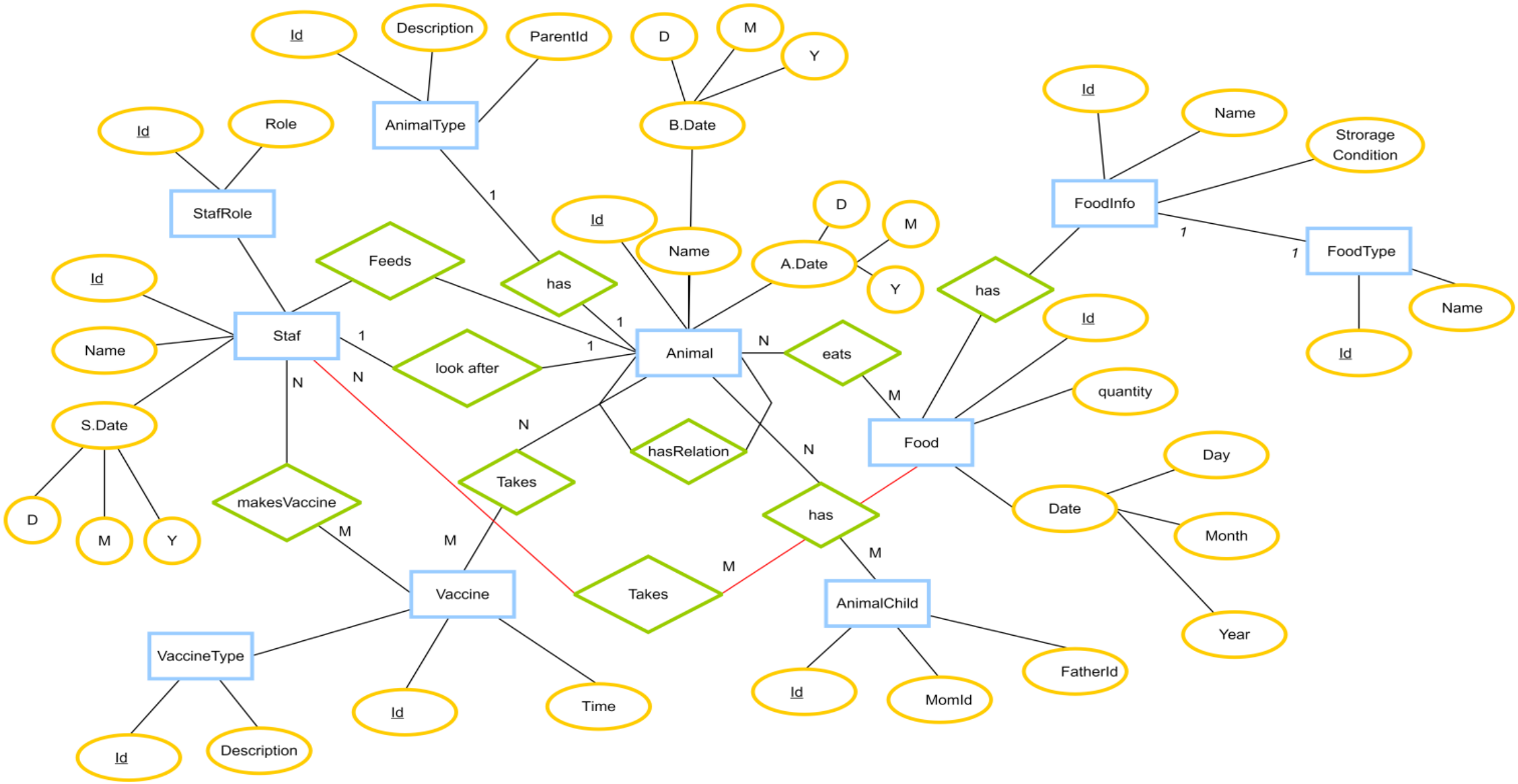
Problem: Hayvanat Bahçesi

Bir hayvanat bahçesi yetkilisi, bakımı altında olan hayvanların güncel bilgilerini takip edebileceği bir program istemektedir. Yetkilinin görmek istediği bilgiler aşağıdaki şekildedir:

- Her hayvana ait doğum tarihi, isimi, türü, eğer hayvanat bahçesinde doğmamışsa kendilerine geliş tarihi gibi bilgiler görüntülenmelidir.
- Her hayvanın dahil olduğu tür bilgisi gösterilebilir olmalıdır.
- Eğer hayvan hayvanat bahçesinde doğdu ise, ebeveyn bilgileri saklanmalıdır.
- Her hayvanın yiyebildiği yiyecek türleri bulunmaktadır.
- Yiyeceklerin tür bilgileri, porsiyon/kalori bilgileri tutulmalıdır. Yiyecekler adet, kg ya da litre bazında farklı şekillerde tutulabilir.
- Her hayvan için bir beslenme planı bulunmalıdır. Beslenme planının içinde öğün bazında tüketilmesi gereken besin ve miktarı bulunmalıdır.
- Her yiyecek için stok bilgileri tutulmalıdır. Ayrıca her yiyeceğin nereden, ne zaman ve ne kadar temin edildiği, fiyat bilgileri raporlanmak istenmektedir. Her hayvanın yiyecek işlemi ile ilgilenen bir personel bulunmaktadır. Bir personel bir ya da daha fazla personel ile ilgilenebilir.
- Hayvanlara dönemlik ya da yıllık olarak farklı aşılar uygulanmaktadır. Bu aşılar farklı zamanlarda ve farklı hayvan türlerine uygulanabilmektedir. Örneğin; Yeni doğan kuzu/oğlak için “Septisemi” aşısı, her 6 ayda bir “Şap hastalığı” aşısı, iç-dış parazit aşısı, çiçek hastalığı aşısı, vb. gibi. Her hayvan türüne uygulanacak aşı farklıdır ve hayvanın yaşına göre aşı dozları değişim gösterebilir.
- Hayvanların aşıları ile ilgilenen bir personel bulunmaktadır. Bu personel hayvanların, aşı bilgilerini görüntüleyebilmekte olup, gelecek ay aşı yapılacak hayvanları da tespit edebilir konumdadır. Görevli personel aşı stoklarını görüntüleyebilmektedir. Ayrıca ihtiyaç halinde aşı temini yapabilmekte, geçmişte hangi firmadan ne kadar aşı aldığını ücretlendirmesi gibi kısıtlarla geçmişe dönük raporlar alabilmektedir.

Problem: Hayvanat Bahçesi

- Hayvanların genel temizlikleri ile ilgilenen farklı personeller bulunmaktadır. Bu personeller sadece belirli hayvan gruplarının kafeslerini temizlemektedirler. Temizlik işlemi her kafes için 2 günde bir yapılmaktadır. Temizlik bilgileri de sistem üzerinde görüntülenebilir olmalıdır. Ayrıca o gün temizlik yapılması gereken kafeslerin listesini görebilmelidir.
- Kafeslerde bir ya da daha fazla hayvan yaşayabilmektedir. Ancak her kafesin bir kapasitesi vardır ve aynı kafeste sadece bir hayvan türü yaşayabilmektedir.
- Her bir yemek görevlisi kendi ilgilendiği hayvan grubunun o günkü öğün listesini ve yemeğin verilip verilmediğini görüntüleyebilmektedir. Ayrıca bu personeller hayvanlara ait geçmiş yemek bilgilerini görebilmektedir.
- Yemek ile ilgilenen personeller kalan yiyecek stoklarını görebilir, Hayvanlara ne kadar süre yeteceğini sistem üzerinden sorgulayabilir durumdadır.
- Hayvanat bahçesinde çalışan her bir kişinin farklı bir görevi vardır. Örneğin; Bir kişi sadece temizlik işi yaparken, bir başkası sadece yemek durumu ile ilgilenmektedir. Bir başka personel ise, hayvanların aşıları ile ilgilenmektedir.
- Hayvanat bahçesi yöneticisi, tüm personellere ait bilgileri görüntüleyebilmektedir.
- Personel bilgileri olarak, kişilerin adı - Soyadı bilgisi, kişinin doğum tarihi, işe başlama tarihi, maaşı, yıllık izin durumu, çalıştığı pozisyon gibi bilgiler tutulmaktadır.
- Hayvanat bahçesi yöneticisi, hayvanların yemek bilgilerini, hangi hayvan ile kimin ilgilendiğini, aşı bilgilerini, hayvanlara ait aşılama bilgilerini, gelecek hafta/ay tüketilecek yiyecek bilgilerini, gelecek hafta/ay kullanılacak olan aşı bilgilerini, kafeslerin temizlik durumlarını ve temizleyen kişileri görüntüleyebilmektedir


Id
Description
ParentId
AnimalType
D
M
Y
B.Date
Id
Role
StafRole
1
Id
Name
A.Date
D
M
Y
Feeds
has
Id
Name
Staf
1
1
look after
1
Animal
N
eats
M
Id
Name
Strorage Condition
FoodInfo
1
1
FoodType
Id
Name
has
Id
quantity
Food
N
N
N
S.Date
D
M
Y
makesVaccine
M
M
Takes
hasRelation
N
has
M
M
Takes
M
Vaccine
AnimalChild
Day
Date
Month
Year
VaccineType
Id
Description
Id
Time
Id
MomId
FatherId

Tablolar

- tbl_FoodType(id, description, parentId)
- tbl_AnimalType(id, info, ClassTypeId)
- tbl_VaccineType(id, description)
- tbl_StaffRole(id, definition)
- tbl_Animal(id, Name, arrivalDate, birthDate, animalType_Id)
- tbl_Cages(id, descpription, capacity)
- tbl_Parents(id, momId, fatherID)
- tbl_Staff(id, name, surname, birthDate, StartDate, staffRole, address, phone)
- tbl_VaccineDosage(id, vacinneTypeId, animalTypeId, time)
- tbl_Staff_Animal(id, staff_id, animal_id, start_date, finish_date)
- tbl_MealTime(id, info)
- tbl_Animal_Food_Schedule(id, animal_Id, food_id, quantity float, mealTime_Id, day)
- tbl_Food_Stock(food_id, stock float)

Karmaşık View: Bakıcıları, baktıkları hayvanları ve bu hayvanlara hangi tarih aralığında baktıklarını listeyen view.

```
CREATE VIEW view_List_Staff_Animal AS
SELECT
        S.id AS Staff_Id,
        S.name AS Staff_Name,
        A.id AS Animal_Id,
        A.Name AS Animal_Name,
        SA.start_date,
        SA.finish_date FROM tbl_Staff AS S
INNER JOIN tbl_Staff_Animal AS SA ON S.id=SA.staff_id
INNER JOIN tbl_Animal AS A ON A.id=SA.animal_id
```

Parametre Alan ve select yapan SP: Hayvanların beslenme programındaki eksik gün sayılarını listeleyen procedure

```
CREATE PROCEDURE sp_animal_food_Schedule_deficent
  @dayNumber   INT
AS    BEGIN
            DECLARE @idInfo AS INT
            DECLARE @name AS NVARCHAR(50)
            DECLARE @dayinfo AS INT
            DECLARE @tmp TABLE
            (           id INT,
                        name NVARCHAR(50),
                        deficentdayCount INT)
             DECLARE MealControl Scroll CURSOR FOR
            SELECT  A.id, A.Name,COUNT(DISTINCT AFS.day)
            FROM tbl_Animal A LEFT JOIN
            tbl_Animal_Food_Schedule AFS ON A.id=AFS.animal_Id
            GROUP BY A.id,A.Name
            OPEN MealControl
            FETCH MealControl INTO @idInfo,@name,@dayinfo
            WHILE (@@Fetch_Status<>-1)
            BEGIN
            IF(@dayinfo=@dayNumber)
                        INSERT INTO @tmp values(@idInfo,@name,0)
                        ELSE
                        INSERT INTO @tmp values
                        (@idInfo,@name,(@dayNumber-@dayinfo))
            FETCH MealControl INTO @idInfo,@name,@dayinfo
End
            CLOSE MealControl
            DEALLOCATE MealControl
            SELECT * FROM           @tmp
END
```

Parametre alan ve hesaplama yapıp sonucunu parametre olarak döndüren SP: Hayvanat bahçesindeki hayvan sayısını döndüren SP

```
CREATE PROCEDURE sp_animalCount
@sayi AS INT OUT
AS
BEGIN
        SELECT @sayi=COUNT(*) FROM tbl_Animal
END
```

Birden fazla tabloya insert işlemini yapan SP: Hayvanat bahçesine yeni gelen hayvanın kendi bilgilerini ve bakıcı bilgilerini ilgili tablolara ekleme işlemi yapan SP

```
CREATE PROC sp_Add_Animal_Staff
@name AS NVARCHAR(50) ,
@arrivalDate AS DATETIME,
@birthDate AS DATETIME,
@animalType_Id AS INT,
@staff_Id AS INT
AS
BEGIN
BEGIN TRANSACTION
                INSERT  INTO tbl_Animal VALUES

        (@name,@arrivalDate,@birthDate,@animalType_Id)
                INSERT INTO tbl_Staff_Animal VALUES
                (
                        @staff_Id,
                        (SELECT id from tbl_Animal
WHERE Name=@name
                                AND
animalType_Id=@animalType_Id),
                                GETDATE(),
                                NULL
                )
                IF (@@error <> 0)
                BEGIN
                        SELECT 'ROLLBACK'
                        ROLLBACK TRANSACTION
                END
        COMMIT TRANSACTION
END
```

Delete işlemini yapan SP: Hayvan tablosundan silme işlemi yapan SP

```
CREATE PROCEDURE sp_DELETE_Animal

@animalId AS INT

AS

BEGIN

        DELETE FROM tbl_Animal

                WHERE id=@animalId

END
```

Tablo döndüren UDF: Hayvan türüne göre hayvanların beslenme programı olup olmadığını liste şeklinde döndüren UDF

```
CREATE FUNCTION udf_Animal_Ver3
(
        @animalType AS INT
)
RETURNS @tmp TABLE
(
        name  NVARCHAR(50),
        ProgramBilgisi  NVARCHAR(50)
)
AS
BEGIN
   INSERT INTO @tmp
        SELECT DISTINCT A.name,
                CASE
                        WHEN AFS.id IS NULL THEN
                                'YOK'
                                        ELSE 'VAR'
                                END AS ProgramiBilgisi
                FROM tbl_Animal A LEFT JOIN
                tbl_Animal_Food_Schedule AFS ON A.id=AFS.animal_Id
                WHERE A.animalType_Id=@animalType
                RETURN
END
```

Scalar değer döndüren UDF: Toplam hayvan sayısını döndüren UDF

```
CREATE FUNCTION udf_Animal_Count()
RETURNS INT
AS
BEGIN
        DECLARE @num AS INT
        SELECT @num=COUNT(*) FROM tbl_Animal
RETURN @num
END
```

Trigger: Personel silinmeden önce, başka tablolarda ilgili veriler var mı kontrol edilen trigger

```
CREATE Trigger deleteStaffTrigger
 ON  tbl_Staff instead of delete
As
Begin
   If Exists
   (select id from tbl_staff_animal where staff_id  in (Select id from deleted) )
   BEGIN
      RAISERROR('Once diger tablodaki veriler silinmeli',16,1)
 ROLLBACK
   END
END
```

Job ve Users

- Otomatik Yedek Alan bir job tanımlanmıştır
- Yönetici
  - Tüm tablolar üzerinde insert,update, delete ve select yapabilir.
- Veteriner
  - Animal tablosu ve vaccine ile ilgili tablolar üzerinde insert,update, delete ve select yapabilir.
- Bakıcı
  - Animal tablosu ve food ile ilgili tablolar üzerinde insert,update, delete ve select yapabilir.(Bakıcı değişikliği tabloları üzerinde sadece yönetici işlem(insert, update, delete) yapabilir. Bakıcılar sadece select işlemi yapabilir)
- Temizlikçi
  - Animal Tablosu ve cages ile ilgili tablolar üzerinde işlem yapabilir.

Dikkat Edilmesi Gereken Noktalar

- Proje sunumlarınızı kendi bilgisayarınız üzerinde projenizi göstererek yapabilirsiniz.
- Projesini sunum şeklinde yapacak olanlar ise; Dersten önce sunumlarını EDS üzerine yükleyip, hoca bilgisayarı üzerinde sunum yapabilirler. Ancak USB ile sunum alınmayacaktır.
- Proje sunumu yapan herkes sistem üzerine gerekli dokümanları yüklemelidir.
- 05.12.2019 tarihinde sunum yapacak olan arkadaşlar Trigger, Store procedure ve UDF'ler için açıklama şeklinde düz metin olarak anlatabilirler.(Örneğin; ... Amacıyla ... Sp'si yapılacaktır yada .... Sorgusunu yapabilmek için UDF tanımlanacaktır gibi) Sonraki haftalarda sunum yapacak olan arkadaşların gerekli yapıları tamamlamış olması beklenir.