**VERİTABANI YÖNETİM SİSTEMLERİ FİNAL PROJESİ**

**Taha**

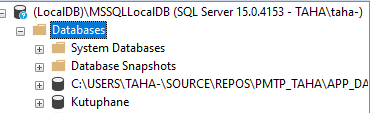
**PALABIYIK**

**95200115**

**1-Veritabanı tasarlama:**

CREATE DATABASE Kutuphane

USE Kutuphane

****

Database Diagram:

**tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu**

CREATE SORGULARI:

CREATE TABLE KITAP (

kitapid INT,

ad VARCHAR(50),

sayfa INT,

yil INT,

tur VARCHAR(30),

PRIMARY KEY (kitapid));

CREATE TABLE SATIS (

satisid INT,

kitapid INT,

musteriisim VARCHAR(50),

satistarih DATE,

PRIMARY KEY (satisid),

FOREIGN KEY (kitapid) REFERENCES KITAP(kitapid));

INSERT SORGULARI:

INSERT INTO KITAP (kitapid, ad, sayfa, yil, tur)

VALUES

(1, 'Lord of the ring', 150, 1980, 'Action'),

(2, 'Beach', 300, 1990, 'Drama'),

(3, 'Hours', 350, 1991, 'Mystery'),

(4, 'Sis ve gece', 330, 1982, 'Drama'),

(5, '1984', 600, 1981, 'Action'),

(6, 'Kabus', 500, 1986, 'Drama'),

(7, 'About a Boy', 180, 1989, 'Action'),

(8, 'Anna Karanina', 200, 1988, 'Action'),

(9, 'Parfume', 230, 1995, 'Mystery'),

(10, 'Rose Red', 110, 1999, 'Action');

INSERT INTO SATIS (satisid, kitapid, musteriisim, satistarih)

VALUES

(1, 10, 'Thomas Harris', '2022-10-05'),

(2, 10, 'Ahmet Ümit', '2022-10-05'),

(3, 7, 'Mario Puzzo', '2022-10-06'),

(4, 6, 'Dan Brown', '2022-10-06'),

(5, 4, 'James Butt', '2022-10-06'),

(6, 1, 'Arthur Venere', '2022-10-06'),

(7, 5, 'Lena Paprocki', '2022-10-07'),

(8, 6, 'Donald Foller', '2022-10-07'),

(9, 9, 'Sue Tollner', '2022-10-09'),

(10,8, 'Simon Morasca', '2022-10-10');

2-WHERE + LIKE

SELECT \* FROM KITAP Where ad LIKE '%a%';

Ekran Çıktısı:

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

İç - İçe Select

SELECT \* FROM KITAP where sayfa =

(SELECT Max(sayfa) from KITAP);

Ekran Çıktısı:

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

Group By + AVG

SELECT tur, AVG(sayfa) as sayfaOrtalama

FROM KITAP

GROUP BY tur;

Ekran Çıktısı:

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

Top + Order By

SELECT TOP 1 \* FROM KITAP ORDER BY sayfa desc;

Ekran Çıktısı:

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

GROUP BY + SUM + COUNT

SELECT tur, SUM(sayfa) AS Sayfatoplam, COUNT(\*) as Kitap

FROM KITAP

GROUP BY tur;

Ekran Çıktısı:

**tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu**

JOIN + GROUP BY + AVG

SELECT KITAP.tur, AVG(KITAP.sayfa) as sayfaOrtalama

FROM SATIS JOIN KITAP ON SATIS.kitapid=KITAP.kitapid

GROUP BY tur;

Ekran Çıktısı:

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

JOIN + GROUP BY + AVG + WHERE

SELECT KITAP.tur, AVG(KITAP.sayfa) as sayfaOrtalama

FROM SATIS JOIN KITAP ON SATIS.kitapid=KITAP.kitapid

Where KITAP.ad LIKE '%g%'

GROUP BY KITAP.tur;

Ekran Çıktısı:

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

MAX-MIN

SELECT Max(sayfa) FROM KITAP;

Ekran Çıktısı:

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

IN

SELECT ad FROM KITAP where sayfa IN

(SELECT Max(sayfa) from KITAP);

Ekran Çıktısı:

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

GROUP BY + HAVING

SELECT tur, AVG(sayfa) as sayfaOrtalama

FROM KITAP

GROUP BY tur

HAVING AVG(sayfa)>100;

Ekran Çıktısı:

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

3)Ekleme (insert) yapan stored procedure

CREATE PROC KitapEkle

@kitapid INT,

@ad VARCHAR(50),

@sayfa INT,

@yil INT,

@tur VARCHAR(30)

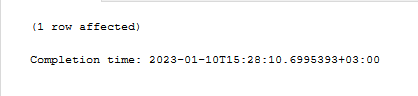
AS

BEGIN

INSERT INTO KITAP values(@kitapid, @ad, @sayfa, @yil, @tur)

END

Ekran Çıktısı:





Like sorgusunu içeren stored procedure:

CREATE PROC kitapArama

@ad varchar(50)

AS

BEGIN

SELECT \* FROM KITAP

WHERE ad LIKE @ad+'%';

END

EXEC kitapArama 'a'

Ekran Çıktısı:

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

4) INSERT TRIGGER

CREATE TRIGGER Kitapeklenen

ON KITAP

FOR INSERT

AS

BEGIN

DECLARE @kitapid INT

SELECT @kitapid=kitapid

FROM INSERTED;

INSERT INTO tblKitapEkleme VALUES(@kitapid,GETDATE());

END

EXEC KitapEkle 14 ,'Yeni Kitap ', 150, 1975 ,'Drama'

EKRAN ÇIKTISI:

**tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu**

DELETE TRIGGER

CREATE TRIGGER KitapSilinen

ON KITAP

AFTER DELETE

AS

BEGIN

DECLARE @kitapid INT

SELECT @kitapid=kitapid

FROM DELETED;

INSERT INTO tblSilinenKitap VALUES(@kitapid,GETDATE());

END

DELETE FROM KITAP WHERE kitapid =14;

EKRAN ÇIKTISI:

**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**5) Case - When**

SELECT ad,

CASE

WHEN sayfa BETWEEN 0 AND 50 THEN 'Kısa Kitap'

WHEN sayfa BETWEEN 50 AND 200 THEN 'Orta Kitap'

WHEN sayfa BETWEEN 200 AND 400 THEN 'Uzun Kitap'

WHEN sayfa >400 THEN 'Çok Uzun Kitap'

ELSE 'Belirsiz'

END sayfaBoyutu

FROM KITAP

EKRAN ÇIKTISI:

****