



Veritabanı Yönetimi Proje Raporu

20011044

Yusuf Enes Kurt

19011093

Fatih Ecevit

19011042

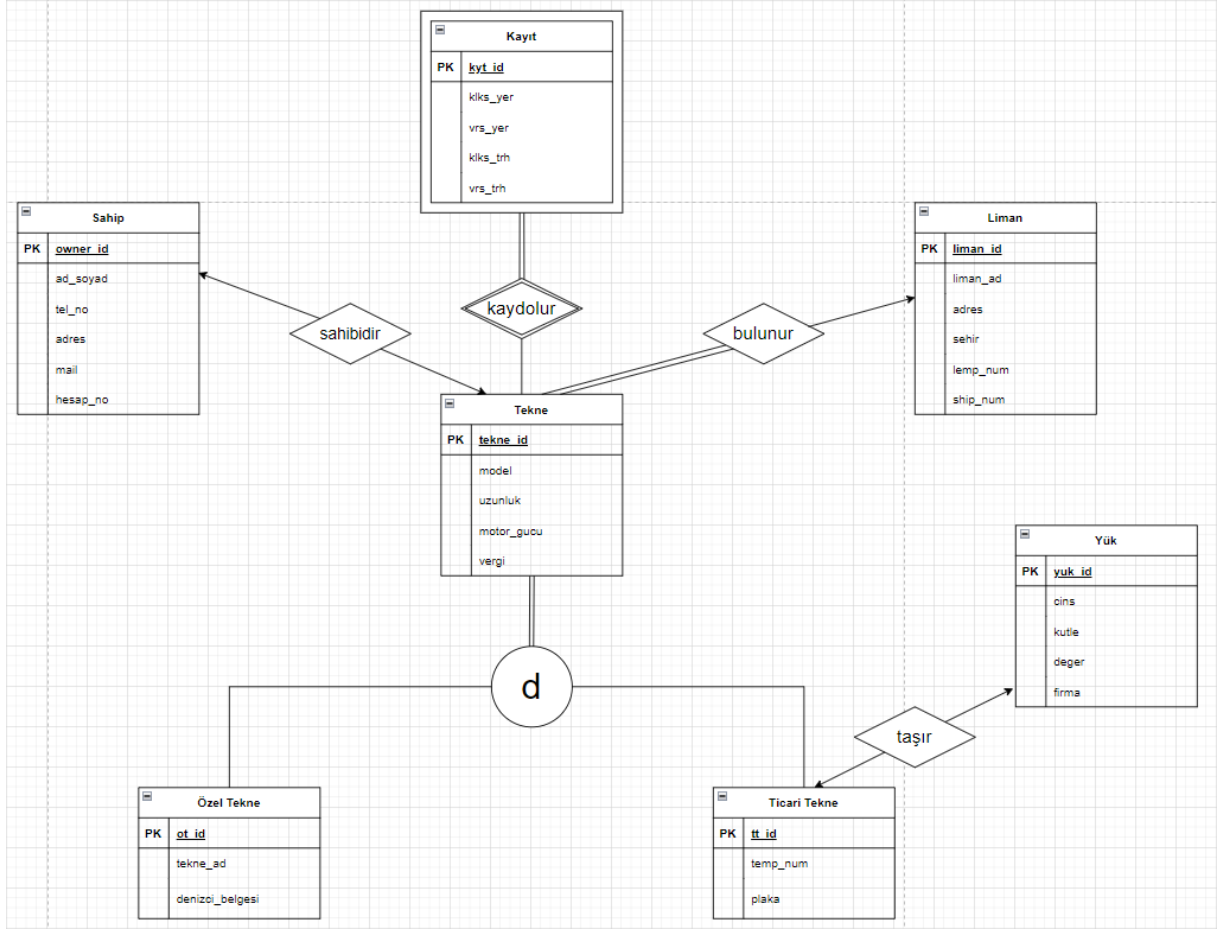
Yusuf Mert Çelikarslan

Ders Yürütücüsü

Dr. Öğr. Üyesi **Mustafa Utku Kalay**

13.01.2022

TASARIM (E-R DİYAGRAMI)



TABLolar

<<TEKNE>> TABLOSU

TEKNE_ID*	MODEL	LIMAN_ID*	OWNER_ID*	UZUNLUK	MOTOR_GUCU	VERGI
100	Dufour	1000	55090358900	7,20	11	100,3
101	Jeanneau	1002	47758075812	6,50	12	200,6
102	Leoard	1002	28595012456	5,00	16	601,8
103	Catana	1001	12598645782	9,40	20	1003
104	Bavaria	1002	28595012456	8,30	12	200,6
105	Benetau	1004	12598645782	8,50	8	0
106	Elan	1005	55090358900	7,50	6	0
107	Hanse	1006	12598645782	10,00	14	401,2
108	Leopard	1000	47758075812	8,20	12	200,6
INTEGER	VARCHAR(32)	INTEGER	CHAR(11)	NUMERIC(4,2)	INTEGER	NUMERIC(6,2)

<<OZEL_TEKNE>> TABLOSU

OT_ID*	TEKNE_AD	DENIZCI_BELGESI
100	Mavi Firtina	1
101	Denizin Sultani	1
102	Gokyuzum	0
103	Kaptan Kosku	1
INTEGER	VARCHAR(40)	CHAR(1)

<<TICARI_TEKNE>> TABLOSU

TT_ID*	TEMP_NUM	YUK_ID*	PLAKA
104	12	10	61TSA1961
105	8	16	66ECO1907
106	7	12	06ANK1881
107	13	10	34IST1453
108	12	11	55SMS1919
INTEGER	SMALLINT	INTEGER	CHAR(9)

<<YUK>> TABLOSU

YUK_ID*	CINS	KUTLE	DEGER	FIRMA
10	Mobilya	400.12	102.3	Istikbal
11	Mobilya	845.56	256.3	Bellona
12	Teknolojik	39.5	456.9	Samsung
13	Teknolojik	56.8	512.8	Apple
14	Insaat	500	436.2	Agaoglu Insaat
15	Insaat	425.8	105.3	Gultekin Insaat
16	Canli	375.65	85.1	Erol Hayvancilik
INTEGER	VARCHAR(30)	FLOAT	FLOAT	VARCHAR(30)

<<SAHIP>> TABLOSU

OWNER_ID*	AD_SOYAD	TEL_NO	ADRES	MAIL	HESAP_NO
55090358900	Fatih Ecevit	5536844160	FSM Yurdu	feceveit@gmail.com	TR12345678901234
47758075812	Yusuf Kurt	5556877456	Istanbul Esenler	ykurt@gmail.com	TR52345678904455
12489563874	Ahmet Unal	5417488966	Yozgat Sorgun	aunal@aoutlook.com	TR45645678901234
28595012456	Ayşe Kılıç	5075158901	Afyon Kabataş	akilic@gmail.com	TR01264567890123
12598645782	Betül Çanak	5557994120	Ankara Çankaya	betulc@gmail.com	TR86264567890123
CHAR(11)	VARCHAR(30)	CHAR(10)	VARCHAR(50)	VARCHAR(40)	CHAR(16)

<<LIMAN>> TABLOSU

LIMAN_ID*	LIMAN_AD	ADRES	SEHIR	LEMP_NUM	SHIP_NUM
1000	Trabzon61	Trabzon Macka Balıkçılar Sokak	Trabzon	80	2
1001	Samsun55	Samsun Isleyen Sokak Caddesi	Samsun	50	1
1002	Rize53	Rize Baskanin Yeri	Rize	70	3
1003	Canakkale18	Canakkale Sehitler Mahellesi	Canakkale	40	0
1004	Istanbul34	Istanbul Eminonu	Istanbul	150	1
1005	Antalya07	Antalya Kemer Bolge Bayii Yani	Antalya	120	1
1006	Alanya07	Alanya Perdeciler Sokak	Alanya	50	1
1007	Izmir35	Izmir Alsancak Bolge Merkezi	Izmir	95	0
1008	Balıkesir10	Balıkesir Gizemli Anayurt	Balıkesir	42	0
INTEGER	VARCHAR(20)	VARCHAR(50)	VARCHAR(20)	INTEGER	INTEGER

<<KAYIT>> TABLOSU

KYT_ID*	GEMI_ID	KLKS_YER	VRS_YER	KLKS_TRH	VRS_TRH
1	103	Trabzon	Samsun	15.10.2022	1.01.2023
2	105	Balikesir	Alanya	11.10.2022	2.01.2023
3	107	Samsun	Izmir	10.08.2022	3.01.2023
4	108	Antalya	Canakkale	6.05.2022	4.01.2023
5	102	Rize	Trabzon	6.09.2022	1.01.2023
6	104	Istanbul	Alanya	11.11.2022	1.03.2023
7	106	Antalya	Balikesir	10.03.2022	1.06.2023
INTEGER	INTEGER	VARCHAR(20)	VARCHAR(20)	DATE	DATE

KOD BLOKLARI

Primary key, foreign key ve silme kısıtları

```
CREATE TABLE ozel_tekne(  
    ot_id int,  
    tekne_ad VARCHAR(40) NOT NULL,  
    denizci_belgesi char(1) NOT NULL,  
    primary key (ot_id),  
    CONSTRAINT fk_ozel FOREIGN KEY  
    (ot_id) REFERENCES tekne (tekne_id)  
    ON DELETE CASCADE  
);
```

Sayı kısıtı

```
CREATE TABLE liman(  
    liman_id int,  
    liman_ad VARCHAR(20) NOT NULL,  
    adres VARCHAR(50) NOT NULL,  
    sehir VARCHAR(20) NOT NULL,  
    lemp_num int check(lemp_num >=0),  
    ship_num int check(ship_num >=0),  
    primary key (liman_id)  
);
```

Insert, update ve delete işlemi

Girilen değere göre ekrana sonuçların listelendiği bir sorgu

View

```
/*view*/  
CREATE VIEW tekne_bilgi  
AS  
SELECT distinct tekne.tekne_id, model, liman_id,owner_id,uzunluk,motor_gucu, vergi, tekne_ad, denizci_belgesi,temp_num,yuk_id,plaka  
FROM tekne natural left outer join ozel_tekne natural left outer join ticari_tekne;
```

Sequence

```
CREATE SEQUENCE tekne_seq
START with 108
INCREMENT BY 1
MINVALUE 1
MAXVALUE 1000;

CREATE SEQUENCE yuk_seq
START with 10
INCREMENT BY 1
MINVALUE 10
MAXVALUE 16
CYCLE;
```

Union veya intersect veya except

Aggregate fonksiyonu ve having ifadesi

3 farklı SQL fonksiyonu ve en az birinde record tanımı-kullanımı

```
CREATE OR REPLACE FUNCTION ort_lemp_num()
RETURNS real AS $$

DECLARE
    ort_iscil numeric;
BEGIN

    SELECT avg(lemp_num) INTO ort_iscil
    FROM liman;
    RETURN ort_iscil;

END;
$$ LANGUAGE 'plpgsql';

CREATE TYPE istikamet AS (gemi_isim VARCHAR(40), kalkis VARCHAR(20), varis VARCHAR(20));

/*record ve fonksiyon*/
CREATE OR REPLACE FUNCTION ozel_tekne_ozet(t_id ozel_tekne.ot_id%type)
RETURNS istikamet AS $$
DECLARE
    bilgi istikamet;
BEGIN
    SELECT o.tekne_ad, klks_yer, vrs_yer INTO bilgi
    FROM ozel_tekne o, kayıt
    WHERE gemi_id = t_id;
    RAISE NOTICE 'Tekne ismi: %, kalkis yeri: %, varis yeri: % ',
        bilgi.gemi_isim, bilgi.kalkis, bilgi.varis;
    RETURN bilgi;
END;
$$ LANGUAGE 'plpgsql';
```

1tane daha yazılacak

Triggerlar ve cursor tanımı-kullanımı

```
--DROP TRIGGER ayni_isim ON ozel_tekne
CREATE TRIGGER ayni_isim
BEFORE INSERT
ON ozel_tekne
FOR EACH ROW EXECUTE PROCEDURE trig_fonk_ayni_isim();

--DROP TRIGGER ayni_plaka ON ticari_tekne
CREATE TRIGGER ayni_plaka
BEFORE INSERT
ON ticari_tekne
FOR EACH ROW EXECUTE PROCEDURE trig_fonk_ayni_plaka();
```

```

--DROP TRIGGER shipnum_arttir ON tekne
CREATE TRIGGER shipnum_arttir
AFTER INSERT
ON tekne
FOR EACH ROW EXECUTE PROCEDURE trig_fonk_shipnum_arttir();

--DROP TRIGGER shipnum_azalt ON tekne
CREATE TRIGGER shipnum_azalt
AFTER DELETE
ON tekne
FOR EACH ROW EXECUTE PROCEDURE trig_fonk_shipnum_azalt();

CREATE OR REPLACE FUNCTION trig_fonk_ayni_isim()
RETURNS TRIGGER AS $$
DECLARE
    yeni_cur CURSOR FOR SELECT ot_id,tekne_ad FROM ozel_tekne;
    eskiLiman INTEGER;
    yeniLiman INTEGER;
BEGIN
    FOR isim IN yeni_cur LOOP
        IF(isim.tekne_ad = new.tekne_ad) THEN
            SELECT liman_id INTO eskiLiman FROM tekne,ozel_tekne WHERE isim.ot_id = tekne_id;
            SELECT liman_id INTO yeniLiman FROM tekne,ozel_tekne WHERE new.ot_id = tekne_id;
            IF(eskiLiman = yeniLiman) THEN
                DELETE FROM tekne WHERE new.ot_id = tekne_id ;
                RAISE NOTICE 'Aynı isimde tekne ekleyemezsiniz';
                RETURN null;
            END IF;
        END IF;
    END LOOP;
    RETURN new;
END;
$$ LANGUAGE 'plpgsql';
CREATE OR REPLACE FUNCTION trig_fonk_ayni_plaka()
RETURNS TRIGGER AS $$
DECLARE
    yeni_cur2 CURSOR FOR SELECT tt_id,plaka FROM ticari_tekne;
    eskiLiman INTEGER;
    yeniLiman INTEGER;
BEGIN
    FOR isim IN yeni_cur2 LOOP
        IF(isim.plaka = new.plaka) THEN
            SELECT liman_id INTO eskiLiman FROM tekne,ticari_tekne WHERE isim.tt_id = tekne_id;
            SELECT liman_id INTO yeniLiman FROM tekne,ozel_tekne WHERE new.tt_id = tekne_id;
            IF(eskiLiman = yeniLiman) THEN
                DELETE FROM tekne WHERE new.tt_id = tekne_id;
                RAISE NOTICE 'Aynı plakada tekne ekleyemezsiniz';
                RETURN null;
            END IF;
        END IF;
    END LOOP;
    RETURN new;
END;
$$ LANGUAGE 'plpgsql';
CREATE OR REPLACE FUNCTION trig_fonk_shipnum_arttir()
RETURNS TRIGGER AS $$
BEGIN
    UPDATE liman SET ship_num=ship_num+1 WHERE liman_id IN(SELECT l.liman_id
        FROM tekne,liman l
        WHERE new.liman_id=l.liman_id);
    RETURN new;
END;
$$ LANGUAGE 'plpgsql';
CREATE OR REPLACE FUNCTION trig_fonk_shipnum_azalt()
RETURNS TRIGGER AS $$
BEGIN
    UPDATE liman SET ship_num=ship_num-1 WHERE liman_id IN(SELECT l.liman_id
        FROM tekne,liman l
        WHERE old.liman_id=l.liman_id);
    RETURN new;
END;
$$ LANGUAGE 'plpgsql';

```