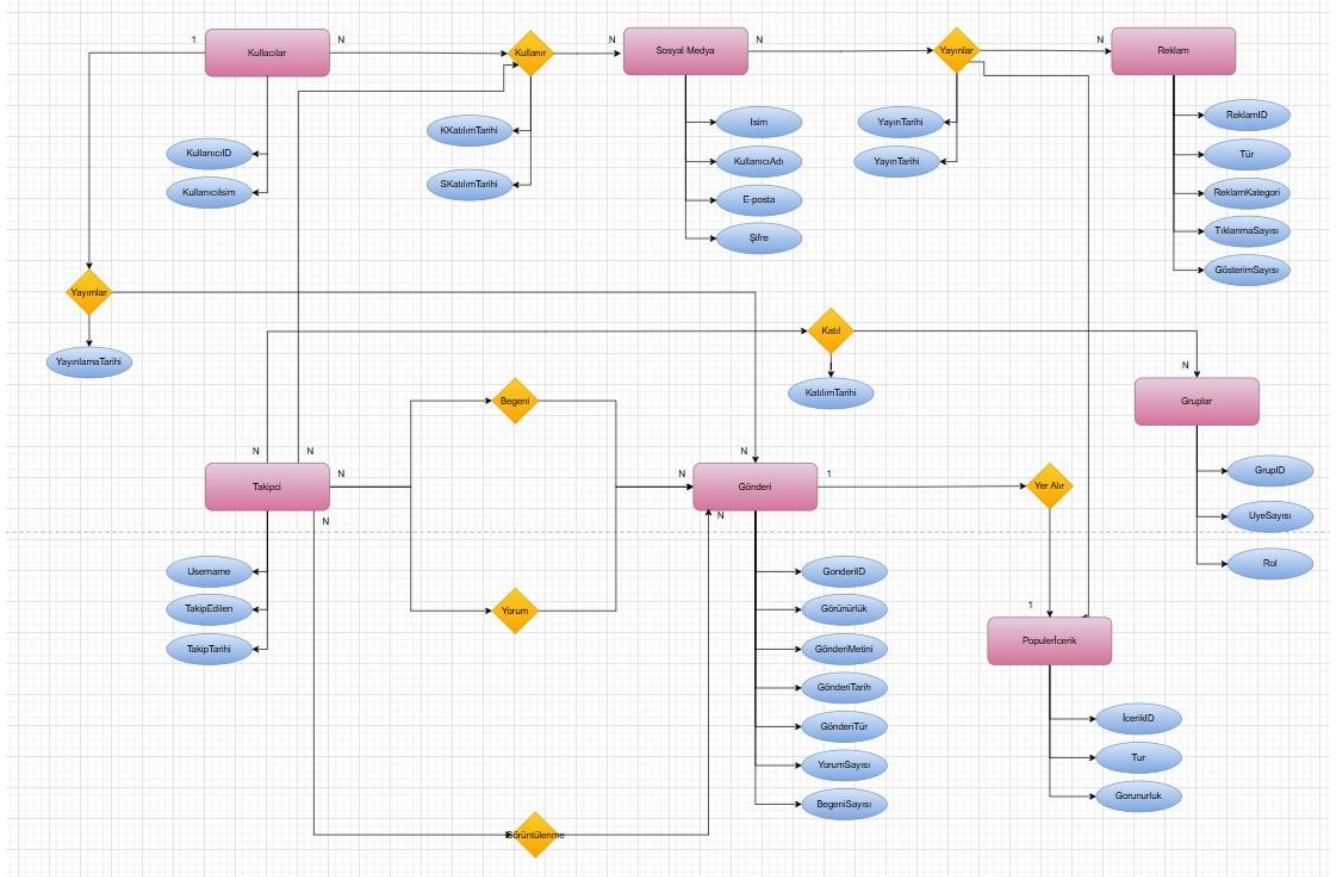


BMÜ329 Veri Tabanı Sistemleri Dersi Dönem Projesi

Gereksinimleri ve E-R Diyagramı Formatı

Proje Başlığı:	Takipçi Davranış Analiz ve Otomasyon Sistemi
----------------	--

GRUP ÜYELERİ		
Numarası	Adı	Soyadı
220260138	Ömer Faruk	Çelik
210260304	Resul	Kaçar
220260124	Muhammed Raşid	Yılan



(Github Adresinde Görsel Ayrıca Paylaşılmıştır.)

Tablolar ve İlişkiler

1. Kullanıcı - Sosyal Medya - Takipçi

İlişki Türü: N-N (Çoktan Çoğa)

Bu ilişkiyi temsil etmek için ara bir tablo kullanılmıştır.

- Kullanici(KullaniciID(PK), KullaniciIsim)
- SosyalMedya(Isim(PK), KullaniciAdi, Eposta, Sifre)
- Takipci(Username(PK), TakipEdilen, TakipTarihi)
- IKullanir(KullaniciAdi(FK), Isim(FK), Username(FK), KatilimTarihi)

2. Kullanıcı ve Gönderi

İlişki Türü: 1-N (Bire Çok)

Bir kullanıcı birden fazla gönderi yapabilir.

- Gonderi(GonderiID(PK), KullaniciID(FK), GonderiMetni, Gorunurluk, GonderiTarihi, GonderiTuru, YorumSayisi, BegeniSayisi)

3. Sosyal Medya ve Gönderi

İlişki Türü: 1-N (Bire Çok)

Bir sosyal medya platformunda birden fazla gönderi olabilir.

- Gonderi(GonderiID(PK), KullaniciID(FK), Isim(FK), GonderiMetni, Gorunurluk, GonderiTarihi, GonderiTuru, YorumSayisi, BegeniSayisi)

4. Gönderi ve Popüler İçerik

İlişki Türü: 1-1 (Bire Bir)

Bu ilişkide gönderi ve popüler içerik bire bir ilişkidir.

- PopulerIcerik(IcerikID(PK), Tur, Gorunurluk)
- IYerAlir(GonderiID(FK), IcerikID(FK), YerAlmaTarihi)

5. Takipçi ve Gönderi

İlişki Türü: N-N (Çoktan Çoğa)

Bir takipçi birden fazla gönderiyi beğenebilir ya da yorum yapabilir.

- **Takipci(Username(PK), TakipEdilen, TakipTarihi)**
- **IBegeni(Username(FK), GonderiID(FK), KullaniciID(FK), Isim(FK))**
- **IYorum(Username(FK), GonderiID(FK), KullaniciID(FK), Isim(FK), YorumMetni)**

6. Gruplar ve Takipçi

İlişki Türü: N-N (Çoktan Çoğa)

Bir kullanıcı birden fazla gruba katılabilir.

- **Gruplar(GrupID(PK), UyeSayisi, Rol)**
- **IKatil(Username(FK), GrupID(FK), KatilimTarihi)**

7. Sosyal Medya, Reklam ve Popüler İçerik

İlişki Türü: N-N (Çoktan Çoğa)

Sosyal medya platformlarında yayınlanan reklamlar ve popüler içerik ilişkisini ifade eder.

- **Reklam(ReklamID(PK), Tur, ReklamKategori, TiklanmaSayisi, GosterimSayisi)**
 - **PopulerIcerik(IcerikID(PK), Tur, Gorunurluk)**
 - **IRYayinlar(ReklamID(FK), IcerikID(FK), Isim(FK), RYayinTarihi, PYayinTarihi)**
-

Normalizasyon(3NF-BCNF Kontrolü Ve Dönüşümü)

=Normalizasyon =



1- Kullanıcı-Sosyal Medya Takipçi

$I_{Kullanır} (\underbrace{KullaniciAdi(Fk)}_R, \underbrace{Isim(Fk)}_A, \underbrace{Uyume(Fk)}_B, \underbrace{KullanimTarih(Fk)}_C, \underbrace{KullanimTarihi(Fk)}_D, \underbrace{KullanimTarihi(Fk)}_E)$

F:
 $ABC \rightarrow DE$
 $AB \rightarrow D$
 $BC \rightarrow E$

Fc:
 $ABC^+ = T\{A, B, C, D, E\}$ Anahtardır.
 $AB^+ = T\{A, B\} \times$
 $BC^+ = T\{B, C\} \times$

BCNF Mi Denetler.

k₁ R₁ R₁ = {R} R₁ = {A, B, C, D, E} 
k₂ R₂ = {A, B, D} R₁ = {A, B, C, E} 
k₃ R₃ = {B, C, E} R₁ = {A, B, C}

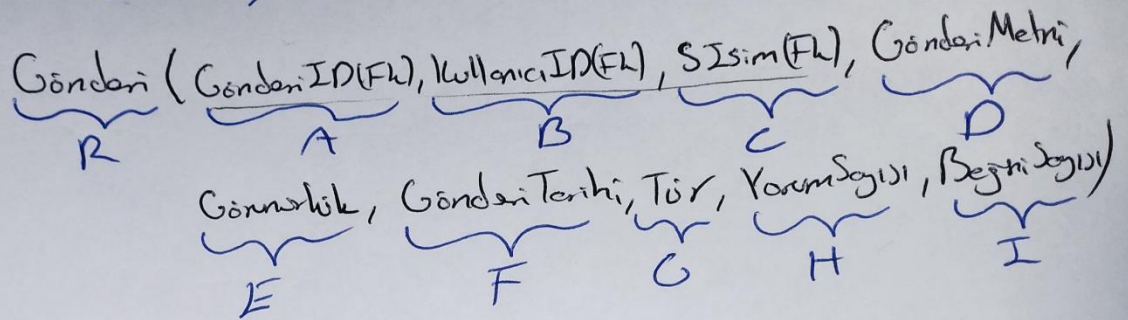
R₁ : Sağ tarafta değer kalmadığı için silinir.

R₂ (A, B, D) , R₃ (B, C, E)

BCNF Formu:

$AB \rightarrow D$
 $BC \rightarrow E$ bulunur.

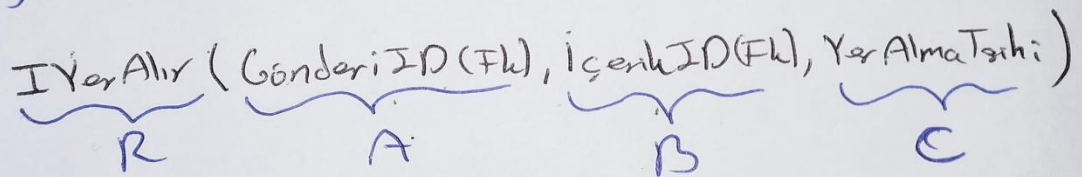
2-) Gönderi → kullanıcı :



F: $ABC \rightarrow DEFGHI$

Fc: $ABC^+ = T\{A, B, C, D, E, F, G, H, I\}$ Anahtar
 BCNF formuna uygundur.

3-) Gönderi - Popüler içerik:



F: $AB \rightarrow C$
 $B \rightarrow C$

Fc: $AB^+ = T\{A, B, C\}$ Anahtar
 $B^+ = T\{B, C\}$

$\left. \begin{array}{ll} K_1 & R_1 = \{R\} \quad R_1 = \{A, B, C\} \\ K_2 & R_2 = \{B, C\} \quad R_1 = \{A, B\} \end{array} \right\}$

BCNF Formu:

$AB \rightarrow C$
 $B \rightarrow C$

3-) Σ Yorum, Σ Begeri:

Σ Korum ($\underbrace{\text{Username (FK)}}_R, \underbrace{\text{Gönderi ID (FK)}}_A, \underbrace{\text{Kullanıcı ID (FK)}}_B, \underbrace{\text{İsim (FK)}}_D, \underbrace{\text{Yorum Metni}}_E$)

F: $ABCD \rightarrow E$

Fc: $ABCD^+ = T\{A, B, C, D, E\}$ Anahtardır
BCNF Formuna uygundur.

Σ Begeri ($\underbrace{\text{Username (FK)}}_R, \underbrace{\text{Gönderi ID (FK)}}_A, \underbrace{\text{Kullanıcı ID (FK)}}_B$)

ABC Primary Key'dir ✓

4-) Gruplar ve Takipçi

Σ Katıl ($\underbrace{\text{Username (FK)}}_R, \underbrace{\text{Group ID (FK)}}_A, \underbrace{\text{Katılım Tarihi}}_C$)

F: $AB \rightarrow C$

Fc: $AB^+ = T\{A, B, C\}$ Anahtardır

BCNF Formuna Uygundur.

5-) Sosyal Medya - Reklam - Popüler İsimler

IR Yayıncılar (Reklam ID (FK), Pazar ID (FK), İsim (FK),
 R Yayıncı Tarihi, P Yayıncı Tarihi)

R
A
B
C
D
E

F:

$ABC \rightarrow DE$
 $AC \rightarrow D$
 $B \rightarrow E$

Fc:

$ABCT = T\{A, B, C, D, E\}$ Anahterdir.
 $ACT = T\{A, C, D\}$
 $B \rightarrow E = T\{B, E\}$

BCNF Algoritması:

k₁ R₁ = {R} R₁ = {A, B, C, D, E}

k₂ R₂ = {A, C, D} R₁ = {A, B, C, E}

k₃ R₃ = {B, E} R₁ = {A, B, C}

BCNF Formu:

$B \rightarrow E$
 $AC \rightarrow D$

BCNF ayrıştırma ile yeniden şekillenen veritabanı tabloları 13 tablo halinde normalize edilmiştir. Yeni Biçimini alan bu tablolarda implementation aşamaları aşağıdaki gibi verilmiştir:

1-) BCNF formatına uygun getirilen tablolar için veri tabanı oluşturuluması.

2-) Oluşturulacak tabloların PK'ları belirlenip gerekli referansların sağlanması.

3-) Bütün Tablolar tam ve eksiksiz oluşturulduktan sonra ana tablolara yapay veri eklenmesi.

4-) Tabloların ilişki durumuna göre birbirini etkileyen ara tablolalar için Triger, Join işlemleri yapıp değerlerin otomatik güncelleme işleminin yapılması.

5-) Tablolar arası ilişkilerin Triger kullanarak birleştirilmesinden sonra veri eklemesi ve sorgusun yapıp sistemin test edilmesi,

Yeni Biçim

= Yeni Biçim =

1- Kullanıcı - Sorgu Medya - Takipçi

I Kullanıcı (Kullanıcı Adı (FK), Şifre (FK), Kullanıcı Adı Tarih, Takipçi (FK), Kullanıcı Tarih)
E

BCNF:

$AB \rightarrow D$

$BC \rightarrow E$

$R_1(A, B, D)$ $R_2(B, C, E)$

$R_1 = \text{Kullanıcı Kullanır}$

$R_2 = \text{Takipçi Kullanır}$

I Kullanıcı Kullanır (A, B, D) Kullanıcı - Sorgu Medya
I Takipçi Kullanır (B, C, E) Takipçi - Sorgu Medya

2- Kullanıcı - Gönderi

$ABC \rightarrow DEFGHI$ olduğu için olduğu gibi

kalır

Gönderi (Gönderi ID (FK), Kullanıcı ID (FK), Şifre (FK), Gönderi Metni, Gönderi Tarih, Tutar, Yorum Sayısı, Beğeni Sayısı)

3-) Gonderi - Populer isik

Σ Yazar Alır ($\underbrace{\text{Gonderi ID (FK)}}_A, \underbrace{\text{isik ID (FK)}}_B, \underbrace{\text{Yazar Alma Tarihi}}_C$)
R

BCNF Formu →

$AB \rightarrow C$ olarak bulunmuştur yine

Σ Yazar Alır ($\text{Gonderi ID (FK)}, \text{isik ID (FK)}, \text{Yazar Alma Tarihi}$)

4-) Σ Yorum, Σ Beğeni BCNF uygun olarak bulunuldu

Σ Yorum ($\text{Username (FK)}, \text{Gonderi ID (FK)}, \text{Kullanıcı ID (FK)}, \text{Sıxım (FK)}, \text{Yorum Metni}$)

Σ Beğeni ($\text{Username (FK)}, \text{Gonderi ID (FK)}, \text{Kullanıcı ID (FK)}$)

5-) Gruplar ve Takipçi

$AB \rightarrow C$ olarak BCNF formuna uygun bulunmuştur.

Σ Katılı ($\text{Username (FK)}, \text{Group ID (FK)}, \text{Katılım Tarihi}$)

6-) Sosyal Medya - Reklam - Popüler İçerik

$R = \{ \text{IRYayinlar} (\text{ReklamID(FK)}, \text{PicarID(FK)}, \text{Sisim(FK)}, \text{RYayinTarihi}, \text{PYayinTarihi}) \}$
 $E = \{ \text{E} \}$

$CB \rightarrow E$ $R_1 (A, C, D)$ olarak BCNF uyarı
 $AC \rightarrow D$ $R_2 (B, E, C)$ oluşturulmuştur.

$R_1 = \{ \text{IRYayinlar} (\text{ReklamID(FK)}, \text{Sisim(FK)}, \text{RYayinTarihi}) \}$

$R_2 = \{ \text{IPYayinlar} (\text{PicarID(FK)}, \text{Sisim(FK)}, \text{PYayinTarihi}) \}$

= Tabloların En Son Halleri =

Gönderiler (GönderiID(FK), GönderiID(FK))

Kullanıcı (KullanıcıID(PK), Kullanıcıİsim)

Sosyal Medya (Sisim(PK), KullanıcıAdı, E-Posta, Şifre)

Takipçi (Username(PK), TakipEdilen, TakipTarihi)

IRKullanıcıKullanır (KullanıcıID(FK), Sisim(FK), KullanıcıKatılımTarihi)

ITakipçiKullanır (Username(FK), Sisim(FK), TakipçiKatılımTarihi)

Gönderi (GönderiID(FK), KullanıcıID(FK), Sisim(FK), GönderiMetni,

Görünürlük, GönderiTarihi, Tür, YorumSayısı, BeğeniSayısı)

IVarAlır (GönderiID(FK), PicarID(FK), VarAlmaTarihi)

IVorum (Username(FK), GönderiID(FK), KullanıcıID(FK), Sisim(FK), YorumMetni,

IBegeni (Username(FK), GönderiID(FK), KullanıcıID(FK))

IKatıl (Username(FK), GroupID(FK), KatılımTarihi)

Popülerİçerik (PicarID(PK), Tür, Görünürlük)

Gruplar (GroupID(PK), ÜyeSayısı, GrupAdı)

MSSQL Kodları TABLE OLUŞTURMA – IMPLEMENTATION

```
use TestDB;
-- Veritabanını Sil (Eğer Varsa)
IF DB_ID('TDAS_DB') IS NOT NULL
BEGIN
    ALTER DATABASE TDAS_DB SET SINGLE_USER WITH ROLLBACK IMMEDIATE;
    DROP DATABASE TDAS_DB; -- Eğer veritabanı varsa, önce tek kullanıcı moduna alır ve
ardından siler
END;
GO

-- Yeni Veritabanını Oluştur
CREATE DATABASE TDAS_DB; -- TDAS_DB adında yeni bir veritabanı oluşturulur
GO

USE TDAS_DB; -- Yeni oluşturulan veritabanı seçilir
GO

CREATE SCHEMA sc; -- "sc" adında bir şema oluşturulur, bu şema altında tüm tablolar yer alacak
GO

-- Kullanici Tablosu: Kullanıcı bilgilerini tutar
CREATE TABLE sc.Kullanici (
    KullanicID INT IDENTITY(1,1) PRIMARY KEY, -- Kullanıcı ID'si otomatik olarak 1'den başlar
ve her yeni satırda 1 artar
    KullanicIsm NVARCHAR(100)
);

-- Sosyal Medya Tablosu: Sosyal medya platformları hakkında bilgileri tutar
CREATE TABLE sc.SosyalMedya (
    SosyalMedyaAdi NVARCHAR(100), -- Sosyal medya platformunun adı (örn. Facebook, Twitter)
    KullanicAdi NVARCHAR(100), -- Kullanıcının sosyal medya platformundaki kullanıcı adı
    PRIMARY KEY(SosyalMedyaAdi,KullanicAdi),
    EPosta NVARCHAR(100), -- Kullanıcının e-posta adresi
    Sifre NVARCHAR(100) -- Kullanıcının sosyal medya şifresi
);

-- Takipci Tablosu: Takipçiler ve hangi sosyal medyayı takip ettikleri bilgilerini tutar
CREATE TABLE sc.Takipci (
    TakipciAdi NVARCHAR(100) PRIMARY KEY, -- Takipçinin adı
    TakipEdilen NVARCHAR(100), -- Takip edilen sosyal medya platformunun adı
    TakipTarihi DATE -- Takip etme tarihi
);

-- PopulerIcerik Tablosu: Popüler içeriklerin bilgilerini tutar (görüntülenme sayısı ile)
CREATE TABLE sc.PopulerIcerik (
    PopulerIcerikID INT IDENTITY(1,1) PRIMARY KEY, -- Popüler içerik ID'si
    IcerikTuru NVARCHAR(50), -- İçerik türü (video, resim, yazı vb.)
    GoruntulenmeSayisi INT -- İçeriğin görüntülenme sayısı
);

-- Gruplar Tablosu: Sosyal medya platformlarındaki grupları ve üyelik bilgilerini tutar
CREATE TABLE sc.Gruplar(
    GrupID INT IDENTITY(1,1) PRIMARY KEY, -- Grup ID'si
```



```
UyeSayisi INT, -- Gruba ait üye sayısı
Rol NVARCHAR(100) -- Grubun rolü (admin, üye, vs.)
);
```

-- KullaniciKullanir Tablosu: Hangi kullanıcıların hangi sosyal medya platformlarında hesapları olduğunu belirtir

```
CREATE TABLE sc.KullaniciKullanir (
    KullanicID INT, -- Kullanıcı ID'si
    KullanicAdi NVARCHAR(100),
    SosyalMedyaAdi NVARCHAR(100), -- Sosyal medya platformunun adı
    KatilimTarihi DATE, -- Katılım tarihi
    PRIMARY KEY (KullanicID, SosyalMedyaAdi, KullanicAdi), -- Kullanıcı ve sosyal medya
platformu kombinasyonu benzersizdir
    FOREIGN KEY (KullanicID) REFERENCES sc.Kullanici(KullanicID), -- Kullanıcı tablosu ile
ilişki
    FOREIGN KEY (SosyalMedyaAdi, KullanicAdi) REFERENCES
sc.SosyalMedya(SosyalMedyaAdi, KullanicAdi) -- Sosyal medya tablosu ile ilişki
);
```

-- TakipciKullanir Tablosu: Takipçilerin hangi sosyal medya platformlarında hesapları olduğunu belirtir

```
CREATE TABLE sc.TakipciKullanir (
    KullanicAdi NVARCHAR(100),
    TakipciAdi NVARCHAR(100), -- Takipçi adı
    SosyalMedyaAdi NVARCHAR(100), -- Sosyal medya platformu
    KatilimTarihi DATE, -- Katılım tarihi
    PRIMARY KEY (TakipciAdi, SosyalMedyaAdi, KullanicAdi), -- Takipçi ve sosyal medya
platformu kombinasyonu benzersizdir
    FOREIGN KEY (TakipciAdi) REFERENCES sc.Takipci(TakipciAdi), -- Takipçi tablosu ile ilişki
    FOREIGN KEY (SosyalMedyaAdi, KullanicAdi) REFERENCES
sc.SosyalMedya(SosyalMedyaAdi, KullanicAdi) -- Sosyal medya tablosu ile ilişki
);
```

-- Gonderi Tablosu: Kullanıcıların sosyal medya platformlarında yaptığı paylaşımları tutar

```
CREATE TABLE sc.Gonderi (
    GonderiID INT IDENTITY(1,1), -- Gönderi ID'si
    KullanicAdi NVARCHAR(100),
    KullanicID INT, -- Gönderiyi paylaşan kullanıcı ID'si
    SosyalMedyaAdi NVARCHAR(100), -- Sosyal medya platformu
    GonderiMetni NVARCHAR(MAX), -- Gönderinin metni (yazılı içerik)
    GonderiTarihi DATE, -- Gönderi tarihi
    GonderiTuru NVARCHAR(50), -- Gönderi türü (görsel, yazılı içerik vb.)
    YorumSayisi INT, -- Gönderiye yapılan yorum sayısı
    BegeciSayisi INT, -- Gönderiye yapılan beğeni sayısı
    Gorunurluk INT, -- Gönderinin gizlilik durumu (örneğin, herkese açık veya yalnızca takipçiler için)
    PRIMARY KEY (GonderiID, KullanicID, SosyalMedyaAdi, KullanicAdi), -- Gönderi, kullanıcı ve
sosyal medya platformu kombinasyonu benzersizdir
    FOREIGN KEY (KullanicID) REFERENCES sc.Kullanici(KullanicID), -- Kullanıcı tablosu ile
ilişki
    FOREIGN KEY (SosyalMedyaAdi, KullanicAdi) REFERENCES
sc.SosyalMedya(SosyalMedyaAdi, KullanicAdi) -- Sosyal medya tablosu ile ilişki
);
```

-- Goruntulenme Tablosu: Takipçilerin gönderileri nasıl görüntülediği bilgisini tutar

```
CREATE TABLE sc.Goruntulenme (
```



```

TakipciAdi NVARCHAR(100), -- Takipçi adı
GonderiID INT, -- Gönderi ID'si
KullaniciID INT, -- Gönderiyi paylaşan kullanıcı ID'si
SosyalMedyaAdi NVARCHAR(100), -- Sosyal medya platformu
KullaniciAdi NVARCHAR(100),
PRIMARY KEY (TakipciAdi, GonderiID, KullaniciAdi), -- Takipçi ve gönderi kombinasyonu
benzersizdir
FOREIGN KEY (TakipciAdi) REFERENCES sc.Takipci(TakipciAdi), -- Takipçi tablosu ile ilişki
FOREIGN KEY (GonderiID, KullaniciID, SosyalMedyaAdi, KullaniciAdi) REFERENCES
sc.Gonderi(GonderiID, KullaniciID, SosyalMedyaAdi, KullaniciAdi) -- Gönderi tablosu ile ilişki
);

-- IcerikYerAlir Tablosu: Popüler içeriklerin gönderilerdeki yer alımlarını tutar
CREATE TABLE sc.IcerikYerAlir (
GonderiID INT, -- Gönderi ID'si
PopulerIcerikID INT, -- Popüler içerik ID'si
KullaniciID INT, -- Gönderiyi paylaşan kullanıcı ID'si
SosyalMedyaAdi NVARCHAR(100), -- Sosyal medya platformu
KullaniciAdi NVARCHAR(100),
YerAlmaTarihi DATE, -- Yer alma tarihi
PRIMARY KEY (GonderiID, PopulerIcerikID), -- Gönderi ve popüler içerik kombinasyonu
benzersizdir
FOREIGN KEY (PopulerIcerikID) REFERENCES sc.PopulerIcerik(PopulerIcerikID), -- Popüler
içerik tablosu ile ilişki
FOREIGN KEY (GonderiID, KullaniciID, SosyalMedyaAdi, KullaniciAdi) REFERENCES
sc.Gonderi(GonderiID, KullaniciID, SosyalMedyaAdi, KullaniciAdi) -- Gönderi tablosu ile ilişki
);

-- Yorum Tablosu: Takipçilerin gönderiler altına yazdığı yorumları tutar
CREATE TABLE sc.Yorum (
TakipciAdi NVARCHAR(100), -- Takipçi adı
GonderiID INT, -- Gönderi ID'si
KullaniciID INT, -- Gönderiyi paylaşan kullanıcı ID'si
SosyalMedyaAdi NVARCHAR(100), -- Sosyal medya platformu
KullaniciAdi NVARCHAR(100),
YorumMetni NVARCHAR(100), -- Yorum metni
PRIMARY KEY (TakipciAdi, GonderiID, KullaniciID,
SosyalMedyaAdi, KullaniciAdi, YorumMetni), -- Takipçi, gönderi, kullanıcı ve sosyal medya platformu
kombinasyonu benzersizdir
FOREIGN KEY (TakipciAdi) REFERENCES sc.Takipci(TakipciAdi), -- Takipçi tablosu ile ilişki
FOREIGN KEY (KullaniciID) REFERENCES sc.Kullanici(KullaniciID), -- Kullanıcı tablosu ile
ilişki
FOREIGN KEY (GonderiID, KullaniciID, SosyalMedyaAdi, KullaniciAdi) REFERENCES
sc.Gonderi(GonderiID, KullaniciID, SosyalMedyaAdi, KullaniciAdi), -- Gönderi tablosu ile ilişki
FOREIGN KEY (SosyalMedyaAdi, KullaniciAdi) REFERENCES
sc.SosyalMedya(SosyalMedyaAdi, KullaniciAdi) -- Sosyal medya tablosu ile ilişki
);

-- Begeni Tablosu: Takipçilerin gönderilere yaptığı beğenileri tutar
CREATE TABLE sc.Begeni (
TakipciAdi NVARCHAR(100), -- Takipçi adı
GonderiID INT, -- Gönderi ID'si
KullaniciID INT, -- Gönderiyi paylaşan kullanıcı ID'si
SosyalMedyaAdi NVARCHAR(100), -- Sosyal medya platformu
KullaniciAdi NVARCHAR(100),

```

PRIMARY KEY (TakipciAdi, GonderiID, KullaniciID, SosyalMedyaAdi,KullaniciAdi), -- Takipçi, gönderi, kullanıcı ve sosyal medya platformu kombinasyonu benzersizdir

FOREIGN KEY (TakipciAdi) **REFERENCES** sc.Takipci(TakipciAdi), -- Takipçi tablosu ile ilişki

FOREIGN KEY (KullaniciID) **REFERENCES** sc.Kullanici(KullaniciID), -- Kullanıcı tablosu ile ilişki

FOREIGN KEY (GonderiID, KullaniciID, SosyalMedyaAdi,KullaniciAdi) **REFERENCES** sc.Gonderi(GonderiID, KullaniciID, SosyalMedyaAdi,KullaniciAdi) -- Gönderi tablosu ile ilişki
);

-- GrupKatilim Tablosu: Takipçilerin gruplara katılım bilgilerini tutar

CREATE TABLE sc.GrupKatilim (
TakipciAdi **NVARCHAR**(100), -- Takipçi adı
GrupID **INT IDENTITY**(1,1), -- Grup ID'si
KatilimTarihi **DATE**, -- Katılım tarihi
PRIMARY KEY (TakipciAdi, GrupID), -- Takipçi ve grup kombinasyonu benzersizdir
FOREIGN KEY (GrupID) **REFERENCES** sc.Gruplar(GrupID), -- Grup tablosu ile ilişki
FOREIGN KEY (TakipciAdi) **REFERENCES** sc.Takipci(TakipciAdi) -- Takipçi tablosu ile ilişki
);

MSSQL TRIGGER JOIN

-- Loglama Tablosu Oluşturma

CREATE TABLE sc.LogTable (
LogID **INT IDENTITY**(1,1) **PRIMARY KEY**,
EventTime **DATETIME** **DEFAULT** GETDATE(),
TableName **NVARCHAR**(100),
Action **NVARCHAR**(50),
Details **NVARCHAR**(MAX)
);
GO

-- Kullanici Tablosu İçin Tetikleyici

CREATE TRIGGER tr_AfterInsert_Kullanici
ON sc.Kullanici
AFTER INSERT
AS
BEGIN
PRINT 'Yeni bir kullanıcı eklendi!';
INSERT INTO sc.LogTable (TableName, Action, Details)
VALUES ('Kullanici', 'INSERT', 'Yeni bir kullanıcı eklendi.');

END;
GO

-- SosyalMedya Tablosu İçin Tetikleyici

CREATE TRIGGER tr_AfterUpdate_SosyalMedya
ON sc.SosyalMedya
AFTER UPDATE
AS
BEGIN
PRINT 'Bir sosyal medya hesabı güncellendi!';
INSERT INTO sc.LogTable (TableName, Action, Details)

```
VALUES ('SosyalMedya', 'UPDATE', 'Bir sosyal medya hesabı güncellendi.');
```

END;

GO

-- Takipci Tablosu İçin Tetikleyici

```
CREATE TRIGGER tr_AfterDelete_Takipci
ON sc.Takipci
AFTER DELETE
AS
BEGIN
    PRINT 'Bir takipçi silindi!';
    INSERT INTO sc.LogTable (TableName, Action, Details)
    VALUES ('Takipci', 'DELETE', 'Bir takipçi silindi.');
```

END;

GO

-- Gonderi Tablosu İçin Tetikleyici

```
CREATE TRIGGER tr_AfterInsert_Gonderi
ON sc.Gonderi
AFTER INSERT
AS
BEGIN
    PRINT 'Yeni bir gönderi eklendi!';
    INSERT INTO sc.LogTable (TableName, Action, Details)
    VALUES ('Gonderi', 'INSERT', 'Yeni bir gönderi eklendi.');
```

END;

GO

-- Beğeni Tablosu İçin Tetikleyici

```
CREATE TRIGGER tr_AfterInsert_Begeni
ON sc.Begeni
AFTER INSERT
AS
BEGIN
    UPDATE sc.Gonderi
    SET BeğeniSayisi = BeğeniSayisi + 1
    FROM sc.Gonderi g
    INNER JOIN Inserted i ON g.GonderiID = i.GonderiID
    WHERE g.KullaniciID = i.KullaniciID;

    PRINT 'Bir gönderiye beğeni eklendi!';
    INSERT INTO sc.LogTable (TableName, Action, Details)
    VALUES ('Beğeni', 'INSERT', 'Bir gönderiye beğeni eklendi.');
```

END;

GO

-- Yorum Tablosu İçin Tetikleyici

CREATE TRIGGER tr_AfterDelete_Yorum

ON sc.Yorum

AFTER DELETE

AS

BEGIN

UPDATE sc.Gonderi

SET YorumSayisi = YorumSayisi - 1

FROM sc.Gonderi g

INNER JOIN Deleted d ON g.GonderiID = d.GonderiID

WHERE g.KullaniciID = d.KullaniciID;

PRINT 'Bir yorum silindi!';

INSERT INTO sc.LogTable (TableName, Action, Details)

VALUES ('Yorum', 'DELETE', 'Bir yorum silindi.');

END;

GO

-- Yorum Tablosuna Veri Eklendiğinde YorumSayisi'ni Arttıran Tetikleyici

CREATE TRIGGER tr_AfterInsert_Yorum

ON sc.Yorum

AFTER INSERT

AS

BEGIN

-- Gonderi tablosundaki YorumSayisi'ni arttırma

UPDATE sc.Gonderi

SET YorumSayisi = YorumSayisi + 1

FROM sc.Gonderi g

INNER JOIN Inserted i ON g.GonderiID = i.GonderiID

WHERE g.KullaniciID = i.KullaniciID;

PRINT 'Bir yorum eklendi!';

INSERT INTO sc.LogTable (TableName, Action, Details)

VALUES ('Yorum', 'INSERT', 'Bir yorum eklendi.');

END;

GO

-- TakipciKullanir Tablosu İçin Tetikleyici

CREATE TRIGGER tr_AfterDelete_TakipciKullanir

ON sc.TakipciKullanir

AFTER DELETE

AS

BEGIN

PRINT 'Bir takipçi bir sosyal medya platformundan ayrıldı!';

INSERT INTO sc.LogTable (TableName, Action, Details)

VALUES ('TakipciKullanir', 'DELETE', 'Bir takipçi bir sosyal medya platformundan ayrıldı.');

END;

GO

MSSQL VERİTABANINA ÖRNEK VERİLER EKLEME

-- Kullanici tablosuna veri ekleme

```
INSERT INTO sc.Kullanici (KullaniciIsim) VALUES  
( 'Ahmet Yılmaz' ),  
( 'Ayşe Kaya' ),  
( 'Mehmet Demir' ),  
( 'Zeynep Şahin' ),  
( 'Emir Can' );
```

-- SosyalMedya tablosuna veri ekleme

```
INSERT INTO sc.SosyalMedya (SosyalMedyaAdi, KullaniciAdi, EPosta, Sifre) VALUES  
( 'Instagram', 'ahmet_yilmaz', 'ahmet@email.com', '123456' ),  
( 'Twitter', 'ayse_kaya', 'ayse@email.com', 'abcdef' ),  
( 'Facebook', 'mehmet_demir', 'mehmet@email.com', 'qwerty' ),  
( 'LinkedIn', 'zeynep_sahin', 'zeynep@email.com', '7890ab' ),  
( 'TikTok', 'emir_can', 'emir@email.com', 'zxcvbn' );
```

-- Takipci tablosuna veri ekleme

```
INSERT INTO sc.Takipci (TakipciAdi, TakipEdilen, TakipTarihi) VALUES  
( 'Ali Veli', 'Instagram', '2023-01-15' ),  
( 'Fatma Gül', 'Twitter', '2023-02-20' ),  
( 'Mustafa Can', 'Facebook', '2023-03-25' ),  
( 'Elif Nur', 'LinkedIn', '2023-04-01' ),  
( 'Burak Öz', 'TikTok', '2023-05-10' );
```

-- PopulerIcerik tablosuna veri ekleme

```
INSERT INTO sc.PopulerIcerik (IcerikTuru, GoruntulenmeSayisi) VALUES  
( 'Video', 5000 ),  
( 'Resim', 10000 ),  
( 'Yazı', 2000 ),  
( 'Video', 7500 ),  
( 'Resim', 12000 );
```

-- Gruplar tablosuna veri ekleme

```
INSERT INTO sc.Gruplar (UyeSayisi, Rol) VALUES  
( 150, 'Admin' ),  
( 200, 'Üye' ),  
( 100, 'Üye' ),  
( 300, 'Admin' ),  
( 50, 'Üye' );
```


BikeStore_SQL-Server-Sample-Database > SQLQuery_2.sql

Run Cancel Disconnect Change Database: TDAS_DB Estimated Plan Enable Actual Plan Parse Enable SQLCMD To Note

```
1 use TDAS_DB;
2 go
3
4 SELECT * FROM sc.SosyalMedya;
```

Results Messages

	SosyalMedyaAdi	KullaniciAdi	EPosta	Sifre
17	Dribbble	kullanici22	email22@example.com	sifre22
18	Dribbble2	kullanici47	email47@example.com	sifre47
19	Dribbble3	kullanici72	email72@example.com	sifre72
20	Dribbble4	kullanici97	email97@example.com	sifre97
21	Facebook	kullanici3	email3@example.com	sifre3
22	Facebook2	kullanici28	email28@example.com	sifre28
23	Facebook3	kullanici53	email53@example.com	sifre53
24	Facebook4	kullanici78	email78@example.com	sifre78
25	Flickr	kullanici20	email20@example.com	sifre20
26	Flickr2	kullanici45	email45@example.com	sifre45
27	Flickr3	kullanici70	email70@example.com	sifre70
28	Flickr4	kullanici95	email95@example.com	sifre95
29	Instagram	kullanici1	email1@example.com	sifre1
30	Instagram2	kullanici26	email26@example.com	sifre26
31	Instagram3	kullanici51	email51@example.com	sifre51
32	Instagram4	kullanici76	email76@example.com	sifre76
33	LinkedIn	kullanici4	email4@example.com	sifre4
34	LinkedIn2	kullanici29	email29@example.com	sifre29
35	LinkedIn3	kullanici54	email54@example.com	sifre54
36	LinkedIn4	kullanici79	email79@example.com	sifre79
37	Medium	kullanici10	email10@example.com	sifre10

TESTLER

```
INSERT INTO sc.SosyalMedya(SosyalMedyaAdi,KullaniciAdi)
VALUES('insta','MRY');
INSERT INTO sc.Kullanici(KullaniciIsim)
VALUES('mmüslüm baba')
```

```
INSERT INTO sc.SosyalMedya(SosyalMedyaAdi,KullaniciAdi)
VALUES('insta','müslüm baba');
```

```
INSERT INTO sc.Takipci(TakipciAdi,TakipEdilen,TakipTarihi)
VALUES('MRY','insta',GETDATE());
```

```
INSERT INTO
sc.Gonderi(KullaniciID,KullaniciAdi,SosyalMedyaAdi,GonderiMetni,GonderiTuru,GonderiTarihi)
VALUES(2,'müslüm baba','insta','Huzurum Kalmkdjlksadi Ulen','Serzeniş',GETDATE());
```

```
INSERT INTO sc.Begeni(KullaniciID,KullaniciAdi,GonderiID,SosyalMedyaAdi,TakipciAdi)
VALUES(2,'müslüm baba', 4,'insta','MRY');
```

INSERT INTO

sc.Yorum(KullaniciAdi,KullaniciID,GonderiID,SosyalMedyaAdi,TakipciAdi,YorumMetni)
VALUES('müslüm baba',2,4,'insta','MRY','İASDçim Yanar Ferdi Baba ahhhh');

```
select * from sc.Takipci;
select * from sc.Kullanici;
select * from sc.SosyalMedya;
select * from sc.Gonderi;
select * from sc.Yorum;
```

ResultsMessagesChart

1	MRY	insta	2025-01-05
2	Rasid Yilan	TikTok	2025-01-05

	KullaniciID	KullaniciIsim
1	1	Ferdi Tayfur
2	2	mmüslüm baba

Results grid

	SosyalMedyaAdi	KullaniciAdi	EPosta	Sifre
1	insta	MRY	NULL	NULL
2	insta	müslüm baba	NULL	NULL
3	TikTok	Ferdi Tayfur	NULL	NULL
4	TikTok	Rasid Yilan	NULL	NULL

	GonderiID	KullaniciAdi	KullaniciID	SosyalMedyaAdi	GonderiMetni	GonderiTarihi	GonderiTuru	YorumSayisi	BegeniSayisi
1	3	Ferdi Tayfur	1	TikTok	Huzurum Kalmadi Ulen	2025-01-05	Serzenis	2	1
2	4	müslüm baba	2	insta	Huzurum Kalmkdjlksadi Ulen	2025-01-05	Serzenis	2	2

	TakipciAdi	GonderiID	KullaniciID	SosyalMedyaAdi	KullaniciAdi	YorumMetni
1	MRY	4	2	insta	müslüm baba	IASDçim Yanar Ferdi Baba ahhhh
2	Rasid Yilan	3	1	TikTok	Ferdi Tayfur	İçim Yanar Ferdi Baba ahh
3	Rasid Yilan	3	1	TikTok	Ferdi Tayfur	İçim Yanar Ferdi Baba ahhhh
4	Rasid Yilan	4	2	insta	müslüm baba	İçim Yanar Ferdi Baba ahhhh

SONUÇ VE ÖZET

Takipçi Davranışı Analiz ve Otomasyon Sistemi Proje Amacı

Takipçi Davranışı Analiz ve Otomasyon Sistemi, kullanıcıların sosyal medya platformlarında yaptığı etkileşimleri analiz ederek içerik üreticilerine veri toplama ve bu veriler doğrultusunda içeriklerini optimize etme fırsatı sunmayı hedefleyen bir programdır. Bu sistem sayesinde kullanıcılar, sosyal medya içeriklerini daha etkili bir şekilde planlayarak hedef kitlelerine daha uygun içerikler sunabilir ve etkileşim oranlarını artırabilir.

Projenin İşlevleri

Kullanıcı Etkileşimlerinin Takibi:

- Kullanıcıların sosyal medya platformlarında yaptığı beğeni, yorum ve takip işlemleri kayıt altına alınır.
- Bu etkileşimlerin tarih ve platform bilgisiyle detaylandırılması sağlanır.

Veri Analizi:

1. Kullanıcı davranışlarının analizi ile hangi içeriklerin daha fazla etkileşim aldığı belirlenir.
2. Beğeni ve yorum sayıları analiz edilerek popüler içerikler tespit edilir.

İçerik Optimizasyonu:

1. Toplanan veriler doğrultusunda, kullanıcıların içerik üretim stratejilerini geliştirmelerine yardımcı olunur.
2. Hangi tür içeriklerin daha fazla ilgi gördüğü verilerle desteklenir.

Raporlama:

1. Kullanıcıların sosyal medya davranışlarına yönelik raporlar oluşturularak verimlilik artırılır.
2. İçeriklerin etkisi, etkileşim oranlarına göre analiz edilir.

Veritabanı Yapısı

Sistemin temel bileşenleri aşağıdaki gibi tanımlanmıştır:

1. Kullanıcılar (Kullanici):

- KullanıcıID
- KullaniciIsim

2. Sosyal Medya (SosyalMedya):

- SosyalMedyaAdi
- KullaniciAdi
- EPosta
- Sifre

3. Gönderiler (Gonderi):

- GonderiID
- KullaniciID
- KullaniciAdi
- SosyalMedyaAdi
- GonderiMetni
- GonderiTuru
- GonderiTarihi
- YorumSayisi
- BeğeniSayisi

4. Yorumlar (Yorum):

- YorumID
- KullaniciAdi
- KullaniciID
- GonderiID
- SosyalMedyaAdi
- TakipciAdi
- YorumMetni

5. Beğeniler (Beğeni):

- BeğeniID
- KullaniciID
- KullaniciAdi
- GonderiID
- SosyalMedyaAdi
- TakipciAdi

6. Takipçiler (Takipci):

- TakipciID
- TakipciAdi
- TakipEdilen
- TakipTarihi

Örnek Veritabanı Sorguları

Projenin veritabanı, ilişkisel bir model üzerinden tasarlanmıştır. Aşağıda örnek SQL sorguları bulunmaktadır:

Kullanıcı Ekleme:

```
INSERT INTO sc.Kullanici(KullaniciIsim)
VALUES ('müslüm baba');
```

Sosyal Medya Kullanıcısı Ekleme:

```
INSERT INTO sc.SosyalMedya(SosyalMedyaAdi, KullaniciAdi)
VALUES ('insta', 'müslüm baba');
```

Gönderi Ekleme:

```
INSERT INTO sc.Gonderi(KullaniciID, KullaniciAdi, SosyalMedyaAdi, GonderiMetni, GonderiTuru, GonderiTarihi)
VALUES (2, 'müslüm baba', 'insta', 'Huzurum Kalmkdjlksadi Ulen', 'Serzeniş', GETDATE());
```

Beğeni Ekleme:

```
INSERT INTO sc.Begeni(KullaniciID, KullaniciAdi, GonderiID, SosyalMedyaAdi, TakipciAdi)
VALUES (2, 'müslüm baba', 4, 'insta', 'MRY');
```

Yorum Ekleme:

```
INSERT INTO sc.Yorum(KullaniciAdi, KullaniciID, GonderiID, SosyalMedyaAdi, TakipciAdi, YorumMetni)
VALUES ('müslüm baba', 2, 4, 'insta', 'MRY', 'İASDçim Yanar Ferdi Baba ahhhh');
```

Verileri Görüntüleme:

```
SELECT * FROM sc.Takipci;
SELECT * FROM sc.Kullanici;
SELECT * FROM sc.SosyalMedya;
SELECT * FROM sc.Gonderi;
SELECT * FROM sc.Yorum;
```

Sonuç

Bu proje, sosyal medya kullanıcılarının davranışlarını analiz ederek içerik üreticilerine yol göstermeyi amaçlamaktadır. Sağlanan veri analitiği ve raporlama özellikleri sayesinde, kullanıcılar etkileşimlerini artırabilecek ve içeriklerini daha verimli hale getirebilecektir. Sistem aynı zamanda işletmelerin sosyal medya stratejilerini güçlendirmeleri için de bir araç olarak kullanılabilir.